COMPUTACION DE ALTO RENDIMIENTO MEDIANTE PROGRAMACION DE PASO DE MENSAJES - MPI

Autor

ZULAY VIVIANA MEDINA VILLAMIZAR

Director

JOSE ORLANDO MALDONADO BAUTISTA

DEPARTAMENTO DE INGENIERÍAS ELÉCTRICA ELECTRÓNICA SISTEMAS Y TELECOMUNICACIONES

FACULTAD DE INGENIERÍAS Y ARQUITECTURA

INGENIERIA DE SISTEMAS

UNIVERSIDAD DE PAMPLONA

PAMPLONA, FEBRERO de 2016

1. INFORMACIÓN GENERAL DEL TRABAJO DE GRADO

Título: COMPUTACION DE ALTO RENDIMIENTO MEDIANTE PROGRAMACION DE PASO DE MENSAJES - MPI							
Nombre Autor: ZULAY VIVIANA MEDINA VILLAMIZAR					C.C. : 1094269231 FIRMA:		
E-mail: zuvimevi1992@hotmail.com					Teléfono: 3138919481		
Lugar de Ejecución del Proyecto: Pamplona, Norte de Santander							
Duración de Proyecto (en semanas): 16							
Modalidad:							
Investigación	X	Práctica Empresarial*		Diplomado		Docencia	
**Deberá adjuntar carta de aceptación de la empresa o copia del convenio o contrato*							

Objetivo General:

- Identificar las características propias de los problemas que pueden ser optimizados mediante la programación de paso de mensajes MPI

Objetivos específicos:

- Estudiar el paradigma de programación de paso de mensajes MPI
- Análizar y caracterizar los problemas que son suceptibles de ser optimizados mediantes este paradigma.
- Seleccionar problemas tipo dentro de la caracterización realizada y análzar el rendimiento de la implementacion en paralelo frente a la serial tradicional.

Valor Total del Proyecto:	
Aceptación director de la tesis: **NOMBRE:** **FIRMA:** ***Acepto y certifico que revisé el presente anteproyecto antes de firmarlo.***	**Autorización director del programa: SI NO** **FIRMA:**
Jurado 1:	**Firma Jurado 1:**
Jurado 2:	**Firma Jurado 2:**
Jurado 3:	**Firma Jurado 3:**

Sustentación Anteproyecto:

Lugar : _____Fecha: _____ Hora_____

Aprobado ☐ **Incompleto** ☐ **Rechazado** ☐

Observaciones del jurado: _____

_____

_____

_____

_____

_____

Firma Jurado 1: _____ Firma Jurado 2: _____ Firma Jurado 3: _____

Realización de Correcciones

Lugar : _____Fecha: _____ Hora_____

Aprobado ☐ **Rechazado** ☐

Firma Jurado 1: _____ Firma Jurado 2: _____ Firma Jurado 3: _____

2. RESUMEN DEL PROYECTO

Con la realización de este proyecto se busca caracterizar diversos problemas tipo basado en el estudio del estándar MPI dado que es una API desarrollada para facilitar la programación en paralelo y/o distribuida que maneja el paradigma de programación de paso de mensajes, aplicándose a multiprocesadores, multicomputadores, redes, entre otros aplicando todos los conocimientos en la implementación de un Clusters de Alto Rendimiento en el cual se desarrollara parte de las prácticas en la busca de resultados.

2.1. Palabras claves:

Computación de alto rendimiento, programación en paralelo, programación de paso de mensajes MPI, multiprocesadores, multicomputadores, Clusters de Alto Rendimiento.

3. PLANTEAMIENTO DEL PROBLEMA.

En la actualidad se busca realzar el alto rendimiento buscando la efectividad, la eficiencia y la eficacia de diversos procesos realizados, aun así con el pasar de los semestres y las asignaturas estudiadas durante los periodos académicos no se estudiaron tipos de API como OpenMP o MPI las cuales se usan en el desarrollo de aplicaciones a ser ejecutadas tanto en paralelo como con memoria distribuida, por tal razón se tomó la decisión de realizar este proyecto y de esta forma fomentar la investigación en nuevas técnicas y/o tecnologías para el desarrollo de soluciones de diversos problemas tipo, teniendo en cuenta que en la universidad contamos con un laboratorio de alto rendimiento se implementará un Clusters de Alto Rendimiento y de esta forma poder realizar diversas prácticas y con ellas observar el rendimiento de dichas APIs .

