



UNIVERSIDAD PONTIFICIA DE SALAMANCA

Campus de Madrid

Facultad de Informática

DOCTORADO EN INGENIERÍA INFORMÁTICA

Programa de Ingeniería del Software
BIENIO 2003-2005

ASIGNATURA:

INGENIERÍA WEB Y SERVICIOS WEB

TRABAJO DE INVESTIGACIÓN:

**NUEVAS TENDENCIAS DE LOS SERVICIOS WEB:
ARQUITECTURA ORIENTADA A SERVICIOS Y
TECNOLOGÍA GRID**

PROFESOR:

Dr. D. Juan Manuel Cueva Lovelle

AUTOR:

César Parejas Llanovarced

Madrid, Abril de 2005

ÍNDICE

INTRODUCCIÓN.....	3
CAPITULO 1: LA INFORMÁTICA DISTRIBUIDA	4
1.1 DEL MODELO CLIENTE/SERVIDOR A LA INFORMÁTICA DISTRIBUIDA.....	4
1.2 EVOLUCIÓN DE LAS ARQUITECTURAS	5
1.3 LA NECESIDAD DE INTEROPERABILIDAD	6
1.4 UNA NUEVA TECNOLOGÍA DE INTEGRACIÓN: LOS SERVICIOS WEB	8
1.5 HACIA EL UTILITY COMPUTING.....	8
CAPITULO 2: LA INFORMÁTICA COMO UNA RED DE SERVICIOS.....	11
2.1 SERVICIOS WEB E INTEGRACIÓN DE APLICACIONES	11
2.2 SOA, UNA ARQUITECTURA PARA LOS SERVICIOS BAJO DEMANDA	13
2.2.1 INTEGRACIÓN APLICACIÓN-CON-APLICACIÓN	14
2.2.2 SODA (DESARROLLO DE APLICACIONES ORIENTADAS A LOS SERVICIOS)	14
2.2.3 LA RED DE SERVICIOS.....	15
2.3 EL USO DE LA TECNOLOGÍA ESTÁNDAR.....	19
CAPITULO 3: GRID COMPUTING	21
3.1 ¿QUÉ ES EL GRID COMPUTING?	21
3.1.1 DEFINICIÓN DE GRID	23
3.2 COMPONENTES DE UN SISTEMA GRID.....	24
3.2.1 ELEMENTOS QUE COMPONEN UN GRID	24
3.2.2 PROCESO DE EJECUCIÓN DE UNA TAREA	25
3.3 BENEFICIOS TECNOLÓGICOS DEL GRID.....	27
3.4 TECNOLOGÍAS GRID E INFORMÁTICA DISTRIBUIDA.....	27
CAPITULO 4: LA INFORMÁTICA GRID COMO UTILITY BAJO DEMANDA.....	29
4.1 COMUNIDADES VIRTUALES.....	29
4.2 EL FUTURO DE LA INFORMÁTICA GRID	31
CAPITULO 5: TECNOLOGÍAS GRID Y SERVICIOS WEB: HACIA UNA PATAFORMA INFORMÁTICA GLOBAL	34
5.1 LAS DOS CARAS DE LA MONEDA	34
5.2 EL VALOR DE LO VIRTUAL	35
5.3 ARQUITECTURA DE SERVICIOS ABIERTOS GRID (OGSA).....	37
CAPITULO 6: PANORAMA ACTUAL DEL SOFTWARE GRID.....	39
6.1 AVAKI DATA GRID, AVAKI COMPUTE GRID, AVAKI COMPREHENSIVE GRID.....	39
6.2 DCGRID	41
6.3 LSF, ACTIVECLUSTER, MULTICLUSTER, PLATFORMGLOBUS.....	42
6.4 GRID MP (METAPROCESSOR)	44
6.5 LIVECLUSTER	45
6.6 INNERGRID	46
6.7 GLOBUS TOOLKIT	49
CONCLUSIONES.....	51
BIBLIOGRAFIA Y REFERENCIAS WEB	52

RESUMEN

Compartición, virtualización y distribución son palabras clave comunes en todos los discursos sobre tecnologías Grid y servicios Web; como comunes empiezan a ser también los protocolos sobre los que ambas propuestas se sustentan (XML, SOAP, UDDI y WSDL). No es casualidad que la lista de los promotores de Grid incluya a los suministradores cuyas estrategias apuestan fuerte por los servicios Web y la virtualización de los recursos, como son Sun Microsystems (Sun One y N1), Microsoft (.Net), Hewlett-Packard (Planetary Computing) e IBM (eLiza). De la convergencia de ambos campos surgirá según muchos, una nueva Internet transformada en plataforma universal.

Palabras Claves: Grid, servicios Web, virtualización, distribución, SOA, SODA, utility computing, OGSA

INTRODUCCIÓN

La nueva generación de arquitecturas basadas en servicios tiene un potencial enorme de negocio gracias a tres tecnologías interrelacionadas que están emergiendo para revolucionar y cambiar Internet tal y como lo conocemos: servicios Web (crear componentes de software reusables como servicios), grid services (capacidad de procesamiento, almacenamiento y memoria bajo demanda), y la web semántica (expresar la información de forma procesable por una máquina). Las tres convertirán a Internet en una plataforma de desarrollo que soporte en tiempo real la creación de servicios personalizados, en contraste con las comunicaciones basadas en contenido estático que caracterizan actualmente su uso.

En el presente trabajo, analizaremos estas dos primeras tecnologías como punto de partida en el advenimiento de Internet como una “utility” que enlace, conecte y sirva de intermediario para todas las transacciones de nuestra sociedad. El ideal de acceder a cualquier información, desde cualquier dispositivo y en cualquier momento será un hecho e Internet será el contenedor universal de aplicaciones que permitirá a las aplicaciones ejecutarse utilizando protocolos estándar.

CAPITULO 1: LA INFORMÁTICA DISTRIBUIDA

Se está produciendo una regeneración de la industria informática, una evolución que cambiará radicalmente el modelo de negocio de este sector, y que afectará a todos sus actores alterando la forma en la que se crea el software, la forma de venderlo, cómo se implanta y donde se ejecuta.

1.1 DEL MODELO CLIENTE/SERVIDOR A LA INFORMÁTICA DISTRIBUIDA

Este cambio se producirá al permitir a las aplicaciones trabajar de forma transparente y colaborativa unas con otras a través de distintos sistemas, plataformas, sistemas operativos y lenguajes de programación.

Para que esto sea posible se necesita operar bajo un entorno distribuido, donde las aplicaciones puedan encontrar automáticamente otras aplicaciones con las que colaborar y construir nuevas aplicaciones con sólo acoplar módulos funcionales que algún día se podrían comprar como servicios.

Un servicio desde el punto de vista informático es una pieza reusable con lógica autocontenida, que sabe como realizar una tarea sin preocuparse de quien y para que ha sido llamado. Los desarrolladores hace tiempo que detectaron la necesidad de servicios. Sin una programación de la lógica reusable se tiene que reescribir el mismo código múltiples veces y después mantenerlo. En los primeros tiempos de la informática, cuando los sistemas se limitaban a una plataforma host, los servicios reusables se llamaban subrutinas o llamadas a programas externos; éstas fueron las primeras versiones de servicios. Con la informática distribuida surge la necesidad de crear servicios reusables que operen a través de distintas máquinas. Con la madurez tecnológica de Internet la noción de servicio pasa a otro nivel.

Asistimos a un cambio de arquitecturas de sistemas, desde los tradicionales modelos cliente/servidor hacia nuevas arquitecturas orientadas al desarrollo de

servicios (Arquitectura Orientada a Servicios SOA. *Service Oriented Architecture*) para afrontar las necesidades actuales tanto en funcionalidad como en ahorro de costes.

1.2 EVOLUCIÓN DE LAS ARQUITECTURAS

La siguiente tabla muestra los principales efectos de los cambios de paradigma.

ARQUITECTURA	MAINFRAME	CLIENTE/SERVIDOR	SOA
PLATAFORMA	Monolítica y centralizada	Homogénea y controlada	Diversa e impredecible
RED	Limitada y cerrada	LAN	Internet
FORMATO DATOS	Opacos e inaccesibles	Binarios y propietarios	Semántica y compartida
TECNOLOGÍA	Sistema operativo	Base de datos	Interfaz
USUARIOS	Departamento TI	Empleados	Socios, clientes y empleados
VALOR DE NEGOCIO	Centralización de datos	Poner los datos al alcance de los usuarios de negocio	Permitir colaboración y agilidad de negocio

Tabla 1. Fuente: The Stencil Group

La transición del modelo cliente/servidor hacia arquitecturas orientadas al desarrollo de servicios nació de la necesidad de interoperabilidad entre aplicaciones y posterior consolidación de la informática distribuida. La historia comenzó hace un par de décadas, cuando las aplicaciones informáticas se ejecutaban en grandes ordenadores. Más tarde aparecieron terminales que se conectaban a estos ordenadores para que los usuarios los utilizaran por medio de comandos escritos en texto. En los años ochenta surge el ordenador personal o PC, en el que los usuarios podían ejecutar sus propias aplicaciones. Durante aquellos años los protocolos de comunicación todavía carecen de relevancia cuando el reto estaba en la comunicación entre aplicaciones.

A principios de los noventa surgen tecnologías basadas en objetos como el Modelo de Objetos Componente de Microsoft (COM, Component Object method) y la Arquitectura de Negociación de Petición de Objetos Comunes (CORBA, Common Object Request Broker Architecture), diseñadas para la escritura y encapsulación de código binario. Permitían el desarrollo de componentes que se

podían invocar desde cualquier aplicación que soportara estas tecnologías de manera estandarizada y sencilla. La falta de interoperabilidad de ambas tecnologías no consiguió acabar con plataformas informáticas constituidas por máquinas independientes.

En los años noventa nacen las redes locales para conectar ordenadores personales, y la conexión entre máquinas empieza a ser una prioridad. Fabricantes y organizaciones que contaban con estructuras propietarias de modelo de objetos las amplían para permitir la comunicación a través de redes.

Como protocolo de cable estándar de CORBA surge IIOP (Internet Inter-ORB Protocol, el protocolo para la comunicación entre ORB); a su vez Microsoft lanza el Modelo de Objetos Componentes Distribuidos (DCOM, Distributed Component Object Model) y, de la mano de Sun Microsystems, aparece otro competidor, la Invocación a Métodos Remotos (RMI, Remote Method Invocation) que utilizarán los usuarios de java.

1.3 LA NECESIDAD DE INTEROPERABILIDAD

Utilizando estos protocolos, una aplicación podía invocar componentes que residían en otros ordenadores de la red, mediante una Llamada a Procedimiento Remoto (RPC, Remote Procedure Call), para invocar éste y enviar respuesta a la aplicación que realiza la llamada. Aunque el avance era significativo, se mantuvo la interoperabilidad entre estas tecnologías, de forma que los usuarios que utilizaban cualquiera de los protocolos sólo podían llamar a servidores compatibles.

La conexión de aplicaciones utilizando estos protocolos se caracteriza por su buen funcionamiento si dichas aplicaciones se encuentran en la misma red local. Pero con la aparición de Internet y en particular de la web, la red creció rápidamente y se volvió extremadamente distribuida y descentralizada. Cambiaron las reglas del juego y lo que funcionaba dentro de una red local no tenía porque hacerlo en la web. ¿Cómo se podrían utilizar entonces los protocolos de aplicación distribuida actuales?

Estos protocolos suelen ser complejos de implantar, requieren de cierta simetría en cuanto que ambos extremos de la comunicación necesitan tener implantado el mismo modelo de distribución de objetos. Se trata de soluciones propietarias que no son compatibles con todos los sistemas operativos y lenguajes, no traspasan los cortafuegos y la industria no acaba por decantarse por uno en particular.

Así las cosas, las aplicaciones y páginas web empezaron como islas de información con enlaces que permitían al usuario moverse de un lugar a otro, pero no era más que una forma de redirección ya que el usuario debía abandonar la información de la primera página para dirigirse a una segunda.

Con la técnica de marcos, el contenido de un sitio web se podía mostrar junto al contenido de otro, gracias a tecnologías como Java applets o los controles Activex. Su limitación es que sólo se conecta la interfaz de usuario, no las aplicaciones. Surgieron otras técnicas para lograr la interoperabilidad entre aplicaciones web, pero no fue hasta mediados de los noventa, con la aparición del Lenguaje de Marcas Extensible (XML, Extensible Markup Language) y su capacidad para expresar estructuras de información y mensajes de manera uniforme y autodescriptiva, cuando se decidió utilizarla para aplicar un formato a los mensajes intercambiados entre sistemas. Esta técnica permite a los usuarios intercambiar mensajes entre sistemas de manera autodescriptiva y extensible independiente del sistema operativo y lenguaje utilizado.

