



**FACULTAD DE CIENCIAS E INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA
DE SISTEMAS E INFORMÁTICA**

TESIS

Implementación de un clúster de alto rendimiento para resolver problemas de alta complejidad en el centro de tecnologías de información y comunicación de la Universidad Nacional de Ingeniería

PRESENTADO POR

Huamani Uriarte, Enrique Lee

ASESOR

Delgado Villanueva, Alexi

Los Olivos, 2019



Facultad de Ciencias e Ingeniería

**Escuela Profesional de Ingeniería
de Sistemas e Informática**

**IMPLEMENTACIÓN DE UN CLÚSTER DE ALTO
RENDIMIENTO PARA RESOLVER PROBLEMAS
DE ALTA COMPLEJIDAD EN EL CENTRO DE
TECNOLOGÍAS DE INFORMACIÓN Y
COMUNICACIONES DE LA UNIVERSIDAD
NACIONAL DE INGENIERÍA**

Tesis

**Para optar el título profesional
de Ingeniero de Sistemas e Informática**

Presentado por:

Huamaní Uriarte, Enrique Lee

Asesor:

Delgado Villanueva, Alexi

Lima - Perú

2019

SUSTENTADO Y APROBADO POR LOS SIGUIENTES JURADOS:

Dr Andrade Arenas, Laberiano Matías.

Mg Vilchez Sandoval, Jesus.

**JURADO 1
PRESIDENTE**

**JURADO 2
SECRETARIO**

Ing Lapa Asto, Ulises.

Dr Delgado Villanueva, Alexi.

**JURADO 3
VOCAL**

ASESOR

Dedicatoria

A mis padres Edgar y Margarita, por el amor incondicional que me dieron; a mi hermano Alexis Huamaní Uriarte, por siempre ser la persona quien me ayudó en las peores circunstancias; y a mi pareja Katherine Ayala Mauricio, quien llena de mucha felicidad en mi vida.

Agradecimiento

A mi familia, por su constante apoyo en todo este tiempo; a mi asesor el Dr. Alexi Delgado Villanueva, al Ing. Ulises Lapa Asto y al Doctor José Fiestas Iquira, quienes me apoyaron desde un inicio para que este proyecto sea posible.

Resumen

En la actualidad existen computadoras de última generación de alta características en hardware; pero al realizar procesos algorítmicos de alta complejidad científica demandan un tiempo extenso para ser resuelto, por ese tipo de limitaciones nacen las llamadas supercomputadoras, que son utilizadas en instituciones o empresas en el área de investigación científica, militar, aeroespacial etc. la cual necesitan manejar una gran cantidad de información donde computadoras ordinarias no pueden realizarlas por sus limitadas características. El Centro de Tecnologías de Información y Comunicaciones de la Universidad Nacional de Ingeniería CTIC-UNI, que se dedica a la ciencia tecnológica a nivel nacional como internacional requiere de un clúster de alto rendimiento para poder realizar procesos algorítmicos de alta complejidad, la cual se unirá con varias computadoras mediante una red de comunicación donde estos resolverán un problema y serán distribuidas a cada ordenador para poderlo resolver de manera paralela ahorrándonos tiempo en tener un resultado, el trabajo de investigación se ha realizado gracias a la aprobación del Fondo Nacional de Desarrollo Científico y Tecnológico FONDECYT en poder dar el apoyo económico en aprobar la realización del proyecto que beneficiará a la comunidad científica en la Universidad Nacional de Ingeniería y contribuir al avance tecnológico del país.

Palabras claves: Red computacional, Supercomputadora, Cluster de alto rendimiento, Paralelismo computacional, HPC.

Abstract

At present computers of last generation exist of high characteristics in hardware, but on having realized processes algorithms of high scientific complexity, they demand an extensive time to be solved, for these types of limitations there are born the called supercomputers, which are used in institutions or companies in the areas of scientific, military, aerospace investigation etc. that need to handle large number of information, where ordinary computers cannot realize them for its limited characteristics. The Center of Technologies of Information and Communications of the National University of Engineering CTIC-UNI, which devote themselves to the sciences technological at national level as international, they need of a high performance cluster to be able to realize algorithmic processes of high complexity, several computers will join by means of a network of communication where these will solve a problem in which they will be distributed to every computer to be able to solve it in a parallel way saving ourselves time in having the result, the research work has been realized thanks to the approval of the National Fund of Scientific Development and Technological FONDECYT in being able to give the economic support in approving the achievement of the project that he will benefit to the scientific community in the National University of Engineering and contributing to the technological progress of the country.

Keywords: computer network, supercomputer cluster, unified process, parallelism

CONTENIDO

INTRODUCCIÓN	1
CAPÍTULO I: ASPECTOS GENERALES.....	3
1.1 DIAGNOSTICO DE LA ORGANIZACIÓN.....	4
1.1.1 Datos de la organización	4
1.1.2 Datos de la institución.....	5
1.1.3 Diagnostico estratégico	5
CAPITULO II:.....	7
PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA.....	7
2.1 PLANTEAMIENTO DEL PROBLEMA DE INVESTIGACIÓN	8
2.1.1 Descripción del problema	8
2.1.2 Formulación del problema general	9
2.1.3 Formulación de los problemas específicos	9
2.2 DEFINICIÓN DE LOS OBJETIVOS DE LA INVESTIGACIÓN:	9
2.2.1 Objetivo general:.....	9
2.2.2 Objetivos específicos:	9
2.3 JUSTIFICACIÓN DE LA INVESTIGACIÓN:	10
2.4 ALCANCES Y LIMITACIONES DE LA INVESTIGACIÓN	10
2.4.1 Alcances	10
2.4.2 Limitaciones.....	11
CAPITULO III: FUNDAMENTO TEÓRICO	12
3.1 ANTECEDENTES	13
3.1.1 Internacionales	14
3.1.2 Nacionales.....	16
3.2 MARCO TEÓRICO.....	16
3.2.1 Supercomputadoras	16
3.2.2 Clúster	18
A. Características.....	18
B. Clasificación de clúster	19
a. Alto rendimiento:.....	19
b. Alta disponibilidad:.....	19
c. Alta eficiencia:	20
C. Componentes de un clúster	20
a. Nodos	20
b. Almacenamiento.....	21
c. Sistema operativo.....	21
d. Conexiones de red	22
e. Elementos auxiliares	23
f. Ambiente de programación paralela.....	24

3.3	MARCO METODOLÓGICO	26
3.4	MARCO LEGAL	27
3.4.1.	Ley general.....	27
3.5	ARQUITECTURA COMPUTACIONAL.....	28
CAPITULO IV: DESARROLLO DEL PROYECTO		29
4.1	LEVANTAMIENTO DE INFORMACIÓN.....	30
4.2.1	Planificación del proyecto	30
4.2.2	Recopilación de información.....	30
4.2	IMPLEMENTACIÓN.....	30
4.2.1	Secuencia de la implementación del proyecto.....	30
4.2.2	Diagrama de la arquitectura computacional.....	30
4.2.3	Armado del rack del HPC	31
4.2.4	Especificación de los nodos.....	33
4.2.5	Implementación de la red computacional	33
4.2.6	Finalización de la arquitectura física del clúster.....	34
4.2.7	Determinación de requerimientos de software.....	35
4.2.8	Instalación del sistema operativo del clúster de alto rendimiento. ...	35
4.2.9	Herramientas utilizadas en la instalación.....	35
4.2.10	Configuración del Script psh.....	36
4.2.11	Configuración del keyscan.sh.....	37
4.2.12	Configuración del Script pscp.....	37
4.2.13	Configuración del nodo maestros	38
4.2.14	Configuración del servidor PXE.....	39
4.2.15	Clonación a los nodos esclavos	40
4.2.16	Configuración nodos esclavos.....	41
4.2.17	Configuración del nuevo nodo agregado paso 1	42
4.2.18	Configuración del nuevo nodo agregado paso 2	43
4.2.19	Configuración del nuevo nodo agregado paso 3	44
4.3	PRUEBAS DE RENDIMIENTO.....	45
4.3.1	Pruebas de rendimiento con simulación de n-cuerpos	46
4.3.2	Algoritmo de pruebas de rendimiento	47
4.3.3	Técnicas de distribución de procesos.....	48
4.3.4	Objetivo de pruebas de rendimiento	49
4.3.5	Comportamiento de prueba de rendimiento.....	49
4.3.6	Obtención de operaciones de coma flotante por segundo con pruebas de rendimiento	50
4.3.7	Obtención de tiempos con pruebas de rendimiento.....	51
4.3.8	Metodología en pruebas de rendimiento	52
A.	Rendimiento de gigaflops vs nodos	53
B.	Rendimiento utilizando 1 nodo.....	54
C.	Rendimiento utilizando 2 nodos.....	55

D. Rendimiento utilizando 4 nodos.....	57
E. Rendimiento utilizando 8 nodos.....	58
F. Rendimiento utilizando 16 nodos.....	59
G. Comparación de rendimiento escalable.....	61
H. Tiempos Vs Nodos	62
a. Tiempo vs nodos con 8000 cuerpos.....	62
b. Tiempo vs nodos con 16000 cuerpos.....	63
c. Tiempo vs nodos con 32000 cuerpos.....	63
d. Tiempo vs nodos con 64000 cuerpos.....	64
e. Tiempo vs nodos con 128000 cuerpos.....	64
5.2 IDENTIFICACIÓN DE CUELLOS DE BOTELLAS EN EL HPC.....	65
5.2.1 Cálculo matemático.....	65
CAPITULO V: ANÁLISIS COSTO Y BENEFICIO	69
5.1 ANÁLISIS DE COSTOS	70
5.2.1 Recursos humanos.....	70
5.2.2 Recursos de hardware.....	70
5.2.3 Recursos de software	71
5.2.4 Otros gastos	71
5.2.5 Costo total.....	72
5.2 ANÁLISIS DE BENEFICIO	73
5.2.1 Beneficios intangibles	73
5.3 ANÁLISIS DE BENEFICIOS.....	74
5.3.2 Análisis VAN Y TIR.....	75
CONCLUSIONES	76
RECOMENDACIONES.....	77
REFERENCIAS	78
GLOSARIO	78
ANEXOS	82

Lista de figura

Figura 1. Lugar de la Institución CTIC-UNI	5
Figura 2. Organigrama nominal y funcional de la Institución CTIC-UNI.....	6
Figura 3. Clúster con ordenadores reciclados	8
Figura 4. Supercomputador STAMPEDE TEXAS.....	17
Figura 6. MIMD <i>Instruction Pool</i>	25
Figura 7. Arquitectura del HPC	28
Figura 8. Diagrama computacional de clústeres.....	31
Figura 9. Armario rack.....	32
Figura 10. <i>Cooler</i> de refrigeración.....	32
Figura 11. Ordenadores HO 800G2.....	33
Figura 12. Nodos esclavos y maestro con red de alta velocidad	33
Figura 13. Implementación física del clúster de alto rendimiento	34
Figura 14. Instalación de <i>Script psh</i>	36
Figura 15. Instalación de <i>keyscan.sh</i>	37
Figura 16. Instalación de <i>Script pscp</i>	37
Figura 17. Configuración nodo maestro	38
Figura 18. Configuración de servidor PXE.....	39
Figura 19. Clonación a nodos esclavos	40
Figura 20. Configuración nodos esclavo	41
Figura 21. Configuración de nuevos nodos esclavo paso 1	42
Figura 22. Configuración de nuevos nodos esclavo paso 2	43
Figura 23. Configuración de nuevos nodos esclavo paso 3	44
Figura 24. Número de cálculos de fuerzas $N(N-1) / 2$	46
Figura 25. Algoritmo paralelo del cálculo de fuerzas	47
Figura 26. Función de cálculo de fuerzas en código secuencial.....	48
Figura 27. Diagrama de pruebas de rendimiento	49
Figura 28. Diagrama prueba de rendimiento (Tiempo)	51
Figura 29. Prueba de rendimiento <i>Gigaflops</i> VS Nodos	53
Figura 30. Prueba de rendimiento TIEMPO VS N-CUERPO con 1 Nodo.....	54
Figura 31. Prueba de rendimiento GFLOPS VS N-CUERPO con 1 Nodo.....	55
Figura 32. Prueba de rendimiento TIEMPO VS N-CUERPO con 2 Nodos.....	56
Figura 33. Prueba de rendimiento GFLOPS VS N-CUERPO con 2 Nodos.....	56
Figura 34. Prueba de rendimiento GFLOPS VS N-CUERPO con 4 Nodos.....	57
Figura 35. Prueba de rendimiento TIEMPO VS N-CUERPO con 8 Nodos.....	58
Figura 36. Prueba de rendimiento GFLOPS VS N-CUERPO con 8 Nodos.....	59
Figura 37. Prueba de rendimiento TIEMPO VS N-CUERPO con 16 Nodos.....	60
Figura 38. Prueba de rendimiento GFLOPS VS N-CUERPO con 16 Nodos.....	60
Figura 39. Prueba de rendimiento TIEMPOS VS N-CUERPOS	61
Figura 40. Tiempos vs Nodos utilizando 8000 cuerpos	62
Figura 41. Tiempos vs Nodos utilizando 16000 cuerpos.....	63

Figura 42. Tiempo vs Nodo utilizando 32000 cuerpos	63
Figura 43. Tiempos vs Nodos con 64000 cuerpos	64
Figura 44. Tiempo vs Nodos con 128000 cuerpos	64
Figura 45. Ejemplo de obtención de resultados en pruebas de rendimiento.....	65
Figura 46. Ejemplo de utilización adecuada de nodos	66
Figura 47. Cuello de botella.....	68

Lista de tabla

Tabla 1. Lista de sistemas operativos utilizados en clúster de alto rendimiento.....	22
Tabla 2: Normativas de rendimiento escalable	26
Tabla 3. Mediciones de flops.....	50
Tabla 4. Cantidad de cuerpos por nodos	52
Tabla 5. Estadística de <i>Gigaflops</i> vs Cantidad de nodos	53
Tabla 6. Estadística de Tiempo, N-cuerpos y <i>Gigaflops</i> con 1 nodo.....	54
Tabla 7. Estadística de Tiempo, N-cuerpos y <i>Gigaflops</i> con 2 nodos.....	55
Tabla 8. Estadística de Tiempo, N-cuerpos y <i>Gigaflops</i> con 4 nodos.....	57
Tabla 9. Estadística de Tiempo, N-cuerpos y <i>Gigaflops</i> con 8 nodos.....	58
Tabla 10. Estadística de Tiempo, N-cuerpos y <i>Gigaflops</i> con 16 nodos ...	59
Tabla 11. Tiempos Vs Nodos.....	62
Tabla 12. Resultados 1 Nodo prueba de rendimiento	67
Tabla 13. Resultados 2 Nodos prueba de rendimiento	67
Tabla 14. Resultados 4 Nodos prueba de rendimiento	67
Tabla 15. Resultados 8 Nodos prueba de rendimiento	67
Tabla 16. Resultado 16 Nodos prueba de rendimiento	67
Tabla 17: Costos de personal	70
Tabla 18: Costo de hardware	70
Tabla 19: Costo de software.....	71
Tabla 20:Otros gastos	71
Tabla 21:Costo total.....	72
Tabla 22:Beneficios intangibles.....	73
Tabla 23. Flujo de caja	74
Tabla 24. Análisis VAN y TIR.....	75

Introducción

Para la implementación del proyecto de investigación se ha tenido que recurrir a la aprobación de FONDECYT en la cual CTIC-UNI presentó una propuesta de proyecto la cual fue aprobada y financiada para su realización en el año 2017. El proyecto fue dividido en 4 tesis debido a su complejidad y nivel investigación, el nombre del proyecto en general tiene como nombre instrumentos de diseño y mantenimiento de sistemas de flujo de transporte urbano utilizando computación de alto rendimiento y tecnologías *Big Data*; la documentación para la aprobación del este proyecto fue enviada y aprobada a mediados del año 2017, inicios y finales del año 2019.

El proyecto que se realizó fue la implementación de un clúster de alto rendimiento que es el conglomerado de computadoras interconectadas entre sí mediante una red de alta velocidad que tendrán un comportamiento único; su funcionalidad principal será procesar algoritmos científicos de alta complejidad que requieren gran capacidad de recursos computacionales, en la cual se utilizaron los núcleos de cada computadora donde estas se integrarán como una sola. La necesidad de haber implementado un clúster de alto rendimiento es debido al tiempo que estas demoran en resolver problemas, donde las computadoras ordinarias no podrían lograrlo por su limitada capacidad con respecto a sus características.

En el capítulo uno se habló sobre los aspectos generales de la institución donde se implementó datos generales y se conoció la misión y visión que brinda la institución.

En el capítulo dos se identificó el problema general y específico, también se encontró la justificación de la investigación en el ámbito teórico y práctico.

En el capítulo tres se explicó el fundamento teórico de nuestro tema de estudio donde se vio los antecedentes nacionales e internacionales y el marco teórico de la investigación.

EL capítulo cuatro; la arquitectura global del clúster de alto rendimiento, el comportamiento del HPC y las pruebas de rendimiento para ver su escalabilidad en rendimiento en utilizar variedad de cantidades de computadores.

En el quinto capítulo se observó los costos de inversión en el proyecto, mano de obra, componentes de la arquitectura etc., esto equivale a indicar si el proyecto cumplió con el dinero estimado propuesto por el FONDECYT, y también se vieron algunas referencias utilizadas para recolectar información.

Podemos concluir mencionando que las tecnologías avanzan a pasos agigantados, de igual manera los procesos computacionales para resolver algoritmos requieren de mayores recursos y gracias a esta arquitectura podremos tener resultados en un menor tiempo.

