

# Sistemas Paralelos sobre Arquitecturas Distribuidas. Cluster / Multicluster / Grid / Cloud Computing.

Armando De Giusti, Marcelo Naiouf, Laura De Giusti, Franco Chichizola, Mónica Denham, Ismael Rodríguez, Adrián Pousa, José E. Pettoruti, Diego Montezanti, Diego Encinas, Luciano Iglesias, Horacio Villagarcía Wanza, Leandro Bertogna, Emmanuel Frati, Victoria Sanz, Fabiana Leibovich .

## Instituto de Investigación en Informática LIDI (III-LIDI)

### Facultad de Informática – UNLP

{degiusti, mnaiouf, ldgiusti, francoch, mdenham, ismael, apousa, josep, dmontezanti, dencinas, li, hvw}@lidi.info.unlp.edu.ar, [mlbertog@uncoma.edu.ar](mailto:mlbertog@uncoma.edu.ar), {fefrati, vsanz, fleibovich}@lidi.info.unlp.edu.ar,

## CONTEXTO

Esta línea de Investigación forma parte de dos de los Subproyectos dentro del Proyecto “Sistemas Distribuidos y Paralelos” acreditado por la UNLP y de proyectos específicos apoyados por CyTED, CIC, Agencia, IBM, Telefónica y Fundación YPF.

En el tema hay cooperación con varias Universidades de Argentina y se está trabajando con la red de Universidades iberoamericanas del proyecto CyTED “Tecnología grid como motor de desarrollo regional”. También el III-LIDI forma parte (desde la Facultad de Informática) de la iniciativa LAGrid (Latin American Grid) de IBM y el proyecto EELA2 (E-infrastructure shared between Europe and Latin America).

## RESUMEN

Esta línea de I/D se enfoca en la especificación, desarrollo y evaluación de sistemas paralelos utilizando esquemas multiprocesador.

Se parte del estudio de las arquitecturas de múltiples núcleos (multicore) y se trabaja también en configuraciones distribuidas como Clusters, Multiclusters, Grids y Clouds.

Los temas fundamentales se relacionan con el estudio de los esquemas multiprocesador

como soporte para diferentes paradigmas de programación paralela, las técnicas de implementación de algoritmos paralelos sobre tales arquitecturas (desde multicores a grid), el desarrollo de modelos para predicción de performance en estas configuraciones, el desarrollo y utilización de middleware para el uso de esquemas multicluster y la resolución de aplicaciones concretas.

**Keywords:** *Sistemas Paralelos. Paradigmas de programación paralela. Multicores. Cluster, Multicluster, Grid y Cloud Computing. Modelos y predicción de performance. Scheduling. Virtualización. Aplicaciones.*

## 1. INTRODUCCION

La investigación en Sistemas Distribuidos y Paralelos es una de las líneas de mayor desarrollo en la Ciencia Informática actual [1][2]. En particular la utilización de arquitecturas multiprocesador configuradas en clusters, multiclusters y grids y soportadas por redes de diferentes características y topologías se ha generalizado, tanto para el desarrollo de algoritmos paralelos como para el de servicios WEB distribuidos [3][4][5]. En la misma línea están los desarrollos incipientes en cloud computing [6] El cambio tecnológico, fundamentalmente a partir de los procesadores multicore, ha im-

puesto la necesidad de investigar en modelos mixtos o híbridos, en los cuales coexisten esquemas de memoria compartida con mensajes [7] [8].

Es importante en este contexto investigar la modelización del comportamiento de esta clase de sistemas paralelos, así como desarrollar nuevos paradigmas para la programación eficiente de aplicaciones.

Por otra parte la heterogeneidad que caracteriza las redes y las arquitecturas Grid, se puede extender también a los procesadores multicore (buscando optimizar la funcionalidad de cada uno de los núcleos), lo que genera restricciones y desafíos para la modelización y también para problemas clásicos como el scheduling y la optimización de los algoritmos [9]. Aquí reaparece el tema del multithreading interno de los procesadores y la posibilidad de su manejo desde la aplicación. [10].

Asimismo los problemas de virtualización crecen en complejidad al considerar sistemas paralelos con componentes multicore. [11].