Las tecnologías para la integración de aplicaciones de empresas basadas en el concepto de Integración de Aplicaciones Empresariales (EAI) propietarias, de conexión rígida y por lo general caras de implantar serán superadas por la tecnología de integración de aplicaciones base del e-business colaborativo [ref. 14]: los servicios web.

1.4 UNA NUEVA TECNOLOGÍA DE INTEGRACIÓN: LOS SERVICIOS WEB

Los servicios web son aplicaciones modulares autodescriptivas que se pueden publicar, ubicar e invocar desde cualquier punto de la red o desde el interior de una red local, basados en estándares abiertos de Internet. Ya no es necesario que el proveedor y el usuario de un servicio web tengan el mismo sistema operativo y utilicen el mismo lenguaje de programación, dado que se basan en estándares aceptados plenamente por la industria, como XML, HTTP y SMTP.

La creación de aplicaciones con servicios web permite migrar de aplicaciones estrechamente ligadas hacia otras menos ligadas, que son más escalables, manejables y extensibles, y menos susceptibles de errores causados por modificaciones, evolucionando hacia un modelo de creación de aplicaciones que pueden descubrirse e integrarse de forma dinámica. En un entorno distribuido, las aplicaciones pueden encontrar automáticamente otras aplicaciones con las que colaborar y construir nuevas aplicaciones con sólo acoplar módulos funcionales que se compran como servicios (pago por uso).

1.5 HACIA EL UTILITY COMPUTING

Aunque el concepto de TI ofrecido como una utility no es nuevo, hoy disponemos de los avances tecnológicos que lo harían posible:

- Menores costes al uso de banda, que permite la creación de redes más veloces que pueden ofrecer servicios a un mayor número de consumidores.
- Contenidos y aplicaciones distribuidas accesibles desde Internet.
- Virtualización de servidores y capacidad de almacenamiento, para compartir infraestructuras reduciendo sus costes.

Pero la llegada del utility computing estará precedida por la aplicación en las empresas de las siguientes tecnologías con el fin de integrar sistemas y plataformas heterogéneas:

- **Tecnologías abiertas** de Internet y el lenguaje Java.

Utilizar estándares abiertos para integrar los sistemas heterogéneos y distribuidos de la empresa la protege para que futuras tecnologías y productos de distintos fabricantes se puedan integrar en su arquitectura de forma sencilla y menos costosa en el futuro. Toda la industria de TI está adoptando una cultura de estándares abiertos para acelerar la integración, en un mundo donde los fabricantes compiten para crear el mejor producto en vez de intentar apropiarse de un segmento del mercado.

- **Servicios Web**

Es una tecnología de integración basada en un conjunto de estándares de Internet que permite la comunicación y ejecución de transacciones entre distintas aplicaciones sin la complejidad de sus predecesoras como CORBA, RMI, DCOM...

- **Grid Computing**

Es una colección de dispositivos informáticos distribuidos disponibles en red para ser utilizados por el usuario o aplicación en forma de supercomputador virtual.

Un grid supera las limitaciones geográficas, organizativas y de arquitectura hardware y software, para ofrecer capacidad de procesamiento ilimitado, colaboración y acceso a la información a cualquiera que esté conectado. Representa la evolución de Internet como tecnología de redes capaz de ofrecer la informática como una utility.

- **Utility computing**

Se aplica de forma conjunta las tecnologías mencionadas, es decir, integramos nuestras aplicaciones y sistemas utilizando estándares abiertos, y creamos servicios web para descubrir y acceder a los recursos disponibles en un grid, lo que obtendríamos serían recursos informáticos factibles de ser compartidos y gestionados globalmente sobre una arquitectura flexible y altamente escalable, preparada para la era on-demand, en el que la informática puede ofrecerse como un servicio en forma de utility.

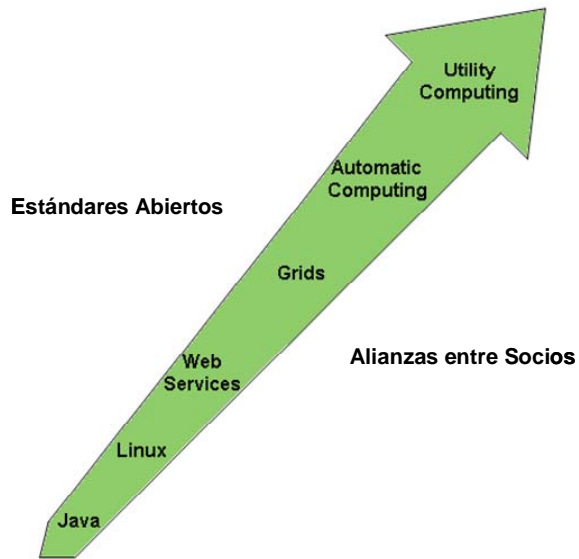


Figura 1. La evolución hacia el Utility Computing. *Fuente: IBM*

CAPITULO 2: LA INFORMÁTICA COMO UNA RED DE SERVICIOS

En un mundo competitivo como el actual el objetivo es crear un entorno de Tecnologías de la Información (TI) que permita a las organizaciones participar de la economía digital donde los procesos de negocio y los sistemas que los soportan son configurados dinámicamente para ejecutar estrategias y tácticas de negocio cambiantes.

Al Departamento de Tecnologías de la Información hoy se le exige potenciar la agilidad y colaboración del negocio, e históricamente cada vez que ha cambiado su rol se ha producido también un cambio en la arquitectura de sistemas, desde las arquitecturas monolíticas y centralizadas propias del mainframe o los formatos de datos propietarios del modelo cliente/servidor, las nuevas arquitecturas tienden a trabajar sobre plataformas distribuidas y heterogéneas, utilizando Internet como red de comunicación, orientadas al desarrollo de interfaces y dirigidas al uso de socios, empleados y clientes.

Estas nuevas arquitecturas reciben el nombre de SOA (Service Oriented Architectures) y representan un modelo de arquitectura informática en el cual pequeños trozos de funcionalidad de una aplicación son publicados, consumidos y pueden ser integrados en otras aplicaciones de la red.

Ofrecen al desarrollador el entorno apropiado donde crear aplicaciones empresariales, publicando la lógica de la aplicación como un servicio.

2.1 SERVICIOS WEB E INTEGRACIÓN DE APLICACIONES

Los servicios Web aparecen como la arquitectura que hace posible la computación distribuida en un área extensa entre sistemas heterogéneos comunicados por Internet. Sus fundamentos son XML, como tecnología base, y sus derivados SOAP, WDSL y UDDI. Esto estándares, junto con cuestiones como la seguridad, la integración con J2EE y los sistemas ya existentes, representan hoy los factores esenciales de los desarrollos de servicios web.

Esta tecnología podría describirse como un paradigma de programación que permite la invocación/llamada remota de objetos de software entre sistemas heterogéneos en Internet, con independencia del tipo de lenguaje. El objetivo de estos servicios –aplicables tanto a los procesos internos de las empresas como al B2B y al B2C- es hacer posible que los sistemas se comuniquen entre sí, reduciendo notablemente los esfuerzos de integración en recursos y tiempos de desarrollo. No hay que olvidar que el trabajo de integrar dos aplicaciones de dos empresas respectivas, una basada en tecnología Microsoft y otra en Java, y donde entra además en juego CORAB/C++, puede prolongarse por espacio de doce meses aproximadamente con medios convencionales. Mientras que utilizando servicios Web, ese periodo puede reducirse a unos pocos días.

En el pasado los programadores debían desarrollar y mantener conexiones entre sistemas; en el futuro las aplicaciones encontrarán otras aplicaciones con las que sea posible colaborar, dando lugar a otras más complejas con tan sólo ensamblar módulos, permitiendo responder de manera rápida a la presión del mercado e incluso ofrecer el software como un servicio de pago por uso.

Pero lo más importante es que, con los servicios Web, distintas aplicaciones pueden colaborar para ofrecer nuevos productos y servicios al mercado en tiempo real, incluso ofrecer los no estratégicos al mercado, de tal forma que no sólo cambiará la forma de desarrollar software empresarial sino los modelos de negocio sobre los que las empresas operan.

En resumen, los servicios Web:

- Se diseñan para permitir que módulos de una aplicación (objetos) se comuniquen con otros y, una vez conectados, ofrezcan sus servicios de información y transaccionales.
- Utilizan XML como formato común para compartir datos e información.
- Se basan en estándares de Internet (UDDI, WDSL y SOAP) para el registro y la comunicación.
- Usan Internet como plataforma central de comunicaciones.

- Los servicios Web significan una forma práctica de implantar SOA, lo que no quiere decir que haya que utilizar todos los formatos y protocolos propuestos como estándares aunque la recomendación es que soporte los principales. Los servicios Web son aplicaciones basadas en red desarrolladas para interactuar con otras aplicaciones usando tecnología y conexiones estándar de Internet para mejorar los procesos de negocio. No significan una ruptura total con las inversiones realizadas, sino una extensión de la funcionalidad de los servidores de aplicaciones y los componentes desarrollados.

Igual que HTML y HTTP consiguieron que las personas pudieran ponerse en contacto unas con otras y buscar cualquier tipo de información en la web, el uso de los servicios Web llevará a Internet a un nuevo nivel, un lugar donde la comunicación entre equipos sea común, así como el desarrollo de aplicaciones basadas en servicios, dando lugar a una colaboración más rica y significativa entre las personas y las empresas.

2.2 SOA, UNA ARQUITECTURA PARA LOS SERVICIOS BAJO DEMANDA

Con los servicios bajo demanda nace un nuevo modelo de arquitectura que extiende la funcionalidad de las ya existentes en la empresa, orientada al desarrollo de servicios y a la gestión de procesos. La finalidad de esta nueva arquitectura es crear una red de servicios (Service Network) con el objetivo de: hacer más sencilla la integración de sistemas, agilizar la adaptación del sistema a los nuevos requisitos funcionales, incrementar la reusabilidad de sus componentes y facilitar el comercio B2B.

Esta red de servicios se apoya en el concepto de la “ubicuidad de los interfaces” [ref. 14] que permite utilizar los mismos mecanismos de comunicación tanto para la integración de aplicaciones dentro de la empresa como para integrar éstas con los sistemas de sus colaboradores. A principios de los años noventa gracias a otra ubicuidad, la que ofrecían los navegadores, y utilizando estándares como HTTP y HTML, fue posible la creación de Internet como medio de comunicación de masas para compartir información entre personas y aplicaciones, con la

ubicuidad de interfaces nace una nueva forma de comunicación aplicación-con-aplicación, esta vez utilizando otro tipo de protocolos como son SOAP, WSDL y UDDI.

A diferencia de la ubicuidad del navegador que impulsó nuevos modelos de negocio como B2C, C2C, B2E, etc., donde al menos uno de los participantes era una persona, la ubicuidad de interfaces no necesita la participación humana y se aplica básicamente para: la integración aplicación-con-aplicación (A2A), la integración con las aplicaciones y sistemas de colaboradores de negocio y SODA (Service Oriented Development of Applications).

2.2.1 Integración aplicación-con-aplicación

La integración entre aplicaciones o EAI (Enterprise Application Integration) se refiere a una generación de soluciones de software que surgieron en los años noventa y que utilizaban tecnología de integración propietaria para: el diseño de flujos de mensajes, la transformación de datos y la creación de adaptadores entre aplicaciones.

A2A significa la evolución de EAI y utiliza para la integración: lenguajes estándares para el modelado de procesos como BPEL4WS, tecnología estándar para la transformación de datos como XSLT y adaptadores estándar como los servicios Web.

La mayoría de los fabricantes de soluciones EAI están actualizando sus plataformas para adoptar estos estándares al mismo tiempo que surgen nuevos fabricantes de nuevas soluciones diseñadas en base a estos estándares desde un principio.

2.2.2 SODA (Desarrollo de Aplicaciones Orientadas a los Servicios)

Utiliza los mismos conceptos que la programación orientada al desarrollo de componentes pero extendiéndola usando un modelo de informática distribuida y buscando la calidad del servicio; pone el énfasis en los mensajes asíncronos, tipos de datos XML y el uso de protocolos estándares en contra del desarrollo de

APIs (Application Programs Interface). Se apoya en una estructura SOA para el diseño y orquestación de las aplicaciones, estas nuevas aplicaciones se caracterizan por una débil integración de sus servicios en tiempo de ejecución. Esto se consigue al compilar sólo los servicios y dejar que un motor de orquestación los ensamble en tiempo de ejecución.

