

**PRUEBAS DE CONECTIVIDAD Y PROCESAMIENTO DISTRIBUIDO CON
CLÚSTERS BEOWULF UNIVERSITARIOS USANDO LA RED NACIONAL
DE TECNOLOGIA AVANZADA (RENATA)**

Alvaro Enrique Ospina Sanjuan
ROBERTO CARVAJAL SALAMANCA

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
UNIVERSIDAD OBERTA DE CATALUÑA
MAESTRIA EN SOFTWARE LIBRE
2008

**PRUEBAS DE CONECTIVIDAD Y PROCESAMIENTO DISTRIBUIDO CON
CLÚSTERS BEOWULF UNIVERSITARIOS USANDO LA RED NACIONAL
DE TECNOLOGIA AVANZADA (RENATA)**

ALVARO ENRIQUE OSPINA SANJUAN
ROBERTO CARVAJAL SALAMANCA

Trabajo de grado presentado para optar el título de:
Magister en Software Libre

Director
Dr. Eduardo Carrillo Zambrano

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
UNIVERSIDAD OBERTA DE CATALUÑA
MAESTRIA EN SOFTWARE LIBRE
2008

Nota de aceptación:

Presidente del Jurado

Jurado

Jurado

Bucaramanga, 07 de Julio de 2008

CONTENIDO

	pág.
INTRODUCCION	10
1. MARCO TEORICO Y ESTADO DEL ARTE	12
1.1 SMP	12
1.2 MPP	12
1.3 SPP	13
1.4 SISTEMAS DISTRIBUIDOS	14
1.5 INTRANET COMPUTING	15
1.6 INTERNET COMPUTING	15
1.7 CLUSTER COMPUTING	15
1.7.1 Definición de Clúster	15
1.7.2 Clúster Beowulf	16
1.7.3 Razones para trabajar con clústeres	20
2. GRID COMPUTING	21
2.1 OBJETIVOS DEL GRID	21
2.2 TOPOLOGIAS GRID	22
2.2.1 Intragrid (InnerGrid)	22
2.2.2 Extragrid (OuterGrid)	23
2.2.3 Intergrid	23
2.3 REQUERIMIENTOS DE UN GRID	24
2.4 ROLES DE USUARIO EN UN GRID	24
2.4.1 Rol de Donante	25
2.4.2 Rol de Administrador	26

2.5 MIDDLEWARE	30
2.5.1 Globus Toolkit	31
3. PRODUCTO FINAL	35
3.1 CONSTRUCCION DE LAS SALA CLUSTER	35
3.1.1 Instalación y configuración del SFE	37
3.1.2 Instalación y configuración del TNP	38
3.1.3 Instalación del resto de terminales nodo de la sala-clúster	39
3.2 INTERCONEXION DE DOS SALA-CLUSTER PARA CREAR UNA INTRAGRID	40
3.2.1 Conexión y comunicación de los dos servidores en la red	40
3.2.2 Configuración y prueba de componentes de Globus	43
3.3 PRUEBAS DE CONECTIVIDAD DEL GRID	45
4. RECOMENDACIONES PARA TRABAJOS FUTUROS	46
5. CONCLUSIONES	47
BILIOGRAFIA	50

LISTA DE TABLAS

	pág.
Tabla 1. Tiempos estimados en la instalación del SFE	38
Tabla 2. Tiempos estimados en la instalación y configuración del TNP	39
Tabla 3. Tiempos estimados de instalación del resto de Terminales nodos	39

LISTA DE FIGURAS

	pág.
Figura 1. Arquitectura de clúster	18
Figura 2. The grid virtualizes heterogeneous geographically disperse Resources.....	23
Figura 3. Intragrid, Extragrid and InterGrid.....	25
Figura 4. Servicios Globus toolkit versión 4 (GT4).....	32
Figura 5. GT4 architecture schematic, showing many (but not all) components.....	35
Figura 6. Capabilities of a GT4 container.....	36
Figura 7. Arquitectura de la Sala-clúster	38
Figura 8. Sala clúster ubicada en el quinto piso del Ed. de Biblioteca de la UNAB.....	38
Figura 9. IntraGrid UNAB.....	46
Figura 10. Relación entre componentes de Java y los servicios que ofrece Globus.....	48

LISTA DE ANEXOS

pág.

ANEXO A. Manual de Instalación de una sala-clúster con Linux Rocks.....

ANEXO B. Manual de instalación y configuración de Globus T.K 4.0.4 para el montaje de una Intragrid

RESUMEN

Palabras claves: Clúster, Grid, Procesamiento distribuido, GNU/Linux – distribuciones para clúster, Computación de alto rendimiento en educación.

Las universidades cuentan con gran infraestructura tecnológica, especialmente de computadores personales de escritorio que están a disposición de estudiantes, docentes y directivos para uso cotidianos en sus labores. Estas salas son constantemente utilizadas en el día y parte de la noche, pero permanecen ociosas el resto del tiempo. Teniendo en cuenta lo anterior y que además existen proyectos en las universidades que requieren alto poder de cómputo, se propone una solución de bajo costo que aprovecha los recursos no solo de una universidad, sino los recursos combinados de dos o más instituciones a través de la utilización de los clústeres de computadores y su interconexión a través de redes dedicadas de alta velocidad para conformar un Grid de computadores. Así mismo, este tipo de solución, relativamente nueva en nuestro medio, abre paso a un nuevo campo de investigación y desarrollo, especialmente en el área del software libre, pilar fundamental para este tipo de soluciones.

Los computadores que conforman un clúster pueden ser utilizados de manera dedicada (solo para usos del clúster) o de manera compartida (en algún momento el computador se usa para el clúster y en otros momentos se usa para trabajos propios de los usuarios normales, configuración que suele denominarse sala-clúster).

Una solución de sala-clúster¹ brinda beneficios a las organizaciones al ofrecer más poder de cómputo de manera económica, toda vez que se utilizan equipos ya existentes en la organización, aprovechando el tiempo ocioso de los mismos para conformar un sistema multiprocesador virtual que permita realizar tareas fuertes de cálculo en los momentos en que las máquinas no estén siendo usadas. Sin embargo estas máquinas seguirán también utilizándose para las tareas diarias de la organización. Por último, las sala-clúster generan un espacio para la capacitación en el manejo, administración y programación sobre plataformas de computación de alto rendimiento en nuestras Universidades.

¹ J. Zuluaga, A. Ospina. "Installation and Configuration of a Clúster-Room as a Low Cost Solution for the Access to Distributed Computing Technologies in Latin America," Conferencia Latinoamericana de Computación de Alto Rendimiento, ISBN 978-958-708-299-9 , agosto 2007

Las ventajas proporcionadas por un clúster de computadores, pueden escalarse para formar un Grid de computadores. Un Grid se forma a través de la interconexión de varios clústeres, ya sean dentro de la misma organización o de varias, utilizando para ello redes de alta velocidad, sometiéndolas a regímenes de tráfico y condiciones de estabilidad excepcionales. Además de traer diversos beneficios sobre distintas áreas de la investigación científica y tecnológica del país (así lo demuestran experiencias exitosas similares en otras naciones latinoamericanas), la computación grid atraería la atención de otras organizaciones internacionales del mismo tipo (Grids en latinoamérica, Norteamérica, Asia y Europa) abriendo posibilidades de colaboración en el área y de participación en grandes proyectos de Grid a mayor escala (e.g. EELA2).

INTRODUCCIÓN

En la actualidad, las Universidades mantienen una infraestructura de cómputo grande y costosa, necesaria para atender las labores académicas y administrativas propias de la organización, pero que durante muchas horas del día y especialmente de la noche permanecen ociosas. Así mismo, es evidente también que las Universidades realizan cada vez y con mayor frecuencia proyectos de investigación que requieren gran poder de cómputo y de almacenamiento, los cuales se alcanzan en la mayoría de las veces a través de la adquisición de poderosas y costosas computadoras que resuelven sus necesidades.

En cuanto a las necesidades de cómputo, existe una solución de relativo bajo costo que utiliza el poder de cómputo ocioso para crear un sistema multiprocesador virtual que permite cumplir con las tareas diarias de la organización, además de las tareas fuertes de cálculo necesarias. La solución se denomina clúster de computadores. Un clúster de computadores balancea la carga de trabajo en un arreglo de computadores administrando tareas conjuntas a través de la red. Esta configuración ofrece muchos beneficios entre los cuales se destaca el incremento en la velocidad de procesamiento, incremento de la confiabilidad de ejecución de las aplicaciones y el aprovechamiento de los recursos de la organización.

Cuando los computadores de una sala de cómputo pública, originalmente concebida para prestar servicios de terminales de trabajo o capacitación, se configuran para que puedan utilizarse como un clúster de cómputo parcialmente dedicado, da origen a una plataforma computacional denominada en lo sucesivo Sala Clúster.

El objetivo de esta tesis es realizar pruebas de conectividad y procesamiento distribuido a través de la interconexión de dos sala-clúster ubicadas en dos universidades diferentes, utilizando para ello la Red Nacional de Tecnología Avanzada (RENATA).

El documento presentado como resultado de este trabajo de grado está organizado en cinco capítulos, a saber:

1. Marco teórico y Estado del Arte. Se presenta en este apartado una breve descripción de las arquitecturas de computación paralela existentes, detallando varias alternativas de cómputo que encajan con las arquitecturas descritas, para finalmente entrar en el detalle de lo que se denomina Clúster Computing.

2. Grid Computing. Este capítulo describe conceptualmente lo que es un Grid de Cómputo: Objetivos, Topologías, Requerimientos, Roles de Usuario y Middleware.

3. Producto Final. Describe el producto del trabajo de grado. Este capítulo se divide en tres grandes secciones: Construcción de las sala clúster, Interconexión de las sala clúster y Pruebas de Conectividad del Grid.

4. Recomendaciones para trabajos futuros. Se presentan aquí varias recomendaciones a tener en cuenta para continuar o avanzar en esta temática.

5. Conclusiones. Finalmente se muestran varias conclusiones referentes al trabajo realizado en este trabajo de grado.

1. MARCO TEÓRICO Y ESTADO DEL ARTE.

En la década de 1980 debido al avance en la investigación, desarrollo e instalación tanto de software como de hardware para la computación paralela, los investigadores de este campo se especializaron en buscar algoritmos, arquitecturas y programas que soportaran simultaneidad.

Entendiéndose la computación paralela como una arquitectura de cómputo con múltiples procesadores, se definen tres arquitecturas existentes:

1.1 SMP

(*Symmetrical Multiprocessing*, multiproceso simétrico). Una arquitectura SMP consta básicamente de varios procesadores que comparten todos los demás recursos del sistema (memoria principal, almacenamiento secundario, periféricos de entrada y salida). En esta arquitectura no se establece distinción entre los procesadores; todos son jerárquicamente iguales y pueden ejecutar tareas indistintamente.

Debido a que esta arquitectura comparte los recursos del sistema, una debilidad yace en la saturación del bus de memoria.

1.2 MPP

(*Massively parallel processing*, Procesamiento masivamente paralelo) es otro diseño de procesamiento paralelo. Para evitar los cuellos de botella en el bus de memoria, MPP no utiliza memoria compartida. En su lugar, distribuye la memoria RAM entre los procesadores de modo que se asemeja a una red (cada procesador con su memoria distribuida asociada es similar a un computador dentro de una red de procesamiento distribuido). Debido a la distribución dispersa de los recursos RAM, esta arquitectura es también conocida como dispersamente acoplada (*loosely coupled*), o compartiendo nada (*shared nothing*).

