

Experiencias con Globus Toolkit

Martín Chuburu

Miriam Lechner

Javier Echaiz

Jorge Ardenghi

Laboratorio de Investigación de Sistemas Distribuidos (LISiDi)
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur, Bahía Blanca (8000), Argentina
{mic,mtl,je,jra}@cs.uns.edu.ar

RESUMEN

Hoy en día el uso masivo de Internet y las redes de área local (LAN) conforman una gran topología distribuida que hace evidente la necesidad de una nueva tecnología que permita aunar recursos de manera consistente y sencilla. La computación *grid* ha emergido como una solución a este planteo.

Esta nueva tecnología posibilitará explotar el potencial de los recursos que se encuentran diseminados a lo largo de la red, a menudo subutilizados por no poder aprovechar la conectividad existente entre ellos.

En tal sentido Globus Toolkit constituye una de las plataformas capaces de lograr esta conectividad, en base al uso de estándares abiertos.

1. INTRODUCCIÓN

De una manera informal e idealizada podemos pensar en *grid* como la posibilidad de crear la ilusión de una única aunque muy potente computadora virtual en base a una colección de sistemas que comparten varias combinaciones de recursos, donde el usuario no tenga la necesidad de ser consciente de la ubicación física de los mismos.

Grid oculta la naturaleza distribuida de los sistemas actuales facilitando el acceso a los recursos. Así como el protocolo IP es el “pegamento” que mantiene a Internet unida, el uso de estándares abiertos lo es de igual forma para lograr la abstracción en el uso de los recursos.

Actualmente las redes que conforman Internet normalmente están protegidas (por cuestiones de seguridad en accesos) por los

denominados *firewalls*, esto hace que no cualquier tipo de comunicación entre recursos sea posible. Los *firewalls* en general no bloquean las comunicaciones HTTP [FGM+99] por lo que *grid* aprovecha la tecnología *Web Services* como canal para que los recursos puedan vincularse.

Entre los beneficios de aplicar computación *Grid*, como ya mencionamos, se encuentra la explotación de recursos subutilizados, la posibilidad de aprovechar una capacidad masiva de CPUs paralelos (normalmente para uso científico), colaboración entre distintas organizaciones en la elaboración de proyectos conjuntos, acceso a recursos especializados (por ejemplo, supercomputadora), balance de carga (atenuación de picos de cómputo migrando procesos a nodos con menor carga, o suspendiendo procesos de menor prioridad), confiabilidad tanto para la replicación de datos y otros componentes como para implementar mecanismos de votación en aplicaciones instanciadas en forma múltiple para obtener resultados mas confiables [JMF+05].

Actualmente podemos diferenciar distintos tipos de *Grid* basados en el nivel logrado de virtualización de los recursos y de la finalidad por la cual se crearon.

En el primer sentido mencionado, se distinguen los siguientes tipos de topologías:

- ▷ *Intragrid*: Es la expresión mas reducida de *grid*, conformado típicamen-

te por máquinas homogéneas conectadas por una red de alta velocidad que comparten archivos usualmente por Networked File System (NFS). Los límites de la misma se encuentran circunscriptos a una sola organización. Esta configuración se denomina comúnmente con el nombre de *cluster*.

- ▷ *Extragrid*: Expande el concepto anterior al conectar dos o más intragrids dentro de una organización. Una de las características principales es que la seguridad se encuentra distribuida entre los intragrids, será necesario establecer relaciones confiables entre ellos. Además es posible la cooperación entre partes de la organización.
- ▷ *Intergrid*: El grid de una organización puede crecer geográficamente para incorporar recursos específicos provistos por otras organizaciones. La conectividad entre grids en esta instancia se produce a través de WANs por lo que la seguridad cobra un papel absolutamente relevante.

En cuanto a la finalidad de uso, podemos distinguir en principio dos modelos de arquitecturas grid:

- ▷ *Grid computacional*: principalmente se utiliza para aplicaciones de gran escala en las que es necesaria alta potencia de cómputo. El grid recolectará ciclos de CPU libres de las máquinas (dedicadas o no) que lo conforman para lograr esta potencia de cómputo a bajo costo.
- ▷ *Grid de datos*: provee una visión integrada del almacenamiento de datos. Estos medios de almacenamientos se seden en porciones y en forma temporal o permanente al grid. Los usos son variados, yendo desde el acceso a archivos sin necesidad de conocer su

ubicación exacta hasta la unificación de bases de datos para conformar una única base de datos global.

Son variados los tipos de aplicaciones que aprovecharán esta tecnología. Aplicaciones secuenciales pueden ser migradas cuando el sitio en el que se originaron está sobrecargado; aplicaciones paralelas podrán disponer de tantos recursos de cómputo como necesiten.

2. GLOBUS TOOLKIT 4.0.1 (GT4)

Globus Toolkit [FK97] es casi un estándar de facto en computación grid; consiste de varios componentes open source, que pueden ser usados como base para implementar un entorno grid.

Algunos componentes tienen su implementación basada en web services mientras que otros no y por compatibilidad algunos tienen ambas implementaciones.

- ▷ *Common runtime components*: son librerías y herramientas necesarias para ambos tipos de implementaciones. Incluye soporte para Java, C y Python.
- ▷ *Componentes de seguridad*: dada la importancia de la seguridad en grid, GT4 provee varios componentes, entre ellos se encuentra un shell para ejecución remota basado en autenticación GSI (Grid Security Infrastructure) y, un paquete para implementar una autoridad certificante del grid. Incluye:
 - WS authentication and authorization.
 - Pre-WS authentication and authorization.
 - Community authorization service (CAS).
 - Delegation service.
 - SimpleCA.
 - MyProxy.