CAPÍTULO I: ASPECTOS GENERALES

1.1 DIAGNOSTICO DE LA ORGANIZACIÓN

1.1.1 Datos de la organización

A. Razón Social: Centro de Tecnologías de Información y Comunicaciones De la Universidad Nacional de Ingeniería.

B. Nombre comercial: CTIC-UNI.

C. Giro del negocio: Investigación

D. RUC: 20517141721

E. Teléfono: (01)4812559

F. Ubicación: Av. Túpac Amaru, Puerta N 5 UNI

G. Fecha inicio actividades: 11 / Noviembre/2008.

H. Reseña histórica:

El Centro de Tecnologías de Información y Comunicaciones de la Universidad Nacional de Ingeniería (CTIC-UNI) fue inaugurado en el año 2008. Es un centro que absorbe las actividades del centro de cómputo de la UNI (CC-UNI) y el Centro de aplicaciones telemáticas (CETEL-UNI). El CTIC-UNI ha sido construido e implementado gracias a una donación de la agencia peruana de cooperación internacional de corea (KOICA).

El equipo que gestionó esta importante donación estuvo conformado por los docentes Ing. Doris Rojas (Líder del proyecto), Lic. Héctor Bedón Monzón, Ing. Samuel Oporto y la Ing. Luz Eizaguirre.

Desde su reciente creación ha sido gestora de importantes proyectos tecnológicos de Impacto Nacional e Internacional como: El satélite chasqui I, el primer dirigible autónomo UNI, aplicaciones biomédicas, robótica, etc. que colocan a la universidad a la vanguardia tecnológica.

1.1.2 Datos de la institución.

La institución está ubicada Av. Túpac Amaru, Puerta N 5 Pabellón R4 dentro de la Universidad Nacional de Ingeniería.



Figura 1. Lugar de la Institución CTIC-UNI

Fuente: Google maps, 2018.

En la figura se muestra la localización del CTIC-UNI, la institución se encuentra dentro la Universidad Nacional de Ingeniería, accediendo por la puerta Nro. 6.

1.1.3 Diagnostico estratégico

A. Misión

Compromiso en la investigación científica y tecnológica a estudiantes y profesionales a nivel nacional e internacional con el reto de contribuir al avance tecnológico del país en busca de integrar esfuerzo con empresas, el estado y la sociedad en general.

B. Visión

Aplicar las tecnologías de información en la Universidad Nacional de Ingeniería con la intención de ser una institución prestigiosa que aplica las últimas tecnologías para resolver problemas actuales a nuestra sociedad tecnológica, con el fin de dar bases de prestigio, calidad y liderazgo institucional.

C. Organigrama

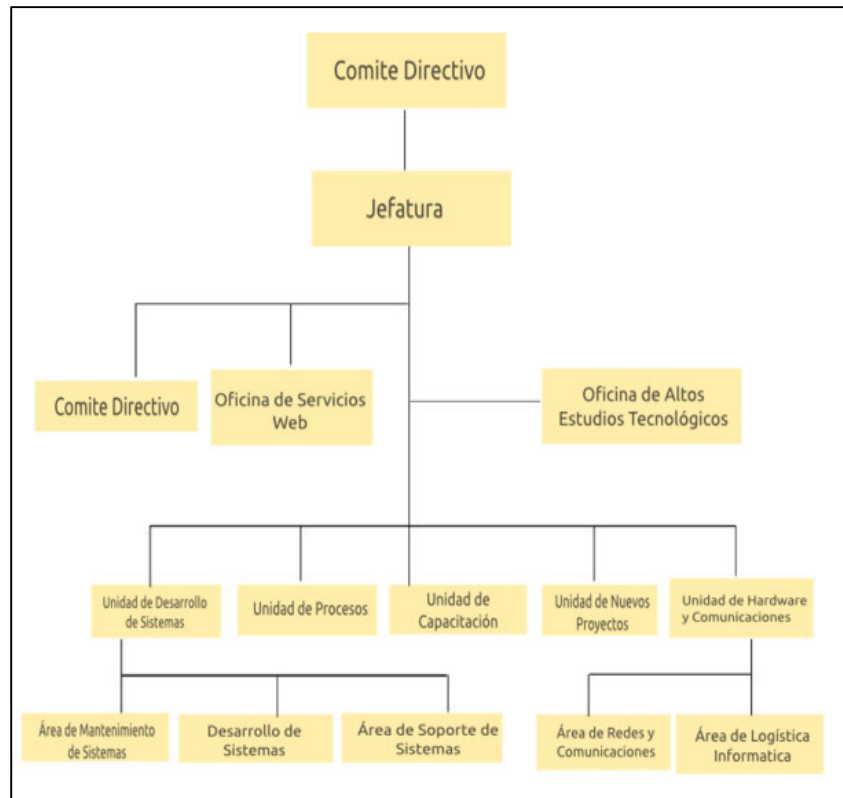


Figura 2. Organigrama nominal y funcional de la Institución CTIC-UNI

Fuente: Área Administrativa CTIC-UNI

La figura nos muestra cómo está organizada la institución CTIC-UNI la información fue extraída del área administrativa de la institución, el proyecto se implementó en el área de Unidad de Nuevos Proyectos.

CAPITULO II:

PLANTEAMIENTO Y FORMULACIÓN DEL

PROBLEMA

2.1 PLANTEAMIENTO DEL PROBLEMA DE INVESTIGACIÓN

2.1.1 Descripción del problema

Cada año las tecnologías de la información crecen de manera potencial y esto conlleva a incrementar la complejidad de los algoritmos.

En la actualidad, CTIC-UNI necesita tener la arquitectura de un clúster de alto rendimiento que sirve como plataforma base para poder realizar procesos de alta complejidad científica; la institución requiere que los procesos sean resueltos en un menor tiempo de lo esperado debido que los algoritmos que se implementó requieren de su inmediata información.



Figura 3. Clúster con ordenadores reciclados

Fuente: Laboratorio 205 CTIC-UNI

En la Figura 3 se muestra un clúster con computadoras recicladas con una cantidad 24 ordenadores, la arquitectura y configuración están mal implementadas incluyendo la característica a nivel hardware que no ayudarán al tener un rendimiento ideal al resolver problemas de alta complejidad.

2.1.2 Formulación del problema general

¿Qué arquitectura debe implementar para poder utilizar todos los recursos computacionales unidos en una sola red?

2.1.3 Formulación de los problemas específicos

P.E.1: ¿Cómo reducir tiempo de procesos de alta complejidad que tardan días en resolverse?

P.E.2: ¿Cómo eliminar los cuellos de botellas al realizar procesos de alta complejidad?

P.E.3: ¿Cómo medir el rendimiento del computador al lograr procesos de alta complejidad?

2.2 DEFINICIÓN DE LOS OBJETIVOS DE LA INVESTIGACIÓN:

2.2.1 Objetivo general:

Implementar una arquitectura computacional y configurar un clúster de alto rendimiento para resolver problemas de alta complejidad en el CTIC-UNI.

2.2.2 Objetivos específicos:

- **O.E.1:** Unir varias computadoras para que se comporten como un único computador con el fin de reducir el tiempo a casi a una 16ava parte de un proceso algorítmico complejo.
- **O.E.2:** Calcular el rendimiento de los ordenadores utilizando un algoritmo matemático para identificar los cuellos de botella y así utilizar una cantidad adecuada de nodos.
- **O.E.3:** Medir el rendimiento del clúster utilizando el algoritmo computacional de N-CUERPOS utilizando cierta de cantidad distinta de nodos para medir el tiempo del proceso.

2.3 JUSTIFICACIÓN DE LA INVESTIGACIÓN:

El proyecto de investigación conlleva a dar una solución a los procesos complejos que se presentan al utilizar algoritmos que utilizan altos recursos computacionales; busca aplicar una técnica para optimizar el tiempo utilizando el paralelismo computacional que se utilizan con normalidad en los supercomputadores.

2.4 ALCANCES Y LIMITACIONES DE LA INVESTIGACIÓN:

2.4.1 Alcances

Con la implementación del clúster de alto rendimiento se conseguirá lograr una arquitectura de hardware de altas características computacionales, estos tendrán diferentes beneficios en cuando se requiera de procesar y/o tener información con altos volúmenes. Entre sus alcances son los siguientes:

- A. **Paralelismo computacional.** Cuando un ordenador comienza a procesar un algoritmo computacional estos utilizan sus CPU y cada uno de sus núcleos procesan el problema de una manera secuencial, con la arquitectura del clúster de alto rendimiento, los procesos dejarán de resolverse secuencial mente y se resolverá utilizando el paradigma de paralelismo computacional.
- B. **Nodos maestros y esclavos.** Los ordenadores estarán siendo controlados por un nodo maestro quien se encarga de distribuir los procesos en las computadoras esclavas.
- C. **Sistema operativo *open source*.** Se utiliza una distribución libre para no tener problemas de licencia o tener que lidiar con virus informáticos que son más propensos a los sistemas operativos con licencia.

2.4.2 Limitaciones

A. Técnicas de *Big Data*:

Los procesos que haga el clúster no tienen nada vinculado a la obtención de gran cantidad de información de la nube.

B. **Inteligencia artificial.** El supercomputador no estará relacionado con inteligencia artificial debido a que la institución no ha requerido de su necesidad.

C. **Algoritmo propio:** No se trabajará con un algoritmo propio para poder hacer procesos de alta complejidad, más si un *Benckmarck* para poder hacer pruebas de rendimiento.

D. **Compras de componentes:** Una de las limitaciones es la compra total de los componentes, debido a que no se logró comprar todos los componentes en un solo pedido, al conseguir todos los componentes en el año 2017 el rendimiento de nuestro proyecto incrementará en el potencial computacional.

CAPITULO III: FUNDAMENTO TEÓRICO

3.1 ANTECEDENTES

Desde la creación de las computadoras cambiaron algunos paradigmas para poder resolver problemas cotidianos del hombre, esto a su vez llego a ser utilizado en varias áreas del conocimiento, llegando a si a sacarle su máximo provecho de esa forma; las computadoras llegaron a ser de uso militar y al querer sacar sus mayores capacidades computacionales nacen los clústeres o su seudónimo supercomputadoras.

Las primeras supercomputadoras fueron usadas para uso militar como escenario histórico en la segunda guerra mundial, donde el servicio de inteligencia de (Los Estados Unidos de América) EEUU infiltró a sus soldados en las filas alemanas, debían descifrar los movimientos del enemigo y como las informaciones eran tan numerosas y se necesitaba celeridad para poder tomar las decisiones lo más rápido posible para conducir a las fuerzas propias. (Hernandez, 2014, p.1)

Las primeras computadoras existentes eran sólo para uso militar y científico no fueron llamadas supercomputadoras debido a que fueron las primeras y no tuvieron antecesores para tener ese prefijo, “Las supercomputadoras del ayer son las computadoras de hoy”, las computadoras desde su primera aparición hasta la actualidad han facilitado al ser humano al obtener resultados óptimos en un tiempo muy corto.

Las supercomputadoras determinadas con este prefijo fueron introducidas en la década de los 70 y fueron diseñada en la compañía *Control Data Corporation* (CDC) por Seymour Cray al crear la primera arquitectura de un supercomputador donde el proceso de varias computadoras se unió en una sola obteniendo un gran poder computacional. (Atul , Bhargavi & Uditnarayan, 2015, p.2)

3.1.1 Internacionales

- A.** Para poder hacer procesos de alta demanda computacional y hallar resultados en muy corto tiempo, se requiere de una computadora de alto rendimiento, en palabras comerciales una supercomputadora que son utilizadas para hacer procesos muy complejos donde computadoras comunes y corrientes no podrían alcanzar en sus procesos computacionales; las supercomputadoras están diseñadas para resolver problemas científicos que abarca una amplia cantidad de necesidades que se necesitan en la actualidad; el costo de comprar una supercomputadora es muy elevado debido a que pocos países poseen una y su demanda crece más.

La computación científica es en nuestra actualidad un tema de mucha relevación en las áreas de ciencias de la computación, la cantidad de usuarios que utilizan técnicas computacionales para resolver algoritmos científicos crece de manera paulatina, la complejidad y el elevado costo computacional de dichos problemas, ha impulsado al desarrollo de procesos avanzados para la computación distribuida y paralela de alto rendimiento (Cabrera y Soler, 2011, p.2)

- B.** La compra de supercomputadoras ensambladas son muy elevados; con el pasar del tiempo presentará problemas de *software* y *hardware* por lo cual a dar soluciones tendría un costo adicional por tener que llamar al técnico especialista, a diferencia de implementarse un HPC que posee elementos que ya conocemos y puede darse mantenimiento adecuado cada cierto tiempo que uno crea conveniente.

La principal de esta arquitectura es que es directamente escalable y presenta un bajo costo para sistemas grandes, la transferencia de datos se realiza a través de la red de interconexión que conecta un subconjunto de procesadores con otro subconjunto dado. (González y Rodríguez, 2008, p.12)

- C.** Implementación un clúster de alto rendimiento nos ahorraría el costo de inversión y mantenimiento.

Los fabricantes de supercomputadoras nos ofrecen usualmente una única opción para la actualización: “compre el nuevo modelo de este año”. Además, ni pensemos lo que representa si algún componente de la supercomputadora se daña fuera de su periodo de garantía. (Milone, Azar y Rufiner, 2002, p.6)

- D.** El HPC realizó un supercomputador cuya función primordial es resolver algoritmos de alta complejidad en corto tiempo; lo explicado por Herbert nos indica que hacer simulaciones implica utilizar un computador de alto rendimiento computacional.

La investigación de Hoeger (2006) “Las aplicaciones intensivas en el cálculo que solo pueden hacerlas las supercomputadoras: Estas son algoritmos computacionales que requieren varios ciclos de máquinas, estas han impulsado el desarrollo de supercomputadores. Son típicas en ciencias e ingeniería, aunque recientemente ha aparecido en otras áreas como simulación financiera y económica. Depende en su totalidad de la velocidad y el procesamiento de punto flotante de los supercomputadores”. (p.5)

- E.** Lo mencionado por Lechtaler relaciona las funciones que hizo en criptoanálisis con la investigación a la problemática actual; la implementación del clúster de alto rendimiento no solamente nos podrá solucionar problemas a simulaciones de sistema vehicular, sino también a resolver diversos problemas científicos.

Las arquitecturas de cómputo distribuidas han sido utilizadas históricamente en problemáticas científicas específicas como cálculos de flujos, predicciones meteorológicas o cálculos astronómicos. Existen múltiples formas de computación distribuida, desde clústeres de procesamiento hasta máquinas paralelas pasando por sistema de cómputo en grilla. (Lechtaler, 2014, p.2)

3.1.2 Nacionales

Las necesidades de tener supercomputadores en el Perú ya están siendo bien requeridas en varias áreas de investigación; tener un conocimiento adecuado de estos equipos en la institución podría también incrementar los ingresos a la institución.

La experiencia técnica por la empresa COOPERAZIONE INTERNAZIONALE-COOPI explica lo siguiente: “Para poder estudiar las interferencias de las construcciones, se debe considerar datos de topología satelital (...) El modelo numérico puede aceptar este tipo de datos. Las limitaciones serían de carácter computacional, se necesitaría una supercomputadora para procesarlo en un tiempo relativamente corto”. (COOPI, PNUD, IRD, Defensa Civil, & Unión Europea, 2010, p.32)

La utilización de supercomputadoras en nuestro país ya no es una idea que se debe considerar en segundo plano, debido al crecimiento de las nuevas tecnologías; en la actualidad se utilizan bases de datos con otros paradigmas para poder extraer información en corto tiempo, un ejemplo claro sería las técnicas de *Big Data* que se aplican en clústeres para poder sacar el mayor rendimiento computacional.

3.2 MARCO TEÓRICO

Para entender mejor este trabajo de investigación daremos a conocer algunas definiciones teóricas sobre el tema de nuestro trabajo de investigación.

3.2.1 Supercomputadoras

Para poder realizar procesos de alta complejidad debemos conocer detalladamente como es su comportamiento del clúster de esa manera poder sacar su mayor provecho, la utilización de supercomputadoras en Perú está siendo escasa debido al alto costo que demandan, el siguiente autor nos hable sobre el uso de supercomputadoras.

Las supercomputadoras son equipos de alto rendimiento muy poderosas que se usan sobre todo para cálculos a nivel de científico y comercial. Para acelerar las operaciones, los componentes de hardware se interconectan para minimizar la distancia que estos tienen que recorrer, también utilizan técnicas especiales para disminuir el calor en los circuitos y evitar un sobrecalentamiento y evitar que ocurra que se quemen los componentes. Su uso se enfoca a ciertas aplicaciones científicas y su mercado también es pequeño, debido a su alto precio que estos demandan. (Morris, 2009, p.120)

Como se indicó en la siguiente cita estas serían las características y comportamiento que tendría un supercomputador, estos a su vez están volviéndose tendencia debida que las empresas desean sacar el mayor provecho para poder obtener procesos en muy corto tiempo.



Figura 4. Supercomputador STAMPEDE TEXAS

Fuente: Texas Advanced Computing Center, 2018

La siguiente figura nos muestra un HPC que utiliza la misma arquitectura computacional que se utilizó en el trabajo de investigación.

3.2.2 Clúster

Un clúster es un conjunto de computadoras interconectadas entre sí mediante una red de alta velocidad, estas a sus veces van a resolver problemas científicos en corto tiempo facilitando así el tiempo de soluciones de problemas

Estos sistemas constituyen una solución flexible, de bajo costo y de gran escalabilidad por aplicaciones que requieren una elevada capacidad de computadora y memoria. Así mismo estos equipos son independientes que ejecutan una serie de aplicaciones de forma conjunta. (Colobran, Arqués y Galindo, 2007, p.34)

La implementación del cluster ayuda a resolver problemas científicos un menor precio que un supercomputador, así mismo el mantenimiento no genero muchos gastos debido que los componentes son fáciles de reemplazar.