3.1. JUSTIFICACION.

Este proyecto se realiza con el fin de incrementar e incorporar nuevos conocimientos en el proceso de creación de desarrollo para soluciones a diversos problemas tipo los cuales se pueden realizar de una forma más eficiente y eficaz utilizando OpenMPI, para obtener el mayor beneficio de la computación de alto rendimiento e implimentado un Clusters de Alto Rendimiento.

4. MARCO TEÓRICO

COMPUTACIÓN DE ALTO RENDIMIENTO

El campo de computación de alto rendimiento (High performance Computing o HPC en inglés) es una herramienta muy importante en el desarrollo de simulaciones computacionales a problemas complejos. Para lograr este objetivo, la computación de alto rendimiento se apoya en tecnologías computacionales como los clusters, supercomputadores o mediante el uso de la computación paralela. La mayoría de las ideas

actuales de la computación distribuida se han basado en la computación de alto rendimiento

CLUSTERS DE ALTO RENDIMIENTO

Un cluster puede estar constituido por dos o más computadores, de los cuales se espera que presente uno o diferentes combinaciones de los siguientes servicios:

- Almacenamiento
- Alta disponibilidad
- Balance de carga
- Alto rendimiento

Almacenamiento

Un cluster de almacenamiento proporciona una imagen de sistema de archivos consistente a lo largo de los servidores en el cluster, permitiendo que los servidores lean y escriban de forma simultánea a un sistema de archivos compartido.

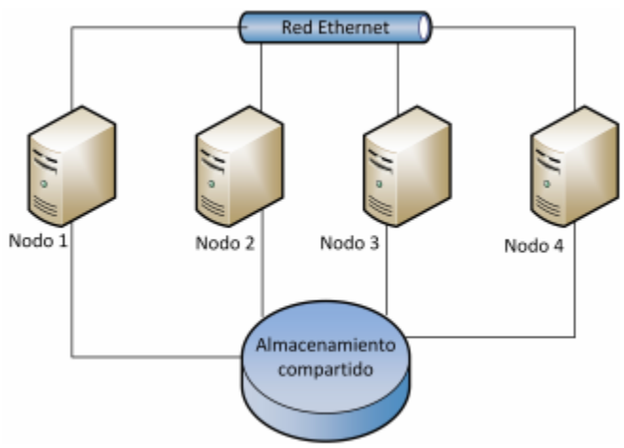

Figura 1: Cluster de Almacenamiento

Un cluster de almacenamiento simplifica la administración de almacenamiento al limitar la instalación de aplicaciones a un sistema de archivos. Asimismo, con un sistema de archivos a lo largo del cluster, un cluster de almacenamiento elimina la necesidad de copias de más de los datos de la aplicación y simplifica la creación de copias de seguridad y recuperación contra desastres.

Alta Disponibilidad

Los cluster de alta disponibilidad proporcionan continua disponibilidad de los servicios a través de la eliminación de la falla por un único elemento y a través del proceso de recuperación en contra de fallos al trasladar el servicio desde el nodo de cluster erróneo a otro nodo completamente funcional.