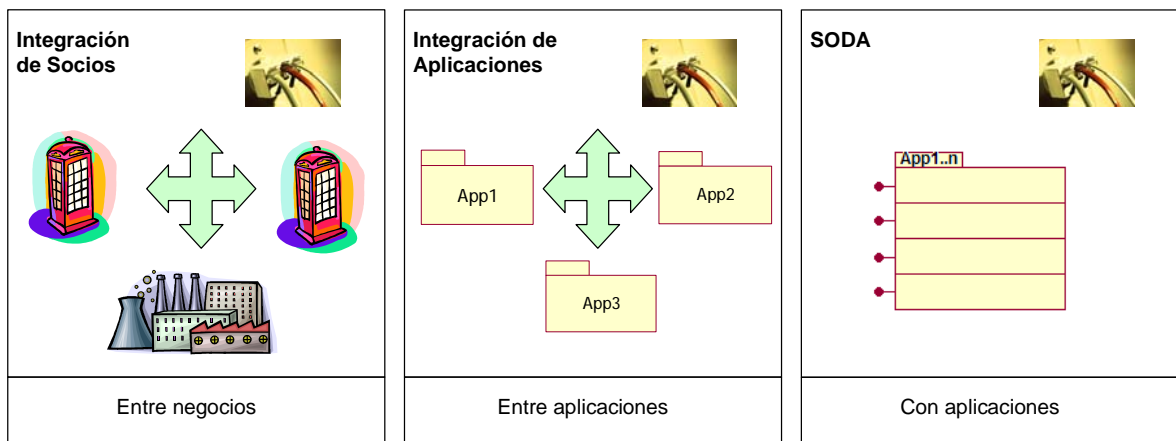


Figura 2. La informática como un servicio

2.2.3 La Red de Servicios

Una red de servicios es cualquier red de aplicaciones que es soportado por una arquitectura SOA y que está compuesta por sus participantes (colaboradores) y por un conjunto de servicios. A diferencia de los modelos de desarrollo tradicionales con aplicaciones cerradas, en una red de servicios es difícil saber dónde empieza y dónde acaba una aplicación, y a que aplicación y red representan la misma entidad.

Esta red se extiende más allá del firewall y se integra con las redes de servicios de otras empresas, esto es posible gracias a la promoción, publicación y soporte de protocolos abiertos usados para la comunicación por organizaciones como WS-i o GXA (Global XML Architecture).

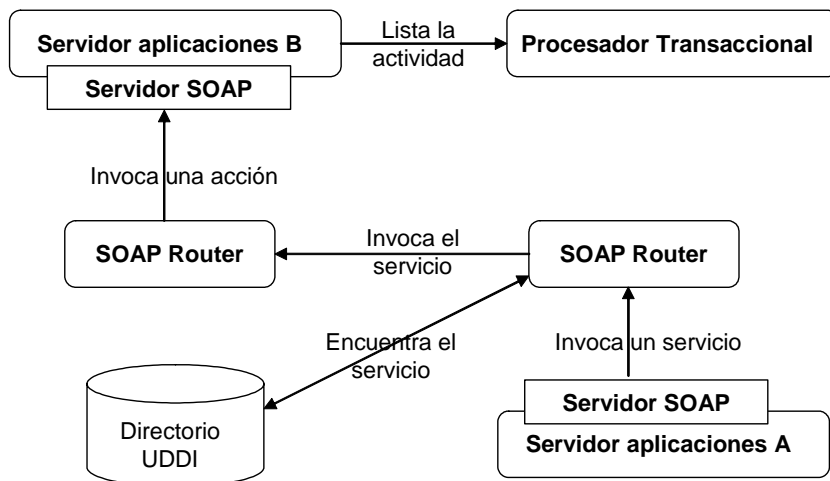


Figura 3. Estructura de una red de servicios

Un servidor SOAP que contiene los servicios web actúa como proveedor de servicios y gestiona los mensajes entre aplicaciones, al ser el participante crítico de esta red, los servidores de aplicaciones como WebSphere han extendido su funcionalidad para soportar esta nueva generación de servidores y junto a interfaces propietarias también ofrecen interfaces abiertos que facilitan la conectividad entre sistemas heterogéneos. Por tanto, los participantes de una red de servicios son: Los servidores de aplicaciones, los servidores y routers SOAP, un procesador de transacciones (Saga), los directorios de servicios (como UDDI) y los motores de orquestación.

¿Por qué desarrollar aplicaciones orientadas a los servicios?

- El mayor problema de las organizaciones actuales es la agilidad de negocio, que es la habilidad de la empresa para gestionar el cambio de las condiciones del entorno (mercado, tecnología, etc.) y utilizarlo en su beneficio. Las tecnologías de la información no pueden convertirse en un cuello de botella en la toma de decisiones de la empresa; los negocios deben mandar sobre la tecnología y no al revés.
- La clave para obtener la agilidad de negocio es una integración eficiente. El mayor cuello de botella tecnológico en la empresa es la integración, conseguir que sus diferentes sistemas “hablen” entre ellos de manera flexible siempre ha sido costoso. Lo que necesitan las empresas es una arquitectura que permita

un débil acoplamiento de sus componentes evitando la necesidad de que los desarrolladores controlen el sistema a ambos lados; en definitiva, permitir a sistemas, aplicaciones y fuentes de datos comunicarse sin preocuparse de los requerimientos técnicos del sistema destino.

- La integración eficiente necesita arquitecturas orientadas al desarrollo de servicios. Estas arquitecturas, basadas en estándares de débil acoplamiento, son un modelo de informática distribuida que concibe el software como un servicio accesible desde la red y que expone su funcionalidad en términos de negocio.
- La encapsulación y composición son claves para el desarrollo de una SOA. Hay que encapsular los componentes software, aplicaciones y sistemas dentro de interfaces de servicios web y luego crear (componer) una capa de abstracción entre el usuario y el software que transforme la funcionalidad de estos pequeños trozos de lógica en servicios de negocio.

Una SOA se apoya en cuatro pilares básicos:

- ***Es una arquitectura distribuida:*** los elementos funcionales de la aplicación se implantan en distintos sistemas y se ejecutan en redes locales remotas.
- ***Se compone de interfaces de débil integración,*** que requieren menos nivel de coordinación y permiten una reconfiguración más flexible.
- ***Las conexiones se basan en estándares y son independientes del fabricante,*** a diferencia de CORBA, DCOM o RMI.
- ***Los sistemas se diseñan desde una perspectiva orientada al proceso,*** los servicios creados representan una actividad dentro de un workflow o proceso de negocio. Un servicio informático bien diseñado deberá describir sus inputs de forma que cualquier otro software pueda saber lo que hace, cómo se invoca su funcionalidad y los resultados esperados.

Todo modelo SOA comprende tres actores principales:

- Un **service provider**, que es el nodo de la red que ofrece la interfaz de un servicio o conjunto de tareas asociadas a un determinado software que contienen lógica empresarial que va a enviarse. Puede representar todos los servicios que ofrece una entidad o la interfaz reusable de cualquier subsistema.
- Un **service requester**, que es el nodo de la red que descubre e invoca a otros servicios de software para componer una solución de negocio que ejecuta llamadas a los procedimientos remotos de un objeto distribuido que es el proveedor del servicio. En algunos casos, el proveedor reside en la intranet local y en otros en algún lugar remoto de Internet.
- Por último, el **service broker**, que es el nodo de la red que funciona como un repositorio a modo de páginas amarillas, de las interfaces de software publicadas por el service provider. Pueden ser entidades de negocio u operadores independientes.

Estos actores interactúan usando básicamente tres operaciones (**figura 4**): publish (publicar), find (encontrar) y bind (adherirse). Los proveedores del servicio lo publican en un registro, donde puede ser encontrado por un posible consumidor del servicio que se adhiere a él.

Este modelo de arquitectura explota la principal característica del software como servicio: que los componentes están emparejados de forma poco rígida, simplificando la adaptación del sistema a los nuevos requisitos.

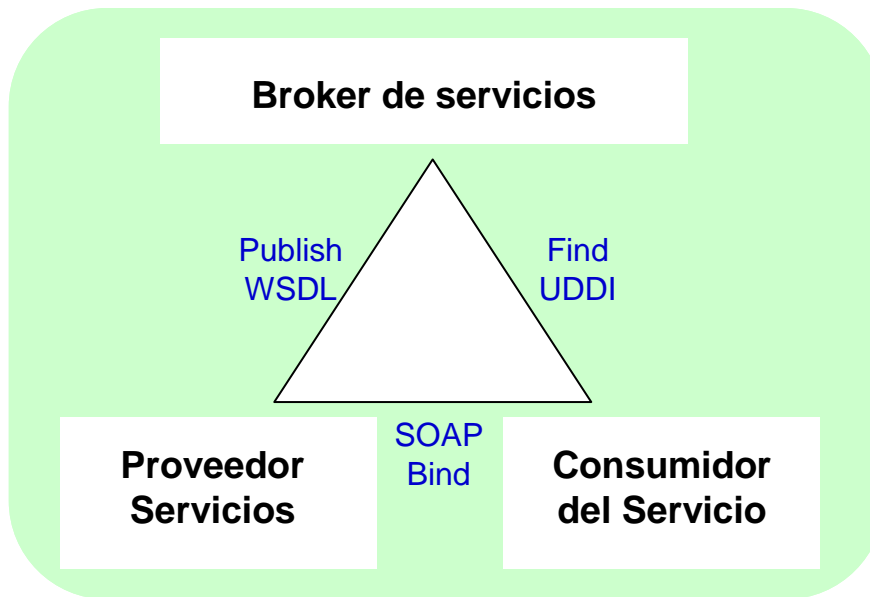


Figura 4. Modelo de arquitectura orientada a servicios (SOA). Fuente: IBM

2.3 EL USO DE LA TECNOLOGÍA ESTÁNDAR

Una SOA es independiente de la tecnología de red, protocolos de transporte, seguridad y otros pequeños detalles de los que se encarga su implementación específica. Los estándares abiertos de Internet trabajan juntos para componer sus servicios y están siendo progresivamente integrados en las plataformas software de los fabricantes. Un aspecto fundamental a la hora de implantar este tipo de arquitectura es utilizar estándares abiertos de Internet, una serie de tecnologías emergentes producto del trabajo conjunto de distintos actores de la industria informática.

Las tecnologías que forman parte de una arquitectura orientada a los servicios son:

- **SOAP.** El Simple Object Access Protocol es un protocolo de comunicaciones de base XML que se utiliza para el intercambio de información entre ordenadores, independientemente del sistema operativo, entornos de programación y modelo de objetos.
- **WSDL.** El Web Service Description Language proporciona una gramática para la descripción de servicios como un conjunto de puntos finales que intercambian mensajes.

- **UDDI.** Universal Description, Discovery and Integration define una especificación simple para los registros empresariales.
- **WSFL.** El Web Services Flow Language es una especificación para definir procesos de negocio.