La utilización del clúster para resolver problemas científicos será en menor tiempo debido a que estos resuelven problemas en paralelo optimizando de esta forma el tiempo de solución.

Como son componentes de conocimiento técnico se puede incrementar la capacidad del clúster debido a que su instalación es sencilla.

A. Características

- Consta de 2 o más computadoras interconectadas por una red de alta velocidad
- Cada computadora necesita procesador memoria e interfaz gráfica que ayudará para la comunicación con la red.
- Necesita un software especializado que ayude a la comunicación entre los nodos.
- Todos los elementos trabajan en conjunto y a la vez de forma independiente para resolver un problema.

B. Clasificación de clúster

En términos generales la palabra clúster hace referencia a la colaboración estratégica entre sectores industriales en la cual colaboran para tener un beneficio en común, en términos técnicos un clúster equivale a indicar la unión de varios computadores estos están clasificados en:

- Alto rendimiento
- Alta disponibilidad.
- Alta eficiencia.

a. Alto rendimiento:

Los clústeres de alto rendimiento o también conocidos como HPC toman como recursos los componentes de hardware, donde su principal función es resolver procesos de alta complejidad.

Según Jiménez y Medina (2014) “Los clúster de alto rendimiento utilizan los nodos interconectados entre sí para ejecutar procesos en paralelo, esto permite que el resultado de un problema pueda obtenerse en menor tiempo.” (p.17)

b. Alta disponibilidad:

Los clústeres de alta disponibilidad cuya función principal es dar la máxima disponibilidad, son capaces de detectar fallos y permitir recuperarse de ellos sin tener inconvenientes con el proceso. Como indica (González, 2014, p.10). “Los clúster de alta disponibilidad no buscan tener una mayor potencia de cálculo si no conseguir un conglomerado de máquinas que brinden servicios donde esta pueda ocuparse de sus tareas si una queda inoperativa”.

c. Alta eficiencia:

Los clústeres de alta eficiencia su función principal es resolver mayor cantidad de tareas en el menor tiempo posible, estas no requieren que su algoritmo sea complejo.

Con respecto los clúster de alta eficiencia “Su objetivos es la de realizar grandes cantidades de tareas en un corto tiempo, la necesidad de procesos no requieren complejidad debido a que estos no resolverán procesos de alta complejidad.” (Sinisterra, Díaz y Ruiz, 2012, p.94)

C. Componentes de un clúster

Un clúster necesitan de varios componentes de hardware y software, sus requerimientos mínimos para que tenga un buen ambiente de trabajo son los siguientes, como lo indica Alcántara (2014) “Los componentes esenciales que deben tener un clúster son los siguientes.

1. Nodos.
2. Almacenamiento.
3. Sistema Operativo.
4. Conexiones de red.
5. Elementos auxiliares: PDU, Conmutador KVM, Rack.
6. Ambientes de programación paralela.” (p.9)

a. Nodos

En términos técnicos un nodo es un componente que forma parte de una red, en este caso serían los ordenadores que utilizará el clúster. Los nodos del clúster pueden consistir de uno a más procesadores, estos equipos deberán tener un ambiente refrigerante para que así puedan trabajar y evitar que se sobrecaliente”.

b. Almacenamiento

Los clústeres necesitan de una unidad de almacenamiento, donde se instalará fundamentalmente el sistema operativo, configuraciones de red para la comunicación entre los nodos esclavos y maestros, es de consideración tener una gran capacidad de espacio debido a la información que llegaran a los nodos esclavos.

En la actualidad todos los ordenadores incluyen entre sus dispositivos las unidades de almacenamiento que son los discos duros (HDD, *Hard Disk Drive*) estos constituyen el medio de información más importante permite recuperar gran cantidad de información, Estos dispositivos son llamados memoria secundaria, debido a la oposición de la memoria principal o memoria RAM. (Rebollo, 2013, p.91).

Los nodos del clúster utilizarán los discos duros como principal recurso para obtener información y estas ser procesadas por la unidad central de procesos CPU, el disco duro siempre estará en comunicación en la CPU y las memorias primarias.

c. Sistema operativo

El conjunto de programas informáticos que dan una administración eficaz a los recursos de una computadora es conocido como sistemas operativos, estos operan apenas se encienda el equipo, debido a que estos gestionan el hardware a múltiples niveles, existen diversidad de sistemas operativos, entre ellos están los sistemas operativos de código abierto y cerrado.

Los clústeres pueden adaptarse a cualquier S.O que se le asigne, el cambio radica en la complejidad de las configuraciones que estas puedan tener, es recomendable implementar un clúster utilizando S.O de

código abierto debido a que se podrá acceder al código fuente donde podrás acceder a la lógica del sistema, a diferencia de los S.O de código cerrado que limitan estos accesos.

El software es fundamental y complejo para los clúster de alto rendimiento a lo contrario como se piensa en la informática convencional, esto deben estar preparados para aprovechar la máxima cantidad de núcleos y otros recursos computacionales que se puedan utilizar, en recomendación se debería de utilizar software que este registrados en la página web de los Top 500, donde se encuentran registradas los supercomputadores más potentes del mundo.(Ocampo y Exequiel, 2017,p.91).

Como fue indicado en la cita anterior podríamos acceder a la página que se dio recomendación para poder seleccionar un sistema operativo que se acoja a nuestras necesidades de uso, en la siguiente imagen apreciaremos lo rescatados por el autor la lista de sistema operativo que se utilizarán en un clúster de alto rendimiento.

Tabla 1. Lista de sistemas operativos utilizados en clúster de alto rendimiento.

Sistema Operativo	Link de descarga
CentOS	https://www.centos.org/download
Cray Linux	Sin datos
Red Hat Enterprise	https://access.redhat.com/downloads
SUSE Enterprise	https://www.suse.com/download-linux/
TOSS	Sin datos

La siguiente tabla es una lista de los sistemas operativos más utilizados en clústeres de alto rendimiento.

d. Conexiones de red

La interacción entre los nodos de un clúster debe ser mediante una red comunicación, donde estos tienen que interactuar de manera fluida sin la interrupción de cuellos de botellas que pueden ocurrir en el transcurso de la

solución de un problema, estos problemas pueden ocasionarse por diferentes factores que se ve con detalle cuando se realicen las pruebas de rendimiento.

La interconexión entre nodo va consistir de tener una de red alta velocidad, siempre respetando las normas de crimpado de cable, para que estos no tengan problema a la hora de su comunicación se sugiere la utilización de Myrinet que es una red de interconexión para clúster de altas prestaciones o una red Ethernet pero siempre midiendo en donde ocurrirían los cuellos de botella al momento de la comunicación si se colocaran un conglomerados de nodos.(Pérez, 2014, p.5)

Cuando mayor sea la cantidad de nodos interconectados entre sí, se va tener que optar de dar un seguimiento a la comunicación que estos tenga mediante su red de comunicación por ellos en propias consideraciones es preferible de utilizar dispositivos de comunicación que eviten los cuellos de botella que puedan ocurrir.

e. Elementos auxiliares

Los elementos auxiliares sirven para darle al clúster una mejor estabilidad y ambiente de trabajo, entre ellos tenemos:

1. PDU (*Power Distribution Unit*).
2. Conmutado KVM.
3. Rack.

Existen muchísimos otros elementos auxiliares, pero estos son considerados lo más importante para que el clúster tenga un ambiente de trabajo adecuado.

- **PDU**

Los PDU o sus siglas en español unidad de distribución de energía es un dispositivo de hardware diseñados para distribuir la energía eléctrica a los

racks de ordenadores y equipos de red ubicados en un clúster.

- **Conmutador KVM**

El conmutador KVM es un equipo que procesa diversas señales de video donde tiene una única salida, los conmutadores en clúster son utilizados para poder acceder desde un monitor a los diferentes nodos para visualizar su interfaz gráfica.

- **Rack**

Los racks permiten sostener o albergar dispositivos tecnológicos y vibran de un ambiente adecuado para que realicen sus trabajos, se sugiere tener un rack integrado con aire acondicionado para que los nodos del clúster no se sobrecalienten.

f. Ambiente de programación paralela.

Para resolver problema de alta complejidad tendremos que recurrir a la programación paralela, al ser ejecutado con esta técnica de programación el problema se dividirá en parte iguales y serán distribuidas a cada nodo esclavo para ser procesados, van a tener que ser paralelos debidos que el clúster sea utilizado correctamente, el objetivo de la paralelización contribuye los siguientes puntos.

1. Reducir el tiempo real de cómputo de un problema de cómputo ordinario.
2. Reducir el costo de cómputo. Esto se logra dividiendo la carga computacional entre varios procesadores distintos, para generar un mayor potencial computacional con una menor inversión en hardware.

La programación paralela utiliza el recurso de todos nodos interconectados entre sí, la programación en paralelo será administrada desde un solo ordenador que

administrará el proceso de todos los nodos conectadas a ellas.

3.2.3 ***Multiple Instruction Multiple Data (MIMD)***

Consistieron en el siguiente paso en la evolución de las arquitecturas paralelas. Fueron lentamente despasando al modelo SIMD. Las computadoras MIMD pueden trabajar asincrónicamente (los procesadores tienen la capacidad de funcionamiento semiautónomo).

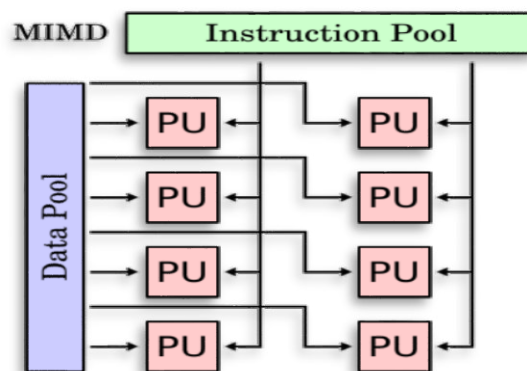


Figura 5. MIMD *Instruction Pool*

Fuente: Allig, 2015

En la figura 5 vemos un diagrama del comportamiento de múltiples instrucciones con múltiples datos, esta es la secuencia que sigue un clúster de alto rendimiento.

3.3 MARCO METODOLÓGICO

El proyecto de tesis está orientado a las áreas de investigación, por ello se explica desde el inicio la arquitectura del clúster hasta el último apartado los pasos para la implementación.

El proyecto tiene como primer punto explicar las características de los componentes, arquitectura del hardware, las configuraciones entre los nodos esclavos y maestro, utilizaremos normativas utilizadas en clústeres para medir la escalabilidad, entre ellas es hacer pruebas de rendimiento utilizando el algorítmico de N-CUERPOS y localización de tráfico de red o también llamado cuellos botella.

Tabla 2: Normativas de rendimiento escalable

Normativas de rendimiento escalable para clústeres de alto rendimiento

- Realización de pruebas de rendimiento con un algoritmo que acepte técnicas de paralelismo computacional para medir el rendimiento con cantidades de distintas de nodos con el fin de obtener el tiempo de solución.
- Utilización del mismo algoritmo y obtener el resultado de comunicación, esta comunicación servirá para restarlo con el tiempo de resultado del algoritmo, si este resultado tiende a ser cero, se concluye que son las cantidades de nodos ideales a utilizar para evitar tráficos habituales de comunicación.

En la siguiente tabla se observa las normativas que se utiliza para detectar si el clúster es escalable y obtener la cantidad de nodos deseadas. “Estas normativas son habitualmente utilizadas en los clústeres de alto rendimiento y nos proporciona asertividad en la escalabilidad de la implementación de estos equipos”(Fiestas, 2014, p.2).

3.4 MARCO LEGAL

El presente proyecto se encuentra dentro de la Constitución Política del Perú, del decreto Supremo Nro. 076-2010- PCM estableciendo las disposiciones referidas a las adquisiciones de computadoras personales que convoquen las entidades públicas, con lo que indica en las leyes en sus artículos siguientes.

3.4.1. Ley general.

Título: Ley Adquisición y adecuación del software y hardware en el estado peruano.

Artículo 2°. La evaluación técnica de los recursos de software y hardware requeridos por la administración pública se sujetará a las normas dictadas por el ente rector del Sistema Nacional de Informática.

Artículo 3°. Para efectos de la presente Ley se adoptan las definiciones:

1 software libre:

Es aquel cuya licencia de uso garantiza las facultades de:

- Uso irrestricto del programa para cualquier propósito.
- Inspección exhaustiva del mecanismo de funcionamiento del programa.
- Confección y distribución de copias del programa
- Modificación del programa y distribución libre tanto de las alteraciones como nuevo programa resultante.

2 software propietario:

Es aquel cuya licencia de uso no permite ninguna o alguna de las facultades previstas en la definición anterior.

3.5 ARQUITECTURA COMPUTACIONAL

La siguiente arquitectura explica la forma en que se implementó el trabajo de investigación.

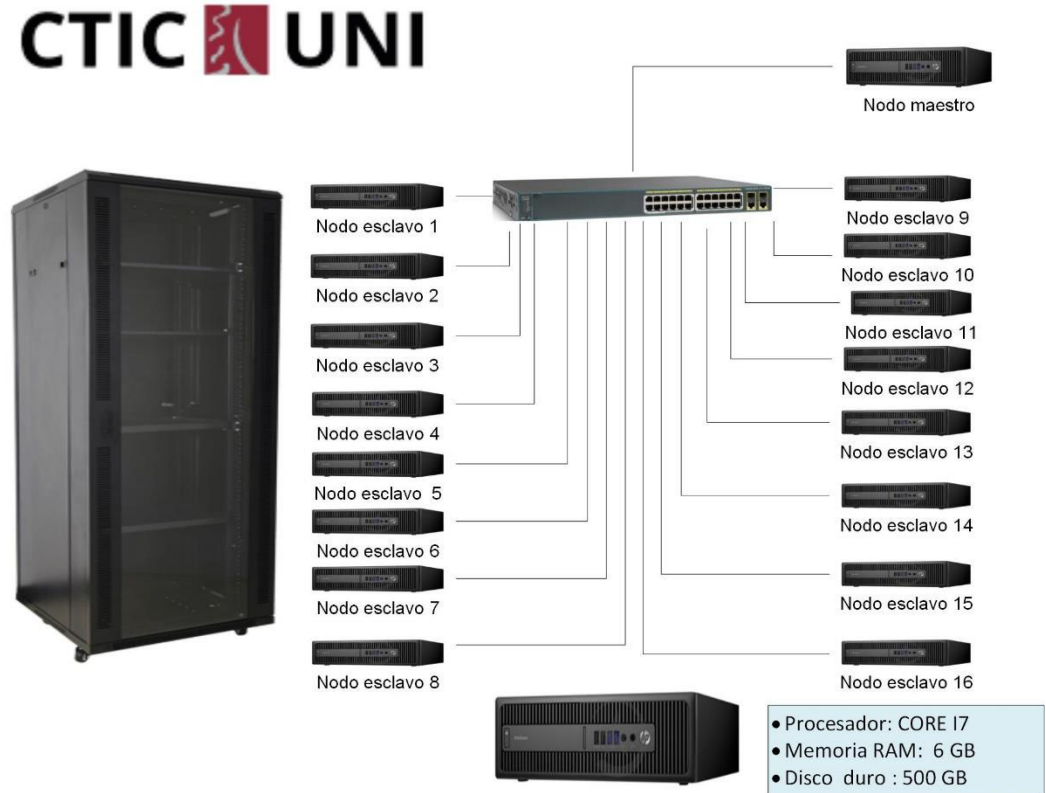


Figura 6. Arquitectura del HPC

Como se ve en la figura 6, vemos 16 nodos esclavos y 1 nodo maestro que están dentro del gabinete, todos los ordenadores tienen la misma característica en hardware, el que administra todas las ordenes es el nodo maestro quien distribuye los procesos a los nodos esclavos, esto lo consigue teniendo comunicación mediante una red alta velocidad (*switch*) que cuenta con más de 17 puertos para ser integrado por cada uno de los nodos esclavos y maestro, la utilización de un switch con gran cantidad de puertos es debido a un posterior crecimiento entre los nodos esclavos .

CAPITULO IV: DESARROLLO DEL PROYECTO

4.1 LEVANTAMIENTO DE INFORMACIÓN

4.2.1 Planificación del proyecto

Se realizó el cronograma de actividades del proyecto el cual corresponde a las actividades, reuniones realizadas para la implementación del proyecto, estas actividades están organizadas por fases que a su vez incluyen un inicio y un final programado (Ver anexo A).

4.2.2 Recopilación de información

- **Modelo de Investigación**

Modelo de cuestionarios (Ver anexo C).

Cargo: Enrique Lee Huamani Uriarte.

- **Modelo de exposición**

Modelo de exposición (Ver anexo D).

Cargo: Enrique Lee Huamani Uriarte.

4.2 IMPLEMENTACIÓN

4.2.1 Secuencia de la implementación del proyecto

En la implementación del proyecto se ha tenido que seguir pasos, estos han servido de ayuda para poder conocer el armado desde un inicio a fin de la implementación, estos nos han ayudado a tener el conocimiento de cómo se implementó un hardware con una arquitectura ideal para poder resolver problemas de alta complejidad. En los siguientes capítulos vamos a explicar los pasos que se realizó en la implementación del clúster, iniciando desde la implementación de hardware y finalizando con las configuraciones de software como la configuración es idéntica a cada nodo del clúster se argumentará a un solo nodo considerándolo como el general.

4.2.2 Diagrama de la arquitectura computacional

En el armado del clúster dio una arquitectura para su adecuado funcionamiento, en la siguiente imagen vemos la arquitectura que se

implementó, este diseño es la arquitectura estándar utilizadas en los HPC.

