Una primera aproximación a la implementación de un clúster para la ejecución de un modelo de predicción climática

Ing. Ronald Gualán
Coordinador

Ronald Marcelo Gualán,
Angel Oswaldo Vázquez,
Oswaldo Francisco Vega
Grupo de Ciencias de la Tierra y del Ambiente,
Universidad de Cuenca

Resumen

Se explora de forma global una de las tecnologías más trascendentes en el campo de la computación a gran escala, los clústeres de computadores. Se parte de una descripción breve de la clasificación existente, dando especial énfasis al tipo Beowulf, que es el de mayor impacto y utilización en el campo de la investigación. Se muestra la necesidad de potencia de cálculo en varias áreas de la ciencia, modelamiento numérico específicamente. Además, se abordan aspectos básicos y necesarios para la implementación de un clúster, tales como la arquitectura de red, el diseño de software y algunas herramientas de administración. Todo esto enfocado hacia la implementación que se está realizando en el Grupo de Ciencias de la Tierra y el Ambiente, con la finalidad de ejecutar un modelo numérico de predicción de clima, WRF.

Palabras clave: Clúster[1], alto rendimiento, Grid, beowulf, computación paralela, MPI.

Introducción

Un clúster puede ser visto como un conjunto de computadores que se comportan como uno solo para el usuario final. En tal virtud, existe una serie de retos que se debe afrontar para conseguir tal comportamiento, siendo los más complicados aquellos relacionados con la diversidad de equipos de hardware, puesto que el objetivo es la utilización de cualquier equipo de cómputo como recurso o nodo. Esta es la idea fundamental de un clúster y una de las razones por la que este tipo de sistema, es una de las opciones más económicas, especialmente en contraste con equipos individuales de la misma gama: servidores, workstation, mainframes, supercomputadores.

En varias ramas de la ciencia, como modelamiento [2] climatológico por ejemplo, se necesita una capacidad de cálculo numérico intensivo. La solución que presenta un clúster es el empleo de dos o muchas máquinas pequeñas (aunque no necesa-

1 El término clúster viene del inglés cluster, cuya traducción al español es conglomerado o conjunto. Se utilizará en el resto del artículo para hacer referencia a ese conjunto o agrupamiento de computadores, ya que el término ha sido acogido, como es común en informática, en el idioma español.

2 El campo de estudio relacionado a la construcción de modelos matemáticos se lo ha catalogado como Computación Científica [1].

riamente) o nodos para que, interconectados mediante una red (Red Rápida de Área Local por ejemplo), puedan servir como un solo sistema que sea capaz de manejar gran cantidad de operaciones con un bajo costo [2], [3], mediante un orquestador de tales nodos llamado Cluster Middleware [4].

Esta tendencia actual para suplir la capacidad computacional requerida constituye la construcción de sistemas más baratos y de propósito general (o al menos un grupo considerable de aplicaciones) en base a componentes fácilmente accesibles localmente (commodities), como son los PCs, de procesador único o múltiple. Además, se pretende que el sistema sea fácilmente expandido, incrementando el número de nodos o la capacidad de los nodos individuales existentes añadiendo memoria y/o procesadores, o mejorando la calidad de la red de interconexión.

Sobre la clasificación de los clústeres, existen básicamente 3 tipos [5]:

- De alto rendimiento (HPC, High Performance Clusters)
- De alta disponibilidad (HAC, High Availability Clusters)
- De alta eficiencia (HTC, High Throughput Clusters)

De estos tres tipos, el clúster a implementar pertenece a los de alto rendimiento. Los clústeres de alto rendimiento, son aquellos en los que se ejecutan tareas que requieren de gran capacidad computacional, grandes cantidades de memoria, o ambos a la vez. El llevar a cabo estas tareas puede comprometer los recursos del clúster por largos periodos de tiempo.