Para tener acceso a la memoria fuera de su propia RAM, los procesadores utilizan un esquema de paso de mensajes análogo a los paquetes de datos en redes. Este

sistema reduce el tráfico del bus, debido a que cada sección de memoria observa únicamente aquellos accesos que le están destinados, en lugar de observar todos los accesos, como ocurre en un sistema SMP. Únicamente cuando un procesador no dispone de la memoria RAM suficiente, utiliza la memoria RAM sobrante de los otros procesadores. Esto permite sistemas MPP de gran tamaño con cientos y aún miles de procesadores.

La debilidad de MPP es que la programación se vuelve difícil, debido a que la memoria se rompe en pequeños espacios separados. Sin la existencia de un espacio de memoria globalmente compartido, ejecutar una aplicación que requiere una gran cantidad de RAM (comparada con la memoria local), es difícil.

Escribir una aplicación MPP también requiere estar al tanto de la organización de la memoria manejada por el programa. Donde sea necesario, se requieren insertar comandos de paso de mensajes dentro del código del programa. Además de complicar el diseño del programa, tales comandos pueden crear dependencias de hardware en las aplicaciones. Sin embargo, la mayor parte de vendedores de computadores han salvaguardado la portabilidad de las aplicaciones adoptando, ya sea un mecanismo de dominio público para paso de mensajes conocido como PVM (Máquina virtual paralela), o un estándar en fase de desarrollo llamado MPI (Interfaz de Paso de Mensajes), para implementar el mecanismo de paso de mensajes.

¿Cómo superar las dificultades de SMP y MPP?

1.3 SPP

(*Scalable parallel processing*, Procesamiento paralelo escalable), es un híbrido de SMP y MPP, que utiliza una memoria jerárquica de dos niveles para alcanzar la escalabilidad. La primera capa de memoria consiste de un nodo que es esencialmente un sistema SMP completo, con múltiples procesadores y su memoria globalmente compartida². La segunda capa de memoria aparece lógicamente ante los nodos como una memoria global compartida, de modo tal que permita la interconexión de dos o más nodos para crear sistemas SPP más grandes.

² Boletín 5. Universidad Internacional del Ecuador. Facultad de Informática y Multimedia [en línea]. <http://www.internacional.edu.ec/academica/informatica/creatividad/uide-bits/uide-bits-05-2003.pdf> [Consulta: Noviembre 11 de 2007]

La memoria de dos niveles reduce el tráfico de bus debido a que solamente ocurren actualizaciones para mantener coherencia de memoria. Por tanto, SPP ofrece facilidad de programación del modelo SMP, a la vez que provee una escalabilidad similar a la de un diseño MPP.

Los desarrolladores empezaron a crear aplicaciones que sobrepasaban los límites de los mejores computadores de arquitectura paralela de la década de 1980, e impulsados por diversos grupos como el HPCG³ de la Universidad Politécnica de Cataluña y SCAI⁴, comenzaron a buscar recursos por fuera de los límites de las máquinas para lograr la resolución de problemas de cómputo cada vez más complejos⁵, y así se logran los primeros pasos hacia los sistemas distribuidos.

El objetivo de estos sistemas es resolver problemas demasiado grandes para cualquier supercomputadora, mientras se mantiene la flexibilidad de trabajar en múltiples problemas más pequeños.

Estas alternativas de cómputo en red, que consiguen rendimientos comparables con las arquitecturas más avanzadas y a un precio más razonable, se popularizan a partir de proyectos tales como CONDOR⁶, y LEGION⁷ lanzados en 1988 y 1993 respectivamente.

A continuación se describen ejemplos de las alternativas de cómputo mencionadas anteriormente.

1.4 SISTEMAS DISTRIBUIDOS

Aquellos sistemas en el que los componentes hardware y/o software ubicados en computadores en red, se comunican y coordinan sus acciones intercambiando mensajes.⁸

³ High Performance Computing Group. Departamento de Arquitectura de Computadores. Universidad Politécnica de Cataluña.

⁴ SCAI : Scientific Computing Associates Inc.

⁵ BERMAN, Fran y otros. Grid Computing; Making the global infrastructure a reality. Wiley, West Sussex, 2003.

⁶ CONDOR High throughput computer.

⁷ LEGION WorldWide Virtual Computer.

⁸ COLOURIS, G., y Otros. Sistemas Distribuidos: Conceptos y Diseño. Addison-Wesley, 3ra edición, 2001.

Se considera el GRID, en este sentido, como una especie de sistema distribuido.

1.5 INTRANET COMPUTING

Unión de la potencia computacional desaprovechada en los recursos hardware distribuidos en una red de área local (la mayoría de los PCs y estaciones de trabajo están frecuentemente ociosos). Su principal ventaja es que, para un determinado tipo de aplicaciones, puede proporcionar rendimientos semejantes a los ofrecidos por los sistemas de alto rendimiento con un coste económico casi nulo.

1.6 INTERNET COMPUTING

Aprovechamiento de la potencia de los recursos distribuidos por Internet siguiendo el modelo cliente/servidor. Actualmente casi todas estas herramientas se limitan a ejecución paramétrica. La ventaja es el gran rendimiento que se puede obtener y sus inconvenientes se deben al bajo ancho de banda y a la escasa seguridad en Internet.

1.7 CLÚSTER COMPUTING

1.7.1 Definición de Clúster

Un clúster consiste en un grupo de computadoras de relativo bajo costo conectadas entre sí mediante un sistema de red de alta velocidad (Gigabit de fibra óptica por lo general) y un software que realiza la distribución de la carga de trabajo entre los equipos. Por lo general, este tipo de sistemas cuentan con un centro de almacenamiento de datos único. Su ventaja fundamental es la mejor relación coste/rendimiento y sus inconvenientes, la dificultad de programación y de mantenimiento.

La base, en términos de ingeniería, detrás del concepto formal de computación clúster, es la conocida Ley de Amdahl, la cual fue publicada en 1967 y se reconoce hoy como documento seminal de la computación paralela⁹.

Los clústeres según su uso se dividen en:

Clúster de Alta Disponibilidad. Busca proporcionar un servicio de la manera más confiable posible. En esta configuración, uno o más equipos del clúster proporcionan el servicio, mientras que los demás funcionan como respaldo. Normalmente efectúan *mirroring* de la información en los equipos de servicio, a fin de mantener una copia actualizada de dicha información. Si alguno de los equipos de servicio llega a fallar, el equipo de respaldo entra en su lugar, de esta manera logrando que el servicio no sea interrumpido.

Clúster de balanceo de cargas. Es un clúster que permite que un conjunto de servidores compartan la carga de trabajo y de tráfico de sus clientes. Un ejemplo de este tipo de clúster es Linux Virtual Server (LVS).

Clúster de alto rendimiento. Busca la resolución de un problema, por medio de la cooperación entre los equipos que lo componen, en el menor tiempo posible. En este sentido es diferente al clúster de alta disponibilidad pues se busca que todos los equipos estén realizando alguna tarea componente de la solución al problema, y comunicándose con los demás nodos.

Los primeros proyectos para utilizar este tipo de configuración en cómputo de alto rendimiento se enfocaron a la explotación de recursos existentes, como los nombrados anteriormente (CONDOR, LEGION). Ejemplo: Clúster Beowulf.

1.7.2 Clúster Beowulf

En 1994, se integró el primer clúster de PCs en el Centro de Vuelos Espaciales Goddard de la NASA, para resolver problemas computacionales que aparecen en

⁹ PEREZ, John. Desarrollo de un clúster computacional para la compilación de algoritmos en paralelo en el Observatorio Astronómico [en línea].
<http://www.usergioarboleda.edu.co/civilizar/revista6/Desarrollo_de_un_clúster_computacional_para_la_compilacion_de_algoritmos_en_paralelo_en_el_Observatorio_Astronomico.doc> [Consulta: Noviembre 9 de 2007].

las ciencias de la tierra y el espacio. Los pioneros de este proyecto fueron Thomas Sterling, Donald Becker y otros científicos de la NASA. El clúster de PC's desarrollado tuvo una eficiencia de 70 megaflops (millones de operaciones de punto flotante por segundo). Los investigadores de la NASA le dieron el nombre de *Beowulf* a este clúster, en honor del héroe de las leyendas medievales, quien derrotó al monstruo gigante Grendel.¹⁰

En 1996, hubo también otros dos sucesores del proyecto Beowulf de la NASA. Uno de ellos es el proyecto *Hyglac* desarrollado por investigadores del Instituto Tecnológico de California (CalTech) y el Laboratorio de Propulsión Jet (JPL), y el otro, el proyecto *Loki* construido en el Laboratorio Nacional de Los Álamos, en Nuevo México. Cada clúster se integró con 16 microprocesadores Intel Pentium Pro y tuvieron un rendimiento sostenido de más de un gigaflop con un costo menor a \$50,000 dólares.¹¹

❖ Características de un Clúster Beowulf

- Los nodos son dedicados al clúster beowulf y no tienen otro propósito.
- La red o redes en las cuales residen los nodos son dedicadas al clúster beowulf y no cumplen otro propósito.
- Los nodos son computadores M²COTS¹². Una parte esencial de la definición de Beowulf (lo cual la diferencia de sistemas MPP producidos en masa empresas como IBM y HP) es que sus nodos de cómputo son comodidades producidas en masa, listas para su uso, y por lo tanto relativamente económicas.
- Los dispositivos de red son componentes M²COTS.
- Los nodos disponen de software de fuente abierta (open source).
- El clúster resultante es usado para computación de alto rendimiento.
- Una característica importante de un *Beowulf* es que las actualizaciones de *hardware* más comunes, como son actualización de procesador, incremento de memoria, o mejora de velocidad de transferencia en la red, no cambian

¹⁰ MERKEY, Phil Beowulf History <<http://www.beowulf.org/overview/history.html>> [Consulta: Noviembre 11 de 2007]

¹¹ WARREN, Michael S. y OTROS [Pentium Pro Inside: I. A Treecode at 430 Gigaflops on ASCI Red, II. Price/Performance of \\$50/Mflop on Loki and Hyglac.](http://www.lanl.gov/papers/sc97/) <<http://loki-www.lanl.gov/papers/sc97/>> [Consulta: Noviembre 11 de 2007]

¹² M²COTS: Mass market commodities off-the-shelf.

el modelo de programación utilizado. Por lo tanto, los usuarios de estos sistemas pueden contar con mejor compatibilidad con equipos futuros.¹³

❖ Arquitectura de un Clúster Beowulf

Se define la arquitectura de un Clúster Beowulf en forma de capas como muestra la figura 1, donde cada capa cumple una función específica.

- **Capa de red:** Es donde se realiza la conectividad de los recursos en el clúster; en esta capa se recomienda utilizar la topología lógica de red en bus.
- **Capa de recursos:** Compuesta por los nodos que son parte del clúster

Figura 1. Arquitectura de clúster.