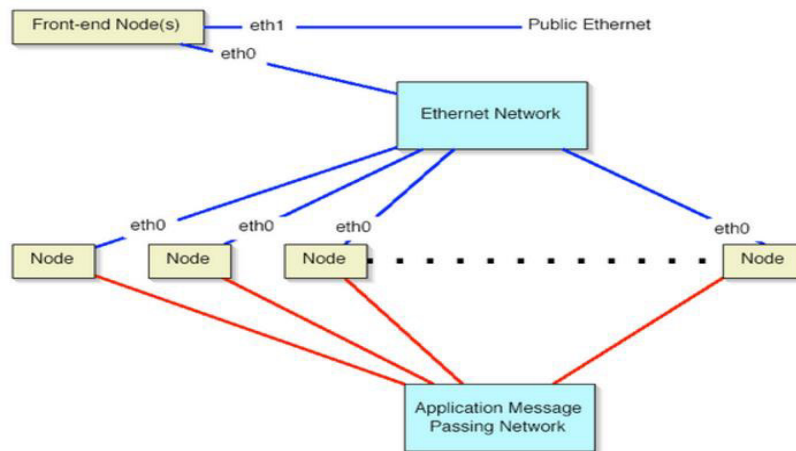


Figura 7. Diagrama computacional de clústeres

Fuente: bairdpetro, 2015

La siguiente figura 7, nos muestra el diagrama estándar del clúster de alto rendimiento, extraído de una página dedicada a la supercomputación.

4.2.3 Armado del rack del HPC

El clúster de alto rendimiento al ser un hardware necesitó de un espacio donde pueda utilizarse, para ello se ha tenido que poner un mueble ideal, en la siguiente imagen se observa el armado del rack, en ella se está incluyendo componentes de ventilación para que el equipo computacional tenga un mejor ambiente de trabajo, el armario rack utilizado es ideal para cumplir una formalidad con los nodos del clúster y tenga un espacio focalizado en su mantenimiento o su posterior incremento de computadores cuando existan nuevos requerimientos por parte de la entidad que nos financió el proyecto de la implementación.



Figura 8. Armario rack.

En la figura 8 nos muestra el armario rack que se utilizó para la implementación del clúster de alto rendimiento.

También se colocó en la parte superior del clúster para que tenga una ventilación ideal y trabaje sin problemas al querer funcionar y no tenga un sobrecalentamiento que ocurre a utilizar mayor demanda de esfuerzo computacional.



Figura 9. Cooler de refrigeración

El cooler incorporado del rack ayudará a una mayor ventilación del clúster de alto rendimiento

4.2.4 Especificación de los nodos

Se utilizó los ordenadores HO 800G2 de HP, la opción a utilizar este modelo de case es debido al poco espacio que ocupa, para saber sobre sus características de hardware ver (Ver anexo F).



Figura 10. Ordenadores HO 800G2

La siguiente figura nos muestra el case que se utilizó en la implementación del clúster, en ella tenemos incorporados los componentes de hardware.

4.2.5 Implementación de la red computacional

Para la comunicación de los nodos esclavos con el nodo maestro, se utilizó un hardware de red de comunicación, este permitió dar y recibir la información, en la siguiente imagen vemos la arquitectura que se realizó con los nodos esclavos.

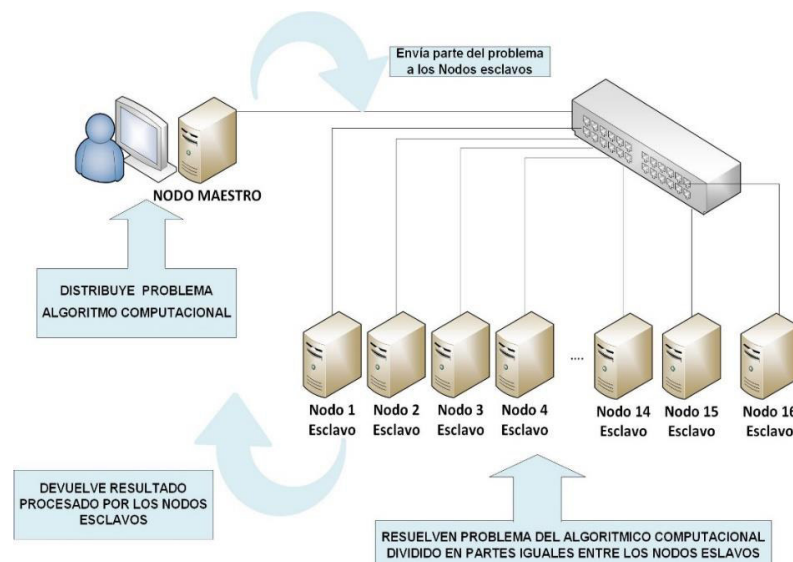


Figura 11. Nodos esclavos y maestro con red de alta velocidad

La figura 11 nos muestra una la red alta velocidad, estas tienen que ser gran eficiencia debido a la comunicación constante que tienen con los nodos.

En la implementación de clúster siempre se tiene que utilizar una red de comunicación, para que los nodos esclavos se comuniquen con el nodo maestro, con el objetivo de poder recibir el problema y poder resolverlo, como se indicó en líneas anteriores se utiliza una red de alta velocidad con características necesarias para que se eviten los posibles cuellos de botellas que puedan ocurrir cuando se ejecute el algoritmo computacional.

4.2.6 Finalización de la arquitectura física del clúster

Teniendo el armado físico del clúster, sigue las configuraciones para cada uno de sus nodos, esto lo veremos el punto 4.2.9 herramientas utilizadas en la instalación, que explica detalladamente las configuraciones a nodos esclavos y maestro.



Figura 12. Implementación física del clúster de alto rendimiento

La figura 12 nos muestra una la red alta velocidad, estas tienen que ser gran eficiencia debido a la comunicación constante que tienen con los nodos.

En la figura 12 observamos el clúster operativo en el CTIC-UNI, siendo utilizada por estudiantes, investigadores y profesionales en temas de computación científica para poder resolver problemas de alta complejidad, el objetivo que se plantea para el año 2019 es utilizar el clúster con un algoritmo inteligente para que resuelva el tráfico vehicular en Lima metropolitana utilizando técnicas de *BIG DATA*.

4.2.7 Determinación de requerimientos de software

La construcción de un clúster de alto rendimiento requiere de tener un Sistema Operativo (S.O) robusto y especializado para soportar las diversas cantidad de procesos que se desea hacer, inicial mente se planteó utilizar un software propietario debido a su conocimiento básico que tiene la mayor cantidad de usuarios, pero por ser un software propietario tendremos limitaciones en hacer modificaciones a nuestra plataforma debido a que son códigos establecidos y nos inhabilitan en adecuarlo o configurar a entorno a nuestra necesidades, para saber más sobre las características de este sistema operativo (Ver anexo H).

4.2.8 Instalación del sistema operativo del clúster de alto rendimiento.

El sistema operativo que se eligió fue el CENTOS ROCKS , es un sistema operativo que ya tiene definido herramientas para la configuración del clúster, con este S.O tendremos 2 maneras para poder configurar el clúster , uno será la manera automática no recomendada debido a las desconexiones que esta sufre por la mala configuración que se dan a los nodos y la forma manual que sirve para darle una configuración ideal al clúster , un aspecto importante a utilizar son las herramientas (*Scripts*) , esto resulta ventajoso debido a la cantidad de nodos a configurar.

4.2.9 Herramientas utilizadas en la instalación

Para la optimización del clúster se utilizó unas herramientas necesarias para optimización de la comunicación, los *scripts* que

ayudan a dar órdenes y copiar ficheros mediante *ssh* desde el nodo maestro hacia los nodos esclavos, también se ha creado una herramienta que genera claves públicas para poder acceder a los servidores esclavos sin tener la necesidad de colocar una contraseña cada vez que se ejecute una orden, se utilizaron los siguientes Script.

- **Script *psh***: Envía órdenes mediante el *ssh* a todos los nodos del clúster.
- **Script *pscp***: Copiara un fichero del master a todos los nodos esclavos.
- **Script *keyscan.sh***: Se encarga de generas clavas públicas de los nodos esclavos del clúster.

Estos son *Scripts* necesarios que se utilizaron para la configuración de los nodos esclavos y maestro del clúster a continuación se muestra los pasos de la configuración que se hicieron al clúster de alto rendimiento.

4.2.10 Configuración del *Script psh*

```
#!/bin/sh

#Script psh. Envía una orden a todos los nodos del Cluster TFM.
i=2
while [ $i -le $CLUSTER_SIZE ]
do
    #Para imprimir por pantalla el nodo y la orden que estamos pasando.
    echo server$i $1;
    #Nos conectamos al nodo y le pasamos la orden que escribo en el terminal.
    ssh server$i $1;
    let i=i+1
done
sleep 1s
echo "Operacion completada satisfactoriamente!!"
echo "*****+*****+*****+*****"
echo "Fin del script psh!!...*****"
```

Figura 13. Instalación de *Script psh*

La figura 13 nos muestra la configuración que se hacen con respecto al Script *psh*, la configuración se está aplicando al nodo maestro.

4.2.11 Configuración del *keyscan.sh*

```
#!/bin/sh

#Script pscp. Para copiar un fichero en todos los nodos del Cluster TFM.
i=2
while [ $i -le $CLUSTER_SIZE ]
do
    echo server$i $1;
    #Copia el fichero que se pasa en el primer argumento ($1),
    #en la ruta que pasa mos como segundo argumento ($2).
    scp $1 server$i:$2;
    let i=i+1
done
sleep 1s
echo "Operacion completada satisfactoriamente!!"
echo "*****+*****+*****+*****"
echo "Fin del script pscp...*****"
echo "*****+*****+*****+*****"
```

Figura 14. Instalación de *keyscan.sh*

La figura 14 nos muestra la configuración que se hacen con respecto al de *keyscan.sh*, la configuración se está aplicando al nodo maestro.

4.2.12 Configuración del *Script pscp*

```
#!/bin/bash

i=2
while [ $i -le $CLUSTER_SIZE ]
do
    ssh-keyscan cluster$i >> /root/.ssh/known_hosts
    let i=i+1
done
```

Figura 15. Instalación de *Script pscp*

La figura 15 nos muestra la configuración que se hacen con respecto al de *keyscan.pscp*, la configuración se está aplicando al nodo maestro.

4.2.13 Configuración del nodo maestro

En primer lugar, se procedió la instalación del S.O al nodo maestro, la estrategia fue en realizar una única instalación, se clonará en todos los nodos la instalación inicial realizada, para volver a realizar el proceso de instalación en cada nodo.

Luego de la instalación del S.O se preparó la imagen que aplicaremos a los nodos esclavos, para lograr esto se instaló un *servidor Network File System* (NFS) que ayudo a conseguir una partición que sirvió de arranque al nodo, a continuación, se muestra el *Script* donde se realizó la configuración.

```
#!/bin/sh

#Copia del sistema principal instalado al NFS.

cp -ax / /nfs/srv/

#Configuracion del fichero /nfs/srv/etc/fstab:

rm -r /nfs/srv/etc/fstab

echo "/dev/nfs      /          nfs defaults    0   0" >> /nfs/srv/etc/fstab

#Configuracion del fichero /nfs/srv/etc/network/interfaces:

rm -r /nfs/srv/etc/network/interfaces

echo "auto lo" >> /nfs/srv/etc/network/interfaces
echo "iface lo inet loopback" >> /nfs/srv/etc/network/interfaces
echo "####" >> /nfs/srv/etc/network/interfaces
echo "####" >> /nfs/srv/etc/network/interfaces
echo "iface eth0 inet dhcp" >> /nfs/srv/etc/network/interfaces

#Eliminar fichero corosync.

rm -r /nfs/srv/etc/default/corosync

echo "*****+*****+*****+*****"
echo "Configuracion completada satisfactoriamente!!*****"
echo "*****+*****+*****+*****"
echo "FIN DEL SCRPT copyNFS.sh*****"
```

Figura 16. Configuración nodo maestro

La figura 16 nos muestra la configuración que se hacen con respecto al de *keyscan. pscp*, la configuración se está aplicando al nodo maestro.

4.2.14 Configuración del servidor PXE

El PXE con sus siglas *Preboot Execution Environment* es un entorno donde se arranca e instala el S.O en una computadora, una vez que se hallan arrancado se procede a la clonación.

```
#Instalacion del servidor PXE.

apt-get install syslinux

cp /usr/lib/syslinux/pxelinux.0 /boot/

#Configuracion del servidor PXE.

mkdir /boot/pxelinux.cfg

echo "DEFAULT linux" > /boot/pxelinux.cfg/default
echo "LABEL linux" >> /boot/pxelinux.cfg/default
echo "KERNEL vmlinuz" >> /boot/pxelinux.cfg/default
echo "APPEND root=/dev/nfs initrd=/initrd_netboot nfsroot=192.168.1.201:/nfs/srv rw" >> /boot/pxelinux.cfg/default

echo "*****+*****+*****+*****"
echo "Configuracion completada satisfactoriamente!!"
echo "*****+*****+*****+*****"
sleep 2s
echo "Fin del script configPXE.sh!!*****"
```

Figura 17. Configuración de servidor PXE

En la figura 17 se muestra la configuración hecha en el nodo maestro como arranque para que pueda comunicarse con los nodos esclavos de clúster

4.2.15 Clonación a los nodos esclavos

A continuación, se muestra la configuración de los nodos esclavos, se realiza una clonación para evitar replicar la instalación uno de los ordenadores.

```
#!/bin/sh

#Clonacion del sistema a los nodos servidores.

#Particionado de los discos de los servidores.
sfdisk -d /dev/sda > /nfs/srv/root/sda.out
sleep 1s

#Creacion tabla de particiones.
./psh "sfdisk -f /dev/sda < /root/sda.out"
sleep 1s

#Formatear la particion de arranque.
./psh "mkfs /dev/sda8"
sleep 1s

#Preparacion de la swap.
./psh "mkswap /dev/sda1"
./psh "swapon /dev/sda1"
sleep 1s

#Clonacion del sistema a los nodos servidores.
./psh "mkdir /mnt/sda8"
./psh "mount /dev/sda8 /mnt/sda8"
./psh "cp -ax / /mnt/sda8"
sleep 2s

echo "*****+*****+*****+*****"
echo "Configuracion completada satisfactoriamente!!*****"
echo "*****+*****+*****+*****"
echo "Fin del script servCLONE.sh...*****"
```

Figura 18. Clonación a nodos esclavos

En la figura 18 se clona los nodos esclavos

4.2.16 Configuración nodos esclavos

```
#Configuracion adicional de los nodos servidores y del nodo maestro.

#Configuracion del fichero /etc/fstab para los nodos servidores.
echo "/dev/sda8 / ext4 defaults 0 0" >> /root/fstab
echo "/dev/sda1 none swap auto,defaults 0 0" >> /root/fstab

#Copia del fichero /root/fstab en los nodos servidores.
./pscp /root/fstab /mnt/sda8/etc

#Configuracion del fichero /etc/network/interfaces de los nodos servidores.
echo "auto lo" >> /root/interfaces
echo "iface lo inet loopback" >> /root/interfaces
echo "auto eth0" >> /root/interfaces
echo "iface eth0 inet dhcp" >> /root/interfaces

#Copia del fichero interfaces en los nodos servidores.
./pscp /root/interfaces /mnt/sda8/etc/network/interfaces

./psh "rm /mnt/sda8/etc/udev/rules.d/70-persistent-net.rules"
sleep 2s

#Instalacion del cargador de arranque en los servidores.
./psh "/root/grub.sh"

#Asignacion del nombre del maquina a los servidores.
./hnode.sh

#Reinicio de la utilidad dnsmasq en modo DHCP.
./start-dhcp

#Reinicio de todos los nodos del cluster TFM.
./psh "umount /mnt/sda8"
./psh reboot

echo "*****+*****+*****+*****"
echo "Fin del script configNODE.sh...*****"
echo "*****+*****+*****+*****"
```

Figura 19. Configuración nodos esclavo

En la figura 19 se configura los nodos esclavos del clúster.

4.2.17 Configuración del nuevo nodo agregado paso 1

El clúster de alto rendimiento es un proyecto escalable, esto quiere decir que podemos incrementar su poder computacional incluyendo nuevos nodos esclavos, la configuración para agregar nuevos nodos son la siguiente.

```
#Preparacion del nodo Master para agregar un nuevo nodo al cluster TFM.
#
#
#Editamos el fichero /etc/hosts para que incluya el nuevo nodo.
#
echo "192.168.1.20$1      server$1      server$1.cluster" >> /etc/hosts

#Editamos el fichero /etc/ethers para que incluya el nuevo nodo.
#
echo "$2:$3:$4:$5:$6:$7 192.168.1.20$1" >> /etc/ethers

#Arrancamos el servidor en modo DHCP + PXE.
#
./start-dhcp-pxe

echo "*****+*****+*****+*****"
echo "Operacion realizada satisfactoriamente!!"
echo "Fin del script prenode.sh..."
```

Figura 20. Configuración de nuevos nodos esclavo paso 1

En la figura 20 Configuración de nuevos nodos esclavos del clúster.