Uno de los enfoques más utilizados es el de un clúster Beowulf, que se ha vuelto muy popular por su relación precio-desempeño, flexibilidad de configuración y actualización, y escalabilidad para proveer un sistema muy robusto [6]. Consiste en la utilización de computadores comúnmente personales, no diseñados precisamente con el fin de utilizarlos para nodos en un sistema paralelo de altas prestaciones, para construir un sistema de cómputo paralelo conectado por una red estándar y que utiliza por lo general software libre u open source . El rápido avance de los microprocesadores, las redes de alta velocidad, y otras tecnologías de componentes han facilitado muchas implementaciones exitosas de este tipo de agrupación [8].
Entre los beneficios que un clúster Beowulf presenta, tenemos los siguientes [9]:

Rentable: se construyen a partir de componentes de productos relativamente baratos, que están ampliamente disponibles.

Al día con las tecnologías: puesto que se utilizan componentes del mercado de masas, es fácil de emplear las últimas tecnologías tanto es software como hardware.

Configuración flexible: se puede adaptar una configuración y asignar el presupuesto de manera óptima para cumplir con los requisitos de rendimiento de las aplicaciones.

Escalabilidad: cuando aumenta el requerimiento de potencia de procesamiento, el rendimiento y el tamaño puede ser fácilmente ampliado añadiendo más nodos de computación.

Alta disponibilidad: cada nodo de cálculo es una máquina individual. El fallo de un nodo de cálculo no afecta a otros

3 En uno de sus ensayos [7], Richard Stallman explica exactamente cuáles son las grandes diferencia entre estos dos conceptos.

nodos o la disponibilidad de todo el clúster.
Compatibilidad y portabilidad: gracias a la estandarización y la amplia disponibilidad de la interfaces de paso de mensajes, como MPI[3] y PVM[4] , la mayoría de las aplicaciones paralelas utilizan estos middlewares estándar. Como un ejemplo práctico, una aplicación paralela con MPI puede ser fácilmente portada de IBM RS/6000 SP2 o Cray T3E a un clúster Beowulf.
Pila de Software para sistemas de computación de alto rendimiento
En un esquema simplificado para manejar un sistema de computación de alto rendimiento, se tienen principalmente tres componentes esenciales que permiten eliminar la complejidad de administración, mientras se provee el software necesario para la ejecución de las aplicaciones, generalmente complejas.
Se tienen como base de esta pila, el sistema operativo (S.O.), cuya elección depende de la aplicación a correr[6] , generalmente GNU/Linux[7] ; el sistema de administración del clúster, cuyo objetivo es la interacción con el usuario y el manejo de computadores físicos independientes y conexiones de red de alta velocidad, permitiendo que los computadores funcionen como un sólo sistema de computación integrado; y las herramientas de programación que consisten de compiladores, librerías y software especial para el desarrollo y prueba de las aplicaciones [10].

Administración del Clúster

La administración correcta de los recursos hardware y software es uno de los aspectos esenciales a la hora de poner en producción un ambiente de computación paralela. Dos elementos cruciales son la calendarización (scheduling) de trabajos (Jobs) y el monitoreo [11], [12].
En cuanto a la calendarización, el estándar MPI [13], por citar el ejemplo más usual, trabaja en el manejo de procesos en varios nodos, sin embargo, se ve limitado a solamente un programa. Se utilizan sistemas de calendarización de procesos como OpenPBS[8] , Condor[9] , Lava[10] y Torque[11] .
Con el fin de tener información de la disponibilidad y operación de los componentes hardware y software, el monitoreo se realiza con algunos sistemas como Cluemon[12] , Nagios[13] , PARMON[14] , Supermon[14] y Ganglia[15] .

Desarrollo de Software

Una vez implementado un sistema de computación paralela, se debe tener claro cómo sacar provecho de ello, cómo escribir programas y qué librerías se necesitan. Para esto se tienen diferentes estándares cuyo objetivo es el diseño e implementación de software que pueda ser ejecutado sobre una plataforma de computación paralela [15]. Entre algunos estándares desarrollados se mencionan PVM y MPI.
PVM (Máquina Virtual Paralela) es una interfaz de paso de mensajes portable que puede ser usado en programas escritos en C, C++ o Fortran y permite que un conjunto de máquinas heterogéneas funcionen como un clúster.
MPI es otro estándar que está siendo usado ampliamente para la programación de