- GSI–OpenSSH.
- ▷ *Data management components*: provee, entre otras cosas, componentes para transferencia de archivos por medio de distintos protocolos y servicios. Incluye:
- GridFTP: posee como una de sus más importantes características la posibilidad de realizar transferencias entre dos nodos (o servidores) remotos desde un tercer nodo, capacidad conocida como *third-party transfer*.
 - Reliable File Transfer (RFT): provee una interfaz para transferir y eliminar archivos. Recibe pedidos via mensajes SOAP a través de HTTP y utiliza GridFTP. También utiliza una base de datos para mantener una lista de transferencias de archivos y sus estados, y es capaz de recuperar un pedido de transferencia que fué interrumpido.
 - Replica Location Service (RLS): permite acceso a información acerca de la ubicación de datos replicados. Este componente puede mapear múltiples réplicas físicas en un solo archivo lógico.
 - OGSA–DAI: habilita una interfaz para acceder a fuentes de datos grid como base de datos relacionales y repositorios XML, a través de lenguajes de consulta como SQL.
 - Data Replication Service (DRS): provee un sistema para realizar réplicas de archivos en un entorno grid, y registrarlas en RLS.
- ▷ *Servicios de monitoreo y descubrimiento*: se encarga principalmente de recolección, distribución, listado, archivado, y otro tipo de información procesada acerca del estado de varios

recursos, servicios, y configuraciones de sistema. La información recolectada es utilizada tanto para descubrir nuevos servicios o recursos, como para monitorear el estado del sistema. Incluye:

- Index service.
 - Trigger service.
 - Aggregator framework.
 - WebMDS.
- ▷ *Administración de ejecución*: provee herramientas para permitir la administración de ejecución en un entorno grid.
- WS GRAM.
 - Community Scheduler Framework 4 (CSF4).
 - Globus Teleoperations Control Protocol (GTCP).
 - Workspace Management Service (WMS).

3. EXPERIENCIAS PRELIMINARES

Para el desarrollo del cluster, el Laboratorio de Investigación de Sistemas Distribuidos (LISiDi) cuenta con nueve computadoras Pentium IV de 3 GHz con 512Mb de memoria RAM cada una y placas de red de 1 Gbps, conectadas mediante un switch. Además una de ellas, que oficia de server, NAT y firewall, posee dos placas adicionales para conexión al exterior (Internet e Internet–2).

En cada máquina se instaló el sistema operativo GNU/Linux, distribución Fedora Core 5 sobre el cual se instalaron todos los componentes de GT4 mencionados en la sección anterior.

Se verificó el correcto funcionamiento de la implementación de GT4 corriendo Grid Services, incluidos como parte del toolkit. Estos Grid Services hacen, por defecto, uso

de la seguridad provista por la implementación de GSI (Grid Security Infrastructure) de GT4.

Además, se comenzó con el desarrollo de aplicaciones simples extendiendo las aplicaciones incluidas en el toolkit.

Trabajos futuros

Existen varios objetivos basados en estas experiencias preliminares. Se desarrollarán aplicaciones basadas en la infraestructura grid implementada.

La configuración del grid también permitirá desarrollar y ensayar algoritmos distribuidos.

Se formarán recursos humanos en la tecnología grid y se acercará de esta manera una nueva herramienta a otros grupos de investigación.

Lograr la integración con un proyecto grid a nivel nacional con las universidades de La Plata, San Luis, Córdoba y Neuquén y, posiblemente, incorporarnos a un proyecto grid a nivel internacional.

REFERENCIAS

- [JMF+05] Bart Jacob, Michael Brown, Kentaro Fukui y Nihar Trivedi. IBM Red Book: Introduction to Grid Computing. 2005.
- [FK97] Ian Foster y Carl Kesselman. Globus: A Metacomputing Infrastructure Toolkit. The International Journal of Supercomputer Applications and High Performance Computing. Vol 11:2. 1997.
- [Fos95] Ian Foster. Designing and Building Parallel Programs. Addison Wesley. 1995.
- [Les93] Bruce Lester. The Art of Parallel Programming. Prentice-Hall. 1993.
- [FK98] Ian Foster y Carl Kesselman: The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers, ISBN 1558604758, 1998
- [BHF03] Fran Berman, Anthony J. G. Hey y Geoffrey Fox: Grid Computing: Making The Global Infrastructure a Reality, Wiley, ISBN 0470853190, Abr 2003.
- [LB05] Maozhen Li y Mark A. Baker: The Grid: Core Technologies, Wiley, ISBN 0470094176. May 2005.
- [Smi05] Roger Smith: Grid Computing: A Brief Technology Analysis, CTO Network Library, 2005.
- [XS05] Q. Xu y J. Subhlok, Automatic Clustering of Grid Nodes, 6th IEEE/ACM International Workshop on Grid Computing, Seattle, WA, Nov 2005.
- [FK01] I. Foster, C. Kesselman y S. Tuecke: The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International J. Supercomputer Applications, 15(3), 2001.
- [CSK+05] A. Chervenak, R. Schuler, C. Kesselman, S. Koranda y B. Moe: Wide Area Data Replication for Scientific Collaborations. Proceedings of 6th IEEE/ACM International Workshop on Grid Computing (Grid2005), Nov 2005.
- [FGM+99] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach y T. Berners-Lee. RFC 2616 Hypertext Transfer Protocol – HTTP/1.1. Jun 1999.