### 1.1 Definiciones básicas

Un *procesador multicore* surge a partir de integrar dos o más núcleos computacionales dentro de un mismo chip [12]. La motivación de su desarrollo se basa en los problemas de consumo de energía y generación de calor que aparecen al escalar la velocidad de un procesador. Un procesador multicore incrementa el rendimiento de una aplicación si la misma divide su trabajo entre los núcleos del mismo, en vez de utilizar un único procesador más potente.

Un *cluster* es un sistema de procesamiento paralelo compuesto por un conjunto de computadoras interconectadas vía algún tipo de red, las cuales cooperan configurando un recurso que se ve como “único e integrado”, más allá de la distribución física de sus componentes. Cada “procesador” puede tener diferente hardware y sistema operativo, e incluso puede ser un “multiprocesador” [13]. Cuando se conectan dos o más clusters sobre una red tipo LAN o WAN, se tiene un *multicluster* [14]. La configuración

más simple a considerar es la conexión de clusters homogéneos sobre una red LAN o WAN, utilizando un sistema operativo común [15].

Un *Grid* es un tipo de sistema distribuido que permite seleccionar, compartir e integrar recursos autónomos geográficamente distribuidos [16]. Un Grid es una configuración colaborativa que se puede adaptar dinámicamente según lo requerido por el usuario, la disponibilidad y potencia de cómputo de los recursos conectados. El Grid puede verse como un “entorno de procesamiento virtual”, donde el usuario tiene la visión de un sistema de procesamiento “único” y en realidad trabaja con recursos dispersos geográficamente [17] [18].

Algunos autores consideran que un Grid es un “*Cluster de Clusters*”, lo que resulta una definición algo restrictiva pero útil para el desarrollo de un proyecto relacionado con Sistemas Paralelos que migren aplicaciones paralelas de Clusters a Grid.

### 1.2 Aspectos de interés

- En un cluster normalmente se configura una única máquina paralela virtual que puede estar ejecutando una aplicación dedicada. Un Grid permite configurar múltiples máquinas paralelas virtuales para varios usuarios/aplicaciones simultáneas. Sin embargo, este concepto de virtualización se está extendiendo a clusters (incluso a multicores) al aumentar el número de núcleos involucrados. [19]
- Tanto clusters como Grids se basan en procesadores heterogéneos. Sin embargo en Grid esta heterogeneidad se extiende a la red de comunicaciones y al tipo de componentes en cada nodo que pueden ser procesadores (mono o multi core), instrumentos, sensores, etc. La heterogeneidad aparece reforzada al considerar el caso de procesadores multicore que son internamente heterogéneos.
- El middleware necesario para Grid es más complejo que el de los clusters. Fundamentalmente, para configurar la máquina paralela virtual es necesario una

etapa de identificación de recursos físicos y su ubicación, así como la caracterización de los procesos usuarios por prioridades. Estos problemas básicos para el scheduling se están extendiendo a clusters y multicores.

- Las herramientas para el desarrollo de aplicaciones requieren un mayor nivel de abstracción en Grid, por la complejidad y variedad de los múltiples usuarios que pueden utilizar la arquitectura [20].
- La definición de los paradigmas de programación paralela y de los compiladores de lenguajes para sistemas paralelos está fuertemente impactada por el desarrollo de los multicores: el tema de la optimización de los recursos a nivel procesador se vuelve esencial, más allá que la aplicación sea de muy alto nivel (tipo transacción WEB contra un servidor multicore).

Es interesante notar que una estructura de multicluster, visualizada como un *número limitado de clusters dedicados que cooperan en una única aplicación paralela*, es un punto intermedio entre clusters y Grid y requiere algunos servicios especiales en su middleware.

Este tipo de configuración es especialmente interesante para las aplicaciones a investigar desde las Universidades que forman parte de este proyecto.

Más aún cuando cada uno de los clusters puede estar formado por computadoras multicore, homogéneas o heterogéneas.

### **1.3 Sistemas Paralelos sobre arquitecturas distribuidas**

El desarrollo de sistemas paralelos sobre arquitecturas distribuidas (normalmente débilmente acopladas y comunicadas por una red heterogénea) presenta algunos desafíos, entre los que pueden mencionarse:

- La heterogeneidad de las comunicaciones y su costo variable según los nodos a conectar dificulta la asignación óptima de tareas a procesadores y el balance dinámico de la carga.