4 http://www.mcs.anl.gov/research/projects/mpi/
5 http://www.csm.ornl.gov/pvm/
6 Se ha tenido una primera aproximación a la instalación de WRF sobre el S.O. Scientific Linux, mostrándose como la primera opción a escoger.
7 http://www.gnu.org/gnu/linux-and-gnu.es.html
8 http://www.mcs.anl.gov/research/projects/openpbs/
9 http://research.cs.wisc.edu/condor/
10 http://lava-scheduler.readthedocs.org/en/latest/
11 http://www.adaptivecomputing.com/products/open-source/torque/
12 https://github.com/cluenet/cluemon
13 http://www.nagios.org/
14 http://www.buyya.com/parmon/
15 http://ganglia.sourceforge.net/

código cuya ejecución se hace paralelo. Existen algunas implementaciones como OpenMPI, MPICH y LAM/MPI que pueden ser utilizadas desde lenguajes de programación como C, C++ o Fortran

Metodología

La metodología empleada para la implementación de un clúster Beowulf para la ejecución del modelo de predicción de clima, WRF , se resume a continuación:

Análisis de requisitos: entre otras cosas, arroja la necesidad de una herramienta computacional con alta capacidad de procesamiento. Ésto para conseguir que el el tiempo de ejecución del modelo de predicción climática WRF[16] disminuya en lo sumo de lo posible. Se busca conseguir lo anterior tratando de emplear varios computadores personales que están disponibles, y de ser posible cualquier computador de la red, que pueda ser utilizado cuando su usuario no lo esté empleando.

Alternativa seleccionada: un sistema tipo clúster de alto rendimiento, Beowulf.

Implementación: uno de los aspecto más importantes para ésto es la selección, entre una gran variedad, de los componentes de software que se van a emplear. Una opción es usar paquetes o suits que integran todo o gran parte del software requerido en una solución. Para este caso, se selecciona una suit de administración de software para clústeres de alto rendimiento llamada Warewulf [9]. Warewulf es un sistema libre que ofrece escalabilidad, configuración, administración, provisión, instalación y monitoreo que integra varias soluciones libres.

Diseño e implementación de la red: los componentes del diseño son: un nodo maestro, que será el front-end del sistema, y será básicamente un PC completo; nodos de cómputo, que constituyen la fuerza computacional del clúster y se refiere a todos los CPUs (no PCs completos) de los que se dispone; y una red local de alta velocidad que permitirá ínter conectar el nodo maestro y los nodos de cómputo.

Instalación de las soluciones de solución adoptada, la mayoría de configuraciones e instalaciones se llevan a cabo en el nodo máster una sola vez.

Instalación del modelo de predicción climática WRF junto con las librerías paralelas, en donde se destaca MPI.

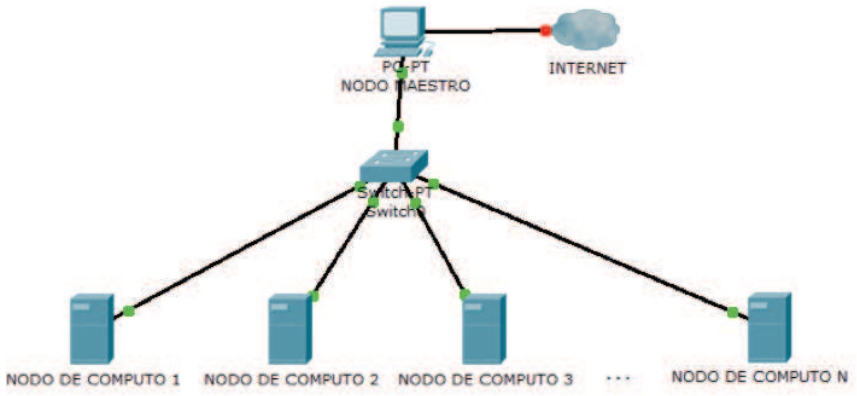
Pruebas: esto se realiza una vez que se ha desplegado, instalado y configurado el sistema, y las aplicaciones a ser empleadas. Para esto se utiliza herramientas de benchmarking que permiten evaluar la capacidad del sistema.

