

**CLUSTER IMPLEMENTATION OF A PROTOTYPE FOR THE RESOLUTION OF A PARTICULAR PROBLEM****IMPLEMENTACION DE UN CLUSTER PROTOTIPO PARA LA RESOLUCION DE UN PROBLEMA PARTICULAR****MSc. Luz Marina Santos Jaimes, Ing. Sergio Peñaloza R., Ing. Esaú Ramón Cruz Cruz.**

**Universidad de Pamplona**, Grupo de Investigación Ciencias Computacionales  
Programa Ingeniería de Sistemas  
Ciudadela Universitaria. Pamplona, Norte de Santander, Colombia  
Email: {lsantos, rsergiop, ercruz}@unipamplona.edu.co

**Abstract:** This paper shows the importance that owns the implementation of clusters as an option of low cost and reasonable return. Shows the main components as well as the mode of operation of a cluster, the software used, different computer architectures today, as well as ways to address a problem through a cluster. This reflects in turn the use of free software tools as *openmosix*, which provide benefits and ease of making a cluster.

**Keywords:** Cluster, Beowulf, SSI, Openmosix.

**Resumen:** Este artículo muestra la importancia que posee la implementación de clusters como una opción de bajo precio y rendimiento razonable. Se muestran los componentes principales así como el modo de funcionamiento de un cluster, el software a utilizar, las diferentes arquitecturas de cómputo actuales, así como las formas de abordar un problema mediante un cluster. Se refleja a su vez el uso de herramientas de software libre como *openmosix*, las cuales proporcionan beneficios y facilidad a la hora de realizar un cluster.

**Palabras clave:** Cluster, Beowulf, SSI, Openmosix.

## 1. INTRODUCCION

En los últimos años la ciencia y la tecnología han tenido un gran avance en diferentes campos, lo cual ha conllevado a que aparezcan problemas cada vez más complejos, enmarcados dentro de lo que han denominado Computación de Alto Rendimiento, estos por arquitecturas computacionales especialmente diseñadas para su tratamiento y que se caracterizan por su alto costo monetario.

Un número significativo de centros de investigación no tienen los recursos suficientes para adquirir estas supercomputadoras, viéndose en la obligación de recortar y en algunos casos

paralizar sus proyectos, debido a que la capacidad de procesamiento necesaria rompe los límites de los recursos con los que cuentan.

Entre las soluciones se encuentra *Grid Computing* enfocada a sistemas que funcionan en diferentes lugares del mundo, es decir de manera distribuida, por otra parte se encuentran los cluster que funcionan de manera más centralizada y privada, pero debido a la facilidad para armarlos y a los costos relativamente bajos en su implementación se han convertido en la arquitectura computacional más usada de los últimos tiempos.

## 2. GRID COMPUTING

El término *Grid Computing* posee diversas definiciones dadas por personas con gran conocimiento en el tema. Se puede ver como un sistema distribuido compuesto por diverso hardware, como un computador de sobremesa, un supercomputador, un cluster, etc. Estos recursos están unidos por redes de alta velocidad diseñadas especialmente para este tipo de sistemas, generalmente estas cubren diversos países y en ocasiones pueden abarcar continentes.

No todo sistema distribuido puede denominarse *Grid*, se deben tener en cuenta algunas condiciones establecidas en [1], las cuales son de gran aceptación en el entorno científico, estas son:

- Coordinar recursos que no están sujetos a control centralizado.
- Usar estándares abiertos, protocolos e interfaces de propósito general.
- Entregar servicios de calidad no triviales.

La tecnología *Grid* posee un valor importante en la actualidad, puesto que esta permitiendo a comunidades científicas intercambiar conocimientos y recursos, pues al idea es compartir poder de cómputo, lo cual sirve como apoyo a la hora de realizar trabajos que exigen el máximo rendimiento de un sistema y que en algunos casos por su naturaleza sólo pueden ser tratados mediante infraestructuras de gran tamaño como *Grid Computing*.

Un *grid* al poseer recursos de diverso tipo y dispersos a nivel mundial debe contar con una política que asegure que estos serán usados de la mejor manera posible, esto es logrado a través de la creación de organizaciones virtuales formadas por las instituciones que desean compartir sus recursos.

