

# Características de *Grids* vs. Sistemas *Peer-to-Peer* y su posible Conjunción

Jorge R. Ardenghi, Javier Echaiz, Karina M. Cenci, Martín Chuburu, Guillermo Friedrich, Rafael B. García, Lionel Gutierrez, Leonardo D. de Matteis, Juan Pablo Caballero<sup>1</sup>

Laboratorio de Investigación en Sistemas Distribuidos (LISiDi)  
Departamento de Ciencias e Ingeniería de la Computación<sup>2</sup>  
Universidad Nacional del Sur

## Resumen

El proyecto Computación Distribuida de Alto Rendimiento y Disponibilidad que se está desarrollando en el LISiDi ocupa varias líneas de atención como: seguridad, problemas de exclusión mutua en sistemas distribuidos, memoria compartida distribuida, movilidad y *grids*.

Dentro del tema de *grids* se abre una línea de estudio y desarrollo que es tratar de compatibilizar las características de los sistemas *peer-to-peer* con los *grids* y tratar de aprovechar lo mejor de ambos mundos.

## Introducción

La computación *grid* es un paradigma que propone el agregado de computación heterogénea, almacenaje y recursos de red geográficamente distribuidos para proveer un acceso generalizado a sus capacidades combinadas, en conjunto se lo llama simplemente *grids*.

Los *grids* de datos [2,6] tratan primariamente de ofrecer servicios e infraestructura para aplicaciones intensivas sobre datos distribuidos que necesitan acceder, transferir y modificar conjuntos de datos masivos almacenados en recursos de almacenaje distribuidos. Para que el usuario logre el mayor beneficio de la infraestructura son necesarias las siguientes capacidades:

- a) Habilidad para buscar entre los numerosos conjuntos de datos disponibles por el conjunto requerido y descubrir los recursos adecuados para acceder a los datos.
- b) Habilidad para transferir conjuntos de datos de gran tamaño entre recursos en el tiempo más corto posible.
- c) Habilidad para que los usuarios manejen múltiples copias de sus datos.
- d) Habilidad para seleccionar los recursos de computación adecuados y procesar los datos en ellos.
- e) Habilidad para manejar los permisos de acceso a los datos.

---

<sup>1</sup> No hay jerarquía en la aparición de los nombres, todos son integrantes del LISiDi. Los e-mail de acuerdo al orden de aparición son: jra@cs.uns.edu.ar, je@cs.uns.edu.ar, kmc@cs.uns.edu.ar, mic@cs.uns.edu.ar, gfried@frbb.utn.edu.ar, rbg@cs.uns.edu.ar, gl@cs.uns.edu.ar, ldm@cs.uns.edu.ar, cjp@cs.uns.edu.ar.

<sup>2</sup> Tel +54 291 4595135, Fax +54 291 4595136

Los *grids*, en general, son estructurados y se han desarrollado y aún se desarrollan *middlewares* con el propósito de establecer una estandarización como Condor [7] y Globus [3].

Las arquitecturas *peer-to-peer* son la base de la operación de sistemas de computación distribuida como Gnutella [4], Seti@home [8], OceanStore [5], etc.

Estas arquitecturas comparten recursos computacionales (ciclos de CPU, almacenaje, contenido, etc) y en general no requieren la intervención de un servidor centralizado.

Las motivaciones de su uso radican en su habilidad para funcionar, escalar y organizarse a si mismo en presencia de una gran población de nodos, redes y fallas de computadoras sin necesidad de un servidor central y la sobrecarga de su administración.

Tiene características inherentes como escalabilidad, resistencia a la censura y control centralizado y creciente acceso a recursos. La administración está repartida. Aceleran procesos de comunicación y reduce los costos de colaboración por medio de una administración de grupos de trabajo.

Resulta difícil lograr un acuerdo sobre una definición correcta de lo que es un sistema *peer-to-peer*. Lo que persiste es la percepción externa del sistema.

Hay dos características de las arquitecturas *peer-to-peer* que la definen:

- Compartir recursos de computación por intercambio directo.
- Su habilidad para manejar lo instantáneo y la conectividad como norma.

Se adopta la siguiente definición [1] sin perjuicio de cualquier otra:

*Sistemas peer-to-peer son sistemas distribuidos consistentes de nodos interconectados capaces de organizarse a si mismo dentro de topologías de redes con el propósito de compartir recursos tales como contenidos, ciclos de CPU, almacenaje y ancho de banda, capaces de adaptarse a las fallas y acomodarse a poblaciones transitorias de nodos mientras mantienen una aceptable conectividad y rendimiento, sin requerir intermediación o soporte de un servidor o autoridad global centralizada.*

Puede hacerse una clasificación de las aplicaciones *peer-to-peer*:

- Comunicación y colaboración (chat, aplicaciones de mensajería instantánea).
- Computación distribuida (potenciar la computación como seti@home, genome@home).
- Soporte de servicio Internet (aplicaciones multicast, indirección, aplicaciones de seguridad).
- Sistemas de Bases de Datos (PIER, Piazza).
- Distribución de contenido (compartir medios digitales y datos entre usuarios como Napster, Publius, Gnutella, Kazaa, Freenet, Oceanstore, MojoNation, PAST, Chord, Scan, FreeHaven, Groove, Mnemosyne)

La distribución de contenidos se pueden agrupar más específicamente en:

- Aplicaciones *peer-to-peer*.

Según objetivos:

- Intercambio de archivos.
- Publicación de contenidos y sistemas de almacenaje.

- Infraestructura *peer-to-peer*.