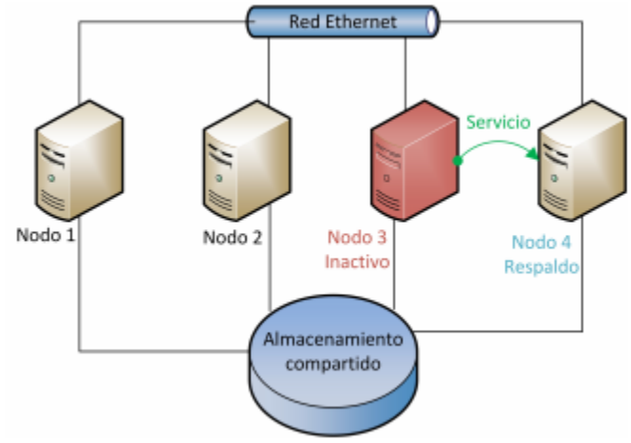

Figura 2: Cluster de Alta Disponibilidad

Generalmente, los servicios en los cluster de alta disponibilidad leen y escriben datos a través de la lectura y escritura a un sistema de archivos montado. Así, un cluster de alta disponibilidad debe mantener la integridad de los datos cuando un nodo recibe el control del servicio desde otro nodo. Los nodos erróneos no son vistos por los clientes fuera del cluster. Los cluster de alta disponibilidad son conocidos también como cluster con recuperación contra fallas.

Balance de Carga

Los cluster de balance de carga responden a peticiones de servicios de red desde diferentes nodos para balancear las peticiones a los largo de los nodos del cluster. El balance de carga proporciona escalabilidad econó- mica porque se puede configurar el número de nodos de acuerdo con los requerimientos de balance de carga.

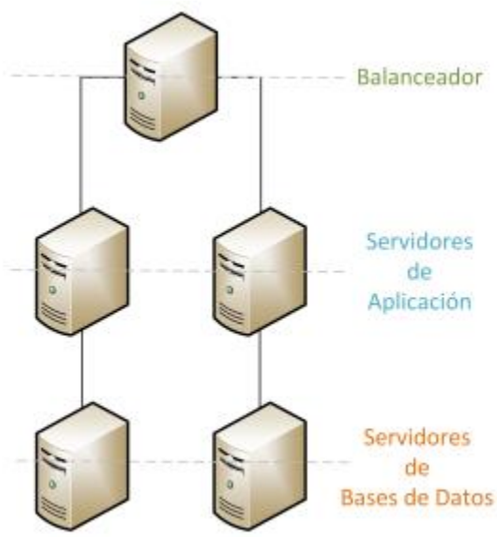

Figura 3: Cluster de Alta Disponibilidad

Si un nodo en un cluster de balance de carga falla, el software de balance de carga detecta la falla y asigna las peticiones a otros nodos en el cluster. Los nodos erróneos en un cluster de balance de carga no son visibles desde los clientes fuera del cluster.

Alto Rendimiento

Los cluster de alto rendimiento utilizan los nodos para ejecutar cálculos simultáneos. Un cluster de alto rendimiento permite que las aplicaciones trabajen de forma paralela,

mejorando así el rendimiento de éstas. Los cluster de alto rendimiento son conocidos como cluster computacionales o computación de red.[1]

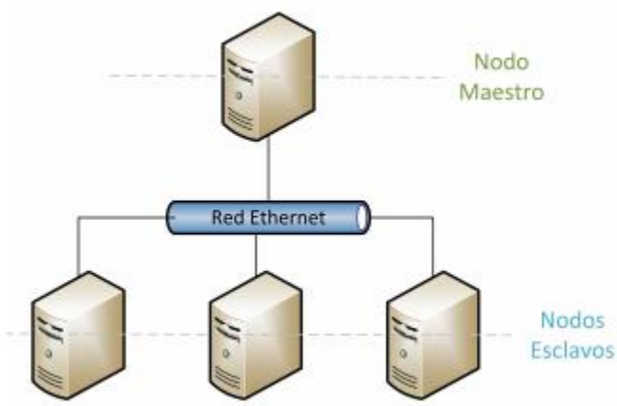

Figura 4: Cluster de Alto Rendimiento

DIFERENTES PARADIGMAS

Ambiente de Programación Paralela Para poder ejecutar programas se necesita de un entorno, en nuestra investigación utilizamos dos diferentes.

1) **MPI**, es un estándar de paso de mensajes que define la sintaxis y la semántica de las funciones de una librería en un lenguaje en específico, en el cual se diseñan programas que explotan a los multiprocesadores o bien a un sistema multinúcleo. MPI tiene soporte para C, C++, C#, Fortran, Ada, Python, OCaml, Java y código ensamblador.