4.2.18 Configuración del nuevo nodo agregado paso 2

```
#Agregar un nuevo nodo servidor al Cluster TFM.

#Particionado del disco de los servidores.
#sfdisk -d /dev/sda > /nfs/srv/root/sda.out
#sleep 1s

#Creacion de la tabla de particiones.
#ssh cluster$1 "sfdisk -f /dev/sda < /root/sda.out"
#sleep 1s

#Formatear la particion de arranque.
ssh server$1 "mkfs /dev/sda8"
sleep 1s

#Preparacion de la swap.
ssh server$1 "mkswap /dev/sda1"
ssh server$1 "swapon /dev/sda1"
sleep 1s

#Clonacion del sistema a los nodos servidores.
ssh server$1 "mkdir /mnt/sda8"
ssh server$1 "mount /dev/sda8 /mnt/sda8"
ssh server$1 "cp -ax / /mnt/sda8"
sleep 1s
```

Figura 21. Configuración de nuevos nodos esclavo paso 2

En la figura 21 se configura de los nuevos nodos esclavos del clúster

4.2.19 Configuración del nuevo nodo agregado paso 3

```
#Configuracion adicional de los nodos servidores y del nodo Master.

#Copia del fichero /root/fstab en los nodos servidores.
scp /root/fstab server$1:/mnt/sda8/etc/

#Copia del fichero /root/interfaces en los nodos servidores.
scp /root/interfaces server$1:/mnt/sda8/etc/network/interfaces

ssh server$1 "rm /mnt/sda8/etc/udev/rules.d/70-persistent-net.rules"

#Copia del fichero /etc/hosts en los nodos servidores.
scp /etc/hosts server$1:/mnt/sda8/etc/hosts

#Instalacion del cargador de arranque en los servidores.
ssh server$1 "/root/grub.sh"

#Asignacion de nombre de maquina a los servidores.
ssh server$1 "echo server$1 > /mnt/sda8/etc/hostname"

#Reinicio de la utilidad dnsmasq en modo DHCP.
./start-dhcp

#Reinicio del nodo del Cluster IFM.
ssh server$1 "umount /mnt/sda8"
ssh server$1 reboot

echo "*****+*****+*****+*****"
echo "Configuracion completada satisfactoriamente!!*****"
echo "*****+*****+*****+*****"
echo "Fin del script addnode.sh...*****"
```

Figura 22. Configuración de nuevos nodos esclavo paso 3

En la figura 22 se configura de nuevos nodos esclavos del clúster

4.3 Pruebas de rendimiento

Cuando tenemos un HPC operativo, listo para hacer operaciones de alta complejidad se debe tener una estadísticas del rendimiento al utilizar ciertas cantidades de recursos computacionales ,antes de poner en marcha las operaciones de nuestro clúster de alto rendimiento ,debemos conocer su eficiencia, si esta escala por la cantidad de nodos o hay un descenso en utilizar hasta cierta cantidad de ellos, de esta manera se puede verificar los posibles cuellos de botellas que ocurran en las mayorías de HPC por diversas razones entre ellas : Mala comunicación entre los nodos esclavos, ineficiencia por la red de alta velocidad , demonios del S.O etc. Se debe conocer con claridad la eficiencia del HPC y evitar problemas inesperados que ocasionaría una ineficiencia en querer obtener resultados en tiempos establecidos.

En las ciencias computacionales un FLOP (o flas o flor/s) es un acrónimo que significa: "*floating point Operations Per Second*" u Operaciones de punto (o coma) flotante por segundo.

Los *FLOPS* son una medida de rendimiento de una computadora, especialmente en el campo científico, en donde se utiliza mucho las operaciones con datos de tipo flotante, para realizar simulaciones precisas y obtener resultados fidedignos. Podemos equiparar esta medida de rendimiento con los conocidos MIPS (Millones de instrucciones por segundo), que es la medida de rendimiento más usual en las computadoras de escritorio.

4.3.1 Pruebas de rendimiento con simulación de n-cuerpos

Las pruebas de rendimiento que utilizaremos en este proyecto será la simulación de N-CUERPOS, este algoritmo es utilizado para la astrofísica y fundamenta los siguientes puntos:

- Número elevado de cuerpos N (e. g. estrellas), lo que hace que el tiempo de cálculo sea de orden N^2 .
- No existe una solución analítica para este tipo de problemas, es decir no hay aproximación disponible para el cálculo directo de fuerzas.
- Requerimiento de memoria es proporcional a N^2 .

El algoritmo realiza el cálculo de interacciones gravitatorias entre N cuerpos (e.g. estrellas en una galaxia).

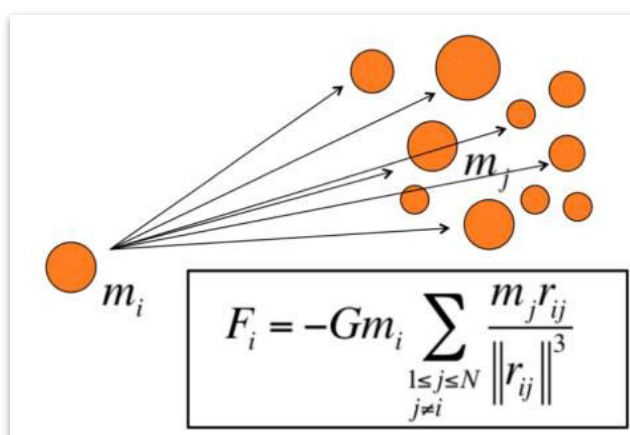


Figura 23. Número de cálculos de fuerzas $N(N-1)/2$

Fuente: (Fiestas, 2014)

Cálculo de fuerzas gravitatorias en un modelo de N cuerpos. F_i es la fuerza calculada por partícula i , m_j es la masa, r_{ij} es la distancia entre partículas vecinas, y G es la constante gravitatoria.

El pseudo código secuencial de cálculo directo de N-CUERPOS se esquematiza en la figura 23, en esta se aprecia el doble bucle en i , j , responsable del cálculo de interacción gravitatoria de todas las partículas.

4.3.2 Algoritmo de pruebas de rendimiento

Se utilizó un algoritmo N-cuerpos para poder realizar nuestras pruebas de rendimiento, el algoritmo está utilizando un bucle donde procesa gran cantidad de información y son resueltas en diferentes tiempos dependiendo la cantidad de nodos esclavos que se utilicen.

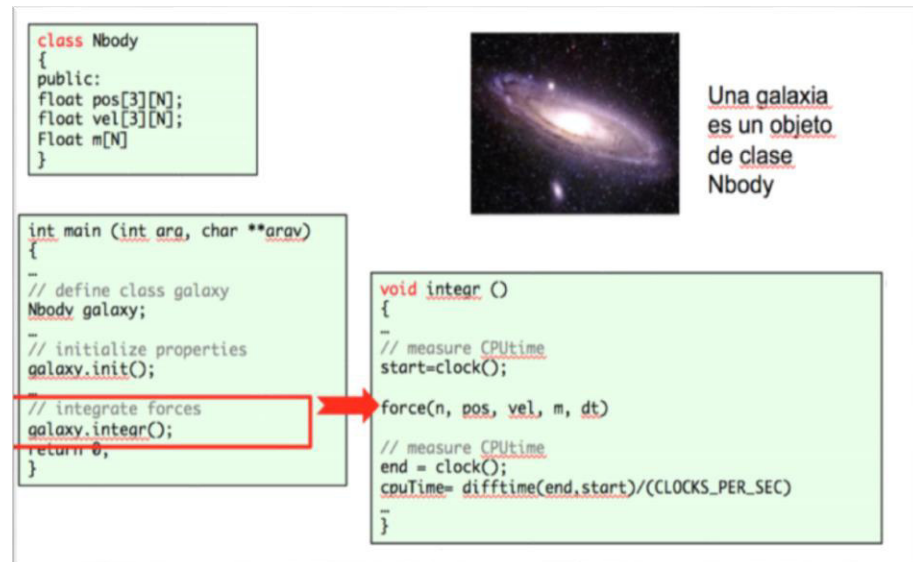


Figura 24. Algoritmo paralelo del cálculo de fuerzas

Fuente: Fiestas, 2011

Cálculo de fuerzas gravitatorias en un modelo de N cuerpos. F_i es la fuerza calculada por partícula i , m_j es la masa, r_{ij} es la distancia entre partículas vecinas, y G es la constante gravitatoria.

El algoritmo utilizó programación paralela, donde su función fue utilizar los nodos esclavos de las computadoras, se utilizaron estas pruebas en 16 computadoras esclavas con la intención de graficar el rendimiento y tiempo que estas tienen al incluir diferentes cantidades de nodos, en el siguiente punto mostraremos los *flops* utilizados en el clúster.

4.3.3 Técnicas de distribución de procesos.

El código utiliza técnicas MPI, el pseudo código con MPI se muestra en la figura 25, en esta se observa el uso de dos directivas MPI. *MPI_CAST* esta es utilizada para distribuir variables N-CUERPOS que son el número total de partículas entre otros procesos, mientras el *MPI_Allreduce* es utilizado para sumar el resultado del cálculo de fuerza de cada proceso y almacenarlo en la variable *force*.

```
void force(int n, float pos[][..],...,float vel[][..], float m[..], float dt)
{
  // sume over i
  for (int i=0; i<n; i++)
  {
    float my_r_x = r_x[i];
    // sume over j
    for (int j=0; j<n; j++)
    {
      if(j!=i) // avoid i=j
      {
        // compute accelerations
        float d = r_x[j]-my_r_x; // 1 flop
        a_x += G*m[j]/(d*d); // 4 flops
      }
    }
    //update velocities
    v_x[i] += a_x*dt; // 2 flops
    // update positions
    r_x[i] += v_x[i]*dt; // 2 flops
  }
}
```

Figura 25. Función de cálculo de fuerzas en código secuencial.

Fuente: Fiestas, 2011

Algoritmo N-Cuerpos con técnicas de distribución de procesos

El software usado para este estudio de rendimiento se llamada phi-GPU, es un código híbrido que está en lenguaje C++, implementado con directivas MPI para hacer el paralelismo efectivo a poder utilizar en supercomputadores, el software ha sido probado en Observatorio Nacional de China (Pekín), en el Centro de Astronomía de Heidelberg (Alemania) y en la Universidad de Berkeley (USA) todas ellas en supercomputadoras de altos requerimientos computacionales.

4.3.4 Objetivo de pruebas de rendimiento

- Obtener resultados reales del rendimiento del equipo a utilizar diversas cantidades de recursos computacionales.
- Saber posibles cambios en utilizar ciertas cantidades de nodos si logra una escalabilidad constante o desciende con una cierta cantidad de recursos.
- Conocer la eficiencia general y específica del HPC, con las diferentes cantidades de nodos establecidas.

4.3.5 Comportamiento de prueba de rendimiento

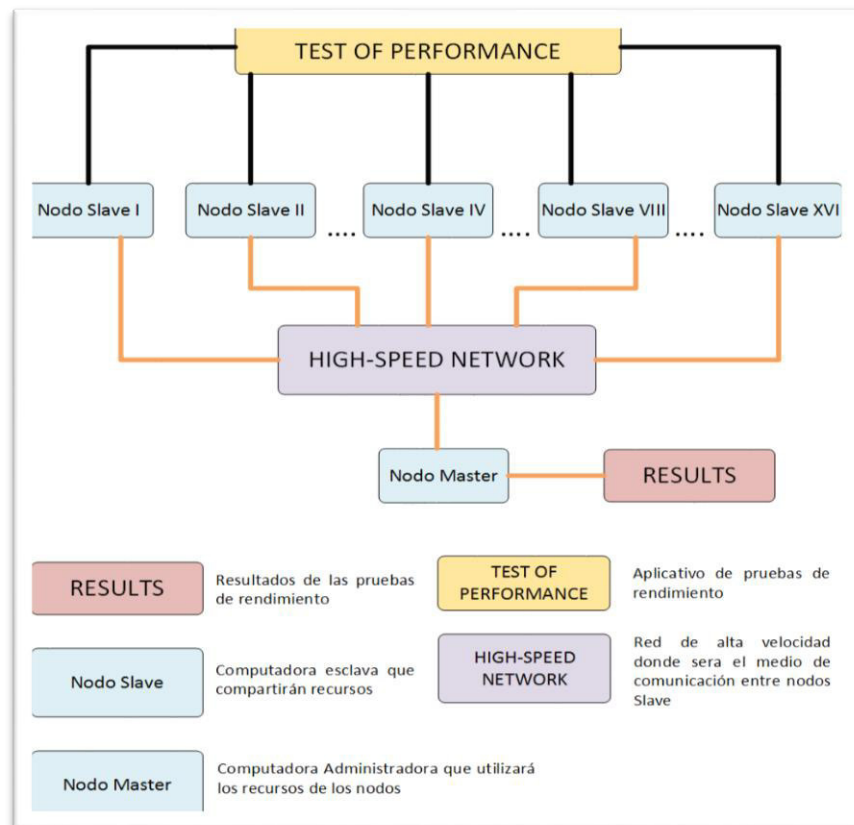


Figura 26. Diagrama de pruebas de rendimiento

En la figura del Diagrama de pruebas de rendimiento que vemos a continuación es el comportamiento que tienen al realizar las pruebas de rendimiento, el bloque "RESULT" son los resultados que se verán desde el nodo Master.

4.3.6 Obtención de operaciones de coma flotante por segundo con pruebas de rendimiento

Tabla 3. Mediciones de *flops*

Nombre	Flops
megaflops	10^6
gigaflops	10^9
teraflops	10^{12}
petaflops	10^{15}
exaflops	10^{18}

En la siguiente tabla vemos las mediciones de cantidad de los *Floating point Operations Per Second*, son las medidas de rendimiento de las computadoras mientras más *flops* existan el cálculo científico es mejor.

Las operaciones de coma flotante por segundo son una medida del rendimiento de una computadora, especialmente para el cálculo científico que requiere un gran uso de operaciones de coma flotante por segundo, su acrónimo es más conocido como *flops*. Las pruebas de rendimiento que utilizaremos nos ayudará a obtener los *flops* correspondiente por cada cantidad de nodo que utilicemos y la cantidad de operaciones que demos a estas, se desea obtener resultados y observar su escalabilidad a incrementar más cantidad de nodos. Obtener los resultados de los *flops* servirá para conocer el potencial computacional del HPC, utilizar cantidades de nodos exactas para aplicativos que no requieren utilizar todos sus nodos para obtener resultados en corto tiempos, ahorrar el trabajo máquina que estas producen a utilizar todos sus recursos computacionales.

4.3.7 Obtención de tiempos con pruebas de rendimiento

Para obtener estadística del tiempo que demora en obtener un resultado con diferentes cantidades de nodos, debemos hacer pruebas de rendimiento utilizando diferentes niveles de complejidad, de esta manera obtendremos resultados del tiempo en utilizar ciertas cantidades de nodos establecidos. La manera que conseguiremos estos resultados es utilizar el potencial de los nodos esclavos, utilizando desde el tiempo que toma en calcular con un nodo y finalmente utilizando los 16 nodos a la vez obteniendo resultados distintos.

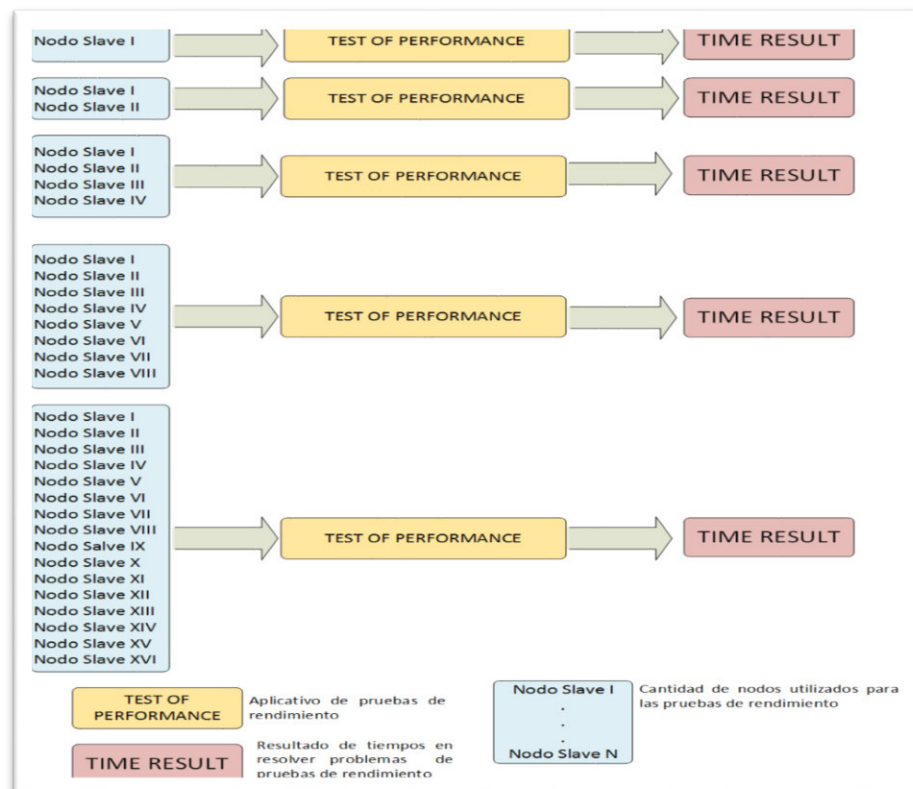


Figura 27. Diagrama prueba de rendimiento (Tiempo)

En la siguiente figura vemos las pruebas de rendimiento utilizadas por ciertas cantidades de nodos, esto es utilizado con intención de ver el tiempo que demora en resolverse con cantidades de nodos establecidas.

4.3.8 Metodología en pruebas de rendimiento

- Se utilizó un aplicativo de simulación en dinámica de cuerpos para obtener tiempos y *Gigaflops* con cantidades de cuerpos establecidos.
- utilizamos una cierta cantidad de cuerpos que serán desde 8000 hasta 128000 partículas en diferentes cantidades de nodos.
- Se organizó los datos obtenidos en tablas estadísticas para la siguiente cantidad de Nodos: Nodo I, Nodo II, Nodo IV, Nodo VIII, Nodo XVI.
- Adjuntaremos los datos obtenidos y compararemos los datos estadísticos con todos los nodos utilizados.