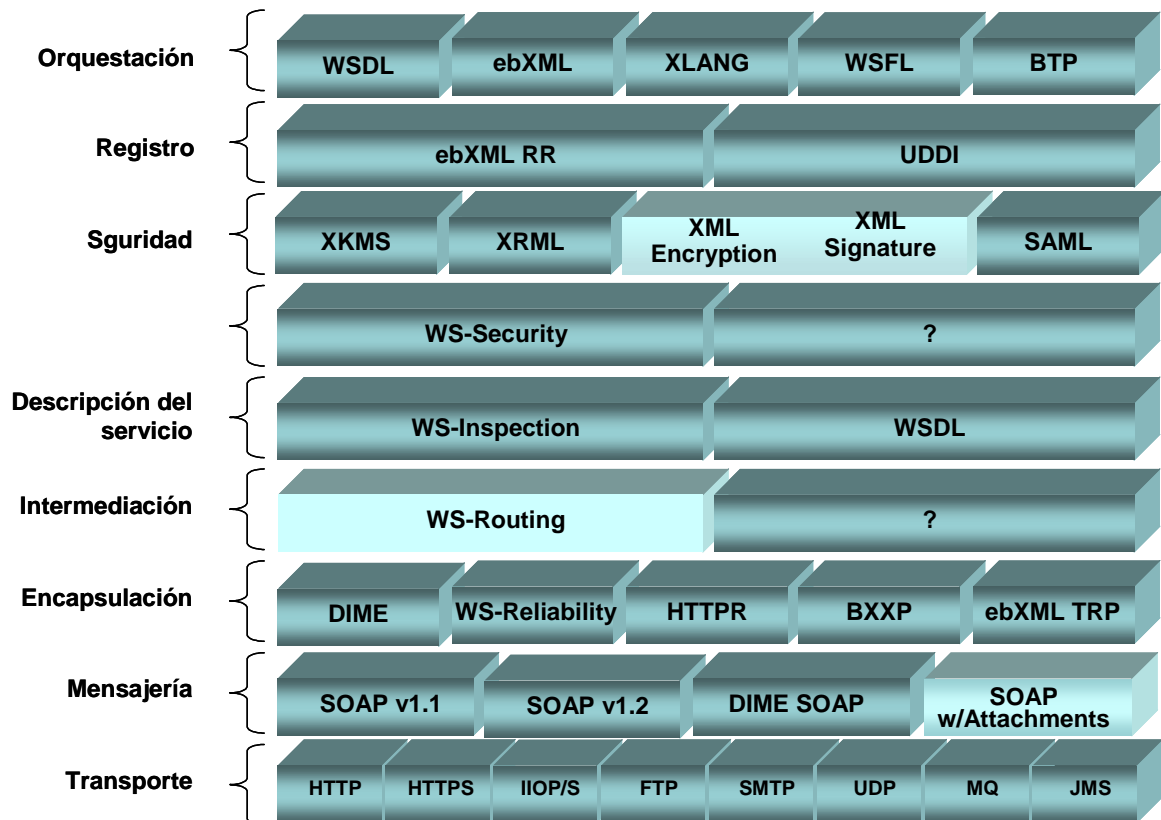


Figura 5. Relación de estándares abiertos de Internet

El desarrollador del sistema trabajará con tecnologías estandarizadas. La comunicación entre los distintos actores de esta arquitectura se realiza por medio de mensajes SOAP, cuyo formato sigue el protocolo XML y se transmiten por http. El proveedor del servicio emplea el estándar WSDL (también en formato XML) para describirlo

CAPITULO 3: GRID COMPUTING

Grid es un término inglés que podemos traducir como “cuadrícula”, “rejilla” o “parrilla”. El término traducido no nos permite intuir por sí solo de qué estamos hablando. En realidad es una evolución muy importante de la informática que, junto al desarrollo de otros conceptos desde el punto de vista de la gestión de la empresa (informática como utility), de la integración en la empresa (servicios web), y del modo de licenciar el software libre (software libre), están dando lugar a una nueva revolución: el e-business colaborativo.

Un grid, al igual que una parrilla o una pared, no es nada por sí mismo y lo es todo por la colaboración de los elementos que lo forman. Desde tiempos inmemoriales, el hombre ha colaborado para alcanzar objetivos que individualmente era imposible conseguir. Es importante resaltar el término colaborar diferenciándolo de compartir: este último hacer referencia a la distribución, al uso de una cosa por una colectividad; por ejemplo, la utilización de una red informática dentro de una empresa para usar las impresoras de manera conjunta. Dentro del ámbito de la computación, el término compartir podría designar el concepto de “informática cliente-servidor”.

La tecnología grid ha superado el test de Christensen para Tecnologías Revolucionarias al ser una tecnología:

- Barata
- Simple
- Que no requiere la sustitución de los sistemas existentes

En los últimos treinta años hemos ido viendo pasar distintos conceptos en la tecnología de redes, desde las primeras de la década de 1970, pasando por la década de 1980 con el descubrimiento de Internet y su posterior eclosión en los 90, hasta el día de hoy con el Grid, los servicios web, XML, P2P.

3.1 ¿QUÉ ES EL GRID COMPUTING?

Cada uno de los conceptos antes enunciados entroncaba una arquitectura asociada al mismo que iba haciendo desaparecer la anterior. La necesidad de una

mayor capacidad de computación para resolver los desafíos científicos y técnicos que nos encontramos en los problemas reales del día a día (estudio del genoma, necesidad de hallar vacunas contra el sida, búsqueda de inteligencia en el espacio exterior, simulaciones de riesgo en evoluciones de mercados financieros...) ha provocado que los problemas sean abordados desde la perspectiva de la utilización y colaboración entre distintos recursos heterogéneos y dispersos geográficamente. Se hace necesario utilizar y maximizar los recursos existentes empleando toda la capacidad productiva que pueden ofrecer.

Aunque la idea de aprovechar recursos de una parte de la red informática que no están utilizando es bastante antigua, aparece en un programa llamado *warm*, que se utilizó en las primeras redes Ethernet del laboratorio de PARC (Palo Alto Research Center), de Xerox, mayoritariamente se ha entendido que podemos centrar el origen de grid en el proyecto SETI@home, que sucedió a otro de la NASA denominado SETI, siglas inglesas de Search for Extra-Terrestrial Intelligence, en el que se buscaban señales de inteligencia extraterrestre por medio de las observaciones de potentes radiotelescopios. El proyecto SETI@home continuaba lo iniciado por la NASA; pero lo que hizo que se considerase el primer grid fue la utilización de un software que se podía descargar de Internet con el objetivo de utilizar ordenadores personales de los internautas que se bajaban el programa con la intención de aumentar la capacidad de procesamiento necesario para analizar las señales proporcionadas por el radiotelescopio de Arecibo (Puerto Rico).

Como resultado del proyecto, más de un millón y medio de personas han permitido que sus ordenadores personales colaboraran en el proyecto. Los resultados de estos colaboradores anónimos de más de 200 países han sido espectaculares: una media de 10 trillones de operaciones por segundo y el equivalente a más de 150.000 años de tiempo de computación.

Los lugares donde usualmente se suelen reconocer que se gestó el concepto de grid son el Laboratorio nacional de Argonne -en la Universidad de Chicago, pero perteneciente al Ministerio de la Energía de Estados Unidos- y la Universidad de California del Sur, en cuyos laboratorios se utilizaban conceptos de

procesamiento en paralelo y computación distribuida con el objetivo de aumentar la potencia de cálculo a partir de la colaboración entre distintos equipos.

Dentro de estos equipos de investigadores, destacan por la influencia que han ejercido en la evolución del pensamiento y estandarización del grid diversos nombres, pero en especial los de Ian Foster y Carl Kesselman, quienes en 1999 iniciaron el proyecto Globus con el objetivo de proporcionar un conjunto de herramientas para el establecimiento de un marco de trabajo y una estandarización que permitiese la evolución del concepto de grid. Uno de los frutos maduros dentro del proyecto Globus es la OGSA (Open Grid Services Architecture) [ref. 13], que se convertirá en el estándar de código abierto para el desarrollo de la nueva generación de futuros grid.

3.1.1 Definición de Grid

Precisamente es dentro de la documentación del proyecto Globus donde podemos encontrar la definición de grid:

“un grid es un tipo de sistema distribuido y de computación en paralelo que permite compartir, seleccionar y agregar recursos distribuidos a través de ‘múltiples’ dominios administrativos basados en los requerimientos de los usuarios respecto a la disponibilidad, capacidad, rendimiento, coste y nivel de servicio de dichos recursos” [ref. 1].

Hoy en día, de hecho la herramienta de código abierto Globus Toolkit se ha convertido en el estándar “de facto” como base de desarrollo de los distintos grid, siendo utilizada en más de 20 proyectos multimillonarios a lo largo y ancho del planeta.

Como ejemplo de proyectos grid podemos citar:

- **Novartis.** La empresa farmacéutica suiza ha creado un grid enlazando más de 2.500 ordenadores personales con el objetivo de aumentar la investigación en la lucha de medicamentos contra distintas enfermedades, como el Alzheimer.

La inversión de esta empresa farmacéutica ha sido de 400.000 dólares USA, pero espera conseguir un ahorro de 2.000.000 de dólares.

Para este proyecto esta compañía se ha basado en la utilización de la tecnología proporcionada por la empresa Platform.

- **Gateway**, empresa dedicada a la fabricación de ordenadores personales. Esta compañía planea unir los 8.000 PC, aproximadamente, que permanecen inactivos durante las noches en sus más de 200 tiendas con el objetivo de formar un Grid computacional que pondrá en alquiler para todas aquellas empresas de distintos sectores que necesiten de su uso. Se planea la utilización de un coste por hora y PC para el cálculo de la factura. Esto permitirá a las compañías traducir un coste fijo en variable, olvidarse de las posibles sobrecapacidades y reducir considerablemente el time-to-market.

Gateway ha utilizado la tecnología suministrada por united Devices, Inc., para la creación del Grid.

- El proyecto **SETI**, de búsqueda de inteligencia artificial.
- El **UK Nacional Grid** del Reino Unido.
- El **Netherlands Nacional Grid** de los Países Bajos.
- El proyecto **DTS**, de IBM (Distributed Terascale Facility).
- **DataGrid**, el proyecto europeo de grid de bases de datos.

En definitiva, con la implantación de un Grid dentro de una compañía u organización podremos aunar un basto grupo de pequeños recursos físicos e individuales para la consecución de uno enorme, virtual y colectivo disponible para todo el conjunto.

3.2 COMPONENTES DE UN SISTEMA GRID

3.2.1 Elementos que componen un Grid

Como se puede ver en la *Figura 6*, un sistema Grid básico de computación tendrá fundamentalmente tres tipos de elementos:

- Elemento/s de planificación y control de los trabajos.

- Elementos encargados de la realización de las tareas.
- Elementos encargados de la administración y gestión de los recursos del sistema.



Figura 6. Elementos básicos de un Grid

Estos elementos pueden estar en servidores u ordenadores de sobremesa, pueden ser dedicados al Grid o por el contrario utilizarse para otras tareas. La administración comprende todo un conjunto de servicios amplios y puede realizarse de forma remota o no. Pero tendremos que encontrar siempre componentes que realicen estos tipos de funcionalidades fundamentales.

La forma en la que implemente este conjunto de elementos variará profundamente según los objetivos a alcanzar en cada caso.

3.2.2 Proceso de ejecución de una tarea

El esquema básico de procesamiento de una tarea en un Grid ya en funcionamiento y con los usuarios autenticados sería el siguiente:

1. El PC del usuario identifica que está ocioso y que puede realizar algún trabajo.
2. El PC envía una petición al servidor de planificación y control de trabajos para que le envíe una tarea.

3. El servidor de planificación y control escoge el trabajo pendiente, parte los trabajos que tiene ahora mismo disponibles en pequeñas tareas y envía al PC del usuario una tarea.
4. La tarea es ejecutada en el ordenador del usuario.
5. Los datos de salida son enviados al servidor que se encarga de recolectar todos los datos que le ha enviado el resto de ordenadores para obtener los resultados globales.

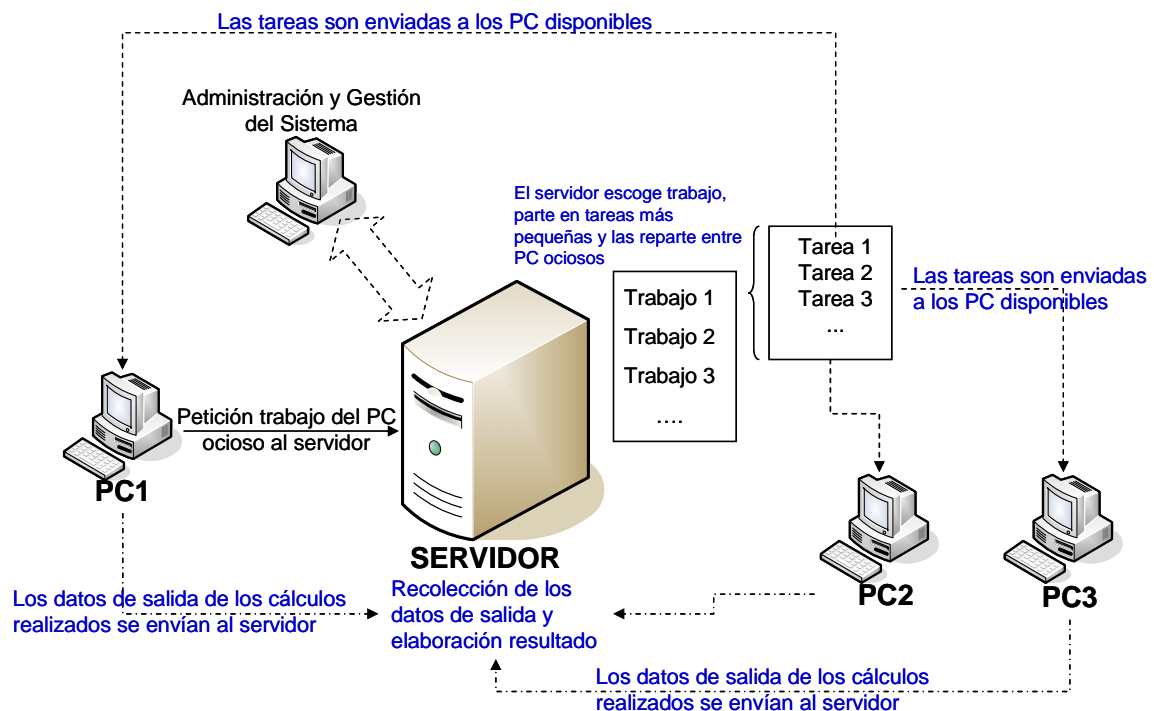


Figura 7. Esquema básico de la ejecución de un trabajo en grid

Lógicamente este esquema es muy básico y puede ser muy diferente. En la realidad es bastante más complejo, pero del estudio de este proceso podemos sacar varias conclusiones:

- Un sistema o una aplicación será tanto más “gridizable” cuanto más se pueda descomponer para trabajar en paralelo.
- En un sistema grid se cuenta con tecnología que no impide el funcionamiento normal del trabajo del usuario. No se ralentizan las tareas que ejecuta el mismo.