Ejecución de WRF y se evaluación del desempeño del clúster, contrastándolo con una ejecución en serie.

Topología

La topología a utilizar se muestra en la fig.1. El nodo maestro utiliza dos tarjetas de red GigaEthernet, la primera para conectarse a Internet, satisfaciendo la necesidad de descargar paquetes u otras herramientas; la segunda para conectarse a nuestra LAN que enlaza los nodos de cómputo.

16 http://www.wrf-model.org/



software: dadas las características de la
Fig. 1: Topología del clúster

Arquitectura

La Fig. 2 indica la Arquitectura del clúster:

La Capa de aplicación contiene aplicaciones secuenciales y aplicaciones paralelas (desarrolladas para ejecutarse en el clúster).

La capa del middleware contiene lo necesario para ejecutar una aplicación; Librerías GNU y MPI, que permitirán enlazar el entorno de programación paralela de la capa superior con compiladores paralelos MPI, PVM, GNU, JAVA, etc; software y herramientas de administración, que facilitan la gestión de archivos, calendario, balanceo de nodos, etc; imagen del sistema operativo, que ofrece a los usuarios el acceso unificado a los recursos del sistema. El nodo maestro es el encargado de compartirlo a través de la red a cada uno de los nodos de computo.

Dentro de cada nodo enlazado a la red, se tiene una interfaz de red (GigaEthernet) y el software necesario para iniciar, mantener y finalizar la comunicación con el nodo Maestro.

La comunicación entre nodos, se realiza a través de una red de alta velocidad GigaEthernet con ayuda de un Switch.

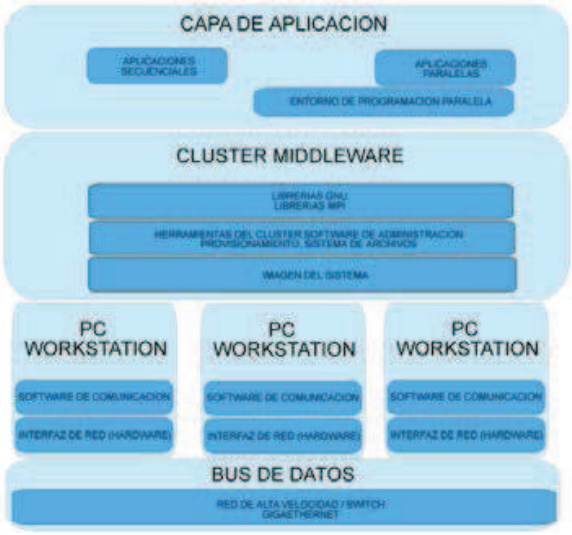


Fig. 2: Arquitectura del clúster

Conclusiones

Gracias a la robustez, capacidad de cálculo intensivo y bajo costo de implementación (motivado por la evolución del software y su libre accesibilidad), los clústeres se han convertido en una solución ampliamente utilizada por una basta cantidad de aplicaciones dentro del campo científico.

Por otro lado, la alta difusión de este tipo de sistemas ha sido impulsado por el amplio desarrollo de aplicaciones paralelas, que a su vez se ha visto simplificado gracias a las estandarización de tecnologías y crecimiento en el desarrollo de software libre.

Para el caso específico del Grupo de Ciencias de la Tierra y del Ambiente, este acercamiento hacia la implementación de un sistema de computación en paralelo, se ha dado para poder aprovechar la capacidad que equipos, quizá no tan potentes, en conjunto (clúster), puedan brindar; además de la experiencia que esto proporciona para que, en el caso de tener a la mano recursos computacionales de mayor capacidad, se pueda implementar un sistema de mayor gama.

Esta primera implementación del clúster dentro del Grupo CTA, busca fomentar la investigación científica en áreas que requieren alta capacidad computacional, optimizando la utilización de los recursos de hardware disponibles, para contar con una herramienta potente y capaz de solventar, en lo sumo de lo posible, esas necesidades.