Tabla 4. Cantidad de cuerpos por nodos

Cantidad de nodos	Cantidad de Cuerpos				
1	8000	16000	32000	64000	128000
2	8000	16000	32000	64000	128000
4	8000	16000	32000	64000	128000
8	8000	16000	32000	64000	128000
16	8000	16000	32000	64000	128000

En la siguiente tabla vemos las cantidades de cuerpos que se utilizaron por nodos esclavos, estas son las cantidades que utilizamos para hacer las pruebas de rendimiento utilizando N-CUERPOS.

A. Rendimiento de *gigaflops* vs nodos

Tabla 5. Estadística de *Gigaflops* vs Cantidad de nodos

NODOS	GIGAFLOPS				
	CANTIDAD DE CUERPOS CELESTES				
	8K	16K	32K	64K	128K
1	1.281	2.664	3.128	6.584	15.144
2	1.327	2.633	4.961	8.114	18.181
4	1.326	2.621	5.114	9.199	19.187
8	1.325	2.644	5.22	9.88	20.421
16	1.549	2.716	5.42	11.77	21.421

En la siguiente tabla se observa los *GIGAFLOPS* obtenidos por cantidades de nodos, con una cantidad de 16 nodos utilizando 128000 cuerpos vemos que obtiene una cantidad de 21.421 *Gigaflops*, estos *flops* son los que se utilizarán para resolver procesos de alta complejidad.

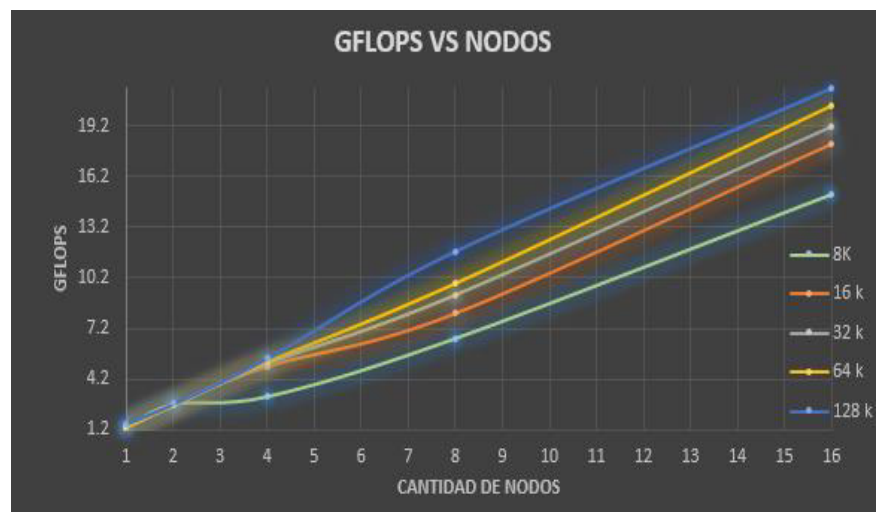


Figura 28. Prueba de rendimiento *Gigaflops* VS Nodos

En la siguiente figura de Prueba de rendimiento *Gigaflops* VS Nodos observamos las pruebas de rendimiento, la cantidad de cuerpos están distribuidas por 8k ,16k, 32k, 64 k y 128k esta gráfica resalta la eficiencia en utilizar más cantidad de nodos para resultados más rápidos por la cantidad *flops* utilizadas.

B. Rendimiento utilizando 1 nodo

Tabla 6. Estadística de Tiempo, N-cuerpos y *Gigaflops* con 1 nodo

TIEMPO	NODOS	N-CUERPOS	GIGAFLOPS
666.3	1	8000	1.281
3402.1	1	16000	1.327
5848.9	1	32000	1.326
14548.9	1	64000	1.325
28448.9	1	128000	1.549

En la siguiente tabla nos muestra el tiempo en obtener resultados con 1 nodo utilizando diferentes cantidades de N-CUERPOS, como observamos en un tiempo de 28448.9 segundos utilizando 1 nodo con 128000 N-CUERPOS se consiguió una cantidad de 1.549 *gigaflops*, mientras más cantidad de procesos se incrementarán más los *gigaflops*

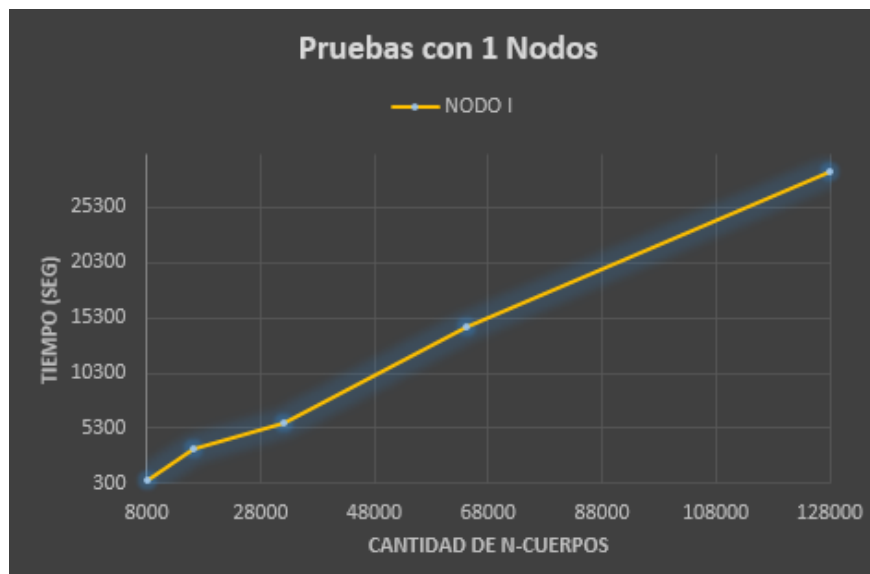


Figura 29. Prueba de rendimiento TIEMPO VS N-CUERPO con 1 Nodo

La siguiente figura nos muestra el tiempo en segundos en que resuelve el problema con 1 Nodo, se muestra que el tiempo que este resuelve es 28448.9 segundos.

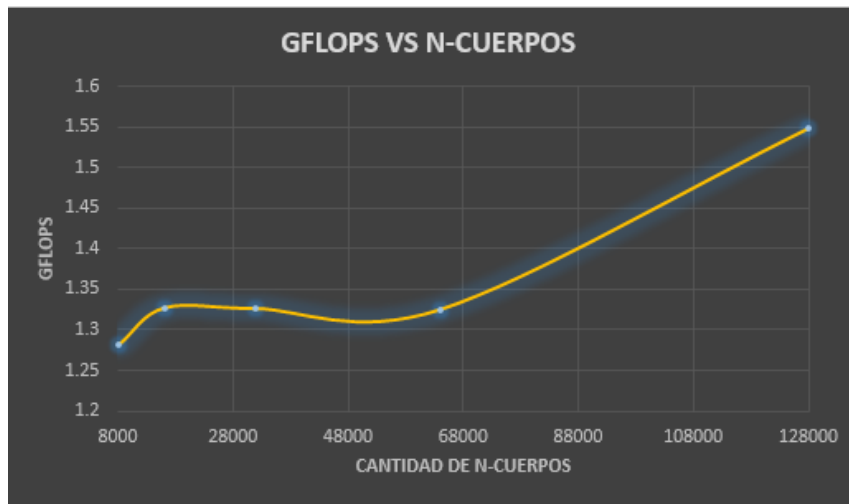


Figura 30. Prueba de rendimiento GFLOPS VS N-CUERPO con 1 Nodo

La siguiente figura nos muestra *gigaflops* utilizando diversa cantidad de N-CUERPOS con 1 Nodo, apreciamos que puede ocurrir un desequilibrio en el aumento de *flops* debido a los cuellos de botellas que se ocasionan por la red de alta velocidad.

C. Rendimiento utilizando 2 nodos

Tabla 7. Estadística de Tiempo, N-cuerpos y Gigaflops con 2 nodos

TIEMPO	NODOS	N-CUERPOS	GIGAFLOPS
331.52	2	8000	2.664
1713.4	2	16000	2.633
3858.8	2	32000	2.621
8807.2	2	64000	2.644
19807.2	2	128000	2.716

En la siguiente tabla nos muestra el tiempo en obtener resultados con 2 nodos utilizando diferentes cantidades de N-CUERPOS, podemos ver que utilizando 2 nodos con 128000 cuerpos podemos obtener 2.716 gigaflops resolviéndolo en un tiempo de 19807.2 segundos.

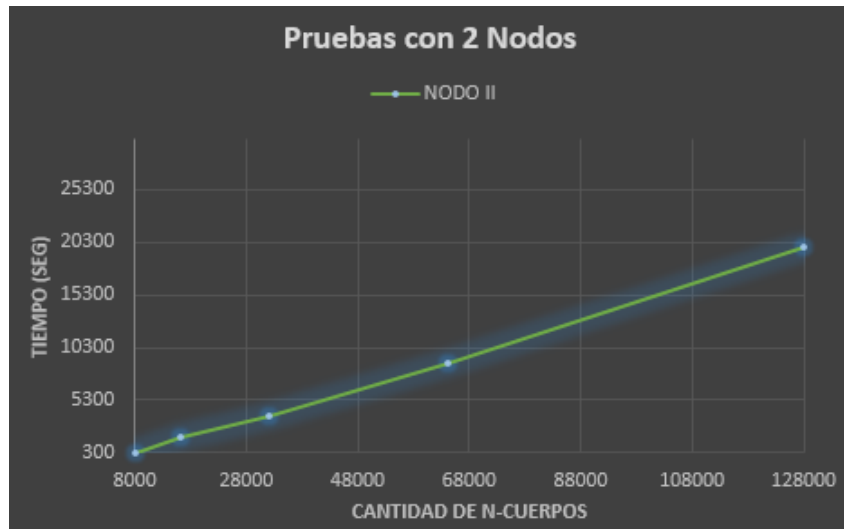


Figura 31. Prueba de rendimiento TIEMPO VS N-CUERPO con 2 Nodos

La siguiente figura nos muestra el tiempo en segundos en que resuelve con 2 Nodos, vemos en la gráfica un crecimiento escalable sin tener problemas con los cuellos de botella.

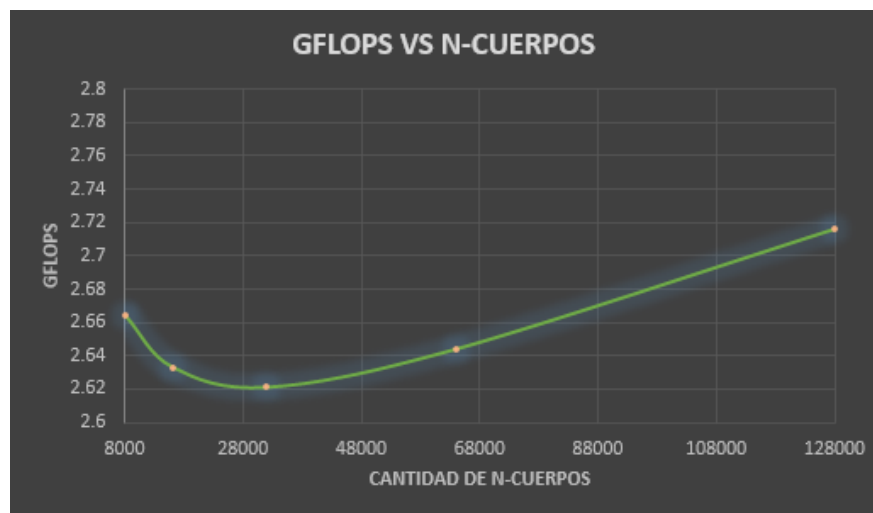


Figura 32. Prueba de rendimiento GFLOPS VS N-CUERPO con 2 Nodos

La siguiente figura nos muestra *gigaflops* utilizando diversa cantidad de N-CUERPOS con 2 Nodos, en la siguiente gráfica observamos una inestabilidad al calcular dichos resultados debido a la red comunicación.

D. Rendimiento utilizando 4 nodos

Tabla 8. Estadística de Tiempo, N-cuerpos y *Gigaflops* con 4 nodos.

TIEMPO	NODOS	N-CUERPOS	GIGAFLOPS
313.3	4	8000	3.128
909.92	4	16000	4.961
2488.7	4	32000	5.114
4458.3	4	64000	5.22
10458.3	4	128000	5.42

En la siguiente tabla nos muestra el tiempo en obtener resultados con 4 nodos utilizando diferentes cantidades de N-CUERPOS, vemos que utilizando una cantidad de 4 nodos con 128000 cuerpo se consiguió obtener 5.42 *gigaflops*.

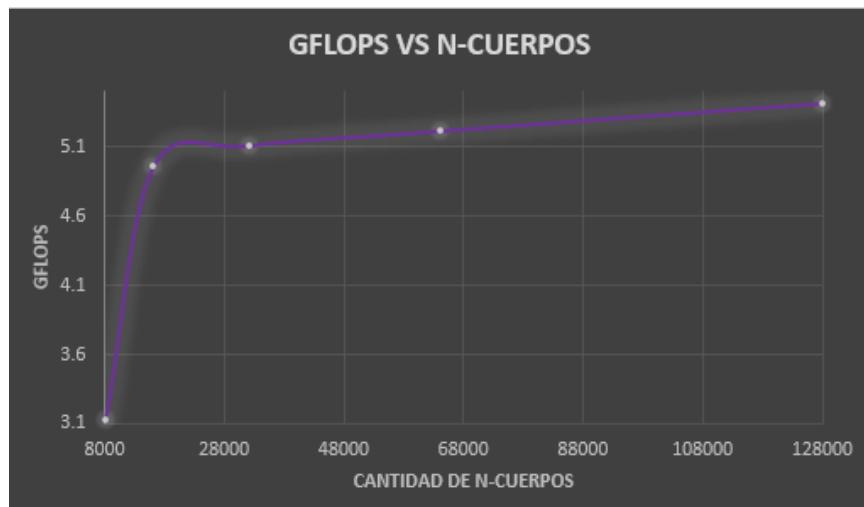


Figura 33. Prueba de rendimiento GFLOPS VS N-CUERPO con 4 Nodos

La siguiente figura nos muestra *gigaflops* utilizando diversa cantidad de N-CUERPOS con 4 nodos, en la gráfica vemos un crecimiento considerable mientras más tiempo transcurre.

E. Rendimiento utilizando 8 nodos

Tabla 9. Estadística de Tiempo, N-cuerpos y Gigaflops con 8 nodos.

TIEMPO	NODOS	N-CUERPOS	GIGAFLOPS
148.48	8	8000	6.584
556.2	8	16000	8.114
1102.7	8	32000	9.199
2355.4	8	64000	9.88
4355.4	8	128000	11.77

En la siguiente tabla nos muestra el tiempo en obtener resultados con 8 nodos utilizando diferentes cantidades de N-CUERPOS, se observa que utilizando 128000 cuerpos se logra obtener 11.77 *gigaflops* en un tiempo de 4355.4 observando desde esta cantidad de nodos el tiempo se reduce a la mitad al realizar el proceso con la misma cantidad de N-CUERPOS que se realizó en la tabla 8.

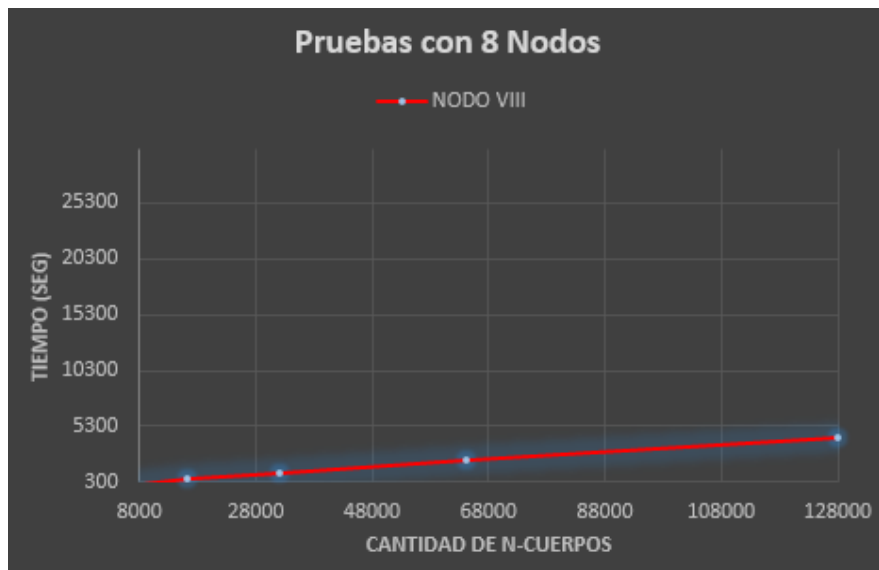


Figura 34. Prueba de rendimiento TIEMPO VS N-CUERPO con 8 Nodos

La siguiente figura nos muestra el tiempo en segundos en que resuelve con 8 nodos, observamos en la gráfica que el crecimiento es escalable.

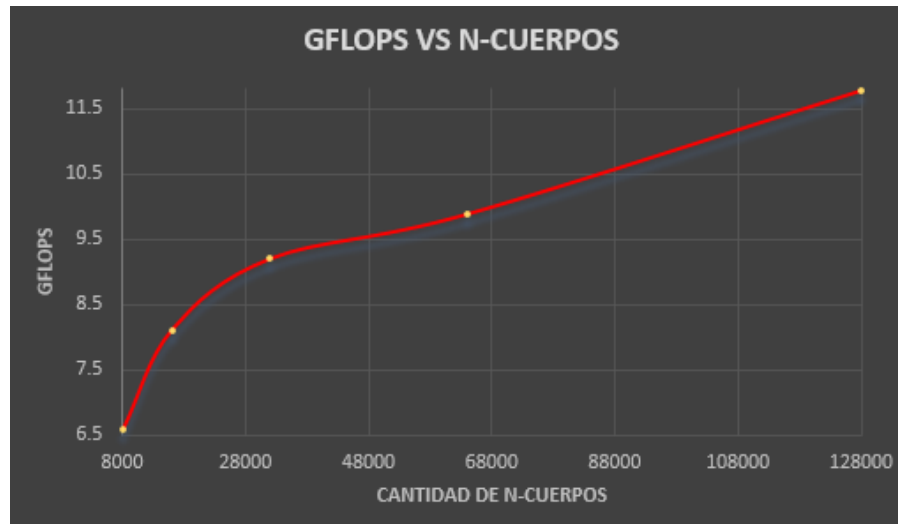


Figura 35. Prueba de rendimiento GFLOPS VS N-CUERPO con 8 Nodos

La siguiente figura nos muestra *gigaflops* utilizando diversa cantidad de N-CUERPOS con 8 Nodos, se observa que el incremento de gigaflops no llega a descender indicándonos que no hay problemas de comunicación con los nodos.