El *grid* es realmente llamativo por su esencia exclusivamente distribuida, que hace que cuente con arquitecturas de diverso tipo, de esta manera se aumentan las capacidades de cómputo, así como también se maximiza el uso de los recursos, produciendo una disminución en los costos.

Esta tecnología a pesar de dar soporte a un gran número de aplicaciones, especialmente de biomedicina, física de partículas, química farmacéutica, industria automotriz entre otras,

posee grandes problemas que aún no han podido ser resueltos en su totalidad por la complejidad que poseen.

El estudio de *Grid Computing* a nivel mundial sigue en avance y aunque Colombia empezó este proceso un tanto tarde respecto a otros países ya se cuenta con una red propia que sirve para compartir recursos y participar en proyectos conjuntos con otros países. Dicha red es denominada RENATA<sup>1</sup>, esta se encuentra auspiciada por el Ministerio de Educación y Colciencias. La siguiente figura muestra la estructura de la red nacional.



Fig. 1. Red RENATA

Tanto en Europa, Norteamérica, Asia y Oceanía se encuentran implementados *grids* de diferente tamaño, que varían según las necesidades de las organizaciones que los componen. A continuación se describe el proyecto de mayor envergadura a nivel mundial.

EEGE<sup>2</sup> (*Enabling Grids for E-Science*): Es la mayor infraestructura *grid* en el mundo, está compuesta por 48 países y tiene como objetivo proveer recursos a las comunidades mundiales de investigación. Entre los proyectos que se están ejecutando en dicho *Grid* se encuentra el LHC (*Large Collider Hadron*), que es un acelerador de partículas de 27 Km de diámetro, ubicado en los límites de Francia y Suiza. El objetivo es analizar entre 10 a 15 años los datos generados por las diversas colisiones producidas por el acelerador.

<sup>1</sup><http://www.renata.edu.co>. Página Oficial de la Red Nacional Académica de Tecnología Avanzada de Colombia.

<sup>2</sup><http://www.eu-egge.org>.

### 3. PROCESAMIENTO PARALELO

Básicamente el procesamiento paralelo busca acelerar la ejecución de un programa dividiéndolo en varios fragmentos que pueden ejecutarse simultáneamente, cada uno en su propio procesador. Un programa que se ejecuta a través de N procesadores podría ejecutarse N veces más rápido que si se utilizara un solo procesador. [2]

En la actualidad existen librerías y lenguajes de programación que permiten codificar un programa para hacer uso intensivo de una máquina de manera paralela, pero es importante resaltar que se deben tener unos conocimientos profundos tanto del sistema como de la aplicación que se pretende paralelizar de manera que tenga un funcionamiento adecuado.

#### 3.1 Teoría de Flynn

En 1966 surgió una forma de clasificar las arquitecturas de modo que se hiciera un tanto más fácil su estudio. Este tipo de computadoras son muy usadas en la actualidad y su característica principal es que los procesadores comparten la misma memoria física. La gran desventaja que poseen es no ser lo suficientemente escalables ya que el bus de interconexión con el que cuentan se satura de manera rápida volviéndose ineficiente a medida que aumenta la cantidad de procesadores.

**3.1.1 SISD (Single Instruction Single Data).** Dentro de este grupo se encuentran las computadoras secuenciales monoprocesador. Una instrucción es ejecutada sobre un solo conjunto de datos en un mismo procesador. Es la arquitectura más básica y dentro de este rango se encuentran las computadoras monoprocesador que no soportan paralelismo.

**3.1.2 MISD (Multiple Data Single Instruction).** Este tipo de arquitectura solo es nombrada de manera teórica, ya que no existe en la actualidad una implementación que logre asemejarse a este tipo de arquitectura.

**3.1.3 SIMD (Single Instruction Multiple Data).** Una instrucción es ejecutada sobre un flujo de datos múltiple en varios procesadores. Los procesadores operan la instrucción al mismo tiempo. Dentro de este conjunto de máquinas están las computadoras vectoriales, estas se caracterizan

por usar agrupaciones o arreglos para tratar sus datos y procesarlos de manera más rápida, algunas de las supercomputadoras actuales pertenecen a este tipo.