- En un grid hay multitud de elementos heterogéneos (diferentes arquitecturas con diferentes sistemas operativos, diferentes dominios de control, etc).

3.3 BENEFICIOS TECNOLÓGICOS DEL GRID

Entre los principales beneficios técnicos que se producen en el core tecnológico de una organización se pueden destacar los siguientes:

- Utilización plena de los recursos disponibles.
- Incremento de la capacidad de procesamiento.
- Virtualización de los datos.
- Aumento de la fiabilidad del sistema.
- Mejora de los procesos de actualizaciones en el sistema.
- Facilidad en la administración y gestión de los recursos tecnológicos.

3.4 TECNOLOGÍAS GRID E INFORMÁTICA DISTRIBUIDA

Grid es un concepto íntimamente relacionado con las tecnologías de código abierto, como Linux o peer-to-peer, y con la tan boga idea de la “virtualización” de los recursos, habiendo llegado a ser considerada como una variante del área amplia del conocido DCE (Distributed Computing Environment). Pero esto de “área amplia” no debe ser tomado a la ligera; podría representar precisamente la diferencia decisiva, que, junto a los enormes esfuerzos de estandarización y simplificación de la gestión que a su alrededor se están desarrollando, diera a Grid el éxito, evitándole seguir los pasos de su predecesora DCE, cuyo amplio despliegue nunca llegó a producirse.

Pudo ser que, simplemente, DCE se anticipara a su tiempo; ahora, la aceptación universal de Internet como plataforma de información distribuida, el creciente éxito de las redes Inter-empresariales (donde los sistemas de diversas organizaciones implicadas en un mismo negocio se cruzan creando comunidades virtuales, con la consiguiente puesta en la red de un número de aplicaciones en continuo aumento) y los servicios Web, han confluído creando un entorno de colaboración donde la existencia de una plataforma básica que lleve la compartición hasta los recursos de procesamiento, almacenamiento, ancho de banda y aplicaciones,

constituye, simplemente, la base idónea. Porque ¿qué mejor para unos servicios basados en componentes que un procesamiento segregable en “componentes”? ¿Y para unos servicios bajo demanda una capacidad bajo demanda? ¿Y para unos recursos de información virtuales y distribuidos que una plataforma informática virtual y distribuida?

CAPITULO 4: LA INFORMÁTICA GRID COMO UTILITY BAJO DEMANDA

La idea es similar a la de la red eléctrica, que proporciona un nivel estándar de servicio a cualquier dispositivo compatible. Se trata de convertir los recursos TIC – en un principio su capacidad de procesamiento, pero ya se habla también de almacenamiento, ancho de banda y aplicaciones– en una utility, y, en su concepción tradicional, se basa en la integración de todos los ordenadores de una empresa o cualquier otra red, sumando sus capacidades, y generando así una potencia informática superior a la de un superordenador. El concepto clave es la colaboración de los diversos sistemas en la creación de un “superordenador virtual” mediante un proceso de comunicación máquina a máquina, transparente al usuario. Los ciclos libres de CPU de todas las máquinas –pueden ser PC, servidores o grandes ordenadores– integradas en la Grid son aprovechadas para la realización de tareas de grandes dimensiones. Para ello basta con cargar en los dispositivos un software cliente específico y en uno de ellos, una máquina que puede ser jerárquicamente igual al resto pero encargada de la coordinación del conjunto, un software servidor.

Así, de la misma manera en que un usuario acciona un interruptor para conectarse a la red eléctrica –analogía frecuentemente utilizada para explicar el funcionamiento de una hipotética Grid global– y conseguir recursos bajo demanda, puede introducirse en una Grid informática accediendo a potencia de computación adicional según sus necesidades utilizando el recurso más barato posible; los ciclos de procesamiento ociosos de las máquinas integradas en el sistema.

4.1 COMUNIDADES VIRTUALES

Considerada tradicionalmente como un modelo de informática distribuida para la compartición de recursos a través de intranets o entre comunidades virtuales, el principal poder de Grid es quizá su potencial como middleware de integración de organizaciones y sitios dispersos en un todo virtual.

El documento “The Anatomy of the Grid”, elaborado por Globus Project (organización de estandarización y desarrollo Grid en la que participan un gran número de suministradores –IBM, Sun, HP y Microsoft, entre muchos otros-), define de manera precisa el segmento de aplicación de este tipo de plataformas, y subraya que esta concepción de la informática nace como una importante nueva área “que se distingue de la informática distribuida convencional por su enfoque en la colaboración de recursos a gran escala y aplicaciones innovadoras”. Como meta prioritaria, este grupo se marca el logro del desarrollo de la tecnología necesaria para permitir la colaboración de recursos, de forma coordinada, segura y flexible, entre grupos dinámicos de individuos, instituciones y recursos, a los que se refiere con la expresión “organizaciones virtuales”. De esta forma, frente al enfoque intracorporativo, adoptado por algunos suministradores, Globus Project se decanta claramente por una aproximación intercorporativa, que, paradójicamente es donde las Grids deben resolver sus más importante escollos, como la autenticación única de acceso a los recursos para cada usuario y el descubrimiento de los mismos.

Por “organización virtual” se entiende, una comunidad multi-institucional y dinámica compuesta por diversas entidades que en un momento dado pueden estar involucradas en un mismo proyecto. Ejemplos de una organización de este tipo serían los proveedores de servicios de aplicaciones, proveedores de servicios de almacenamiento y consultores vinculados por un fabricante de coches para crear un entorno de evaluación; o también un equipo de control de crisis junto con las bases de datos y sistemas de simulación que puedan utilizarse para definir una respuesta a una situación de emergencia. Una comunidad de esta clase puede concretarse en formas muy variadas, tanto en su propósito como en su tamaño, tiempo de duración y escritura, pero todas ellas coincidirán en la necesidad de una plataforma capaz de soportar relaciones de compartición altamente flexibles, desde cliente/servidor a peer-to-peer. Serían preciso, además, un control riguroso y preciso de los recursos compartidos, que pueden ser desde aplicaciones, ficheros y datos, hasta ordenadores, sensores y redes.

Anatomy of the Grid explica que su preocupación más que la compartición de intercambio de ficheros, es el acceso directo a ordenadores, software, datos y

otros recursos, como la requerida por diversas estrategias de solución de problemas y brokering de recursos de forma colaborativa que están emergiendo en la industria, ciencia e ingeniería. Debe ser altamente controlada, con una cuidadosa y clara definición por los proveedores y consumidores de recursos de lo que es lícitamente compartible, a quién le está permitiendo el acceso y bajo qué condiciones se produce la compartición.

Según el documento, las actuales tecnologías de informática distribuida no podrían resolver estas necesidades. De hecho, las tecnologías Internet de hoy se centran en el intercambio de información y comunicación entre ordenadores, pero no proporcionan enfoques integrados de un uso coordinado de los recursos entre los múltiples sitios. Los intercambios business to business resuelven la compartición de información (a menudo a través de servidores centralizados), como las tecnologías de empresas virtual, aunque en este caso, la compartición si puede finalmente extenderse a las aplicaciones y a los dispositivos físicos.

Paralelamente, las tecnologías DCE en la empresa, como CORBA y Enterprise Java, hacen posible compartir los recursos, pero dentro de una única organización; y, aunque los proveedores de servicios de aplicaciones (ASP) y de almacenamiento (SSP) permiten a las empresas externalizar sus recursos, lo hacen de una forma poco flexible, generalmente vinculando al cliente vía una VPN.

Control sin merma de la flexibilidad y escalabilidad es, pues, según Globus Project, el punto hacia donde grid debe dirigir sus esfuerzos con el fin de satisfacer las necesidades de este nuevo tipo de empresas cruzadas nacidas en el seno de Internet. Unas metas que, con tan solo una breve reflexión, traen a la mente las perseguidas por los estándares desarrollados alrededor de los servicios Web (UDDI, SOAP, XML, WSCI, etc).

4.2 EL FUTURO DE LA INFORMATICA GRID

Pero, a poco que se deje volar la imaginación, resulta razonable pensar que el futuro depara a Grid un alcance mucho mayor que el recogido en "The Anatomy of

the Grid”, con un desarrollo parejo a la virtualización de cualquier tecnología de la información, algo por lo que apuesta la inmensa mayoría de la industria. Grid está estrechamente relacionado con lo que Vernon Turner, vicepresidente de Grupo Global Enterprise Server Solutions de IDC, considera que será la próxima revolución de la tecnología de servidor [ref. 6]. Esto ocurrirá, según explica, cuando el concepto tradicional que vincula los servicios a un multiprocesador simétrico (SMP) concreto sea sustituido por una red de “nodos stateless”, donde los servicios podrán ser descubiertos a medida y cuando los recursos del servidor sean requeridos, dando lugar a un centro de datos mucho más flexible y escalable.

La infraestructura de CPD de próxima generación será un sistema dinámico que actuará como una Grid construida para el mundo comercial. “El servidor que conocemos está formado por tres elementos claramente distinguibles: procesamiento, entrada/salida (E/S) y almacenamiento. Con el nuevo paradigma se producirá una segregación de estas funciones”. La arquitectura bus (E/S) evolucionará de forma que las señales puedan ser enviadas sobre distancias mayores. Al mismo tiempo, los recursos de mantenimiento y servidor se virtualizarán para formar parte de una red de nodos sin una única ubicación física y carentes de inteligencia alguna excepto cuando realizan una llamada de servicios. “Si el usuario tiene un nuevo requerimiento de negocio, simplemente tomará el nodo y solicitará la aplicación de su elección”.

Entre las ventajas de este nuevo modelo destaca claramente su potencial para evitar los problemas de sobreprovisionamiento en el centro de datos. Ya en estos momentos, generalmente los recursos se encuentran sobredimensionados, ya que los administradores de redes no encuentran otra forma de capacitar sus infraestructuras para hacer frente a los picos de carga. Pero con la augurada llegada de la informática “pervasive”, predecir cual debe ser la capacidad para soportar tales picos se convertirá en una tarea no ya difícil, sino imposible.

Frente a esto, la virtualización de la tecnología puede construir la única solución haciendo posible la asignación de un servicio a cua

quiera de los miles de nodos sobre la red, y garantizando así, al mismo tiempo, una disponibilidad 24x7. Antes, no obstante, habrán de resolverse un conjunto de importantes cuestiones relacionadas con la gestión del entorno. “Será necesario gestionar, ya no el entorno, sino los servicios; y no podrá tratarse de una tarea manual. El sistema habrá de alcanzar un elevado grado de sofisticación tecnológica, con algoritmos que permitan manejar el control de acceso y la asignación de recursos”. Ejemplos de este tipo de herramientas de gestión incluyen algoritmos predictivos, intermediación (brokering) y limpieza de nodos (node cleansing), esta última será la responsable de depurar el nodo de “restos” de la antigua tarea antes de que un nuevo servicio pueda llamarlo, algo fundamental para evitar conflictos de servicio.

Pero sea cual sea la magnitud de los esfuerzos necesarios antes de alcanzar la meta, para cualquier defensor del desarrollo de la Sociedad de la Información merecerán la pena. Cuando se piensan en las posibilidades a las que Grid abre la puerta, la sensación es vertiginosa: aplicaciones corporativas (intranets), intercorporativas (extranets) e incluso servicios públicos prestados por una nueva generación de proveedores de servicios. Internet ha supuesto sin duda alguna una revolución social de dimensiones titánicas en cuanto a instrumento de comunicación universal. La era de la información parece contar con grandes reservas de creatividad y promete no dejar de sorprendernos. Lo que es una red global de comunicación podría estar a las puertas de convertirse en una auténtica plataforma de networking universal y “democrática” (peer-to-peer), accesible a cualquier ser humano que disponga de un terminal.