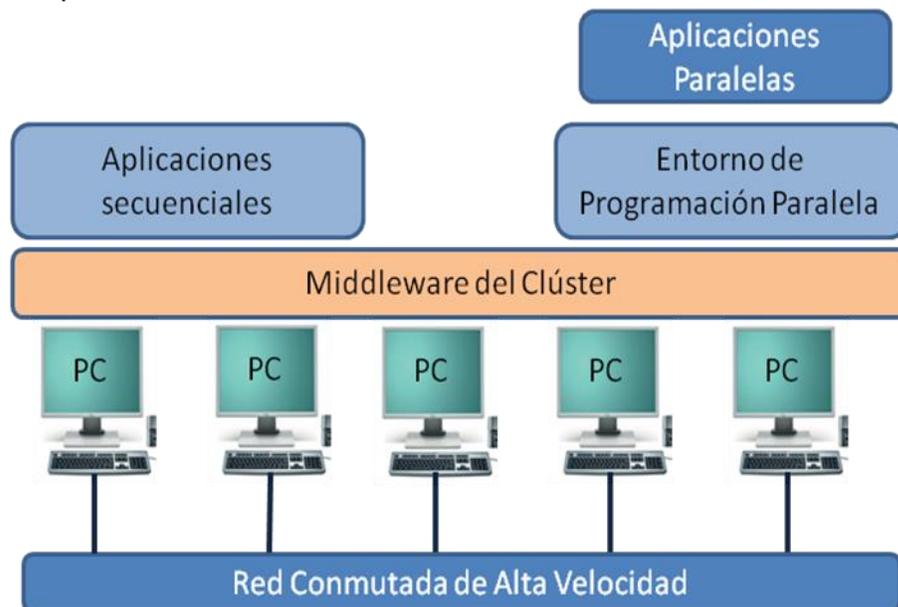


Imagen tomada de Procesamiento paralelo, organización y arquitectura de computadores y reconstruida por los autores del proyecto

¹³ CONTRERAS, Gerardo. Clúster de alta disponibilidad [en línea].
<http://www.puntogmx.net/wordpress/wp-content/uploads/2007/04/clúster_ha.pdf> [Consulta: Noviembre 11 de 2007]

- **Capa de Middleware:** Provee herramientas para la optimización y mantenimiento del sistema: migración de procesos, *checkpoint-restart* (congelar uno o varios procesos, mudarlos de servidor y continuar su funcionamiento en el nuevo host), balanceo de carga y tolerancia a fallos. También brinda escalabilidad, ya que debe poder detectar automáticamente nuevos servidores conectados al clúster para proceder a su utilización. Igualmente, esta capa provee al clúster de una interfaz única de acceso al sistema, denominada SSI (*Single System Image*), la cual genera la sensación al usuario que utiliza un único computador muy potente.

El middleware recibe los trabajos entrantes al clúster y los redistribuye de manera tal que el proceso se ejecute más rápido y el sistema no sufra sobrecargas en un servidor. Esto se logra mediante políticas definidas en el sistema (automáticamente o por un administrador) que le indican dónde y cómo debe distribuir los procesos, por un sistema de monitorización, el cual controla la carga de cada CPU y la cantidad de procesos en él.

- **Capa de entornos y herramientas de programación paralela:** Aquí se realizan actividades de análisis de rendimiento, administración, programación y depuradores paralelos. En esta capa ocurre el paso de mensajes entre nodos por medio de librerías como MPI, PVM, Sockets o SysV IPC.
- **Capa de aplicaciones:** En esta capa es donde interactúa el administrador del clúster. Aquí se encuentran aplicaciones paralelas o distribuidas y secuenciales.¹⁴

La desventaja del clúster es que su escalabilidad es limitada a nodos geográficamente visibles y por políticas de seguridad, deben ser totalmente confiables; de igual forma, los trabajos realizados en los nodos son dependientes.

La evolución del clúster es la computación Grid, la cual se puede entender como la unión de dos o más clústeres; ejemplo de esto es el proyecto SETI@Home, que consiste en el procesamiento de señales de radio para buscar una prueba de

¹⁴ *Arquitectura Clúster. Clústers. Arquitecturas Distribuidas.*
 <<http://www.ac.uma.es/educacion/cursos/informatica/ArqDist/pdfs/04-Clusters.pdf>> [Consulta: Noviembre 11 de 2007]

inteligencia extraterrestre, que puede ser considerado como el clúster distribuido más grande existente con aproximadamente 5.2 millones de nodos participantes¹⁵.

1.7.3 Razones para trabajar con Clústeres

Existen diferentes razones para realizar clústeres pero la más importante es que el procesamiento de grandes volúmenes de información sea más eficiente y rápido, que si se realizara en un solo sistema. Las ventajas en tiempo de cómputo hacen que, aun así, este tipo de soluciones para el cómputo de altas prestaciones (HPC, *High Performance Computing*) sean consideradas muy atractivas y en constante evolución.¹⁶

Un clúster trabaja generalmente en una red de área local (LAN) y si las máquinas están en un espacio cercano físicamente logran una buena comunicación; en un concepto mas extenso se encuentra el Grid que trabaja en redes de área extensa (WAN). Una Grid es un tipo de sistema paralelo y distribuido que permite compartir, seleccionar y añadir recursos que se encuentran distribuidos a lo largo de dominios administrativos "múltiples" basados en su disponibilidad, capacidad, rendimiento, costo y calidad de servicio que requiere un usuario.¹⁷ Algunas personas llaman a Grid como el Clúster de Clústeres.

15 SETI@Home. Acerca de SETI@Home. < http://setiathome.berkeley.edu/sah_about.php> [Consulta: Noviembre 11 de 2007]

16 Administración Avanzada de GNU /Linux. JORBA, Esteve. p 415-432.

17 LIZARRA, Carlos. Clúster de Linux [en línea]. Disponible en internet en: <http://clusters.fisica.uson.mx/clusters_de_Linux.htm>, [visitado el 10/09/2007].

2. GRID COMPUTING.

El concepto de Grid fue introducido oficialmente por Ian Foster y Carl Kesselman (investigadores en ciencia computacional) en 1998, aunque no era la primera vez que se hablaba de acceso por demanda de datos y servicios computacionales. El enfoque inicial de computación Grid, definía que los usuarios (clientes y aplicaciones) ganan acceso a recursos de cómputo, como procesadores, almacenamiento, información y aplicaciones, a medida que lo necesitan sin importancia o conocimiento de la localización de estos recursos o las tecnologías, hardware, sistemas operativos, detrás de éstos.

Desde que el término Grid fue introducido por primera vez, en la comunidad Grid se han iniciado investigaciones y proyectos que muestran que un Grid es una arquitectura de computación distribuida de alto rendimiento que busca compartir y coordinar recursos de hardware y software para optimizar el procesamiento y almacenamiento de tareas de cómputo intensivas ubicadas en una red geográficamente amplia, como se observa en la Figura 2.

2.1 OBJETIVOS DEL GRID

Al implementar una Grid es importante que cualquier hardware que tenga conexión a la red pueda ser integrado al Grid y su conexión no debe ser limitada por el área geográfica en la que se encuentre. El Grid virtualiza recursos heterogéneos geográficamente dispersos. (Ver Figura 2)

Esta red debe ser altamente confiable y disponible en cualquier momento (todos los días a toda hora), además debe tener la capacidad de brindar o compartir todo tipo de recursos (hardware, software, datos, dispositivos de almacenamiento) a cualquier tipo de dispositivo electrónico moderno que haga uso de las Tecnologías de la Información (TI). Asimismo, un Grid debe soportar dichos dispositivos, manteniendo siempre suficiente capacidad de cálculo, su transparencia al usuario (en conectividad) y su uso sencillo o fácil manejo, pues está pensado para usuarios finales.

Figura 2. Conceptualización de un Grid

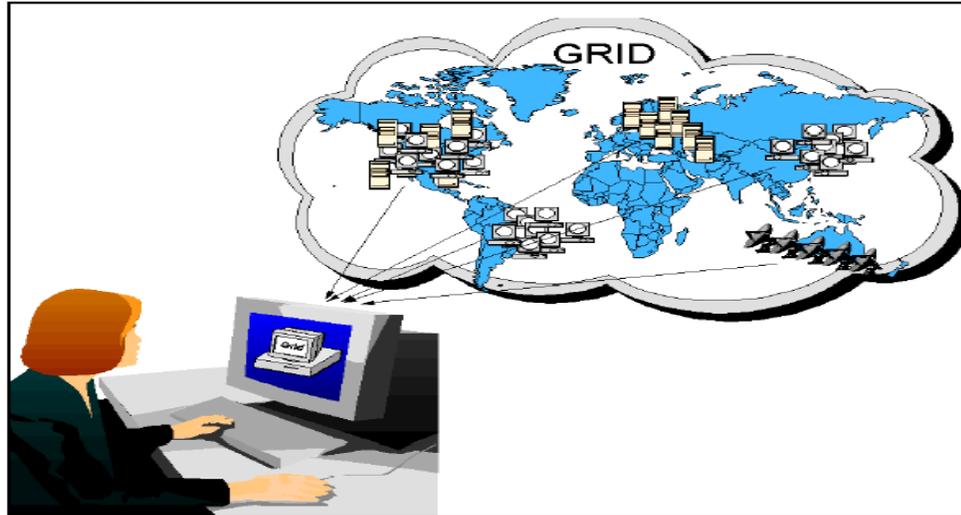


Imagen tomada de "Introduction to Grid Computing with Globus", IBM.

2.2 TOPOLOGIAS GRID

Las topologías Grid hacen referencia a los distintos tipos de Grid que podemos encontrar de acuerdo a los servicios que éste ofrezca y su arquitectura física. Existen tres tipos de Grid, los cuales se explican a continuación. (Ver Figura 3)

2.2.1 Intragrid (InnerGrid)

Esta arquitectura es la más sencilla de las tres topologías: existe en una sola organización (red privada) y posee los servicios básicos de un Grid. La organización puede tener un número de computadores que compartan un dominio de seguridad común. Las principales características de un IntraGrid son: un solo proveedor de seguridad, un ancho de banda alto en la red privada (LAN) y siempre disponible. Un IntraGrid provee una relativa fijación estática de los recursos computacionales y la habilidad de compartir fácilmente datos entre sistemas Grid¹⁸.

¹⁸ IBM Redbooks, Introduction to Grid Computing with Globus. Disponible en internet <URL: <http://www.redbooks.ibm.com/abstracts/sg246895.html?Open>>, [visitado 02/07/2007].

Figura 3. Intragrid, Extragrid e InterGrid.

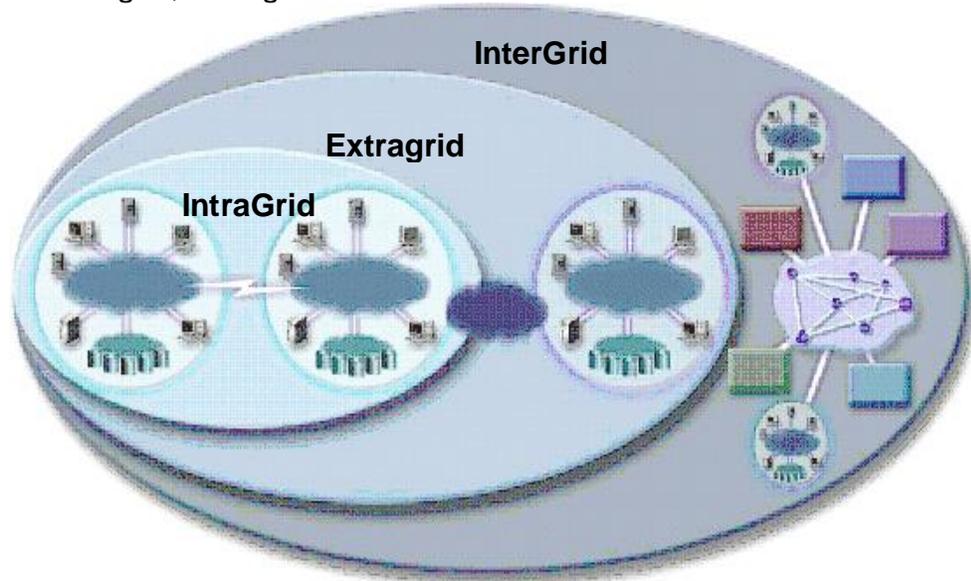


Imagen tomada de “Introduction to Grid Computing with Globus”, IBM.