**3.1.4 MIMD (Multiple Data Multiple Instruction).** En este tipo de arquitectura un conjunto de procesadores múltiples actúan sobre múltiples datos. Poseen diferentes configuraciones de memoria, lo cual hace que dentro de este tipo de arquitectura exista un subconjunto más que será descrito a continuación:

**SMP (Symetric MultiProcessing):** Este tipo de computadoras son muy usadas en la actualidad y su característica principal es que los procesadores comparten la misma memoria física. La gran desventaja que poseen es no ser lo suficientemente escalables ya que el bus de interconexión con el que cuentan se satura de manera rápida volviéndose ineficiente a medida que aumenta la cantidad de procesadores.

**Multicomputador de Memoria Distribuida:** Este tipo de sistemas se caracteriza por poseer una memoria para cada procesador, lo cual disminuye el problema ocasionado al tener que acceder simultáneamente a una única memoria.

Los procesadores se despliegan en el sistema mediante una red de interconexión. Su principal beneficio es la escalabilidad que proporcionan. A pesar de ser escalables se debe tener en cuenta la red utilizada, pues a medida que crece el sistema esta disminuye su efectividad.

#### 3.2 Clusters

La constante reducción en el costo del hardware ha promovido la masificación de este tipo de arquitecturas que poseen una característica especial y es que pueden ser construidas con hardware convencional lo cual reduce en gran medida los costos ocasionados al comprar piezas exclusivas.

Un cluster básicamente es un conjunto de máquinas unidas por medio de una red de interconexión, que generalmente es de tipo Ethernet, pero que puede variar tanto en precio como en rendimiento según el tamaño del cluster. Cada máquina del cluster posee su propia memoria física, permitiendo a los cluster denominarse también como sistemas multicomputadores.

### 3.2.1 *Reseña Histórica.*

El primer clúster fue implementado en 1994 y surgió como solución a los problemas computacionales que poseían los investigadores para realizar sus predicciones en el proyecto de ciencias de la tierra y el espacio.

Esto llevo a Donald Becker, Thomas Sterling y su equipo de trabajo a unir 16 máquinas Intel 486DX a través de una red tipo Ethernet con velocidad de 10Mbits / seg, algo normal para la época. Este recibió el nombre de *Beowulf*, a partir de ahí toda agrupación de máquinas construida con PC comerciales recibe esta denominación. [4]

### 3.2.2 *Beneficios de un Clúster*

- **Reducción de Costos:** La implantación de un clúster es de gran ayuda, especialmente en instituciones con bajos presupuestos, ya que adquirir una supercomputadora de tipo Cray o similar resulta en el desembolso de una gran suma de dinero, lo cual disminuye la posibilidad de realizar el tratamiento de problemas que posean las instituciones.
- **Incremento en el Rendimiento:** Al poseer una arquitectura de este tipo se podrán tratar problemas que exigen altos recursos, especialmente de CPU y memoria, de una mejor forma y un costo relativamente bajo, impulsando de esta manera diversos proyectos de investigación.
- **Escalabilidad:** Es un factor de importancia en la computación de alto rendimiento, los cluster tienen la capacidad de ser altamente escalables, pues crecen en tamaño según las necesidades que se vayan teniendo, aumentando de esta manera la capacidad de procesamiento.
- **Memoria Distribuida:** En un clúster cada una de las máquinas que lo componen poseen su memoria física propia, lo cual les permite liberarse de problemas como la sincronización que tendría que hacerse en el caso de sistemas configurados físicamente con memoria compartida, es decir de manera centralaa
- **Reutilización de Máquinas:** Este es quizás el principal beneficio, puesto que no se tendrá que recurrir en la mayoría de los casos a adquirir computadoras nuevas, ya que se podrá hacer uso de las máquinas existentes en un laboratorio, universidad, institución, etc.
- **Apoyo al Medio Ambiente:** Aunque suene un tanto raro, al hacer uso de máquinas

convencionales y en algunos casos obsoletas, se evita el tener que arrojar gran cantidad de materiales electrónicos, que son altamente tóxicos y contaminantes para la naturaleza.