Esta será probablemente la única base sólida capaz de sacar al e-business y a la informática omnipresente (pervasive computing) de su estancamiento, convirtiendo en factible un objetivo cuyas bonanzas los agentes tecnológicos no dejan de proclamar y que, pese a ello, aparece siempre envuelta en una neblina de irrealidad que en estos momentos comienza a levantarse aunque el proceso se adviene prolongado. Ahora la cuestión es si estamos preparados para controlar el enorme potencial que una Grid de alcance mundial encerraría; nuestra Internet de siempre ya nos hacía seguir a todos –especialmente a los organismos

responsable de su control- una contrarreloj agotadora sobre terrenos desconocidos.

CAPITULO 5: TECNOLOGÍAS GRID Y SERVICIOS WEB: HACIA UNA PATAFORMA INFORMÁTICA GLOBAL

Hasta ahora paralelos, el mundo de las tecnologías grid y de los servicios Web (Web Services) han encontrado su punto de intersección en las mentes de muchos estrategas tecnológicos, y es precisamente en la confluencia de ambos donde nace lo que muchos gurús consideran nada menos que la transformación de Internet de plataforma de comunicación en plataforma informática global. “Hewlett-Packard considera que Grid es el paso siguiente de la Internet como plataforma de comunicación. La Red permitía el acceso a la información y su intercambio, pero no utilizaban los recursos informáticos de otros usuarios, mientras que Grid hace posible el uso de los sistemas de cálculos remotos para tareas intensas que requieren capacidades de supercomputación.

5.1 LAS DOS CARAS DE LA MONEDA

La relación entre ambos ámbitos es de doble sentido. Hasta ahora, el énfasis se ponía en el papel que los Servicios Web podrían representar como un medio de acceso a Grids a través de portales convirtiendo éstas en un segmento más donde desarrollar la nueva generación de servicios. Pero recientemente han comenzado a levantarse voces que cierran el círculo, proclamando una “Grid de Grids” como el entorno idóneo para el despliegue de Servicios Web en su más amplio sentido y para la informática omnipresente (pervasive computing). En un futuro informático global de tales características, los picos de carga serán cada vez más imprevisibles, por lo que una red flexible, capaz de aportar capacidad bajo demanda como si de una utility se tratara adquiere cada vez más la condición de necesidad. Resumiendo, podrían decirse que los servicios Web son a las aplicaciones lo que Grid es a la infraestructura subyacente.

Según Donald Ferguson, arquitecto jefe de IBM WebSphere, “una de las cosas que han quedado absolutamente claras en los últimos meses es la sinergia

existente entre informática Grid y Web Services. Estos aportarán a las Grid los aspectos de integración de aplicaciones, particularmente a medida que iniciativas como Global Grid evolucionen, pero Grid será el eje de una integración de amplio alcance. Actualmente está evolucionando para convertirse con el tiempo en el middleware y en el sistema operativo que albergará los servicios Web. En ese momento, la Grid tendrá menos que ver con el almacenamiento y los ciclos CPU disponibles, y más con las funcionalidades que ofrecerá en forma de servicios” [ref. 6].

Sin lugar a dudas, un importante paso adelante hacia esta Grid Global fuente de servicios bajo demanda ha sido la arquitectura OGSA de Globus Project, donde la comunidad Grid ha concretado un conjunto de protocolos de acceso a servicios como identidad y seguridad. OGSA utiliza protocolos típicos de Web Services, como XML y WSDL (Web Services Description Language) para expresar contenidos, SOAP (Simple Object Access Protocol) en el envío de mensajes, y UDDI (Universal Description, Discovery and Integration) como tecnología estándar de directorios. Facilitará, además, la creación de servicios Web basados en componentes que unan sitios ubicados en dominios administrativos y mensajes diferentes.

5.2 EL VALOR DE LO VIRTUAL

Entre las aplicaciones de Grid se incluyen la utilización de recursos informáticos bajo demanda, el análisis de datos a través de múltiples sitios, la ingeniería colaborativa a través de diversas compañías y la integración de cadenas de suministro, y todo ello basándose en la idea de la virtualización de la tecnología, otra de las tendencias que goza de unanimidad generalizada entre los agentes del mercado. Sun Microsystems, uno de los principales propulsores de los servicios Web a través de su estrategia Sun ONE, dio una gran iniciativa de virtualización integrando sus diversos esfuerzos en esta área bajo el nombre *N1* [ref.12], estrategia cuyo grueso descansa precisamente en la combinación de servicios Web e informática Grid.

Sentido similar tiene la iniciativa Planetary Computing de Hewlett-Packard y eLiza de IBM. En todas ellas, los diversos elementos de una red informática empiezan a ser vistos naturalmente destinados a trabajar como un todo. Desde este punto de vista, resulta cada vez más desfasado hablar del almacenamiento por sí mismo, de los servidores como elementos discretos, o de la red como algo independiente de los recursos informáticos a ella conectados. La unión de la totalidad de estos componentes, permite, según los defensores de la virtualización, maximizar el rendimiento del conjunto disparando sus niveles de flexibilidad y escalabilidad.

En muchos aspectos similares a la informática peer-to-peer, gracias a su confluencia con los servicios Web Grid está dando lugar al nacimiento de un nuevo paradigma de red global. El despliegue de IPv6, debido a su capacidad para aumentar exponencialmente el número de direcciones IP, viene a resolver el último problema lógico para la transformación de Internet en una plataforma de informática Universal peer-to-peer donde cada dispositivo disponga de una dirección y tenga la posibilidad de fundirse en un todo en el que la suma de las fuerzas sea siempre superior a la de cualquiera de los sistemas integrados. Ahora, los escollos para el logro de una Grid Global se restringen a desarrollos tecnológicos –como mecanismos de autenticación y seguridad más robustos y unificados, sistemas con mayores capacidades de autogestión, etc.- y, otro tipo de factores, incluida la inversión en nuevos negocios capaces de explotar el potencial de esta “Nueva Internet” o el compromiso de los operadores para ofrecer Grid como un servicio.

Sin duda alguna, Grid encaja con el modelo seguido en la construcción de Internet. Pero también existe quien considera que la visión de Grid como la metamorfosis de Internet es exagerada. Sería desde este punto de vista una aplicación específica para determinados fines, tan solo una forma más que la red, en su versatilidad, es capaz de adoptar, sin que por ello se vea afectada su naturaleza.

5.3 ARQUITECTURA DE SERVICIOS ABIERTOS GRID (OGSA)

OGSA (Open Grid Services Architecture) es una propuesta para estandarizar la siguiente generación de plataformas para el Grid basándose en el concepto de servicio web. Aunque todavía está en proceso de cambio, OGSA ya proporciona alguna funcionalidad básica para construir Grids computacionales: introspección, registro de eventos, gestión del ciclo de vida, asignación de nombres y creación de servicios.

OGSA se ha convertido en el estándar de facto para las infraestructuras grid actuales. De hecho, es la arquitectura adoptada por la nueva versión (3.0) de Globus Toolkit (GT): el entorno de desarrollo de grids más ampliamente utilizado en la comunidad de investigación de tecnologías grid. Los servicios de OGSA, denominados Servicios Grid, son un tipo especial de Servicios Web que permiten a las organizaciones la provisión de acceso externo implícito a sus recursos de computación (y no solamente a sus datos, como sucede actualmente con los servidores Web). Los Servicios Grid ocultan la forma en que las organizaciones implementan la funcionalidad de los servicios que ofrecen y los recursos que emplean para ello. Con este modelo, OGSA aboga por un escenario en que las aplicaciones grid ofrecen su funcionalidad mediante la composición o concatenación de accesos a Servicios Grid proporcionados por organizaciones diferentes y, posiblemente, dispersas. Por lo tanto, el problema de la selección de recursos inherente a la computación grid se desplaza, en cierto modo, hacia el problema de la selección de Servicios Grid.

Son ocho las categorías de alto nivel de servicios especificados por OGSA:

- **Servicios de Infraestructura:** permite la comunicación entre recursos compartidos removiendo barreras de comunicación, tales recursos como: cómputo, almacenamiento, aplicaciones, etc.
- **Servicios de Gestión de Recursos:** permite monitorear, reservar, desplegar y la configuración de recursos Grid basados en la calidad de requerimientos de servicio.

- **Servicio de Datos:** permite el movimiento de datos donde se necesite ya sea administrando replicas de copias, ejecución de consultas y actualizaciones, y transformación de datos dentro de un formato si es requerido.
- **Servicios de Contexto:** describe los requerimientos y uso de políticas por cada cliente que utiliza el grid, permitiendo el/la optimización de recursos basada en el servicio de requerimientos.
- **Servicios de Información:** es muy eficiente en el acceso a la información del grid y sus recursos, incluyendo el estatus y fiabilidad de un recurso particular.
- **Servicios de gestión propia:** soporta la atadura de los niveles de estado de servicio con la mayor automaticidad, para reducir los costos y complejidad de la gestión del sistema.
- **Servicios de Seguridad:** refuerza las políticas de seguridad dentro de una VO (Virtual Organization), promoviendo la compartición de recursos seguros y la apropiada autenticación y autorización de usuarios.
- **Servicios de gestión de ejecución:** permite el flujo de acciones de trabajo simple y mas complejo para ser ejecutadas incluyendo: colocación, aprovisionamiento y gestión del ciclo de vida de tareas.

Globus Project actualmente centra sus esfuerzos en añadir nuevas herramientas a su arquitectura OGSA con el propósito de lograr una mayor integración con tecnologías de servicios Web y J2EE (Java 2 Enterprise Edition), así como un más amplio soporte de Base de Datos.

CAPITULO 6: PANORAMA ACTUAL DEL SOFTWARE GRID

En este capítulo realizaremos brevemente un análisis comparativo de los principales productos de software que se ofrecen hoy en día para la creación de un grid.

6.1 Avaki Data Grid, Avaki Compute Grid, Avaki Comprehensive Grid

Plataformas	Windows NT/2000, Linux, Aix, Solaris, IRIX, Tru64
Tipos de Grid	Computación, datos
Web	www.avaki.com
Compañía	Avaki Corporation
Sede	Cambridge (Massachusetts, Estados Unidos)

Características básicas:

- Estos productos ofrecidos por Avaki Corporation constan fundamentalmente de tres componentes:
 - Bootstrap Host
 - Service Host
 - Command Clientes
- Estos productos están orientados a la utilización de recursos dedicados al grid. Esto es, a diferencia de otros productos en los que no hace falta una dedicación plena de los servidores al grid, en este caso la idea es aumentar el rendimiento y la utilización de información compartida de forma eficiente por la utilización de los recursos orientados a su utilización dentro del grid. Otra aproximación sería la utilización de los recursos existentes para maximizar su utilización, pero no aumentar la misma mediante la incorporación de recursos dedicados.

- Un punto a destacar es su fortaleza que ofrece al crear un grid de datos para compartir la información entre equipos separados geográficamente y con datos que están en plataformas heterogéneas.
- A la hora de compartir la información, el usuario autorizado se autentifica en el grid y puede consultar y copiar los datos a los que tiene acceso. Las políticas de acceso se pueden definir a un nivel sumamente granularizado, lo que les proporciona gran flexibilidad a la hora de permitir quine tiene acceso y a que recurso. Para el usuario, la utilización es totalmente transparente, ya que no tiene que conocer donde se dispone físicamente el fichero, sino que accede a un directorio virtual del grid en donde se almacena la información disponible para su perfil, independientemente de que ésta esté en distintos orígenes de datos.
- Avaki expone como caso práctico el de una de las primeras diez empresas farmacéuticas estadounidenses que lo ha utilizado para facilitar un espacio virtual común para compartir los datos entre distintos equipos de investigadores que estaban en laboratorios muy alejados entre sí. Con la solución de Avaki se creó una base de datos federada, a la que los usuarios, según su perfil, podían acceder a través de un directorio único. Los usuarios se tienen que registrar en el data grid creado para acceder a los dicheros de la misma forma que se accede a un fichero en el disco duro.
- Avaki expone que la compañía ahorró un 75% por la adopción de un grid que utilizase los servicios que corrían en Linux en vez de utilizar el sistema que se habían planteado originalmente –adquisición de equipos de archivado, sin incluir los costes de administración y mantenimiento-.

