Arquitectura paralela multifuncional basada en PCs

Multifunctional Parallel Architecture based on PCs

J. A. Corrales, J. Ranilla y S. García Carbajal

Presentamos una arquitectura que convierte un laboratorio, inicialmente concebido para el desarrollo de proyectos fin de carrera, en una sala de computación de altas prestaciones, mediante el empleo de MOSIX y Linux Terminal Server Project

Resumen

En este trabajo presentamos una arquitectura que convierte un laboratorio, inicialmente concebido para el desarrollo de proyectos fin de carrera, en una sala de computación de altas prestaciones, mediante el empleo de MOSIX y Linux Terminal Server Project. El uso de estas herramientas, de distribución gratuita, permite obtener un rendimiento similar al de grandes equipos multiprocesador usando los recursos disponibles, de bajo coste, sin precisar complejas labores de administración.

Palabras clave: Cluster MOSIX, LTSP, Computación de altas prestaciones, Paralelismo.

Summary

This work presents a multi functional architecture that permits the use of a common laboratory initially conceived to be a MsC project development space as a high performance parallel computing machine. The use of MOSIX and Linux Terminal Server Project converts our PC room in a cluster able to reach the speed of expensive parallel machines, and gives us the possibility to use interesting parallel computing libraries like MPI. Additionally, our PhD. students can benefit from the computing power of the cluster.

Keywords: MOSIX's cluster, LTSP, High Performance Computing, Parallelism.

1.- MOSIX. Un S.O. para permitir la formación de clusters de bajo coste

MOSIX está formado por un conjunto de ampliaciones a BSD/OS, que implementa un conjunto de algoritmos adaptativos para la utilización conjunta de recursos, y un mecanismo predictivo de migración de procesos indicado para la computación basada en clusters. La granularidad de la distribución de carga es el *proceso UNIX*. De esta forma, un usuario puede ejecutar aplicaciones paralelas iniciando múltiples procesos en un nodo del cluster, y permitir que el sistema ubique cada proceso en el nodo cuya disponibilidad sea mayor. Las ampliaciones BSD que incorpora MOSIX están implementadas sobre el núcleo del sistema operativo, y son totalmente transparentes al nivel de aplicación. Sus principales características son:

- Migración predictiva de procesos.
- Balanceo dinámico de la carga.
- Mecanismos de gestión de memoria compartida.
- Control descentralizado y autonomía.
- Escalabilidad.

MOSIX [1, 2] está pensado para ser ejecutado sobre clusters basados en máquinas Pentium conectadas por una LAN estándar, o redes de comunicación rápida. Dependiendo del tipo de aplicación, la configuración de un cluster MOSIX varía desde un pequeño conjunto de *pcs* conectados a través de Ethernet, hasta sistemas de gran rendimiento, formados por máquinas de gran potencia conectadas a través de redes Gigabit, de gran escalabilidad. Por ejemplo, un cluster con 32 nodos Pentium a 200 Mhz puede obtener potencias de cálculo próximas a 2 GigaFLOPS.

Utilizando esta herramienta, hemos configurado un cluster formado por 24 máquinas a 600 Mhz ejecutando inicialmente Linux RedHat 7.2, posteriormente modificado para permitir el arranque en

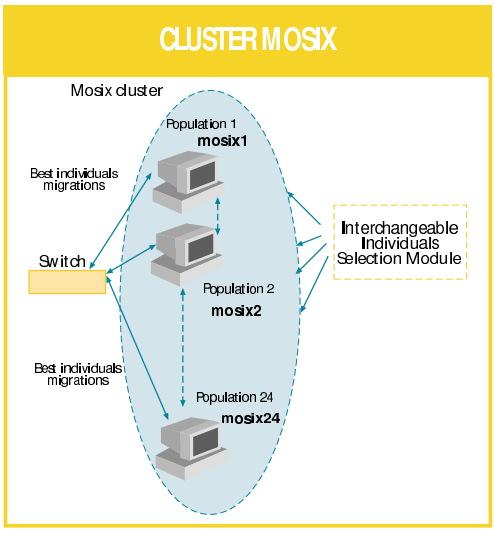
modo MOSIX. Este cluster está siendo utilizado por parte de los investigadores que desarrollan su trabajo en campos como la programación genética, el chequeo de modelos o la visión artificial, debido a los altos requerimientos de potencia computacional que presentan estas disciplinas. Independientemente de la posibilidad de disponer de un cluster de elevada potencia computacional, la sala no ha dejado de utilizarse para su aplicación original como laboratorio de desarrollo de proyectos fin de carrera, al coexistir en cada nodo los sistemas operativos necesarios para cada supuesto. De esta forma, el cluster utiliza al máximo los recursos disponibles en la sala en cada momento.