F. Rendimiento utilizando 16 nodos

Tabla 10. Estadística de Tiempo, N-cuerpos y *Gigaflops* con 16 nodos

TIEMPO	NODOS	N-CUERPOS	GIGAFLOPS
138.82	16	8000	15.144
548.54	16	16000	18.181
732.83	16	32000	19.187
948.14	16	64000	20.421
2048.4	16	128000	21.421

En la siguiente tabla nos muestra el tiempo en obtener resultados con 16 nodos utilizando diferentes cantidades de N-CUERPOS, observamos que con 16 nodos con 128000 cuerpos utiliza 21.4214 y el tiempo para resolver este problema es de 2048.4 segundos, un incremento considerable si lo comparamos con la tabla 7 que el tiempo fue 4355.4 segundos, sacando una conclusión que el tiempo sí se reduce utilizando más cantidad de nodos esclavos.

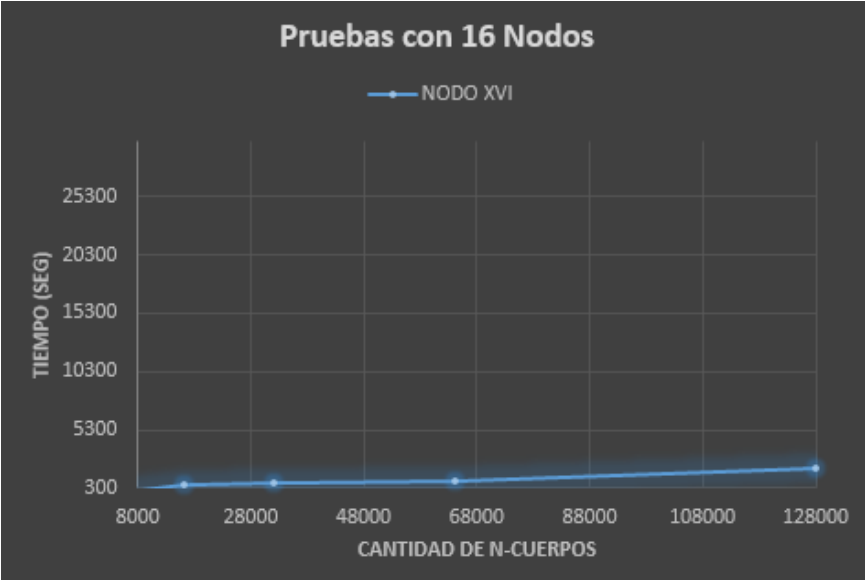


Figura 36. Prueba de rendimiento TIEMPO VS N-CUERPO con 16 Nodos

La siguiente figura nos muestra el tiempo en segundos en que resuelve con 16 Nodos, vemos en la gráfica que el tiempo que resuelve un problema es mínimo por ello podemos ver que en curva el crecimiento no cambia mucho.

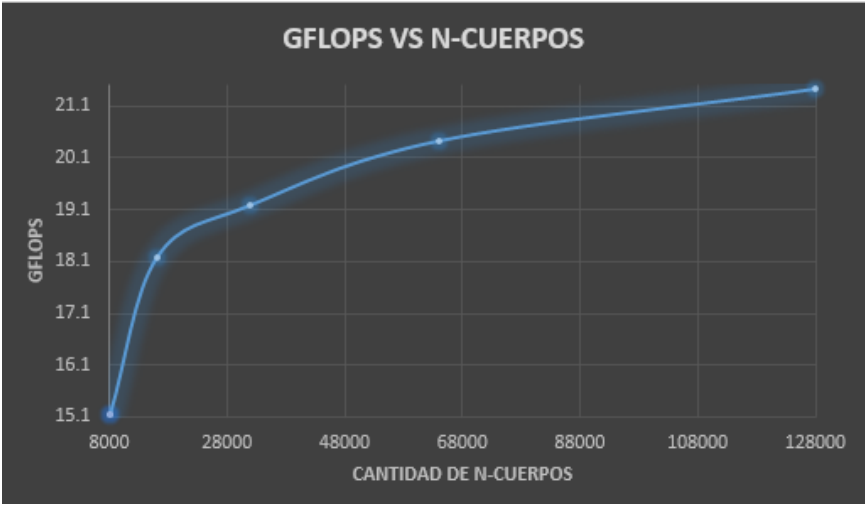


Figura 37. Prueba de rendimiento GFLOPS VS N-CUERPO con 16 Nodos.

La siguiente figura nos muestra *gigaflops* utilizando diversa cantidad de N-CUERPOS con 16 Nodos, observamos el crecimiento crece considerablemente.

G. Comparación de rendimiento escalable

Para saber su eficiencia, hemos adjuntado todos los nodos en una sola tabla estadística comprobando el tiempo que demora en resolver ciertas cantidades de N-CUERPOS.



Figura 38. Prueba de rendimiento TIEMPOS VS N-CUERPOS

En la siguiente figura vemos el tiempo que toma en resolver un problema con diferentes cantidades de N-Cuerpos con distintas cantidades de nodos, tenemos desde utilizar un nodo hasta utilizar 16 de ellos, en la figura se ve una diferencia gigante en el tiempo en utilizar más nodos.

H. Tiempos Vs Nodos

A continuación, mostraremos información correspondiente de Tiempo VS Nodos con el objetivo de poder medir el tiempo que el cluster demora en resolver problema con cantidad de nodo esclavos distintos

Tabla 11. Tiempos Vs Nodos

CUERPOS	1 nodo	2 nodos	4 nodos	8 nodos	16 nodos
8k	666.3 Seg	331.52 Seg	313.3 Seg	148.48 Seg	138.82 Seg
16k	3402.1 Seg	1713.4 Seg	909.92 Seg	556.2 Seg	548.54 Seg
32k	5848.9 Seg	3858.8 Seg	2488.7 Seg	1102.7 Seg	732.83 Seg
64k	14548.9 Seg	8807.2 Seg	4458.3 Seg	2355.4 Seg	948.14 Seg
128k	28448.9 Seg	19807.2 Seg	10458.3 Seg	4355.4 Seg	2048.4 Seg

En la Tabla 11 nos muestra el tiempo que demora en resolver un proceso con respecto a la cantidad de nodos que se le incluya, como vemos en la tabla a mayor cantidad de nodos el tiempo se reduce.

a. Tiempo vs nodos con 8000 cuerpos

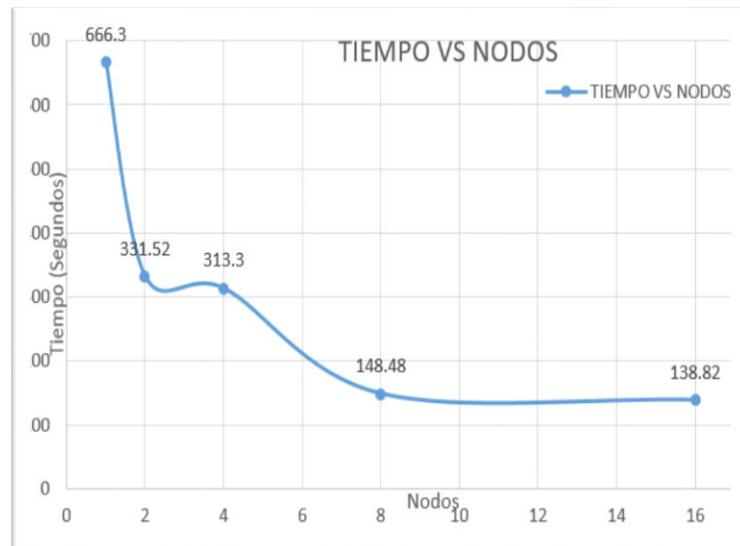


Figura 39. Tiempos vs Nodos utilizando 8000 cuerpos

Figura que muestra el tiempo que demora en resolver un problema con 8000 cuerpos celestes.

b. Tiempo vs nodos con 16000 cuerpos

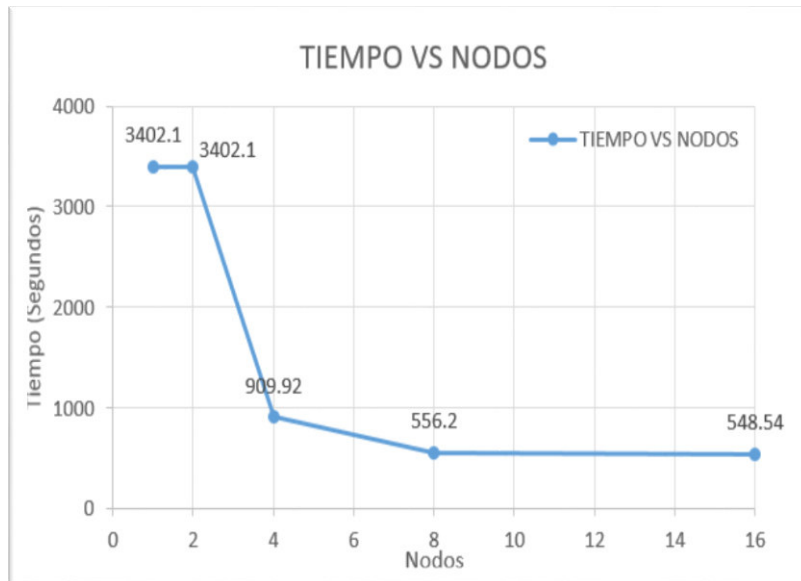


Figura 40. Tiempos vs Nodos utilizando 16000 cuerpos.

Figura que muestra el tiempo que demora en resolver un problema con 16000 cuerpos celestes.

c. Tiempo vs nodos con 32000 cuerpos

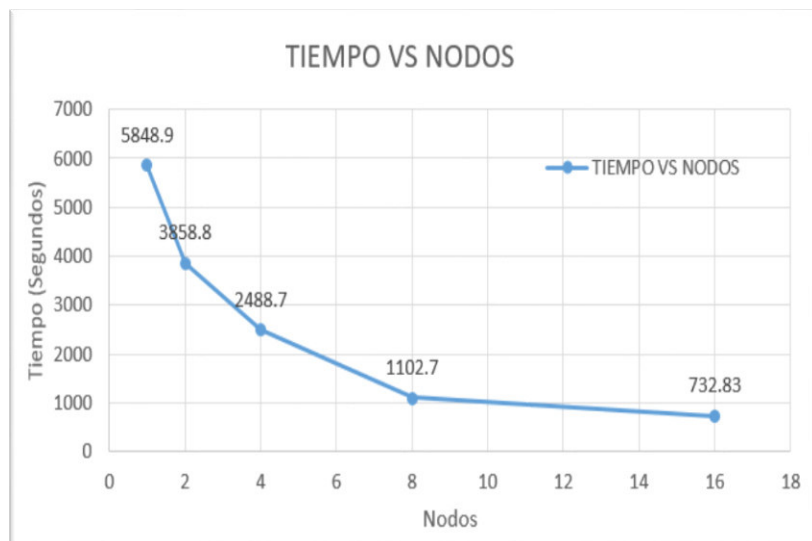


Figura 41. Tiempo vs Nodo utilizando 32000 cuerpos

Figura que muestra el tiempo que demora que demora en resolver un problema con 32000 cuerpos celestes.

d. Tiempo vs nodos con 64000 cuerpos

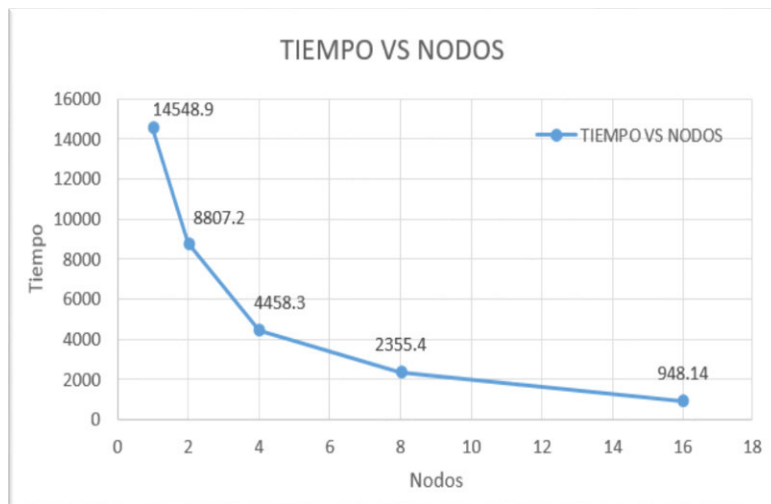


Figura 42. Tiempos vs Nodos con 64000 cuerpos

Figura que muestra el tiempo que demora que demora en resolver un problema con 64000 cuerpos celestes.

e. Tiempo vs nodos con 128000 cuerpos

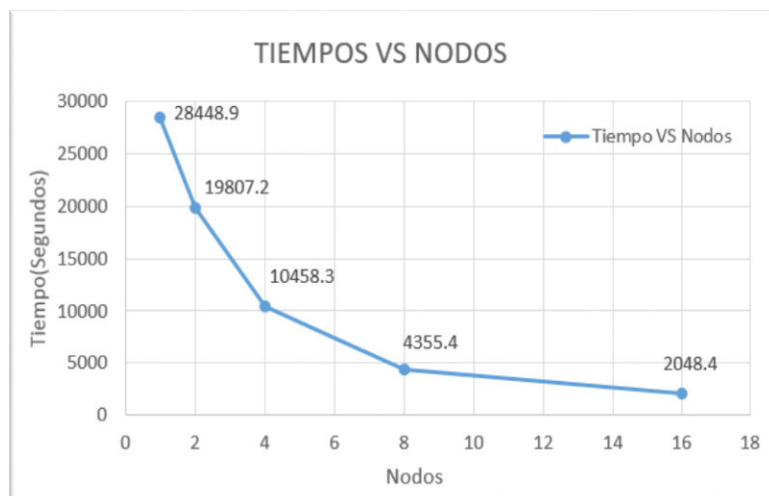


Figura 43. Tiempo vs Nodos con 128000 cuerpos

Figura que muestra el tiempo que demora que demora en resolver un problema con 128000 cuerpos celestes

5.2 Identificación de cuellos de botellas en el HPC

Cuando se mencionan los cuellos de botellas se refieren a las actividades que disminuyen la velocidad de los procesos, incrementan los tiempos de espera y reducen la productividad, trayendo como consecuencia final diversos problemas. En los puntos anteriores con respecto a pruebas de rendimiento podemos tener un alcance donde ocurrirían estos desbalances y cuantos nodos son adecuados a ser utilizados.

Gracias a las pruebas de rendimiento podemos utilizar un cálculo matemático y precisar la cantidad adecuada de nodos a utilizar en nuestro HPC, este resultado será ideal y nos da una gran posibilidad de evitar cuellos de botellas posteriores que ocurran por diversas razones.

5.2.1 Cálculo matemático

Aplicaremos un cálculo matemático para poder obtener los posibles de cuellos de botella que podríamos encontrar en el clúster

```
Rank of the processor 02 on slave05
Rank of the processor 03 on slave06
Rank of the processor 00 on slave03
Rank of the processor 01 on slave04

Begin the calculation of phi-GPU4 program on 004 processors

N      = 065536      eps      = 1.000000E-04
t_beg  = 0.000000E+00  t_end  = 2.500000E-01
dt_disk = 1.000000E+00  dt_contr = 1.250000E-01
eta     = 1.500000E-01  eta_BH  = 1.500000E-01

Initialized, 6.016456 Gflops
0.0000E+00  0.000E+00  0.000E+00  -4.966682E-01  2.485637E-01  -2.481044E-01  -0.000000E+00  4.264118E+01
1.2500E-01  5.899E+03  2.970E+06  -4.964831E-01  2.483762E-01  -2.481069E-01  9.878770E-06  2.229299E+03
2.5000E-01  1.165E+04  5.918E+06  -4.963434E-01  2.482365E-01  -2.481069E-01  3.472802E-10  4.436249E+03

Timesteps = 11653  Total sum of integrated part. = 5918200  n_act average = 508

Real Speed = 5.220 GFlops
      sec_tot  usec/step  ratio
scan : 4.4797E-01  3.8442E+01  1.0048E-04
pred : 6.0487E+00  5.1907E+02  1.3567E-03
jsend: 1.8589E-03  1.5953E-01  4.1696E-07
isend: 0.0000E+00  0.0000E+00  0.0000E+00
force: 4.3646E+03  3.7455E+05  9.7898E-01
recv : 0.0000E+00  0.0000E+00  0.0000E+00
comm : 8.4598E+01  7.2598E+03  1.8975E-02
corr : 2.6308E+00  2.2577E+02  5.9010E-04
tot  : 4.4583E+03  3.8259E+05  1.0000E+00
```

Figura 44. Ejemplo de obtención de resultados en pruebas de rendimiento.

La Figura nos muestra un ejemplo de que datos extraeremos de las pruebas de rendimiento.