2.2.2 ExtraGrid (OuterGrid)

Estos tipos de Grid también se conocen como “partner Grids”, porque se extienden fuera del dominio de administración de la empresa, conectándose a dos o más Extragrid. Se ocupan de compartir recursos, incluyendo a aquellos socios u organizaciones externas con las cuales ya se ha establecido una relación de negocios. Involucra generalmente más de un proveedor de seguridad, un nivel de complejidad de manejo alto y cuenta con conectividad remota (WAN).

2.2.3 InterGrid

La interGrid permite compartir recursos computacionales y datos de almacenamiento a través de una red pública, en colaboración con otras empresas. La información o datos de la InterGrid son en principio global y públicos. Así mismo las aplicaciones, deben ser modificadas por una audiencia global y deben cumplir con unos requisitos o características principales que las clasifiquen como tales. Estas características incluyen seguridad dispersa, múltiples organizaciones

dentro de ella y conectividad remota presente (WAN). Una InterGrid necesita integración dinámica de aplicaciones, recursos y servicios¹⁹.

2.3 REQUERIMIENTOS DE UN GRID

Los requerimientos básicos de un Grid se pueden determinar de acuerdo a tres características importantes presentes en él:

- ❖ **Componentes y dispositivos:** En cuanto a los recursos computacionales (software, CPU, memoria, tiempo de procesamiento) un Grid debe permitir compartirlos y acceder a éstos, además debe permitir también compartir los datos. Respecto al hardware es importante que cualquier dispositivo tenga acceso a la red; estos dispositivos pueden ser servidores, nodos, clientes, entre otros, los cuales pueden estar dispersos geográficamente, conectados a diferentes redes y con diferentes administradores, sistemas operativos y hardware en sí mismo.
- ❖ **Servicios:** De acuerdo a los servicios que ofrece un Grid es importante contar con autenticación, políticas de autorización, descubrimiento de recursos, ubicación de recursos, acceso de datos remoto, alta velocidad de transferencia de datos, manejo de recursos y de fallas, monitoreo y detección de intrusos, garantizar el rendimiento, manejo de cuentas y pagos, adaptación, flexibilidad y evolución para la implementación de nuevos servicios.
- ❖ **Necesidades:** De acuerdo a las necesidades actuales que la computación de alto rendimiento exige, es importante contar con disponibilidad y confiabilidad en el ancho de banda, además de una red global (Internet), capacidad de almacenamiento de datos y manejo de información creciente.

2.4 ROLES DE USUARIO EN UN GRID

Para empezar a construir un entorno Grid es importante tener en cuenta los diferentes roles de usuario que existen y se manejan dentro de un Grid; estos roles de usuario dependen del tipo de actividades y privilegios que tenga y

¹⁹ Joshy Joseph, Mark Ernest y Craig Fellenstein. Evolution of grid computing architecture and grid adoption models, disponible en internet <URL:
<http://www.research.ibm.com/journal/sj/434/joseph.html>>, [visitado 02/09/2007].

desarrolle cada uno. Existen dos tipos de usuario: rol de Donante y rol de Administrador.

2.4.1 Rol de Donante

Este tipo de usuario primero tiene que enrolarse en el Grid e instalar en su propia máquina el software de Grid proveído para la conexión, además tiene la opción de vincular su máquina como un donante del Grid.

Para enrolarse en el Grid el usuario requiere de una autenticación para propósitos de seguridad; esta autenticación solicita el establecimiento de una identidad para el usuario, lo cual se puede lograr a través de la Autoridad Certificadora (Certificate Authority - CA). Este proceso de identificación debe realizar algunos pasos para asegurarse que el usuario es quién dice ser.

Una CA es una entidad de confianza cuya principal función es la de certificar la autenticidad de los usuarios. La CA emite y firma certificados digitales que actúan como prueba de que la llave pública correcta esté asociada a un usuario. De esta forma a través de la confianza de una tercera parte, el que confía en la Autoridad Certificadora, confía también en la llave del usuario.

Una CA puede ser pública o privada. Las CA públicas emiten los certificados para la población en general (aunque a veces están focalizadas hacia algún colectivo en concreto) y además firman CA de otras organizaciones. Las CA privadas pueden emitir certificados para una población, pero no firman CA de otras organizaciones. Una lista de Autoridades Certificadoras puede encontrarse en el sitio Web http://www.dmoz.org/Computers/Security/Public_Key_Infrastructure/PKIX/Tools_and_Services/Third_Party_Certificate_Authorities/

El CA crea un certificado especial disponible para que el software evalúe la verdadera identidad de un usuario Grid y sus solicitudes. Posteriormente se siguen pasos similares para la identificación de la máquina donante si el usuario desea vincularla.

Una vez el usuario y/o la máquina están autenticados, se provee el software al usuario para que lo instale en su máquina con el propósito de que éste se vincule al Grid como un donante. Este software puede ser pre-configurado automáticamente por el sistema de administración del Grid para conocer la

dirección de comunicación de los nodos del Grid y la información de identificación de los usuarios y las máquinas. De esta manera la instalación puede tratarse simplemente de hacer un click por parte del usuario reduciendo la interacción requerida de éste al mínimo.

En algunos casos de instalaciones automatizadas, el usuario tiene la opción de elegir: el límite de recursos que quiere donar al Grid, el tiempo que su máquina puede ser usable en el Grid y otras políticas relacionadas con esto. Además puede necesitar informar al administrador del Grid, sobre cuáles IDs de usuario de otras máquinas le pertenecen en el Grid.

- ❖ **Preguntar y enviar trabajos.** El usuario usualmente enviará preguntas para realizar un chequeo y saber qué tan ocupado se encuentra el Grid, qué recursos hay disponibles y cómo están progresando los sub-trabajos del trabajo enviado al Grid. El sistema Grid provee generalmente una interfaz gráfica con herramientas de líneas de comando para que el usuario pueda realizar sus chequeos.

2.4.2 Rol de Administrador

El uso del Grid como un usuario administrador, describe una serie de actividades a desarrollar que se deben tener en cuenta como: la planeación del Grid, su seguridad, la instalación del software, la administración de la vinculación de usuarios y donantes al Grid, la CA, la administración de recursos y los datos que se compartirán en el Grid. (Este tipo de actividades se pondrán en práctica para la construcción de la Intragrid).

- ❖ **Planeación.** En el caso de la planeación es importante que el administrador entienda los requerimientos de la organización y el uso que se le va a dar al Grid para así poder escoger la mejor tecnología de Grid que satisfaga estos requerimientos.

Se sugiere que el administrador comience primero por la configuración de un pequeño Grid para aprender y obtener información suficiente sobre su instalación y administración antes de tener que enfrentarse a situaciones más complicadas con una Grid de mayor tamaño.

El uso de un Grid a menudo nace de la necesidad de incrementar los recursos de algún tipo. Frecuentemente se ve como algunas organizaciones presentan excesos de capacidad de un recurso en particular, por lo cual, una de las primeras consideraciones es el hardware disponible y cómo se interconectará a través de una LAN o WAN. Posteriormente las organizaciones pueden añadir más hardware para aumentar las capacidades del Grid.

Es importante entender las aplicaciones que van a ser utilizadas en el Grid. Sus características pueden afectar las decisiones de la manera de elegir y configurar el hardware y su conectividad.

- Seguridad. La seguridad es muy importante en la planeación y mantenimiento del Grid, tal vez mucho más que en cualquier entorno convencional de computación distribuida. Los usuarios del Grid tienen la capacidad de ejecutar programas que mueven muchos datos y esto es potencialmente inseguro, debido a la cantidad de virus y troyanos que pueden ejecutarse en este medio. Por ello es importante entender cuáles componentes del Grid deben ser rigurosamente protegidos.
- Organización. Las consideraciones tecnológicas son importantes en la construcción de un Grid. Sin embargo, las cuestiones organizacionales y de negocio pueden ser igualmente importantes. Es importante comprender cómo los departamentos en una organización interactúan, operan y contribuyen con el conjunto.

A menudo, hay barreras construidas entre los departamentos y proyectos para proteger sus recursos en un esfuerzo por aumentar la probabilidad de éxito en el tiempo. Sin embargo, replanteando algunas de estas relaciones se puede encontrar que compartiendo recursos se puede beneficiar a toda la organización. Por ejemplo, un proyecto que tenga un calendario ajustado y un presupuesto limitado, puede no tener los recursos necesarios para resolver el problema. Un Grid pueda dar a este proyecto una medida adicional de seguridad, proporcionando un margen adicional de recursos y la capacidad necesaria para terminar el proyecto. Del mismo modo, un proyecto que esté en sus etapas iniciales, cuando los recursos no están siendo plenamente utilizados, pueden ser capaces de realizar donaciones para otros proyectos que los necesiten. Un Grid también ofrece capacidad a la administración de la organización para ver el panorama más amplio y reaccionar con mayor rapidez en el traspaso de recursos, prioridades y políticas de utilización.

❖ **Instalación.** La instalación del Grid es muy rigurosa y metódica y de manera general se define en 3 pasos:

- El sistema Grid elegido debe ser apropiadamente instalado y configurado en un grupo de máquinas. Estas máquinas deben estar conectadas usando redes con suficiente ancho de banda. Es de vital importancia entender, que si algún grupo de equipos falla, el Grid debe seguir trabajando con otro grupo de ellos sin ningún problema.
- Después de la instalación, el software del Grid, necesita estar configurado para las direcciones e IDs de la red local. Esto lo hace el administrador con un acceso de *root*.
- Una vez que el Grid sea operacional, existe la posibilidad de instalar software y datos adicionales tanto en el Grid como en las máquinas donantes.

❖ **Administración de la vinculación de usuarios y donantes al Grid.** Esta es una actividad importante que debe vigilar el administrador, pues él es el responsable de controlar los derechos que tienen los usuarios en el Grid, los derechos de acceso que tienen las máquinas donantes, los trabajos que se vayan a correr en determinadas máquinas donantes del Grid ejecutados por usuarios especiales, entre otros.

❖ **Autoridad Certificadora (CA).** Un Grid está diseñado para compartir recursos y ejecutar código, por lo tanto, una CA es fundamental para garantizar mayores niveles de seguridad. Si la seguridad del Grid se ve comprometida de alguna manera, se convierte en un terreno fértil para virus, troyanos y otros ataques.

La CA es uno de los aspectos que mantiene fuerte la seguridad del Grid, y el administrador tiene la opción de usar un CA externo u operar uno por si mismo.

Las principales responsabilidades del CA son:

- Identificar positivamente entidades que piden los certificados
- Emitir, derogar y archivar los certificados
- Proteger el Certificate Authority Server
- Mantener un espacio de nombres únicos para los dueños de certificados.
- Servir a aquéllos que necesitan autenticar las entidades con los certificados firmados.
- Encargarse del proceso de inicios de sesión en el ambiente Grid

En pocas palabras, una autoridad de certificación se basa en un sistema de encriptación de clave pública. En este sistema, las claves se generan en pares, una clave pública y una clave privada. Cualquiera puede ser usada para encriptar datos de tal forma que la otra se requiere para descifrar.

La clave privada es custodiada por el propietario y nunca debe ser revelada a nadie. La clave pública se da a cualquier persona que la necesite. Una autoridad de certificación se utiliza para mantener y garantizar la pertenencia de estas claves públicas.