- Proveen *frameworks* de aplicación y servicios básicos
- Ruteo y localización
- Anonimato

- Administración de la reputación

Para la localización y ruteo de objetos la red funciona sobre alguna red física, ésta es una red solapada (*overlay network*).

La estructura de la topología y el grado de centralización de la red solapada, el ruteo y los mecanismos de localización que emplea para mensajes y contenidos es crucial para la operación del sistema porque afecta a:

- la tolerancia a las fallas
- automantenimiento
- adaptabilidad a las fallas
- rendimiento
- escalabilidad
- seguridad

Este último punto sobre seguridad los nodos de la red no deben considerarse confiables y no se pueden hacer suposiciones sin considerar su comportamiento. Se analiza el almacenaje y ruteo seguro, dentro de este, el asignamiento seguro de identidades de los nodos, la evasión de nodos maliciosos, el mantenimiento de tablas de ruteo y el reenvío seguro de mensajes

El control de acceso, autenticación y manejo de identidades generalmente son ignorados en los sistemas *peer-to-peer*.

### **Propuesta de trabajo.**

Los sistemas *peer-to-peer* y *grid* son dos propuestas de computación distribuida, ambas concernientes con la organización de recursos compartidos en sociedades computacionales en gran escala.

Los *grids* son sistemas distribuidos que habilitan el uso coordinado en gran escala de recursos distribuidos geográficamente, basados en la persistencia, infraestructuras de servicios estandarizadas, frecuentemente con una orientación al alto rendimiento.

En tanto que un sistema *grid* se incrementa en escala comienza a requerir soluciones para la autoconfiguración, tolerancia a las fallas y escalabilidad para lo cual la investigación en *peer-to-peer* tiene mucho para ofrecer.

Los sistemas *peer-to-peer*, por otro lado, se enfocan en tratar con la instantaneidad, las poblaciones transitorias, la tolerancia a las fallas y la autoadaptación.

Hoy en día, sin embargo, los desarrolladores de *peer-to-peer* han trabajado principalmente en aplicaciones integradas más que en buscar definir protocolos comunes e infraestructuras estandarizadas para interoperabilidad.

En resumen, se puede decir que la computación *grid* apunta a infraestructura pero no a fallas, mientras que *peer-to-peer* apunta a fallas pero no a infraestructura

En adición a esto, la forma de compartir inicialmente apuntada por *peer-to-peer* tiene una limitada funcionalidad, proveyendo una distribución de contenidos global y un espacio de archivos compartidos sin ninguna forma de control de accesos.

En tanto que las tecnologías *peer-to-peer* avancen hacia aplicaciones más sofisticadas y complejas, tales como distribución de contenido estructurado, colaboración por desktop y

computación en la red, se espera que habrá una fuerte convergencia entre *peer-to-peer* y computación *grid*.

El resultado será una nueva clase de tecnologías combinando elementos de ambos, lo cual apuntará a escalabilidad, autoadaptación y recuperación de fallas al mismo tiempo que provee una infraestructura persistente y estandarizada para interoperabilidad.

Los problemas abiertos que se presentan son:

- 1) Diseño de nuevas formas de ubicación, ruteo y estructura de datos, hash distribuidos y algoritmos para maximizar el rendimiento, seguridad y escalabilidad, sea cual fuera arquitectura.
- 2) Anonimato más seguro, más seguridad (más eficiente) y esquemas resistentes a la censura.
- 3) Agrupación semántica de la información (Web Semántica).
- 4) Convergencia de *peer-to-peer* y *grids*.

## **Bibliografía**

- [1] Androutsellis-Theotokis, S., Spinellis, D.; A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Computing Surveys*, #4, vol 36, dec 2004.
- [2] Chervenak, A., Foster, I., Kesselman, C., Salisbury, C., and Tuecke, S. 2000. The Data Grid: Towards an architecture for the distributed management and analysis of large scientific datasets. *J. Net. Comput. Appl.* 23, 3, 187--200.
- [3] Foster, C. Kesselman, The Globus Project: A Status Report, *Proceedings of the Seventh Heterogeneous Computing Workshop*, p.4, March 30-30, 1998.
- [4] Gnutella 2003. The Gnutella web site: <http://gnutella.wego.com>.
- [5] Kubiawicz, J., Bindel, D., Chen, Y., Eaton, P., Geels, D., Gummadi, S., Weatherspoon, H., Weimer, W., Wells, C., and Zhao, B. 2000. Oceanstore: An architecture for global-scale persistent storage. In *Proceedings of ACM ASPLOS*.
- [6] Hoschek, W., Jaén-Martínez, F. J., Samar, A., Stockinger, H., Stockinger, K., Data Management in an International Data Grid Project, *Proceedings of the First IEEE/ACM International Workshop on Grid Computing*, p.77-90, December 17, 2000.
- [7] Litzkow, M., Livny, M. and Mutka, M. Condor-A Hunter of Idle Workstations. In Proc. 8<sup>th</sup> Intl. Conf. on distributed Computing Systems, pages 104-111, 1988
- [8] SetiAtHome 2003. The seti@home project website. <http://setiathome.ssl.berkeley.edu>.
- [9] Stoica, I., Balakrishnan, H., Kaashoek, M. and Karger D., and. 2003. Looking up data in p2p systems. *Comm. ACM* 46, 2 (Feb.), 43-48.
- [10] Venugopal, S., Buyya, R. and Ramamohanarao, K.; A Taxonomy of Data Grids for Distributed Data Sharing, Management and Processing. *ACM Computing Surveys*, #1, vol 38, mar 2006.