6.2 DCGrid

Plataformas	Windows NT/2000, XP
Tipos de Grid	Computación
Web	www.entropia.com
Compañía	Entropia, Inc.
Sede	San Diego (California, Estados Unidos)

Características básicas:

- Es un producto orientado a que corra bajo la plataforma Windows, tanto desde el punto de vista de los servidores como de los clientes.
- DCGrid cuenta con tres componentes:
 - DCGrid Scheduler
 - DCGrid Manager
 - DCGrid Clients
- El primero de ellos es el encargado de planificar los trabajos que se lanzan en el sistema grid, que son enviados desde el scheduler al cliente. El segundo es el encargado de administrar y gestionar los componentes del sistema grid. Por último el tercero es el software que se instala en los distintos equipos que se vayan a utilizar para ayudar a ejecutar las tareas encomendadas al grid. Solamente puede estar corriendo un job (trabajo) al mismo tiempo.
- Como en cualquier sistema grid, no entorpecen la labor del usuario, ya que se ejecutan cuando el equipo esté en un rendimiento mínimo.
- Se cuenta con una interfaz web para administrar y controlar el grid. Bajo este software puede correr cualquier Win32.
- Como elemento destacable del software de Entropía hay que señalar que el sistema cuenta con una encriptación tanto de ficheros como de aplicaciones que permite una doble protección tanto del cliente a la

aplicación como de la aplicación al cliente. Este sistema reserva un espacio para que corra la aplicación y además la aplicación no puede acceder a ningún fichero que no esté fuera de ese espacio reservado.

- Este software proporciona un nivel de seguridad a tres niveles:
 - Seguridad a nivel de aplicación, manteniendo la aplicación totalmente independiente del entorno operativo del PC. Esto permitiría la completa seguridad del PC mientras los procesos de la aplicación no puede acceder a ningún fichero que esté fuera de ese espacio reservado.
 - Seguridad a nivel de sistema: las aplicaciones tienen que ser protegidas del posible mal uso que se puedan hacer de los recursos desde la red local.
 - Seguridad a nivel de tareas: los datos de la tarea tienen que ser encriptados así como sus resultados para salvaguardar su integridad.

6.3 LSF, ActiveCluster, MultiCluster, PlatformGlobus

Plataformas	AIX, Solaris, IRIX, Tru64, HP-UX, Linux, Windows NT/2000
Tipos de Grid	Computación
Web	www.platform.com
Compañía	Platform Computing
Sede	Toronto (Ontario, Canadá)

Características básicas:

- En la plataforma Grid de Platform hay una parte de recursos dedicados exclusivamente al cluster y otra parte que es la utilización de los PC para ayudar a la ejecución de tareas. Es pues una plataforma orientada tanto a recursos dedicados exclusivamente al grid como en el caso de los productos de Avaki, como la utilización de los recursos existentes en busca

de ciclos que no se estén utilizando para aumentar la capacidad de computación.

- Su plataforma de sistema grid se consigue con la conjunción de tres productos: LSF, ActiveCluster y MultiCluster.
- El LSF cuenta con tres principales componentes:
 - Master Host
 - Server Host
 - Client Host
- El LSF como tal, por sí solo, no es un sistema de computación grid propiamente dicho, sino que se utiliza para la creación de un cluster. Por cada cluster es necesario que haya un Master Host, que es responsable de la planificación de los distintos trabajos dentro del cluster.
- Los Server Host son máquinas en las que se ejecutan los trabajos enviados por el planificador y corren los denominados software LSF.
- Aunque el LSF se utiliza para clusters de servidores básicamente, junto al producto ActiveCluster se pueden utilizar los recursos de la empresa como ordenadores personales. ActiveCluster se instalará en las estaciones de trabajo de la empresa con sistema operativo Windows y permitirá la utilización de las mismas para la ejecución de trabajos. El encargado de enviar los trabajos a los clientes será un Server Host al que se le haya instalado el ActiveCluster.
- Por último, con el software MultiCluster se podrán interconectar y mandar y recibir trabajos entre distintos cluster LSF.
- Así, si por ejemplo un grupo de trabajo está sobrecargado puede enviar a otro grupo de trabajo que no tenga una carga excesiva una parte de las tareas o recibirlas cuando tenga recursos ociosos y otro departamento esté al límite de su capacidad de cálculo.

- Un punto a destacar también es que esta compañía de software es de las primeras que tiene un producto en el que se incorpora el Globus Toolkit (herramienta facilitada por el proyecto Globus para la construcción de un sistema grid).
- El software de Platform está instalado en clientes como Astra-Zeneca, JPMorgan Chase, Ford, Toshiba.

6.4 Grid MP (MetaProcessor)

Plataformas	Rd Hat Linux, Windows 98/Me/NT/2000/XP
Tipos de Grid	Computación
Web	www.ud.com
Compañía	United Devices
Sede	Austin (Texas, Estados Unidos)

Características básicas:

- El grid de United Devices se basa en la utilización de los recursos existentes, no en la dedicación exclusiva de recursos para construir un grid que aumente la capacidad de la empresa.
- La base del producto de United Devices denominado MetaProcessor Platform está compuesta por los siguientes elementos:
 - MP Server
 - MP Agents
- El MP Server corre bajo Red Hat Linux y necesita de una base de datos DB2 en donde se almacena información sobre estadísticas, usuarios en el sistema y demás parámetros del grid. Además cuenta con una consola web en la que se puede administrar el sistema de forma remota.
- Los ordenadores clientes necesitan tener instalado el MP Agents para poder ejecutar los trabajos que son planificados, enviados y recogidos por

el MP Server. Estos trabajos se ejecutan sin interrumpir el normal funcionamiento que está realizando el usuario en su PC, siendo para él transparente la clonación de su ordenador para el grid. Únicamente se puede ejecutar un trabajo a la vez en la máquina cliente.

6.5 LiveCluster

Plataformas	Solaris, Linux, Windows
Tipos de Grid	Computación
Web	www.datasynapse.com
Compañía	DataSynapse
Sede	Nueva York (Estados Unidos)

Características básicas:

- Este producto se compone básicamente de tres niveles de software:
 - Application Enablers o Drivers
 - GridServer
 - Engines
- En el primer nivel encontramos los Drivers que proveen la interfaz entre las aplicaciones y el LiveCluster. Los Drivers son los encargados de enviar trabajos al servidor y recoger los resultados.
- En el nivel inferior aparecen los Engines, software que puede ir en servidores o estaciones de trabajo y que son los encargados de realizar los trabajos encomendados por el GridServer. En estos elementos también se puede monitorizar el estado del sistema y el mantenimiento remoto del mismo a través de una interfaz web.
- El GridServer es el encargado de planificar y administrar el sistema. En el GridServer hay dos capas: por un lado, la capa que se encarga de la planificación y poner en cola las subtareas, denominada Broker, y por otro, los Directors, que actúan como enrutadores del software.

- El sistema funciona utilizando los momentos en los que los ordenadores cliente no están siendo utilizados a plena capacidad. En ese momento, el Engine pide la ejecución de un trabajo, que le es enviado desde el servidor. Este software solamente se utiliza en el caso de que el ordenador esté disponible para ejecutar un trabajo, ya que, si el usuario necesita utilizarlo, entonces se interrumpe el trabajo. El servidor envía el trabajo interrumpido a otro Engine para que lo termine. Este sistema se diferencia de otros en los que es el servidor el que manda la petición al cliente para que lo ejecute. La ventaja de que la petición parta del cliente es que, al aumentar el número de clientes ociosos que puedan realizar tareas, el servidor puede replanificar trabajos o enviar nuevos a los clientes, es decir, a más Engines disponibles, más carga de trabajo. Éste es un enfoque dinámico de asignación de trabajos.
- El LiveCluster de DataSynapse provee un conjunto de API que están disponibles en Java, C++, C# y XML. Este software también permite la integración de servicios web a través de las SOAP API.

6.6 InnerGrid

Plataformas	AIX, Solaris, Linux, Windows NT/2000
Tipos de Grid	Computación
Web	www.gridsystems.com
Compañía	GridSystems
Sede	Palma de Mallorca (España)

Características básicas:

- GridSystem es la empresa española de alcance europeo a la vanguardia en tecnología grid.
- En el sistema proporcionado por GridSystems se encuentran tres niveles:
 - Servidor
 - Ordenadores Maestros

- Ordenadores Esclavos

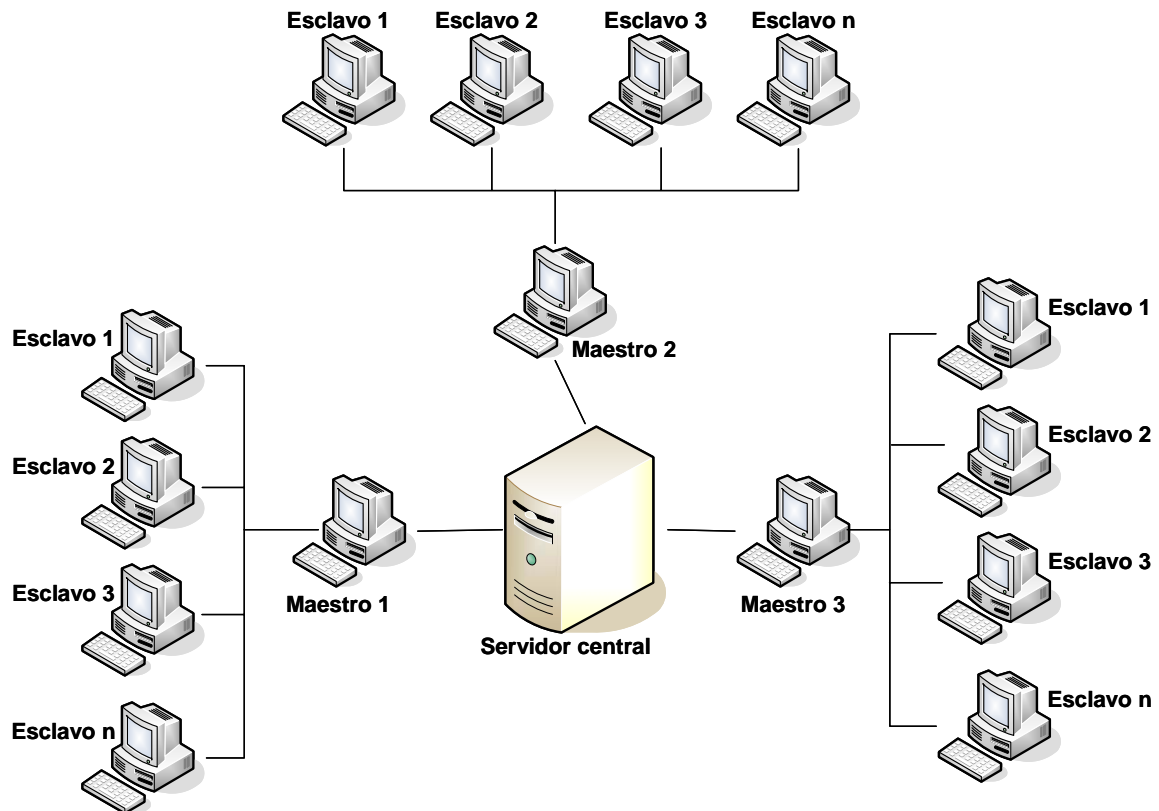


Figura 8. Arquitectura básica Grid de tres niveles de InnerGrid. Fuente GridSystems

- El servidor central será el encargado de distribuir y planificar las tareas.
- Por su parte, los ordenadores Maestros se encargarán de lanzar distintos trabajos que serán fraccionados y partidos para que se ejecuten por los ordenadores Esclavos. El trabajo del usuario no se verá ralentizado, ya que los cálculos ejecutados en su máquina se realizarán únicamente cuando ésta lo permita, porque tenga suficiente capacidad para ello. Si una máquina no está disponible, automáticamente se reasignará su trabajo a otra. Por último, el sistema se encargará de recoger todos los datos de salida de los diferentes Esclavos y generará el resultado final.
- La transmisión de los datos se realizará a través de protocolos seguros, como HTTPS y SSL, con lo que se garantiza la integridad de la misma a través del cifrado.
- La administración del sistema puede realizarse remotamente a través de una interfaz web que permite la gestión de todos los recursos del sistema,

como la asignación de prioridades en los trabajos, la creación de grupos, usuarios, roles, etc.

- En otro esquema de funcionamiento (ver figura 10), los ordenadores Esclavos pueden intercambiar resultados intermedios entre sí. Para ello es necesaria la instalación del código ejecutable en cada uno de los Esclavos.
- InnerGrid tiene conectores con aplicaciones de uso común, como la hoja de cálculo Excel o programas muy utilizados, como Maya para la renderización.