Cuando un usuario utiliza su clave privada para cifrar, el receptor utiliza la clave pública correspondiente para descifrar. El receptor sabe que sólo los usuarios de su clave pública podrán descifrar el mensaje correctamente. Sin embargo, cualquier persona puede interceptar este mensaje y descifrarlo porque cualquier persona puede obtener la clave pública del originador.

Si el originador del mensaje encripta doblemente el mensaje con su clave privada y con la clave pública del destinatario, se forma un enlace de comunicación seguro. El receptor utiliza su clave privada clave para descifrar el mensaje y, a continuación, usa la clave pública del remitente para el segundo descifrado. El receptor sabe que si se descifra el mensaje correctamente, entonces sólo el remitente podría haber enviado el mensaje y por otra parte, el remitente sabe que sólo el receptor podrá descifrar la misma.

La mejor de este sistema es que no hay necesidad de enviar una clave de cifrado del remitente al receptor, como se hace en los sistemas convencionales de encriptación, ya que cualquier manipulación no autorizada de la comunicación la pone al descubierto.

❖ **Administración de recursos y los datos que se compartirán en el Grid.** Otra responsabilidad del administrador es controlar los recursos del Grid; esto incluye dar los permisos a los usuarios Grid, usar los recursos, identificar aquellas organizaciones que requieran de software adicional, reducción en exceso de hardware o reducción de costos, entre otros.

El tema de compartir datos requiere de mucha atención, pues para pequeños Grid puede ser una tarea fácil, usando sistemas de archivos de red existentes, bases de datos o protocolos de transferencia de datos estándar. Cuando una Grid crece y los usuarios llegan a depender de un solo almacén de datos, es importante que el administrador considere procedimientos que permitan mantener copias de respaldo y réplicas de la información para improvisar cambios. Toda la administración de recursos aplica sobre los datos de un Grid²⁰.

Es importante aclarar que cuando un Grid se compone de dos o más organizaciones estas se agrupan mediante organizaciones virtuales. Las organizaciones virtuales (VO) facilitan el diseño de grandes aplicaciones distribuidas que involucran el uso intensivo de recursos compartidos, la construcción de una organización virtual presenta consideraciones de administración como la admisión o revocación de miembros, labor crítica debido su naturaleza distribuida²¹.

El aspecto más importante a tener en cuenta consiste en que los diversos sitios que conforman la malla están normalmente administrados por diferentes organizaciones, cada una con sus propios mecanismos y políticas de seguridad, presentando mayor dificultad que si los componentes de la malla estuvieran localizados en una sola red de área local²².

2.5 MIDDLEWARE

El Middleware es un software puente que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas y dispersos geográficamente; trabaja en las capas de aplicación y

²⁰ IBM Redbooks, Introduction to Grid Computing. Disponible en internet <URL: <http://www.redbooks.ibm.com/abstracts/sg246778.html?Open> >, [visitado 25/02/2008]

²¹ Wu BY (reprint author), Tsing Hua, Internet and network economics, proceedings lecture notes in computer science 3828: 959-968, 2005

²² Palmieri, Francesco, Emerging trends in information and communication security, proceedings lecture notes in computer science 3995: 45-59, 2006

las capas inferiores (sistema operativo y red). Los middleware para Grid más utilizados son: Globus Toolkit, gLite y otros como Gria, OGSA-DAI, Gedeon, rocks.

2.5.1 Globus® Toolkit.

Es un software de código abierto fundamental para la construcción de Grids que permite a los usuarios de una forma segura y homogénea compartir poder de cómputo, datos y otras herramientas sin importar la ubicación geográfica. Globus es uno de los middlewares más utilizados para la tecnología Grid y gracias a éste se han creado y dirigido diversos e importantes proyectos en el mundo.

Globus Toolkit ha crecido con la iniciativa Open Source, similar al proyecto GNU/Linux, lo que ha permitido que los usuarios lo utilicen con más entusiasmo y le hagan mejoras, para hacer a Globus cada vez más atractivo. Hasta el momento hay 3 versiones de Globus Toolkit: la 1.0 de 1998, la 2.0 de 2002 y la 4.0 con nuevos y mejores paquetes que las anteriores. Basta con estas 3 versiones para decir que Globus Toolkit es el estándar por defecto para la construcción de Grid, aunque no el único.

La Figura 4 muestra de manera general todos los servicios disponibles en Globus Toolkit versión 4, que incluye servicios de seguridad, administración de datos, ejecución de administración, servicios de información y tiempos de ejecución comunes.

Las herramientas que se han generado con Globus Toolkit y sus versiones, se han utilizado alternadamente para desarrollar una amplia gama de infraestructuras "Grid" y aplicaciones distribuidas. Se resumirán a continuación las características principales del último lanzamiento, la web de servicios basada en GT4, que proporciona mejoras significativas sobre los lanzamientos anteriores en términos de robustez, funcionamiento, utilidad, documentación, rendimiento de los estándares, y funcionalidad.

El software por sí mismo, tiene un conjunto de librerías y programas direccionados a problemas comunes cuando se construyen sistemas de servicios y aplicaciones distribuidas. Provee una variedad de componentes y de capacidades, incluyendo las siguientes:

Figura 4. Servicios Globus toolkit versión 4 (GT4)

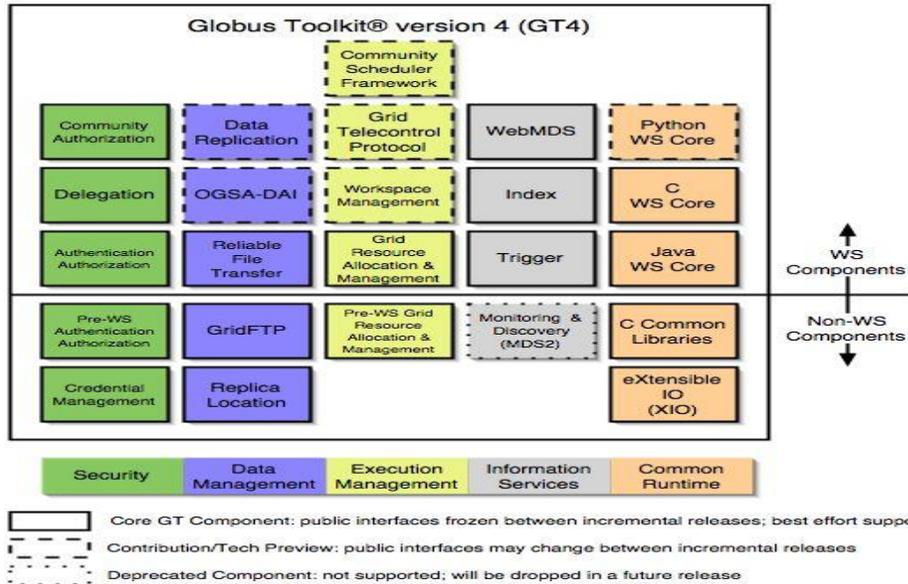


Imagen tomada de "Globus Toolkit Version 4: Software for Service-Oriented Systems".

- Un conjunto de implementación de servicios enfocado a la administración de su infraestructura.
- Herramientas para la construcción de nuevos servicios Web en Java, C y Python.
- Una infraestructura de seguridad basada en poderosos estándares.
- APIs de clientes (en diferentes lenguajes) y programas de líneas de comandos para tener acceso a estos servicios y capacidades.
- Documentación detallada de sus variados componentes, interfaces, y cómo pueden ser usados para construir aplicaciones.

Estos componentes permiten un ecosistema rico de elementos y de herramientas de construcción o interoperación con los componentes GT y una amplia variedad de usos dentro muchos dominios.

GT4 hace un extenso uso de mecanismos de servicios Web para definir sus interfaces y la estructura de sus componentes. Los servicios Web proveen flexibilidad, extensibilidad, y mecanismos basados en XML, para descubrir e invocar servicios de red; adicionalmente su documentación orientada de protocolos está bien adaptada a un conjunto de interacciones que muchos afirman son preferibles para sistemas distribuidos robustos.

Estos mecanismos facilitan el desarrollo de arquitecturas orientadas a servicios, sistemas y aplicaciones estructuradas como servicios de comunicación, en donde se describen las interfaces de servicios, las operaciones son invocadas, el acceso asegurado, entre otros.

Las aplicaciones para usuarios finales se refieren únicamente a dominios específicos con operaciones tales como tasación de listas o análisis de una secuencia de genes: Este tipo de computación últimamente requiere la manipulación y administración de infraestructura de dispositivos físicos como: computadores, sistemas de almacenamiento e instrumentación.

GT4 proporciona un conjunto de servicios de Infraestructura Grid que implementa interfaces de administración computacional, almacenamiento, y otros recursos. En muchos despliegues o versiones de Globus (ej. TeraGrid, Open Science Grid, LHC Computing Grid, China Grid, APgrid), estos servicios son desarrollados para soportar un rango de aplicaciones, cada una de las cuales ejecuta su propio código de aplicaciones específicas que se encargan de realizar estos servicios.²³

En la figura 5, se muestran varios aspectos de la Arquitectura esquemática de GT4 y varios de sus componentes. Destacando lo siguiente:

- Un conjunto de implementaciones de servicios (el fondo de la mitad de la figura).
- Tres contenedores que se pueden utilizar para recibir desarrollos de usuarios escritos en Java, Python y C.
- Un sistema de librerías que permite que los programas en JAVA, Python y C invoquen operaciones tanto de GT4, como de servicios de desarrollo para usuarios.

Es importante observar que con GT4 los clientes pueden interactuar con diferentes servicios de la misma forma, lo cual facilita la construcción de sistemas complejos, facilita su interoperabilidad y anima la reutilización del código.

²³ Foster Ian, Globus Toolkit Version 4: Software for Service-Oriented Systems, disponible en internet: <URL:<http://www.globus.org/alliance/events/sc05/GT4.pdf>>, [visitado 22/08/2007].

Figura 5. Arquitectura esquemática de GT4 y varios de sus componentes.

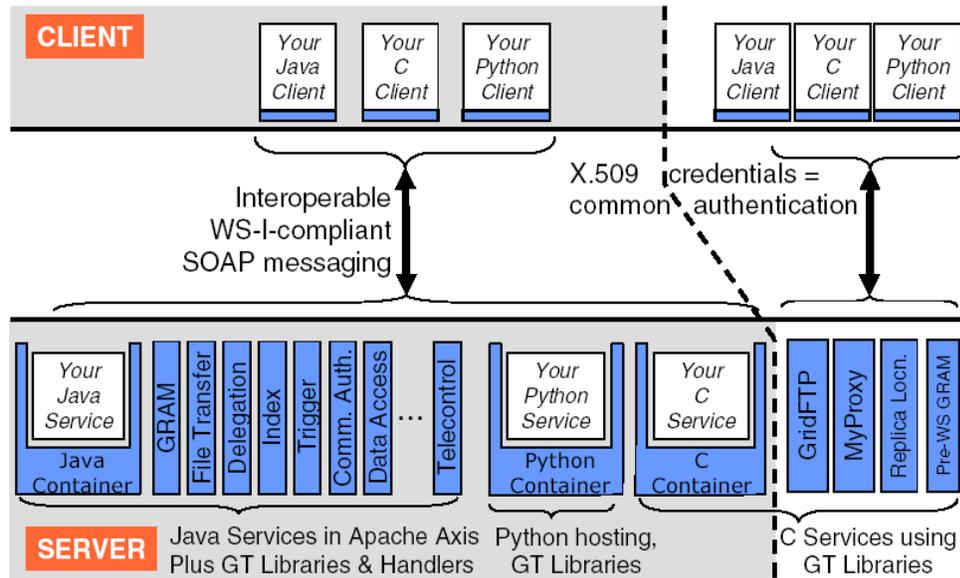


Imagen tomada de “Globus Toolkit Version 4: Software for Service-Oriented Systems”.