2) **Octave GNU**, es un software libre enfocado a los cálculos matemáticos. Es el equivalente al software privativo de Mathworksr MATLABr. Este software ejecuta scripts que sean compatibles con MATLABr, es decir que es compatible con archivos de extensión .m, es un lenguaje interpretado. Actualmente octave tiene muchas funciones nativas para el procesamiento paralelo, entre algunas se puede nombrar parfor, que básicamente es un bucle for pero que utiliza varios procesadores para realizar la ejecución.[2]

Open MPI (OMPI): La capa de MPI es la capa de abstracción más alto, y es el único expuesto a las aplicaciones. La API de MPI se implementa en esta capa, al igual que todos los mensajes que pasa semántica definida por el estándar MPI.
Desde la portabilidad es una exigencia principal, la capa de MPI soporta una amplia variedad de tipos de redes y protocolos subyacentes. Algunas redes son similares en sus

[1] https://es.wikipedia.org/wiki/Computaci%C3%B3n_de_alto_rendimiento

[2] Journal - IEA 2014, Facultad de Ingeniería – UMSA, Daniel Jiménez, Andres Medina, http://www.revistasbolivianas.org.bo/pdf/jit/n14/n14_a04.pdf

características y abstracciones subyacentes; algunos no lo son.

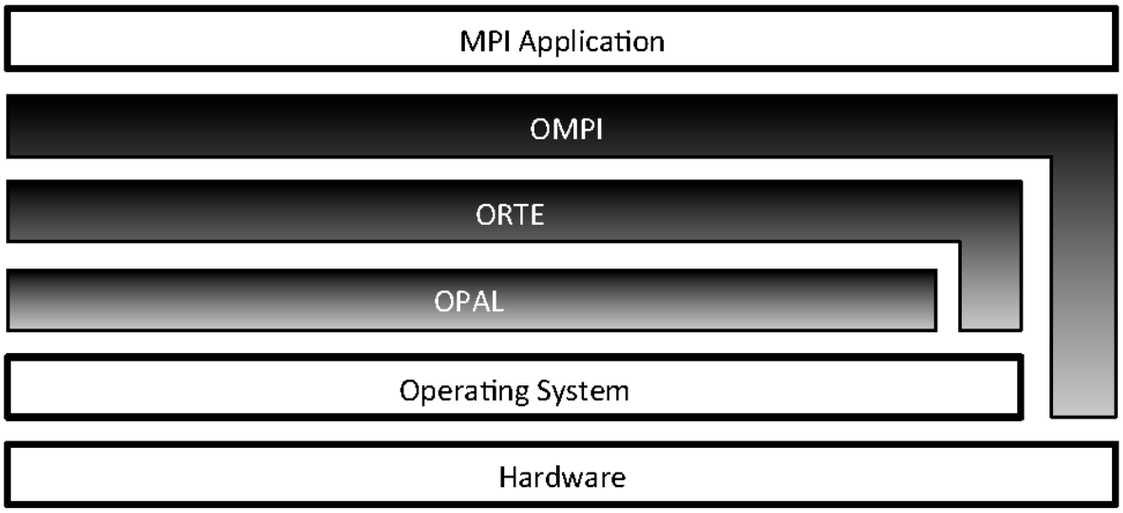

Figura 5: capa de abstracción de vista arquitectónico abierto MPI mostrando sus tres capas principales: OPAL, orte y OMPI

Aunque cada abstracción es en capas en la parte superior de la de abajo, por motivos de rendimiento de las capas orte y OMP pueden pasar por alto las capas de abstracción subyacentes e interactuar directamente con el sistema operativo y / o hardware cuando sea necesario (como se representa en la figura 5). Por ejemplo, la capa OMPI utiliza métodos OS-derivación para comunicarse con ciertos tipos de hardware NIC para obtener el máximo rendimiento de las redes.