- **Uso de Software Libre:** Otro de los factores de gran incidencia en el surgimiento y el auge actual de los clusters se debe a que la mayor parte de software existente es de libre distribución, lo cual reduce en gran magnitud los costos referentes a licencias de software propietario que en algunos casos llegan a superar el costo de los equipos.

### 3.2.3 *Clasificación de los Clusters*

Los cluster presentan su clasificación en base a las tareas sobre las cuales se encuentran orientados. Partiendo de esto se encuentran dos tipos de cluster [5]:

- **Cluster de Alto Desempeño:** Tiene como función el tratamiento de problemas que consumen grandes cantidades de recursos computacionales.
- **Cluster de Alta Disponibilidad:** Su objetivo principal es permitir a una aplicación estar el mayor tiempo activa y ser confiable, para esto se usan dispositivos redundantes dentro del cluster que permitan su actividad continua

### 3.2.4 *Beowulf*

Dentro de los cluster de mayor uso se encuentra este tipo que se caracteriza porque una de sus máquinas actúa como maestra y ordena a las demás que deben hacer, mientras que estas últimas se encargan de realizar las tareas de computo correspondientes.

### 3.2.5 *Sistemas SSI*

Este tipo de sistemas es de importancia en la actualidad, se caracterizan por mostrarse al usuario como un solo sistema capaz de manipular las tareas que le sean impartidas de manera autónoma y descentralizada. Esta es la diferencia básica con los cluster Beowulf. Dentro de este tipo de sistemas se encuentra el software usado en el proyecto, este recibe el nombre de OpenMosix el cual es un parche para el kernel que brinda a un conjunto de máquinas la posibilidad de trabajar conjuntamente. La tarea fundamental de Openmosix es migrar aplicaciones de la manera más uniforme posible, esto lo logra a través de un conjunto de algoritmos diseñados para ese fin.

### 3.2.6 Software necesario para un Clúster

Antes de realizar cualquier configuración se debe tener el parche de openmosix, las herramientas de herramientas de área de usuario para monitorearlo y las herramientas graficas si desea re alizar un seguimiento de manera gráfica.

### 3.2.6 Configuración de OpenMosix

El cluster se implemento sobre un conjunto de máquinas conectadas a través de una red LAN a un switch Fast Ethernet.

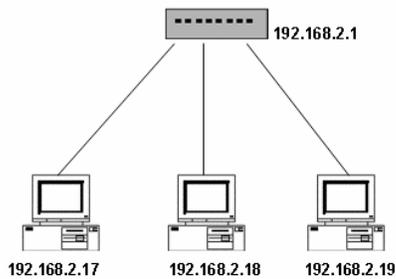


Fig.2. Asignación de Direcciones IP

Sobre cada una de estas fue instalado y compilado el parche de openmosix. Para el reconocimiento de las máquinas dentro de openmosix existen dos formas bien sea a través del demonio de auto configuración o mediante la modificación del archivo de texto /etc/host al cual se agregan cada una de las direcciones IP de las máquinas.

Si se realiza de manera automática con el demonio se procede a digitar en la consola omdiscd y luego se presiona showmap.

Para saber si las máquinas se encuentran funcionando se puede hacer uso de una de las herramientas de usuario más importante mosmon.

Se digita mosmon en la consola, para este caso aparece lo siguiente:

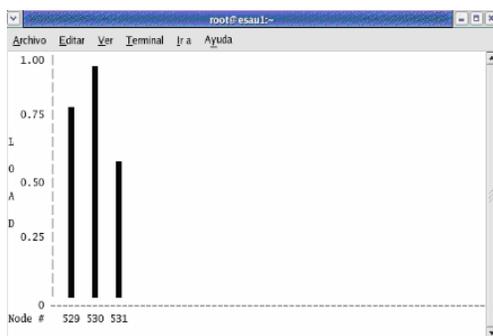


Fig. 3. Monitor Mosmon

### 3.2.7 Pruebas Realizadas

Se realizó una serie de pruebas, pero la que mejores resultados proporciono fue la conversión de formato wav a mp3, esto fue realizado instalando el conversor Bladeenc y convirtiendo las canciones de un CD-ROM.