- La incorporación de procesadores con diferente potencia y velocidad, que en algunos casos aparecen como multiprocesadores con memoria compartida, hace especialmente complejo el problema del balance de carga y muy crítica la localidad de datos y/o procesos.
- Los modelos para predicción de performance son complejos y agregan la incertidumbre del ancho de banda efectivo en el caso de emplear una conexión vía red WAN no dedicada.
- La granularidad óptima a emplear depende de la relación entre potencia de cómputo local y remota. Muchas veces la configuración efectiva de los nodos remotos a utilizar no es conocida a priori.
- El modelo cliente-servidor (paradigma muy empleado en algoritmos paralelos sobre clusters) se torna ineficiente al incrementar el número de nodos. Esto requiere la reformulación de algoritmos que ejecutan sobre clusters y grids. En particular el paradigma “peer to peer” resulta atractivo para investigar, así como esquemas mixtos dentro del mismo sistema paralelo [21].
- Las herramientas de software más generalizadas en clusters están evolucionando para ser utilizables en multicluster y Grid (ej. MPI o PVM) ya que tienen restricciones al tratar de emplearlas en topologías que conectan diferentes redes.
- A su vez, a nivel de procesadores multi núcleo es necesario replantear los temas de scheduling, migración de procesos, comunicaciones sincrónicas/asincrónicas y desarrollo de lenguajes y compiladores que exploten la arquitectura.

## **2. LINEAS DE INVESTIGACION y DESARROLLO**

- ❖ Arquitectura de procesadores multicore y su software de base. Modelos y Caracterización de performance.
- ❖ Sistemas distribuidos y paralelos. Cluster, multicluster, grid y cloud

computing. Modelos y Caracterización de performance.

- ❖ Paradigmas para el desarrollo de algoritmos paralelos sobre arquitecturas multiprocesador distribuidas.
- ❖ Optimización de algoritmos sobre arquitecturas distribuidas heterogéneas.
- ❖ Lenguajes y compiladores para procesamiento paralelo.
- ❖ Modelos de arquitectura híbrida (memoria compartida localizada con mensajes sobre una red). Predicción de performance.
- ❖ Análisis (teórico y práctico) de los problemas de migración y asignación óptima de procesos y datos a procesadores. Métricas de paralelismo.
- ❖ Administración y monitorización de recursos en arquitecturas distribuidas sobre las que se implementan sistemas paralelos. Análisis de rendimiento.
- ❖ Tolerancia a fallas y seguridad en cluster, multicluster, grid y cloud.
- ❖ Optimización de las arquitecturas y procesadores, considerando la migración de algoritmos a hardware.
- ❖ Aplicaciones de tratamiento masivo de datos y simulación paramétrica sobre arquitecturas multiprocesador distribuidas.

### **3. RESULTADOS OBTENIDOS / ESPERADOS**

- ✓ Estudiar el modelo de comportamiento de arquitecturas multicore (homogéneas y heterogéneas), su inserción en sistemas paralelos y la optimización de algoritmos sobre ellas.
- ✓ Simulación de arquitecturas (tipo multicore) y análisis de performance, incluyendo el software de bajo nivel. Estudio de la migración de software a hardware en tales arquitecturas.
- ✓ Modelizar el comportamiento de clusters y multiclusters homogéneos y hete-

rogéneos, sobre redes LAN y WAN, haciendo predicción de performance.

- ✓ Estudiar los temas de scheduling y virtualización a todo nivel de arquitecturas multiprocesador.
- ✓ Estructurar un GRID vinculando Universidades del país y del exterior (está activo)
- ✓ Estudiar predicción de performance de sistemas paralelos sobre arquitecturas distribuidas, realizando estudios experimentales para optimizar los mismos.
- ✓ Desarrollar primitivas de comunicaciones orientadas a cómputo paralelo en multicluster.
- ✓ Estudiar el overhead introducido por el middleware de Grid y como reducirlo para aplicaciones específicas.
- ✓ Investigar aplicaciones concretas de procesamiento masivo de datos y simulación paramétrica sobre Grid.
- ✓ Investigar la especificación e implementación de WEB services eficientes sobre Grid.