Para finalizar, en la Figura 6, se muestran las capacidades o habilidades con que cuenta la nueva versión de Globus Toolkit (GT4) en cuanto a aplicaciones de usuario.

Figura 6. Capacidades del contenedor GT4

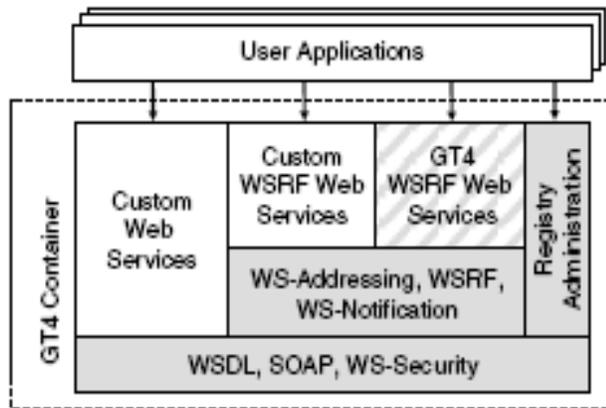


Imagen tomada de “Globus Toolkit Version 4: Software for Service-Oriented Systems”.

3. PRODUCTO FINAL

El desarrollo del proyecto se divide en tres grandes etapas:

- Construcción de las sala-clúster
- Interconexión de las sala-clúster.
- Pruebas de conectividad del Grid.

3.1 CONSTRUCCIÓN DE LAS SALA-CLÚSTER

Los equipos de las sala-clúster implementadas cuentan con dos sistemas operativos instalados en cada uno de ellos: un sistema operativo utilitario (Windows XP) y el sistema operativo del clúster (Rocks), configurados de tal manera que en el momento del arranque muestre los sistemas operativos instalados y se carga solo el seleccionado por el usuario. Este procedimiento de instalación y arranque se conoce también como doble booteo, dual boot, dual booting o doble arranque.

Este tipo de arranque se utiliza en distintas situaciones como:

- Cuando se necesitan ejecutar aplicaciones que no corren en un mismo sistema operativo.
- En un proceso de aprendizaje de un sistema operativo como por ejemplo GNU/Linux, donde los usuarios desean conocer las características que brinda éste, sin abandonar completamente el uso de otro sistema operativo, como por ejemplo Windows.
- Cuando se desea realizar una migración de datos sencilla.

En este proyecto el arranque dual se utilizó para el acceso a los dos o más sistemas operativos que se necesiten para conformar el sala-clúster, pudiendo los usuarios acceder a las aplicaciones del sistema operativo del clúster, sin dejar atrás las funciones y aplicaciones básicas de los computadores que lo conforman.

El arranque dual tiene varios problemas, de los cuales el más conocido es la incompatibilidad de sistemas de archivos entre los sistemas operativos. Es por ello que cuando se intenta instalar algún sistema diferente al que ya se encuentra en el equipo, en algunos casos esta instalación formateará todo el disco duro con su sistema de archivos (este es el caso de ROCKS), dejando muchas veces inhabilitado el sistema que funcionaba inicialmente; en el procedimiento de la sala-clúster se resuelve este problema.

Una sala-clúster típica cuenta con un servidor Frontend y dos o más nodos como se muestra en la Figura 7. Esta sala-clúster se utilizará para los fines básicos de una sala de informática (Windows y otras aplicaciones) y además se encontrará a disposición de proyectos que requieran alto poder de cómputo.

Para llevar a cabo el proceso de construcción de las dos sala-clúster fue de gran importancia tener en cuenta ciertos factores como:

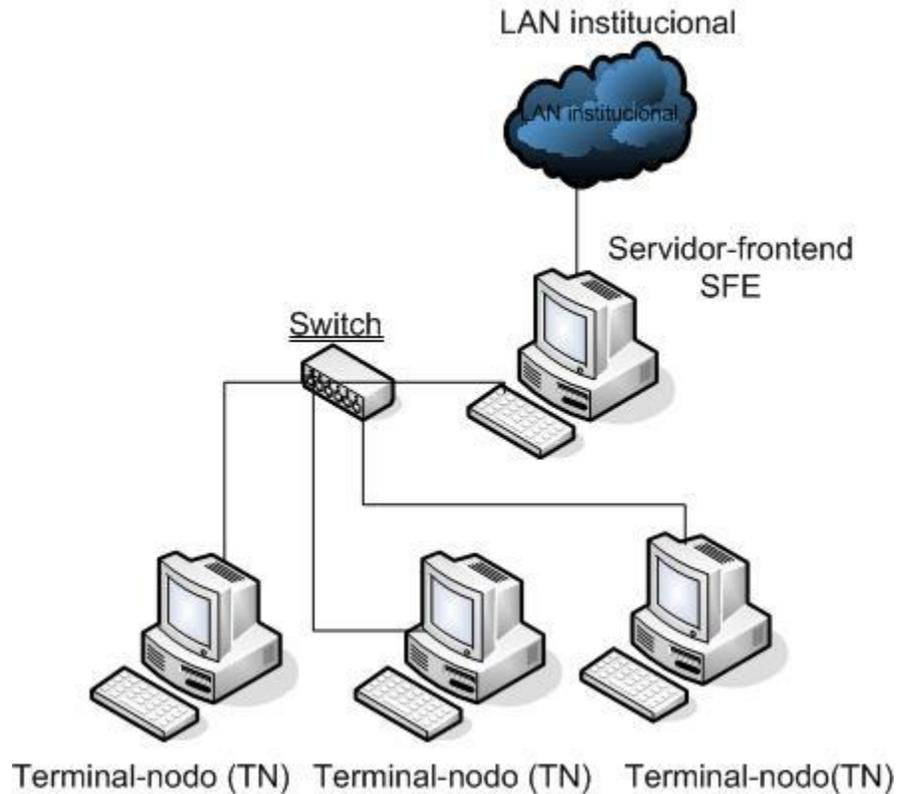
- El tiempo que se requiere para realizar el montaje.
- La disponibilidad y tráfico de la red local (LAN) con la que se va a trabajar.
- La capacidad en hardware de los equipos, sobre todo del equipo designado como servidor Frontend (SFE).

A continuación se muestra información acerca de diferentes periodos de tiempo que se tomaron durante el proceso de creación de una sala-clúster. Este proceso se puede dividir en tres grandes pasos:

- Instalación y configuración del SFE
- Instalación y configuración del terminal nodo prueba (TNP)
- Instalación del resto de terminales nodo de la sala-clúster

Para más información sobre el proceso consultar el **Anexo A** Manual de Instalación de una sala – clúster con Linux - Rocks.

Figura 7. Arquitectura de la Sala-clúster.



Fuente: Autores del Proyecto

3.1.1 Instalación y configuración del SFE

En el momento de instalar el sistema operativo Linux Rocks, la congestión de equipos navegando en la red es un factor que afecta el tiempo de instalación pues durante este proceso se deben descargar componentes de instalación de la Web. Por lo tanto, el tiempo de instalación del SFE varía si se utiliza una LAN compartida o dedicada.

Se hicieron dos pruebas: La primera fue instalar el SFE durante el día, donde se presentaba mayor tráfico en la red y el acceso a Internet era compartido por todos los equipos de la Universidad, afectando la velocidad de transferencia de datos. La segunda prueba fue instalar el SFE durante la noche con un acceso menos compartido y una velocidad de transferencia más alta.

Los resultados fueron:

Tabla 1. Tiempos estimados en la instalación del SFE

Etapas	Tiempo estimado
Instalación durante el día	1 hora y 40 minutos aprox.
Instalación durante la noche	50 minutos aprox.

Fuente: Autores del Proyecto

3.1.2 Instalación y configuración del TNP

En este paso se realiza la inserción e instalación de Rocks en el primer terminal nodo de la sala, a través de la red privada del clúster que se creó utilizando un switch Cisco Catalyst 2950 de 24 puertos 100 BASE-TX; el tiempo utilizado para este proceso es de 40 minutos aproximadamente. Después, se debe realizar la instalación del sistema operativo utilitario de la sala, que en este proyecto es Windows XP. Su instalación sigue el proceso típico de Windows, pero se deben tener en cuenta lo siguiente:

Si se hace una instalación de Windows XP con sus componentes básicos, y de éste se crea una imagen copia que será replicada en los demás nodos de la sala, el periodo de tiempo es menor que si se hace una instalación de Windows XP con otros programas que se requieran para el trabajo cotidiano de la sala.

A continuación se muestran los tiempos tomados para estas dos situaciones:

Tabla 2. Tiempos estimados en la instalación y configuración del TNP

Tipo de Instalación	Tiempo estimado
Instalación de Windows XP con componentes básicos	2h y 40 min aprox.
Instalación de Windows XP con componentes básicos y otros programas *	5 horas aprox.

Fuente: Autores del Proyecto

* Los programas que se instalaron fueron: Dreamweaver 8, MySQL, Packet Tracer 4.0, Java 2 SE Development Kit y ARES.

3.1.3 Instalación del resto de terminales nodo de la sala-clúster

La instalación de los demás terminales nodos, depende de la cantidad de equipos que compartan la red privada del clúster, paso que se realiza después de haber instalado y configurado en TNP. Para ver más información acerca de los pasos que se llevan a cabo en esta etapa, consultar el **Anexo A** “Manual de instalación de una sala-clúster con Linux – Rocks”, en la sección “4.7 Instalación de resto de Nodos”.

Las pruebas que se realizaron suponen una transferencia compartida de red por 2 terminales nodos (TN) a la vez y una transferencia compartida de red por 3 terminales nodos a la vez.

Los resultados fueron los siguientes:

Tabla 3. Tiempos estimados de instalación del resto de Terminales nodos

Tipo de Instalación	Tiempo estimado
Transferencia compartida de red por 2 TN a la vez	2h y 50 min aprox.
Transferencia compartida de red por 3 TN a la vez	5 horas aprox.

3.2 INTERCONEXIÓN DE DOS SALA-CLÚSTER PARA CREAR UN INTRAGRID

El procedimiento realizado para el proceso de interconexión de dos sala-clúster internas, tiene como base la documentación encontrada en la Web acerca de otros proyectos en los cuales se han realizado montajes de Grid, pero con sistemas operativos diferentes a ROCKS, algunos de ellos como UBUNTU y FEDORA. También existe documentación acerca del manejo, instalación y configuración de Globus Toolkit 4, disponible en el sitio Web de Globus: <http://www.globus.org/toolkit>.

De acuerdo a lo realizado, durante la configuración de Globus se deben llevar a cabo dos grandes pasos para el montaje y funcionamiento de la Intragrid.

- Conexión y comunicación de dos servidores en la red.
- Configuración y prueba de componentes de Globus.

Es importante aclarar que en este trabajo solo se pudo culminar satisfactoriamente el desarrollo del primer paso de configuración de Globus.