El procedimiento a seguir para montar un cluster MOSIX es muy sencillo.

1) La ejecución de un fichero *rpm*, permite la selección, en el momento del arranque, de la imagen mosix del sistema operativo Linux presente en la máquina.
2) En cada una de las máquinas que forman el cluster deberá modificarse el fichero *mosix.map*, del directorio */etc*, de forma que contenga la lista de todas las posibles máquinas a incluir en el cluster, en forma de direcciones *ip*.

MOSIX se adapta perfectamente a las necesidades de un departamento en el que los investigadores demandan, cada vez más, la existencia de máquinas cuyas prestaciones son superadas en pocos meses por PCs de mercado de gama alta

En resumen, podemos afirmar que MOSIX se adapta perfectamente a las necesidades de un departamento en el que los investigadores demandan, cada vez más, la existencia de máquinas cuyas prestaciones son superadas en pocos meses por PCs de mercado de gama alta. Hasta el momento, el cluster del que disponemos en el Departamento de Informática de la Universidad de Oviedo ha sido empleado en el desarrollo de dos modelos alternativos de programación genética multinicho aprovechando las capacidades de memoria compartida y posibilidad de creación de múltiples procesos.



2.- Linux Terminal Server Project

Si bien MOSIX crea abstracciones donde los procesos se ejecutan de forma concurrente, y paralela, esta ejecución es independiente. En otras palabras, no soporta un nivel adecuado de acoplamiento y sincronismo para la resolución conjunta de un problema por varios procesos (y procesadores): *Computación Paralela*. En este campo trabajan varios grupos de investigación de nuestro departamento, además de los contenidos teórico/prácticos de asignaturas optativas de último curso de Ingeniería en Informática.

Para permitir la ejecución de algoritmos paralelos, sin olvidar la polivalencia y la multifuncionalidad, se adoptó una solución basada en Linux Terminal Server Project (LTSP en adelante) [3] sobre Linux RedHat. Con LTSP no es necesario instalar software en los nodos de proceso ya que todas las necesidades quedan cubiertas por un equipo que actúa como servidor, del cual obtienen su *kernel* (Diskless Workstations). Además, la administración del servidor, un HP dual en nuestra sala, únicamente precisa la puesta a punto de unos pocos servicios (TFTP, NFS, NIS y DHCP), que por otra

No hay límite al número de equipos que pueden emplearse de esta forma, basta añadir nuevos conmutadores al armario de comunicaciones

parte son muy habituales en este tipo de entornos.
El conjunto así obtenido permite su uso en cualquiera de las tres opciones indicadas. Esto es, en un instante de tiempo concreto coexisten en el laboratorio equipos funcionando como plataformas W2000, otros formando un cluster (MOSIX) y el resto como terminales LTSP para cálculo paralelo.

Según las necesidades se puede reescalar el conjunto sin más que reiniciar en el entorno adecuado los equipos que sean necesarios en ese momento. No hay límite al número de equipos que pueden emplearse de esta forma, basta añadir nuevos conmutadores al armario de comunicaciones.

Referencias

[1] A. Barak; O. La'adan; A. Silo: *Scalable Cluster Computing with Mosix for Linux,* 1999.

[2] A. Barak; O. La'adan: The {MOSIX} Multicomputer Operating System for High Performance Cluster Computing, *Future Generation Computer Systems*, vol. 13, no. 4-5, pp. 361-372, 1998.

[3] J. McQuillan and others: *Linux Terminal Server Project,* http://www.ltsp.org, 2002.

José Antonio Corrales
(ja@lsi.uniovi.es)
José Ranilla
(ranilla@aic.uniovi.es)
Santi García Carbajal
(carbajal@lsi.uniovi.es)
Departamento de Informática
Universidad de Oviedo