### **4. FORMACION DE RECURSOS HUMANOS**

En esta línea de I/D existe cooperación a nivel nacional e internacional. Hay 7 Investigadores realizando su Doctorado en Argentina y 1 en el exterior. Asimismo hay 2 Tesistas de Maestría y 2 de Especialista en curso, y 4 alumnos avanzados están trabajando en su Tesina de Grado de Licenciatura.

En cooperación con las Universidades miembros del proyecto CyTED "Tecnología Grid como motor de desarrollo regional" se ha implementado una Especialización en Cómputo de Altas Prestaciones y Tecnología Grid que se inició en 2007 en la UNLP con Profesores europeos y de las 4 Universidades vinculadas al proyecto.

## 5. BIBLIOGRAFIA

1. Grama A, Gupta A, Karypis G, Kumar V. "Introduction to parallel computing". Second Edition. Pearson Addison Wesley, 2003.
2. Jordan H, Alagband G. "Fundamentals of parallel computing". Prentice Hall, 2002.
3. Dongarra J, Foster I, Fox G, Gropp W, Kennedy K, Torczon L, White A. "The Sourcebook of Parallel Computing". Morgan Kauffman Publishers. Elsevier Science, 2003.
4. Z. Juhasz (Editor), P. Kacsuk (Editor), D. Kranzlmuller (Editor). "Distributed and Parallel Systems: Cluster and Grid Computing" (The International Series in Engineering and Computer Science). Springer; 1 edition (September 21, 2004).
5. Di Stefano M. "Distributed data management for Grid Computing". John Wiley & Sons Inc (29 Jul 2005).
6. Miller Michael Web Based applications that change the way you work online". Que Publishing. USA 2009.
7. Mc. Cool Michael "Programming models for scalable multicore programming" <http://www.hpcwire.com/features/17902939.html> (2007)
8. Lei Chai, Qi Gao, Dhabaleswar K. Panda. "Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core System". IEEE International Symposium on Cluster Computing and the Grid 2007 (CCGRID 2007), pp. 471-478 (May 2007).
9. Suresh Siddha, Venkatesh Pallipadi, Asit Mallick. "Process Scheduling Challenges in the Era of Multicore Processors" Intel Technology Journal, Vol. 11, Issue 04, November 2007.
10. Marek Olszewski, Jason Ansel, Saman Amarasinghe. "Kendo: Efficient Deterministic Multithreading in Software". Architectural Support for Programming Languages and Operating Systems (ASPLOS '09)
11. Lei Chai Qi Gao Dhabaleswar K. Panda. "Understanding the Impact of Multi-Core Architecture in Cluster Computing: A Case Study with Intel Dual-Core System" 7 th. IEEE International Symposium on Cluster Computing and the Grid. CCGRID 07 – Brasil Mayo 2007.
12. Thomas W. Burger. Intel Multi-Core Processors: Quick Reference Guide. [http://cachewww.intel.com/cd/00/00/23/19/231912\\_231912.pdf](http://cachewww.intel.com/cd/00/00/23/19/231912_231912.pdf)
13. Foster I., Kesselman C., Kaufmann M. "The Grid 2: Blueprint for a New Computing Infrastructure". The Morgan Kaufmann Series in Computer Architecture and Design. 2 edition (November 18, 2003).
14. A. E. De Giusti. "Tutorial Grid Computing". June 2006.
15. Grid Computing and Distributed Systems (GRIDS) Laboratory - Department of Computer Science and Software Engineering (University of Melbourne). "Cluster and Grid Computing". 2007. <http://www.cs.mu.oz.au/678/>.
16. Berman F., Fox G., Hey A. "Grid Computing: Making The Global Infrastructure a Reality". John Wiley & Sons (April 8, 2003).
17. The Globus Alliance: <http://www.globus.org>
18. Grid Computing Infocentre: <http://www.gridcomputing.com/>
19. Bertogna M., Grosclaude E., Naiouf M., De Giusti A., Luque E. "Dynamic on Demand Virtual Clusters in Grids" 3rd Workshop on Virtualization in High-Performance Cluster and Grid Computing. VHPC 08 – España. Agosto 2008. (Springer Verlag en prensa).
20. Berman F., Fox G., Hey A. "Grid Computing: Making The Global Infrastructure a Reality". John Wiley & Sons (April 8, 2003).
21. Ghosh S. "Distributed System. An Algorithmic Approach". Chapman & Hall/CRC Computer and Information Science Series.