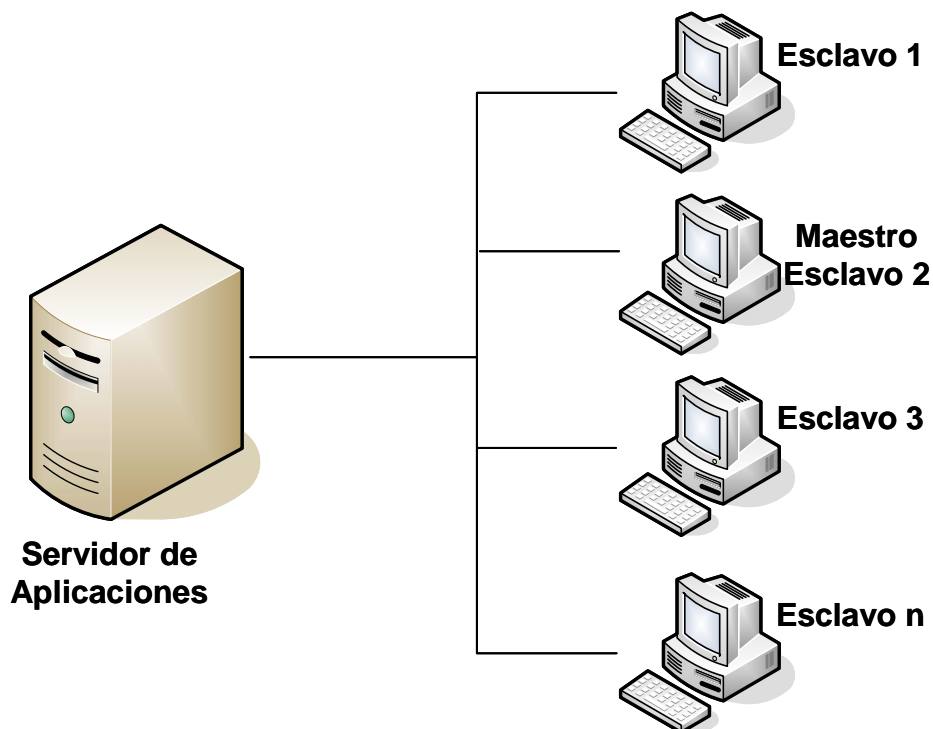


Figura 9. Intercambio de resultados intermedios en servidores esclavos con InnerGrid

6.7 Globus Toolkit

El proyecto Globus, como se comentaba anteriormente, proporciona una arquitectura abierta para la construcción de un grid.

Este proyecto, cuyos principales exponentes son Ian Foster y Carl Kesselman, suministran una herramienta, el Globus Toolkit, que contiene una serie de componentes de código abierto para el desarrollo de un grid.

Los componentes que facilita el Globus Toolkit están enfocados en tres vías principales [ref. 2]:

- Gestión de los recursos del sistema
- Información de la marcha y estado del sistema
- Manejo y transmisión de los datos en el sistema

Todo ello está envuelto en una capa de seguridad para el desarrollo de un grid seguro.

Para dar servicio a los tres puntos anteriores, Globus Toolkit proporciona un conjunto de componentes, que aparecen en la figura 11.

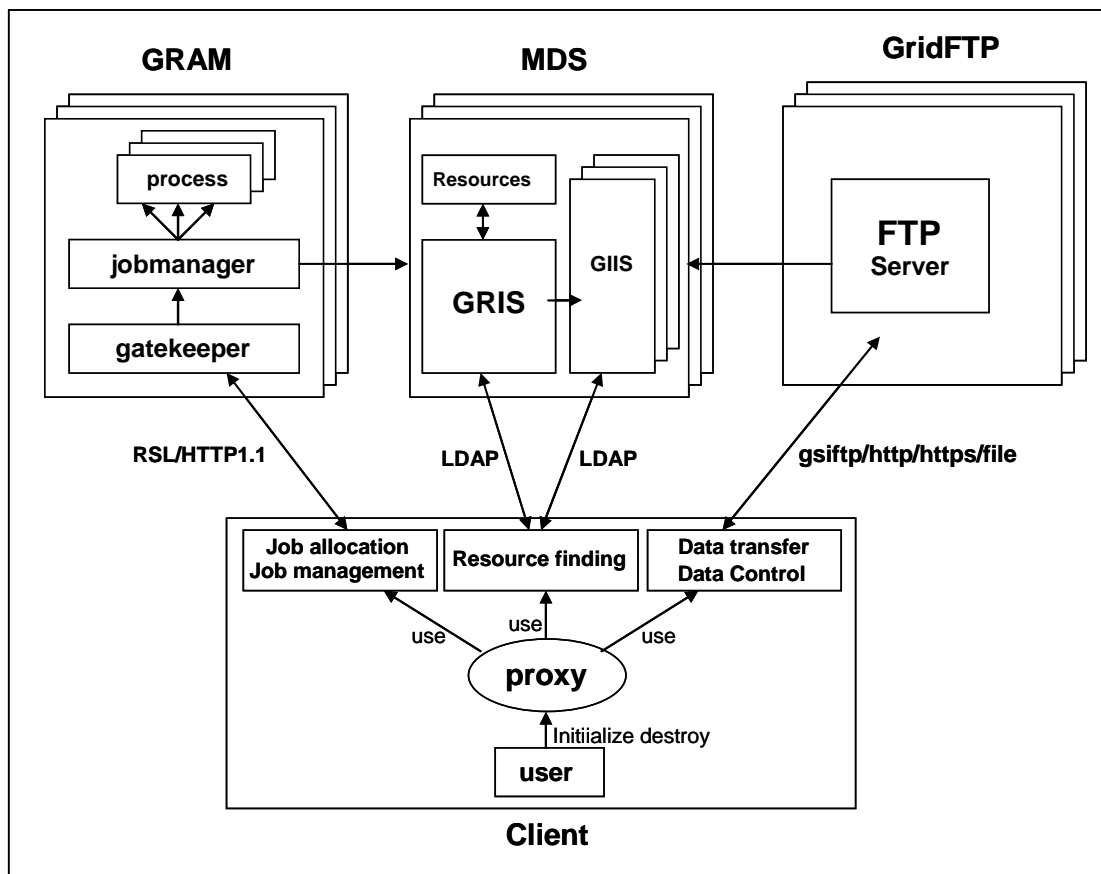


Figura 10. Fuente: Introduction to Grid Computing with Globus. IBM Redbooks

La gestión de los recursos del sistema viene soportada por:

- GRAM (Grid Resource Allocation Manager)
- GASS (Global Access to Secondary Storage)

El sistema de información MDS (Monitoring and Discovery Service) tiene dos componentes básicos:

- GRIS (Grid Resource Information Service)
- GISS (Grid Index Information Service)

La transmisión de los datos se encarga a GridFTP.

El envoltorio de seguridad lo proporciona GSI (Grid Security Infraestructure).

Por último, Globus Toolkit proporciona dos componentes para la personalización y construcción de propias aplicaciones para el grid:

- Developer's Toolkit
- APIs

El Developer's Toolkit, también conocido como Commodity Grid, permite el desarrollo en los siguiente lenguajes y tecnologías:

- Java
- Python
- Perl
- Matlab
- CORBA
- Servicios web

En resumen, el Globus Toolkit puede suponer una alternativa al software comercial en la construcción de un sistema grid.

CONCLUSIONES

1. Aunque los términos SOA y servicios Web se utilizan indistintamente para referirse a la informática como un servicio, verificamos que los servicios Web son una de las formas, quizás la más viable, de alcanzar los beneficios de una arquitectura orientada al desarrollo de servicios. Cualquier plataforma para desarrollar software como un servicio debe ofrecer de forma nativa soporte para los estándares principales asociados a los servicios Web, como SOAP y WDSL, gestión transaccional, seguridad y orquestación.
2. Estamos a un paso de que Internet no sólo distribuya páginas, sino aplicaciones y los servicios web serán los encargados de intercambiar datos entre ellas. Con los servicios web se podrán automatizar muchas de las actividades repetitivas de la empresa como la tramitación de un pedido de compra, con el consiguiente ahorro de tiempo que esto supone.
3. No existe una herramienta de software óptimo para el desarrollo de un Grid. Para cada necesidad se debe adoptar una solución personalizada. Es por ello que no se puede generalizar apuntando que un software es mejor que otro, sino que para cada empresa y en cada caso en concreto se deberá estudiar cuál es la herramienta de software más apropiado para una solución particular.
4. La estructura que se ha seguido en el análisis comparativo de los productos de software para desarrollo Grid es el de una ficha por cada herramienta de software evaluado, de las que se facilita un breve resumen teniendo en cuenta diferentes características.
5. El proyecto Globus ofrece código abierto para la construcción de un grid y un conjunto de herramientas para su administración.
6. La industria necesita entrar en la próxima etapa de Internet, consistente en su transformación en una plataforma informática. El principal valor de Grid reside en la capacidad de virtualización de la infraestructura que encierra.

BIBLIOGRAFIA Y REFERENCIAS WEB

1. The Anatomy of the Grid. Enabling Scalable Virtual Organizations.
Ian Foster, Carl Kesselman, Steven Tuecke
Globus.org, <http://www.globus.org/alliance/publications/papers/anatomy.pdf>
Fecha de Consulta: Feb 2005
2. The Globus Toolkit 4 Programmer's Tutorial,
Borja Sotomayor,
Copyright © 2004, 2005. University of Chicago - Department of Computer
Science. <http://gdp.globus.org/gt4-tutorial/multiplehtml/>
Fecha de Consulta: Marzo 2005
3. Web Services Resource Framework: Impact on OGSA and the Grid Computing
Roadmap
By Dr. David F. Snellong, Fujitsu Laboratories of Europe.
Global Grid Forum. Spring 2004. Vol 2, Issue 1
http://www.ggf.org/documents/Grid_Connections_Spring_2004.pdf
4. Tecnologías Grid: La Metamorfosis de Internet
Revista: Comunicaciones World, Nro. Setiembre 2002. Marta Cabanillas.
<http://www.idg.es/hemeroteca/buscar.asp?cat=6&subcat=1&busqueda=SI&clave=0&AbsPage=1&CBusqueda=&TxtFind=Metamorfosis&Buscar=esta&B1=Buscar>
5. The Grid's place in the service-oriented architecture
Revista Computerworld, Nro. Marzo 2005. Ian Foster.
<http://www.computerworld.com/databasetopics/data/story/0,10801,97919,00.html>
6. El futuro de Internet: Compartir y agregar
Revista Byte, Nro Abril 2003. Tema de Portada. <http://www.byte.com>
7. La empresa sin límites: Bases para el e-business de próxima generación
Revista Network world, Nro Marzo 2004. Informe. <http://www.networkworld.com/>
8. Utility Computing. La virtualización llega al centro de datos.
Informe Comunicaciones World. Nro. Octubre 2003. Alhambra Eidos
<http://www.alhambra-eidos.com>
9. Redbooks paper
Fundamentals of Grid Computing
Copyright IBM Corp. Nov. 2002. Viktors Berstis.
<http://www.redbooks.ibm.com/redpapers/pdfs/redp3613.pdf>
10. The Web Service Discovery Architecture
Wolfgang Hoschek
Novembre 2002. Proceedings of the 2002 ACM/IEEE conference on
Supercomputing

<http://portal.acm.org/citation.cfm?id=762780&coll=ACM&dl=ACM&CFID=45839169&FTOKEN=74716227>

11. A Service-Oriented Grid Computing Platform and Its Framework
Wei Jie, Tiany Zang, Terence Hung
Institute of High Performance Computing, 1 Science Park Road, Singapore
http://www.ngp.org.sg/grid2004/papers/Jie%20Wei_%20zang%20tian%20yi_0078.pdf
12. Sun N1 Grid Services
<http://sun.com/service/n1gridservices>
13. Introduction to Grid Computing with Globus
Copyright IBM Corp. Dic. 2002. International Technical Support Organization
<http://lbn.com/redbooks>
14. E-Business Colaborativo
Ignacio García Valcárcel, Eduardo Munilla Calvo
Fundación Confemetal Copyright IBM Corp. Edición 2003
<http://www.fundacionconfemetal.com>
15. Open Grid Services Architecture: Leveraging Web Services for Grid Computing
Syslab. Condensed Research Note. January 1, 2004
<http://www.infosys.com/Technology/CRN-0401-01.pdf>