Cada capa está construida en una biblioteca independiente. La biblioteca orte depende de la librería OPAL; la biblioteca OMPI depende de la biblioteca orte. La separación de las capas en sus propias bibliotecas ha actuado como una herramienta maravillosa para la prevención de violaciones de abstracción. En concreto, las aplicaciones dejarán de enlace si una capa intenta incorrectamente a utilizar un símbolo en una capa superior. Con los años, este mecanismo de aplicación de abstracción ha salvado a muchos desarrolladores de desenfoque inadvertidamente las líneas entre las tres capas. [3]

5. DELIMITACION

5.1 OBJETIVO GENERAL

- Identificar las características propias de los problemas que pueden ser optimizados mediante la programación de paso de mensajes MPI.

5.2 OBJETIVOS ESPECÍFICOS

[3] http://www.aosabook.org/en/openmpi.html

- Estudiar el paradigma de programación de paso de mensajes MPI
- Análizar y caracterizar los problemas que son suceptibles de ser optimizados mediantes este paradigma.
- Seleccionar problemas tipo dentro de la caracterización realizada y análzar el rendimiento de la implementacion en paralelo frente a la serial tradicional.

5.3. **ACOTACIONES**

Los problemas tipos serán seleccionados de otras áreas de afines a las líneas de los investigadores del grupo CICOM: tratamiento de imágenes, modelado y simulación, tratamiento de imágenes, etc.

6. TAREAS Y CRONOGRAMA DE ACTIVIDADES

1) Estudio de los conceptos básicos de Computación de alto rendimiento
2) Análisis de los conceptos básicos en Programación de paso de mensajes con OpenMPI. Estudio y compilación de de problemas basicos. Programacion multiproceso.
3) Exploración de herramientas de desarrollo, compiladores, IDEs.
4) Implementación de un cluster prototipo. Análisis de herramientas disposibles. Selección de tecnología adecuada. Instalación y puesta en marccha
5) Estudio en profundidad de OpenMPI: funciones de comunicación, envío y recepcion de mensajes
6) Identificación de características de los problemas suceptibles de optimizar
7) Selección de problemas tipo y ejemplos de prueba. codificación, pruebas y análisis de rendimiento.
8) Documentación y compilación del informe.
9) Socializacion de resultados

6.1 Cronograma y descripción de Actividades.

ACTIVIDADES	SEMANAS																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1																		
2																		
3																		
4																		
5																		
6																		
7																		
8																		
9																		

Tabla 6.1. Cronograma y descripción de Actividades

7 RESULTADOS/PRODUCTOS ESPERADOS Y POTENCIALES BENEFICIARIOS

7.1. Impactos esperados a partir del uso de los resultados:

Tabla 7.1. Impactos esperados después de finalizar el proyecto

Impactos	Plazo	Indicador	Supuestos
Sociales	corto	Socialización de resultados	Los resultados serán socializados ante el semillero de investigación, los estudiantes del problema y miembros de grupos afines a grupo CICOM, con el objeto de incetivar el uso y profundización en la computación de alto rendimiento
	Mediano	Publicación de resultadatos	La documentación generada y el reporte técnico estará disponible para consulta en linea desde la web del programa, con el fin de facilitar la continuidad del trabajo.
	Mediano	Desarrollo de proyectos de aplicación	Con la caracterización de problemas y la socialización de los mismos a la comunidad se espera determinar problemas de aplicación en los diferentes grupos de investigación que puedan ser abordados desde el paradigma

BIBLIOGRAFIA

- ✓ http://www.aosabook.org/en/openmpi.html
- ✓ https://es.wikipedia.org/wiki/Computaci%C3%B3n_de_alto_rendimiento
- ✓ Journal - IEA 2014, Facultad de Ingeniería – UMSA, Daniel Jiménez, Andres Medina, http://www.revistasbolivianas.org.bo/pdf/jit/n14/n14_a04.pdf
- ✓ http://digeset.ucol.mx/tesis_posgrado/Pdf/Felix_Ortigosa_Martinez.pdf
- ✓ http://opengear.es/soluciones/par-deploiement/high-performance-computer/