Según las primeras pruebas realizadas hasta el momento, el rendimiento presentado por el clúster de prueba para la ejecución de algoritmos programados para su ejecución en paralelo, es casi proporcional al número de núcleos de los nodos de cómputo que se han empleado. Resulta además sencillo incorporar nuevos nodos de cómputo al clúster, ya que basta con conectarlos a la red y habilitar su arranque en red.

Agradecimientos

Este estudio está siendo posible gracias al desarrollo de los proyectos.

SENESCYT PIC-11-728 y SENESCYT PIC-11-715 dentro del Grupo de Ciencias de la Tierra y del Ambiente de la Dirección de Investigación (DIUC) de la Universidad de Cuenca. Los autores agradecen al apoyo logístico que se está brindando para la conclusión del proyecto.

Bibliografía

[1] G. Em Karniadakis and R. M. Kirby Ii, Parallel Scientific Computing in C++ and MPI. Cambridge University Press, 2003.

[2] A. Lazalde, "Historia de la Tecnología: Clúster Beowulf, la supercomputadora de los pobres," ALT1040. 09-Nov-2011.

[3] "COMPUTACION CON CLUSTERS DE COMPUTADORES PERSONALES," 15-Oct-2012. [Online]. Available: http://clusterfie.epn.edu.ec/clusters/. [Accessed: 15-Oct-2012].

[4] clusterbuilder.org, "Cluster Middleware," LinuxHPC.org/Cluster Builder 1.3. [Online]. Available: http://www.clusterbuilder.org/software/cluster-middleware.php. [Accessed: 14-Oct-2012].

[5] S. Taherian, "Open Source Real-Time OS (RTEMS) on SCI based Compute Clusters," University of Dublin, Dublin, 2003.

[6] T. Sterling, Beowulf Cluster Computing with Linux. MIT Press, 2001.

[7] R. M. Stallman, "Why Open Source Misses the Point of Free Software," in Free Software, Free Society: Selected Essays of Richard M. Stallman, 2nd ed., vol. 1, 1 vols., Boston, MA 02110-1335: Free Software Foundation, 2010, pp. 83–89.

[8] J. Hsieh, "High-Performance Computing with Beowulf Clusters | Dell," High-Performance Computing with Beowulf Clusters, 02-Oct-2012. [Online]. Available: http://www.dell.com/content/topics/global.aspx/power/en/ps2q00_beowulf?c=us&l=en&cs=555. [Accessed: 02-Oct-2012].

[9] Warewulf Project, "Warewulf web page," Warewulf, 27-Sep-2012. [Online]. Available: http://warewulf.lbl.gov/trac. [Accessed: 27-Sep-2012].

[10] Appro, "Software Suite for Easy HPC Cluster Management | Appro," APPRO Supercomputer Solutions. [Online]. Available: http://www.appro.com/products/software/hpc_software_stack/. [Accessed: 14-Oct-2012].

[11] Escuela Politécnica Nacional, "Clusters :: Administración del Cluster." [Online]. Available: http://clusterfie.epn.edu.ec/clusters/Definiciones/definiciones4.html. [Accessed: 15-Oct-2012].

[12] IBM, "High performance Linux clustering, Part 2: Build a working cluster," IBM, 27-Oct-2005. [Online]. Available: http://www.ibm.com/developerworks/linux/library/l-cluster2/. [Accessed: 15-Oct-2012].

[13] B. Barney, "Message Passing Interface (MPI)." Lawrence Livermore National Laboratory.

[14] M. J. Sottile and R. G. Minnich, "Supermon: A High-Speed Cluster Monitoring System," In Proc. of IEEE Intl. Conference on Cluster Computing, pp. 39–46, 2002.

[15] J. Greenseid, "Linux Clustering Software," free code, 01-Jun-2002. [Online]. Available: http://freecode.com/articles/linux-clustering-software. [Accessed: 15-Oct-2012].

"Nuestra recompensa se encuentra en el esfuerzo y no en el resultado. Un esfuerzo total es una victoria completa".

Gandhi