3.2.1 Conexión y comunicación de dos servidores en la red

Como se mencionó anteriormente existen dos roles de usuario dentro de un entorno Grid: el Rol de Administrador y el Rol de Donante (ver sección 2.4) Teniendo en cuenta esto, se muestran a continuación tres tipos de usuario que se clasifican dentro de estos dos Roles y que participan en el proceso de instalación y configuración de la Intragrid. Estos son:

❖ **Autoridad Certificante (Certificate Authority - CA):** Es un único servidor que toma el Rol de Administrador y tiene numerosas responsabilidades en la Intragrid, de las cuales la más importante, es mantener y proporcionar la seguridad identificando, aceptando y rechazando solicitudes de certificados realizadas por los diferentes tipos de usuarios que pueden acceder a un entorno Grid.

Hay dos tipos de certificados que son usados dentro de un entorno Grid. Uno se utiliza para identificar usuarios dentro de un Grid y otro para identificar servidores Grid.

❖ **Usuario:** Son los usuarios Grid que necesitan un *Certificado de Usuario* para identificarse dentro del Grid y este certificado valida su “nombre de usuario” dentro del Grid y no su estación de trabajo; por ejemplo para un usuario llamado Roberto Carvajal, el certificado tendría la siguiente forma:

“/O=Grid/O=GridPrueba/OU=prueba.dominio.com/CN=Roberto Carvajal”

Donde *prueba.dominio.com*, es el nombre y el dominio al que pertenece el clúster, que en este caso sería: nombreClúster.unab.edu.co

❖ **Servidor:** Son los servidores Grid que necesitan un *Certificado de Servidor* para identificarse dentro del Grid; este certificado registrará el *Fully qualified domain name* del servidor en su certificado. Por ejemplo si el nombre del servidor es: servidor.unab.edu.co, el certificado de usuario se verá de la siguiente forma: “/CN=Service/servidor.unab.edu.co”.

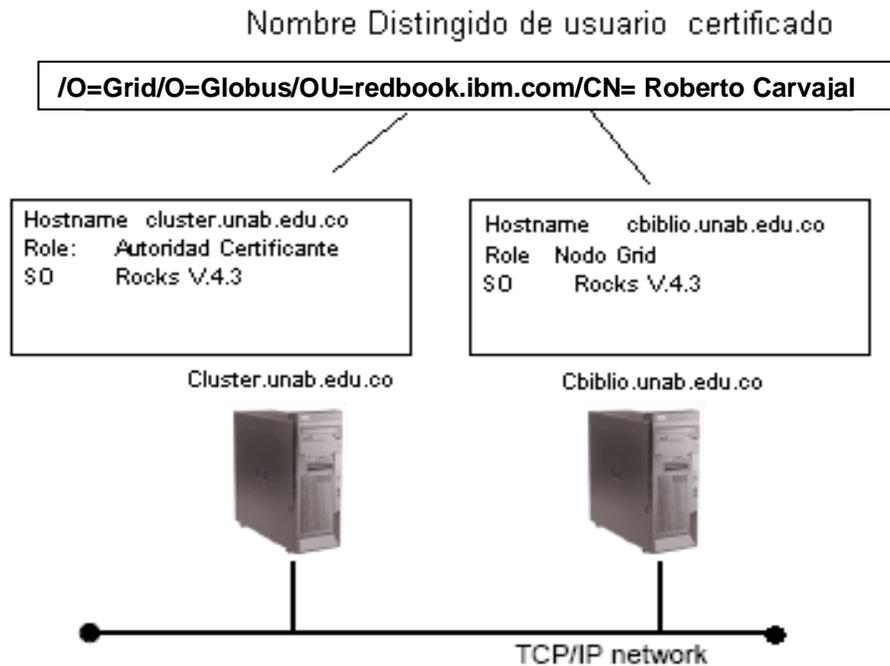
Para trabajar con los certificados de usuario y de servidor, éstos deben ser firmados primero por el CA.

A continuación se explican de manera general los pasos que se llevaron a cabo para lograr la conexión y comunicación de las dos sala-clúster con Globus. Para ver información más detallada consultar el **Anexo B** “Manual de Instalación y Configuración de Globus 4.04 para el montaje de una Intragrid”.

La Figura 9 detalla la estructura utilizada para la interconexión de las dos sala-clúster en la Universidad Autónoma de Bucaramanga a través de la red de área local de la Universidad.

Nota: Es recomendable que el servidor que se designe como Autoridad Certificante sea el equipo que cuente con las mejores condiciones de Hardware, pues como se explicó anteriormente, este es el que se hace cargo de la seguridad y buen funcionamiento de la IntraGrid validando todos los usuarios que quieran acceder a su entorno.

Figura 9. IntraGrid UNAB



Fuente: Autores del Proyecto

Antes de empezar a construir el entorno Grid lo primero que debe hacerse es sincronizar el tiempo del sistema de todas las máquinas de la LAN con un servidor de tiempo NTP y verificar que los puertos 8080, 8443 y 2811 estén abiertos (asignados para componentes de Globus). Para ver más información acerca de los pasos que se deben llevar a cabo, consultar el Anexo B “Manual de instalación y configuración de globus 4.04 t.k para el montaje de una Intragrid”, en las secciones “5.3 Sincronizar todos los host con un servidor NTP” y “5.4 Configuración del firewall”.

El siguiente paso es hacer la solicitud de un paquete de certificados desde el equipo designado como CA. Este paquete será usado en los nodos Grid.

Luego, desde el nodo Grid se copia la clave pública de la Autoridad Certificante a ese equipo y se instala el paquete CA. Para poder hacer uso de algunos servicios de Globus es necesario crear un certificado de host y una clave privada de host de ese equipo. El *certificado de host* creado se debe enviar al CA y éste se encargará de firmarlo y devolverlo de nuevo al nodo Grid.

Hecho esto, se procede a realizar la solicitud de un *Certificado de Usuario* el cual será aquel que tendrá acceso a la IntraGrid. Para esto se creó un usuario con el nombre de “buser” y se realizó desde él la solicitud del Certificado de Usuario la cual debe ser enviada al host CA para que éste la firme y la copie de vuelta en el nodo Grid. Para probar su funcionalidad se puede teclear desde el usuario “buser” el comando “grid-proxy-init”, el cual se encarga de verificar las credenciales del usuario.

Finalmente se debe crear un mapa Grid entre un usuario de Grid autenticado (en nuestro caso buser) y un usuario local del equipo CA. Este paso es un requisito para versiones iguales o superiores a Globus Toolkit 4.

Con este paso se culmina el proceso de conexión y comunicación de los dos servidores con lo que se tiene ya un usuario válido que puede acceder desde un nodo Grid al equipo CA a través de la LAN y por medio de Globus.

3.2.2 Configuración y prueba de componentes de Globus

Globus Toolkit es una colección de componentes de código abierto y a partir de la versión 4 este middleware brinda un soporte de servicios Web basado en implementaciones de muchos de sus componentes los cuales se clasifican dentro de cinco categorías:

- Componentes comunes en Tiempo de Ejecución
- Seguridad
- Administración de Datos
- Servicios de Información
- Gestión de Ejecución.

Dentro de estas categorías de servicios, encontramos 5 componentes importantes para el desarrollo y montaje de la IntraGrid que permiten probar y usar el entorno creado, administrando y permitiendo el paso de datos dentro de la Grid. Estos son:

❖ Componentes comunes en Tiempo de Ejecución

Java WS Core: Este componente contiene herramientas y APIs que implementan estándares de Java y que actúan como componentes base para varios servicios que tiene Globus por defecto. Ver Figura 10.

Figura 10. Relación entre componentes de Java y los servicios que ofrece Globus

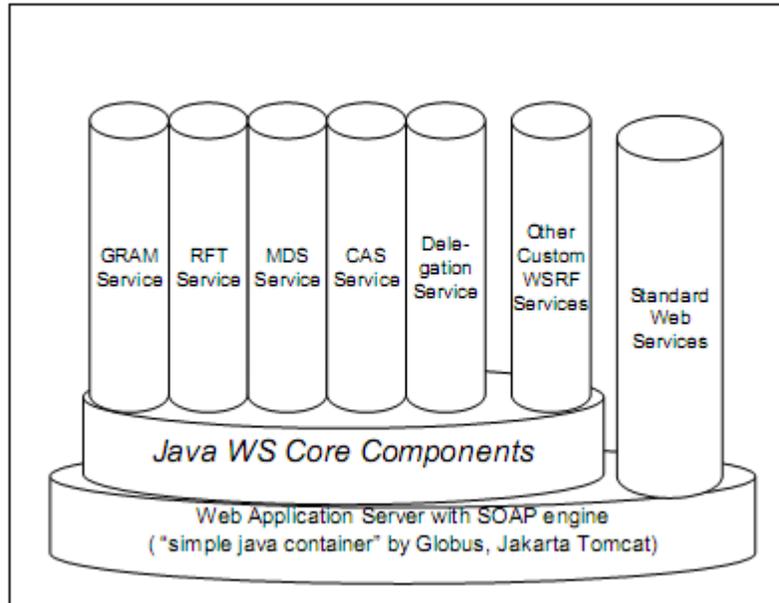


Imagen tomada de "Introduction to Grid Computing: Common runtime components".

❖ Componentes de Administración de Datos

GridFTP: Este componente es muy importante en el entorno Grid porque provee seguridad y transferencia fiable de datos entre los hosts del Grid. Su protocolo se extiende al bien conocido estándar FTP (*File Transfer Protocol* – Protocolo de Transferencia de Archivos) para proveer características adicionales incluyendo soporte de autenticación a través de las Interfaces de Servicio de Grid (GSI).

RFT (*Reliable Transfer Service* – Servicio de Transferencia Confiable): Es un componente de Globus que se encarga de la transferencia de archivos fiable en el Grid, ofreciendo una interfaz de servicio Web para transferencia y eliminación de archivos. Este servicio utiliza GridFTP en el desarrollo de sus procesos.

❖ Componentes de Gestión de Ejecución

WS GRAM: Es un servicio de Grid que provee ejecución remota y administración de estado de trabajos atendiendo solicitudes enviadas por hosts remotos.

- **Monitoreo y descubrimiento de servicios**

MDS4: los servicios MDS se encargan de la recolección, distribución, indexación, almacenamiento y procesamiento de alguna otra información sobre el estado de diversos recursos, servicios y configuraciones del sistema. La información recopilada se usa para descubrir nuevos recursos o servicios o para monitorear el estado del sistema.

Finalmente, es importante tener en cuenta que estos 5 componentes quedaron instalados en el clúster cuando se llevó a cabo la instalación de Java y el middleware Globus 4.04. y es necesario realizar la prueba de funcionamiento y configuración de cada uno de ellos.

Para más información sobre cualquiera de estos componentes consultar el manual de usuario de Globus disponible en el sitio Web <http://www.globus.org>.

3.3 PRUEBAS DE CONECTIVIDAD DEL GRID

Finalmente, la idea después de tener conectados los dos sala-clúster (Intragrid en la UPB e Intragrid en la UNAB) era lograr un Extragrid entre las Universidades, utilizando la RED NACIONAL DE TECNOLOGIA AVANZADA (RENATA), pero debido al cambio de proveedor por parte del gobierno y el poco uso de la Red, no existió conexión entre enero y junio de 2008, tiempo vital para el desarrollo del proyecto de grado, lo que ocasionó, por causas ajenas a los autores, que no se llegará a la conexión del Extragrid. De todas maneras, teniendo conectados los sala-clúster en Intragrid, no debería existir ningún problema en reflejarlo a través de RENATA, cumpliéndose así el principal objetivo del proyecto, de conectar dos clúster.