Openmosix trabaja migrando procesos por tanto la manera de poner a trabajar la aplicación Bladenc fue creando instancias de cada una de las canciones que se deseaban convertir.

Previo a este paso se deben convertir cada una de las canciones que se desean convertir en formato wav, fue utilizado cdparanoia que viene incluido en la distribución Linux. Se digita en consola la siguiente orden: `cdparanoia -vBz`

Luego de tener convertidas las canciones en formato se procede a hacer uso de Bladeenc para la conversión a mp3. Primero se realiza en una computadora y luego en el cluster. Se digita en la consola:

```
bladeenc nombre pista1.wav nombre pista2.wav
```

Para realizar la ejecución en el cluster se deben ejecutar tantos procesos según el número de canciones para este caso ocho:

```
bladeenc nombre pista1.wav & bladeenc nombre  
pista1.wav & ..... ox_08.wav &
```

La siguiente tabla refleja la reducción del tiempo de ejecución de las tareas de compresión de las pistas en el cluster.

Tabla 1. Diferencias de Tiempo de Ejecución

No. Máquinas	Tiempo de Ejecución
1 Computadora	11,0986 minutos
2 Computadoras (Cluster)	10,10323 minutos
3 Computadoras (Cluster)	8,35595 minutos

## 5. CONCLUSIONES

El uso de tecnologías como *Grid Computing* y los cluster de computadoras son una gran oportunidad para nuestras instituciones, ya que permiten fortalecer la investigación y hacer un uso racional

de los recursos de cómputo; por tanto es indispensable inmiscuirse en estas tecnologías que a pesar de su complejidad traen grandes beneficios.

La implementación de un cluster en un ambiente de uso intensivo, requiere de un estudio previo y en profundidad de cada uno de los componentes, desde el hardware a usar, pasando por la tecnología de red utilizada, el software, hasta llegar a la aplicación que hará uso de los recursos del cluster.

El paralelismo puede resultar una muy buena opción a la hora de mejorar el desempeño de una aplicación, pero esto no asegura que se tendrán mejores resultados y en algunos casos puede resultar peor, conllevando a una pérdida de tiempo esencial y por ende a un desperdicio de recursos, por tanto se debe escoger muy bien la infraestructura paralela a utilizar de manera que se reduzcan en la mayor medida posible las dificultades que puedan surgir.

Las herramientas que proporciona el software libre son muy valiosas si se comparan con la robustez que presentan y el ahorro que trae consigo realizar implementaciones basadas en estas.

## 6. RECOMENDACIONES

Un cluster es una excelente alternativa para el tratamiento de problemas que consumen amplios recursos computacionales de manera barata y con un buen rendimiento, además al ser implementados con hardware convencional, pueden usarse máquinas que en algunos casos se creen obsoletas, por esto sería aconsejable contar con un laboratorio enfocado a la utilización de este tipo de arquitecturas.

Para obtener mejores resultados sería adecuado tener completamente dedicadas las máquinas a trabajos concernientes en el cluster.

Es recomendable usar junto a OpenMosix, librerías especiales para realizar paralelismo como MPICH, con el fin de observar el comportamiento presentado por el cluster.

En el tratamiento de problemas algunos no pueden ser resueltos a través de cluster debido a su gran tamaño y a su naturaleza, por tanto se recomienda seguir investigando en tecnologías emergentes como *Grid Computing*.

## REFERENCIAS

- [1]. Foster Ian; Kesselman C; Nick J; Tuecke. S. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. <http://www.globus.org/alliance/publications/papers/ogsa.pdf>
- [2]. <http://www.tldp.org/HOWTO/Parallel-Processing-HOWTO.html>
- [3]. Areli García López, Julián J. Delgado y Salvador Castañeda Metodologías de Paralelización en la Supercomputadora CICESE2000.
- [4]. Hargrove, William W. Hoffman Forrest M., Sterling Thomas. El superordenador Beowulf. Investigación y ciencia. 2001. <http://www.pangea.org/~jbardina/index.htm>
- [5]. <http://clusterfie.epn.edu.ec/clusters/Definiciones/definiciones.html>