4. RECOMENDACIONES PARA TRABAJOS FUTUROS

Para trabajos futuros, basándose en los resultados del proyecto de la conexión de dos clúster Intragrid, puede lograrse la conexión en Extragrid, ya sea con el uso de la RED NACIONAL DE TECNOLOGIA AVANZADA (RENATA), o de Internet (InterGrid), entre universidades e instituciones que necesiten computación de alto rendimiento para proyectos de ciencias e ingeniería. Se deben realizar pruebas sistemáticas de rendimiento y estabilidad de la red apuntando especialmente a la determinación de sus propiedades en relación con los retos que ofrece la computación de alto rendimiento en este tipo de redes.

Se deja entonces abierta la puerta y el punto de partida, de lo que podría constituirse en una iniciativa de Grid Nacional de Colombia, sin dejar de lado que la importancia radica no en la infraestructura o plataforma del grid, sino en las aplicaciones y problemas que se pueden resolver con él.

Uno de los retos es poder establecer políticas, compromisos de uso y tarificación en Extragrid en donde varias instituciones colocan sus recursos de cómputo y redes para resolver problemas, que incluso pueden ser no comunes.

5. CONCLUSIONES

- El procesamiento paralelo sobre redes de computadores es una excelente alternativa para aprovechar los recursos de máquinas y redes de datos para aumentar la capacidad de cómputo, con el propósito de resolver problemas complejos que demandan este potencial.
- Como resultado del proyecto realizado se generaron las primeras bases y recomendaciones para el establecimiento, configuración y operación de Intragrids, fundamentales en el proceso de instalación de Extragrids o InterGrids.
- Una arquitectura de cómputo de tipo clúster requiere de un sistema operativo robusto, multiproceso, multiusuario y con características deseables de facilidad de uso; se requiere además de un middleware que actúe entre el sistema operativo y las aplicaciones con la finalidad de proveer al clúster una interfaz única de acceso al sistema (SSI), herramientas de optimización y mantenimiento del sistema y escalabilidad. En este punto juega un papel fundamental el Software Libre, toda vez que se encuentran disponibles en el mercado sistemas operativos y middleware de libre uso. En el desarrollo de este proyecto se utilizó la versión de GNU/Linux Rocks versión 4.3 y el middleware Globus Tool Kit 4.04 (Globus T.K)
- Si una institución inicialmente no cuenta con los recursos necesarios para poder implementar clústeres dedicados, es posible utilizar los recursos existentes, como ya planteó uno de los autores desde hace algún tiempo, mediante el uso de sala-clúster
- Con los recursos existentes de cómputo, se instaló la primer sala-clúster en la Universidad Autónoma de Bucaramanga y se actualizó la versión del clúster de la Universidad Pontificia Bolivariana de Medellín, permitiendo generar un espacio para la capacitación en el manejo, administración y programación sobre plataformas de computación de alto rendimiento.
- En una entidad universitaria, las sala-clúster brindan además de los beneficios de poder de cómputo, beneficios económicos ya que los costos de

implementación son relativamente bajos al utilizar máquinas que ya tiene la institución, utilizando el tiempo ocioso de salas de informática para conformar un sistema multiprocesador virtual que permita realizar tareas fuertes de cálculo en los momentos en que las máquinas no estén siendo usadas, sin perder de vista que estos computadores deben seguir utilizándose para las tareas diarias de la organización.

- Durante el proceso de instalación y configuración de una sala-clúster se identificaron algunos factores críticos que deben tenerse en cuenta y sobre los cuales se concluye lo siguiente:
 - El tiempo de montaje del clúster es variable y depende de la carga que presente en determinados momentos la red a la que pertenecen los equipos que van a conformar el clúster, lo cual indica que debe disponerse de tiempo suficiente para realizar los montajes, ya que aunque el proceso no es complicado, si conlleva un tiempo considerable.
 - Los requisitos de hardware son indispensables para el montaje del clúster. Esto permite concluir que no todas las máquinas que disponga la institución pueden ser utilizadas para conformar un clúster de procesamiento. Existen requerimientos mínimos en cuanto a memoria principal de los nodos, espacio en disco duro, interfaces de red y procesador, entre otros, que deben ser atendidos antes de comenzar una instalación.
 - La sincronización del tiempo en el sistema es uno de los factores más importantes para la conformación de un grid de cómputo toda vez que la generación y envío de certificados a los usuarios del grid requieren de esta característica. Antes de empezar a construir el entorno Grid lo primero que debe hacerse es sincronizar el tiempo del sistema de todas las máquinas de la LAN con un servidor de tiempo NTP (Network Time Protocol) y verificar que los puertos 8080, 8443 y 2811 estén abiertos, ya que serán asignados a componentes de Globus. Para este proyecto, hubo necesidad de instalar un servidor NTP local.
- El middleware (Globus T.K) utilizado para la conformación del grid está referenciado en Internet pero de una manera escasa y generalizada, lo cual implicó adaptar su funcionalidad al sistema operativo Rocks, constituyéndose éste en uno de principales retos para el desarrollo del proyecto.

- Durante el desarrollo del proyecto se hicieron pruebas con varias versiones de Globus T.K lo cual permite a los autores de este trabajo concluir lo siguiente: Si se desea actualizar la versión de Globus T.K sin usar el Roll de Grid de Rocks, es necesario verificar la compatibilidad de las aplicaciones de Java y Apache Tomcat que son indispensables para el buen funcionamiento del middleware y de otros servicios que presta Globus Toolkit.

BIBLIOGRAFÍA

Alonso, José. Programación de aplicaciones paralelas con MPI (Message Passing Interface) [en línea]. UPV/EHU (España). [Citado 28/07/2007] publicado 1 enero de 1997. Disponible en internet <URL:<http://www.sc.ehu.es/acwmialj/edumat/mpi.pdf>>.

Aplicada y Computación, visitada el 28 junio 2007, Disponible en internet <URL:<http://www.csm.ornl.gov/pvm/>>.

Barros Hernández Carlos Jaime. GRID Computing Pasado - Presente y Futuro de la computación de alto rendimiento [online], versión del año 2005. Facultad de Ingeniera de Sistemas de la UNAB (Bucaramanga- Colombia). [Citado 15/08/2007], disponible en internet <URL: <http://fis.unab.edu.co/Noticias/GRIDUNAB.pdf>>.

Cárdenas Montes, Miguel, Desarrollo de Computación Grid Basado en Software Libre. Publicado 29 de mayo de 2007. Actualización 31 de Mayo de 2007. [Citado 28/07/2007], Disponible en internet <URL:<http://www.ceta-ciemat.es/es/files/ComputacionGrid.pdf>>.

EGEE Enabling Grids for E-sciencE, Proyecto EGEE, [citado 21/08/2007], Disponible internet <URL: <http://public.eu-egee.org/intro/>>.

EGEE-II, gLite: La nueva generación de Middleware para el Grid de EGEE, Actualización: 15 de mayo de 2006. [Citado 28/07/2007], Disponible en internet <URL: www.eu-egee.org/sheets/es/es_glite_mb-1.pdf/download>

Ferreira Luis, Berstis Viktors, Armstrong Jonathan, et al. Introduction to Grid Computing with Globus [online]. 2 ed., IBM Redbooks September 2003, [citado 20/06/2007], disponible en internet <URL:<http://www.redbooks.ibm.com/abstracts/sg246895.html?Open>>.

Foster Ian, Globus Toolkit Version 4. Software for Service-Oriented Systems [online]. Publicado en 2005, Department of Computer Science (University of Chicago), [citado 22/08/2007], disponible en internet <URL:<http://www.globus.org/alliance/events/sc05/GT4.pdf>>.

Joseph Jorba Esteve y Remo Suppi Boldrito. Administración avanzada de GNU/Linux [Universidad virtual de Catalunya], 1 ed., Marzo 2004, [citado 19/09/2007], disponible en internet <URL: http://www.uoc.edu/posgrado/matricula_abierta/img/871.pdf>.

Fuentes, Ricardo. UNIDADES CENTRALES MULTIUSUARIO: Número y especialización de los procesadores [online]. D^a Gloria Nistal Rosique versión 2.0, Ministerio de Administraciones Públicas (España). [Citado 20/06/2007], disponible en internet <URL: <http://www.csae.map.es/csi/silice/Hw-cpu4.html>>.

GRID Colombia, Orígenes, Última Actualización Julio 25, 2006, [citado 7/09/2007], < <http://urania.udea.edu.co/grid-colombia/origin.php>>.

Hoffman Forrest M. High Performance Computing: An Introduction to Parallel Programming with Beowulf, [citado 17/09/2007]. Actualización 30 de Mayo 2007. Disponible en internet: URL:<http://www.climatemodeling.org/~forrest/osdj-2000-11/>>.

JORBA, Esteve. Administración Avanzada de GNU /Linux. 1 ed. s.l: Eureka Media, 2004. P 415-432. ISBN 84-9788-116-8.

Joshy Joseph, Mark Ernest y Craig Fallenstein. Evolution of grid computing architecture and grid adoption models, IBM Systems Journal, Volume 43, Number 4, 2004 -Grid Computing, [citado 02/09/2007], disponible en internet <URL: <http://www.research.ibm.com/journal/sj/43-4/joseph.html>>.

LIZARRA, Carlos. Clústers de Linux [en línea]. Universidad de Sonora (México). Actualización: 20 de Marzo de 2003. [Citado 10/09/2007] Disponible en internet <URL: http://clusters.fisica.uson.mx/clusters_de_Linux.htm>.

Lucas, Ricardo. Grid Colombia [en línea]. [Citado 2/09/2007]. Disponible en internet <URL: http://www.ziki.com/en/people/ricardolucas/archives/3254850-grid-colombia?feed_type=10&refresh=1>.

Martínez, Víctor. Tecnología Grid para la Detección de Cáncer de Mama y Cuello Uterino por Medio de Procesamiento de Imágenes En Clarc Conferencia latinoamericana de computación de alto rendimiento (2007: Santa Marta). Bucaramanga. *UIS, Colombia*. 7.

Merkey, Phil. Beowulf History [en línea]. Beowulf Project. [Citado 15/07/2007] Disponible en internet <URL: <http://www.beowulf.org/overview/history.html>>.

Ramiro Mateus Julián y William Brasset David. La globalización: sus efectos y bondades, Fundación Universidad Autónoma de Colombia [online], revista Economía y Desarrollo - Marzo 2002, vol. 1, N° 1, [citado 7/09/2007], <<http://www.fuac.edu.co/revista/M/cinco.pdf>>.

PVM (Parallel Virtual Machine) [en línea]. Departamento de Matemática Aplicada y Computación. [Citado 15/07/2007]. Departamento de Matemática.

Solano Franco, Víctor. En red, la unión hace la fuerza [en línea]. Red de universidades, red de oportunidades. [Citado 29/07/2007] publicado 15 octubre 2006. Disponible en internet <URL: <http://www.universia.net.co/galeria-de-cientificos/noticias-de-la-ciencia-en-colombia/en-red-la-union-hace-la-fuerza/-clusters-y-grids-en-la-his.html>>.

Stallings, William. Organización y Arquitectura de computadores: Procesamiento paralelo. Séptima Edición. Ciudad: Editorial Pearson Prentice Hall, 2006. p 668. ISBN: 8489660824.

Universidad de Alicante, Dpto. de Ciencia de la computación e Inteligencia Artificial, Computación de Altas Prestaciones y Paralelismo, [citado 12/07/2007], disponible en internet <URL: <http://www.dccia.ua.es/cp/>>.

Universidad de Buenos Aires, Departamento de Computación, Formas de Acoplamiento, [citado 27/06/2007] disponible en internet <URL: <http://www-2.dc.uba.ar/materias/so/datos/cap06.pdf>>.