El cálculo que se aplicará es el siguiente:

$$\text{Total_Computo} = (\text{tot} - \text{comm})^2$$

$$\text{Total_Computo} = (4.583\text{E} + 03 - 8.4598\text{E} + 01)^2$$

$$\text{Total_Computo} = 4.50\text{E} + 03^2\text{jk}$$

Cuando “*Total_Computo*” interceptan con “*Tot*” encontramos la cantidad de nodos adecuados a utilizar en el HPC pasado esta intersección, hay gran posibilidad que ocurra un cuello de botella.

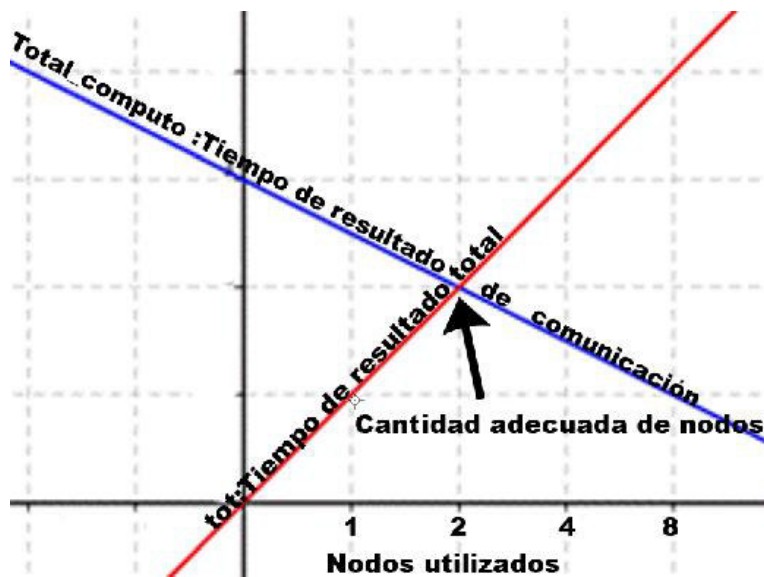


Figura 45. Ejemplo de utilización adecuada de nodos

En la siguiente figura vemos un ejemplo en cómo encontrar las cantidades adecuadas de nodos a utilizar y los posibles cuellos de botellas, el punto de intersección nos indica la cantidad adecuada de nodos que se debe utilizar pasada esa cantidad hay la probabilidad de que ocurran los cuellos de botella.

A continuación, los datos extraídos en las pruebas de rendimiento, con los 16 nodos disponibles calcularemos la intersección y los posibles cuellos de botella pasadas el rango explicado en la figura 45, las siguientes tablas son los datos de *tot*, *comm* y el *Total_computo*, que son los valores para poder calcular la intersección y encontrar cuellos de botellas en el clúster de alto rendimiento

Tabla 12. Resultados 1 Nodo prueba de rendimiento

NODO 1	tot	comm	Total Cómputo
8k	7.65E+02	7.34E-02	7.65E+02
16k	3.40E+03	1.40E-01	3.40E+03
32k	7.65E+03	2.75E-01	7.65E+03

La Tabla 12 nos muestra el total de cómputo que realiza 1 nodo con diferentes cantidades de cuerpos celestes

Tabla 13. Resultados 2 Nodos prueba de rendimiento

NODOS 2	tot	comm	total Cómputo
8k	3.32E+02	1.40E+01	3.18E+02
16k	1.71E+03	4.08E+01	1.67E+03
32k	3.86E+03	3.93E+01	3.82E+03

La tabla 13 nos muestra el total de cómputo que realiza 2 nodos con diferentes cantidades de cuerpos celestes

Tabla 14. Resultados 4 Nodos prueba de rendimiento

NODOS 4	tot	comm	total Computó
8k	3.13E+02	6.87E+01	2.45E+02
16k	9.10E+02	7.18E+01	8.38E+02
32k	1.98E+03	7.98E+01	1.90E+03

La tabla 14 nos muestra el total de cómputo que realiza 4 nodos con diferentes cantidades de cuerpos celestes

Tabla 15. Resultados 8 Nodos prueba de rendimiento

NODOS 8	Total	Comm	total Cómputo
8k	1.48E+02	6.85E+01	8.00E+01
16k	5.56E+02	1.70E+02	3.86E+02
32k	1.10E+03	2.82E+02	8.21E+02

La tabla 15 nos muestra el total de cómputo que realiza 8 nodos con diferentes cantidades de cuerpos celestes

Tabla 16. Resultado 16 Nodos prueba de rendimiento

NODOS 16	Total	Comm	Total_Computo
8k	1.39E+02	9.23E+01	4.65E+01
16k	5.49E+02	3.57E+02	1.91E+02
32k	7.33E+02	3.57E+02	3.75E+02

La tabla 16 nos muestra el total de cómputo que realiza 16 nodos con diferentes cantidades de cuerpos celestes

CUELLO DE BOTELLA

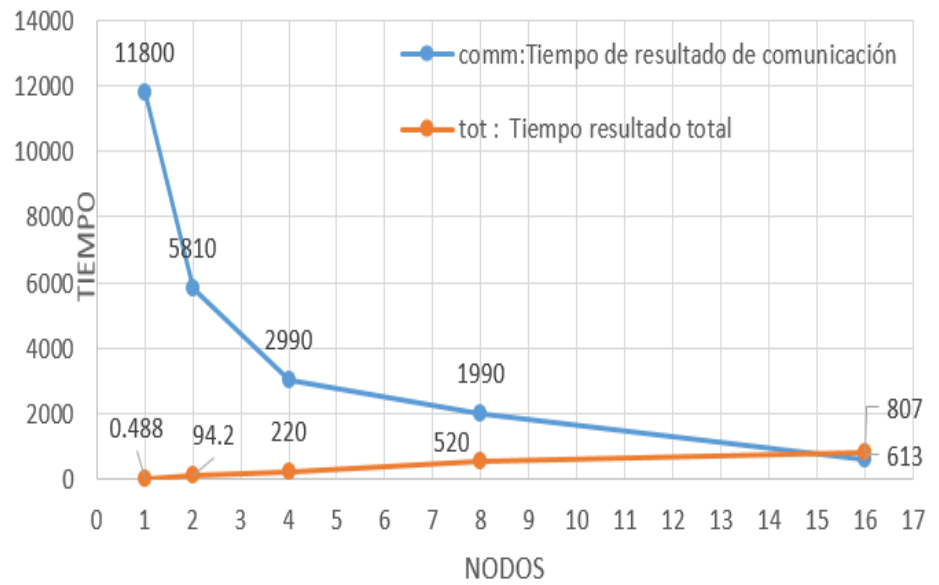


Figura 46. Cuello de botella

En la siguiente figura en la intersección entre el comm y el tot, viene hacer la cantidad ideal en utilizar los nodos, pasada esa intersección ocurren los cuellos de botella, esto quiere decir que la escalabilidad bajaría utilizando más cantidad de los recursos utilizados con anterioridad.

CAPITULO V: ANÁLISIS COSTO Y BENEFICIO

5.1 ANÁLISIS DE COSTOS

5.2.1 Recursos humanos

El financiamiento otorgado por el FONDECYT incluye el costo de mano de obra, con ello se pudieron cubrir diferentes gastos utilizado en la implementación del proyecto.

Tabla 17: Costos de personal

DESCRIPCION	HORAS	MESES	MONTO
Tesis Investigador	1440	12	S/12,000.00
Integración de componentes general de hardware.	100	1	S/1000.00
TOTAL			S/ 13,000.00

En la siguiente tabla nos muestra el costo de la investigación del clúster de alto rendimiento y la mano de la implementación de todos los componentes de hardware.

5.2.2 Recursos de hardware

Tabla 18: Costo de hardware

Descripción hardware	Cantidad	Precio unitario	Monto
Procesadores Core i7	8	S/560.00	S/4,480.00
Memoria Ram 7 Gb	8	S200.00	S /640.00
Placa madre	8	S/350.00	S/2800.00
Fuente de poder	8	S/180.00	S/1440.00
Cables UTP	50	S/.10	S/500.00
Switch	1	S/.1800	S/.1800
Sistema de ventilación	1	S/.220	S/.220
Armario racks	1	s/.1000	s/.1000
TOTAL			S/12,880.00

5.2.3 Recursos de software

Tabla 19: Costo de software

Nombre de software	Descripción	Cantidad	Precio unitario	Monto
S.O Open Source CENTOS	Software libre.	8	S/0.00	S/0.00
Drivers de componentes de Hardware	Driver de reconocimiento de hardware.	8	S/80.00	S /640.00
TOTAL				s/640.00

En la siguiente tabla nos muestra el costo de software utilizados en la implementación del proyecto.

5.2.4 Otros gastos

Tabla 20:Otros gastos

Descripción	Precio unitario
Impresión de documentos y tramites de formulación del proyecto	S/1900.00
Gasto de exposiciones UCH, URP y UNI sobre el proyecto de tesis.	S/1000.00
TOTAL	S/2900.00

En la siguiente tabla nos muestra otros gastos en la implementación del proyecto.

5.2.5 Costo total

Tabla 21:Costo total

Descripción	Precio unitario
Recursos humanos	S/13,000.00
Recursos de hardware	S/12,880.00
Recursos de Software	s/640.00
Otros gastos	S/2900.00
TOTAL	S/29,420.00

En la siguiente tabla nos muestra gasto total en la implementación del clúster de alto rendimiento.

5.2 ANÁLISIS DE BENEFICIO

5.2.1 Beneficios intangibles

Luego de haber realizado pruebas de rendimiento con un software escalable y haber utilizado procedimientos matemáticos, para encontrar donde nuestros nodos ya no escalan, podemos encontrar beneficios intangibles en la implementación del proyecto, en la siguiente tabla vemos los beneficios intangibles con la implementación del HPC y sin ella.

Tabla 22: Beneficios intangibles

BENEFICIOS	CON HPC	SIN HPC
Disminuir el tiempo de procesos de alta complejidad utilizando algoritmos paralelos	2048.4 segundos en dar un resultado de movimiento de cuerpos celestes.	28448.9 segundos en dar un resultado de movimiento de cuerpos celestes
Disminuye los cuellos de botellas al querer hacer procesos de alta complejidad.	15.4 segundos en resolverse el error con el clúster de alto rendimiento.	28000 segundos en resolverse el error
Beneficia la toma de decisiones más rápida debido a los resultados en obtenido en muy poco tiempo.	Si	No
Los procesos de alta complejidad son resueltos óptimamente gracias a la técnica de paralelismo computacional.	Con la implementación de un cluster de alto rendimiento podemos utilizado técnicas de paralelismo.	Con un ordenador realizar técnicas de paralelismo computacional no es recomendable.

5.3 ANÁLISIS DE BENEFICIOS

5.3.1 Desarrollo del flujo de caja

Tabla 23. Flujo de caja

MESES	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
INGRESOS:																			
Utilización del HPC		1000	1000	1000	1000	1500	1000	1500	2000	2000	5000	5000	4000	3000	4000	5000	9500	4000	4000
INVESTIGACION																			
Importe CTIC-UNI		3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
TOTAL DE INGRESOS		4000	4000	4000	4000	4500	4000	4500	5000	5000	8000	8000	7000	6000	7000	8000	12500	7000	7000
EGRESOS:																			
-Costo de desarrollo	29,420																		
-Pago Tesista investigador		1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	0	0	0	0	0	0
-Costo de implementación HPC		0	0	0	0	0	0	1210	1210	1210	1210	1210							
- Otros gastos		0	0	0	0	0	0	0	0	290	290	290	290	290	290	290	290	290	290
TOTAL DE EGRESOS:	29,420	1,000	1,000	1,000	1,000	1,000	1,000	2,210	2,210	2,500	2,500	2,500	2,500	1,500	1,500	1,500	1,500	1,500	1,500
Flujo de Caja (Ingreso Neto)	-29,420	3,000	3,000	3,000	3,000	3,500	3,000	2,475	2,975	2,685	5,685	5,685	4,685	4,685	5,685	6,685	11,185	5,685	5,685
Costo beneficio	29420	-26420	-23420	-20,420	-17420	-13920	-10920	-8630	-5840	-3340	2160	7660	12160	16660	22160	28660	39660	45160	50660

En la siguiente tabla vemos el flujo de caja del resultado del financiamiento del proyecto, como vemos en el mes 10 vemos un retorno positivo de lo invertido.

5.3.2 Análisis VAN Y TIR

Tabla 24. Análisis VAN y TIR

PROMEDIO DE INVERSION (18 MESES)

Tasa de descuento	10%
VAN	S/.1,754.37
TIR	11%

En el presente cuadro, se observa el VAN y TIR, nos muestra el detalle de los beneficios y la inversión a 18 meses

- VAN: Como podemos ver el VAN nos da un resultado mayor a la inversión lo cual podemos ver que el proyecto es viable.
- TIR: El retorno de la inversión es favorable podemos ver un 11% de la ganancia que tiene la implementación y los gastos de mantenimiento.

Conclusiones

- Se optimizó los procesos de alta complejidad, juntando varios computadores logrando que los procesos sean en paralelo; gracias a la arquitectura HPC utilizada en el proyecto conseguimos reducir el tiempo a 2048.4 segundos a comparación de antes de la implementación que fue de 28448.9 segundos, por lo que de esta manera se dio un gran beneficio a la comunidad que se dedica a querer encontrar resultados emergentes en tiempo muy corto.
- Se logró encontrar los cuellos de botella que ocurren cuando se utilizan más de 16 nodos esclavos utilizando un algoritmo matemático que nos mostró la cantidad de 112 núcleos que se utilizó para que el cluster opere con normalidad; al pasar esa cantidad de nodos el rendimiento computacional baja debido a la red de alta velocidad.
- Se logró hacer pruebas de rendimiento para ver el potencial computacional utilizando diferentes cantidades de nodos utilizando el algoritmo de N-Body; gracias a este algoritmo se pudo ver el crecimiento computacional de los nodos con 16 nodos esclavos obtuvieron un total de 21.421 *gigaflops* utilizando 128000 cuerpos celestes en un tiempo de 2048.4 segundos.

Recomendaciones

- Se recomienda tener conocimiento en programación en paralelo para que pueda sacar una máxima utilidad del proyecto implementado; ya que sin él no sacaremos provecho de todos sus núcleos.
- Se recomienda leer el capítulo de pruebas de rendimiento donde se explica cómo se pudo localizar los cuellos de botellas cuando se usa una cierta cantidad de nodos.
- Se recomienda comprender el algoritmo N-CUERPO para poder hacer mediciones a futuro con lo que corresponde a medir el rendimiento computacional del cluster.

Referencias

- Alcántara, L. (2014). *Instalación y configuración de un cluster de alta disponibilidad con reparto de carga* (Tesis de maestría). Recuperado de https://riunet.upv.es/bitstream/handle/10251/46800/Memoria-TFM-Lenin_Alcantara.pdf?sequence=1
- Atul, G., Bhargavi, G., & Uditnarayan, K. (2015). Study of Supercomputer ' s Architecture , Application and Its Future Use. *Super Computer Technologies of Mathematical Modeling*, [1-5]. Recuperado de https://www.researchgate.net/publication/260984373_Study_of_Supercomputer's_Architecture_Application_and_Its_Future_Use/download.
- Cabrera, L., y Soler, E. (2011). Propuestas de ingeniería de Software para la computación de alto rendimiento. *Serie Científica de la Universidad de las ciencias informáticas*, 4(3), 1-13. Recuperado de <http://publicaciones.uci.cu/index.php/SC>
- Castro, A., Repetto, A., Bianchi, M., Cipriano, M., Arroyo, A., Cicerchia, C. y Malvacio, E. (2014). *ULTRACOM: computación de alto rendimiento para criptoanálisis*. Recuperado de <http://sedici.unlp.edu.ar/handle/10915/42390>
- Colobran, M., Arquéz J. y Galindo, E. L. (2007). *Administración de sistemas operativos en red*. España: UOC.
- Fiestas, J. (2014). Construcción e implementación de un Clúster con máquinas PCs recicladas. *Revista de la Facultad de Ciencias de la UNI*, 14(1), 50-57. Recuperado de <https://www.ctic.uni.edu.pe/images/pdflaboratorios/bigdata/paperCruz03.pdf>
- González, S. (2014). Cluster alta disponibilidad sobre plataforma GNU/LINUX (VERITAS) (Tesis de maestría) Recuperado de <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/29841/6/sbarrioTFC0114memoria.pdf> .
- González, E. R. y Rodríguez, A. (2008). *Diseño e implementación de un cluster tipo beowulf para el desarrollo de cómputo científico avanzado* (Tesis de grado). Recuperado de <https://tesis.ipn.mx/jspui/bitstream/123456789/5012/1/DISENOIMPLEMENT.pdf>
- Hinojosa, C. R. (2014). Generaciones de las computadoras. Recuperado de <https://www.sutori.com/story/generaciones-de-las-computadoras-de0193f0-4258-4124-89f8-9ec9ae070825>
- Hoeger, H. (2014). *TB interviewing and contact tracing*. Recuperado de <https://docplayer.es/35633616-Introduccion-a-la-computacion-paralela.html>
- Instituto Nacional de Defensa Civil. (2010). Estudio SIRAD: recursos de respuesta inmediata y de recuperación temprana ante la ocurrencia de un sismo y/o tsunami en Lima metropolitana y Callao. Recuperado de http://sirad.indeci.gob.pe/static/SIRAD_Publicacion_ES.pdf

- Jiménez, D., y Medina, A. (2014). Cluster de alto rendimiento. *Facultad de Ingeniería - UMSA Cluster*, 16-27. Recuperado de http://www.revistasbolivianas.org.bo/pdf/jit/n14/n14_a04.pdf
- Milone, D. H., Azar, A., y Rufiner, H. L. (2002). Supercomputadoras basadas en "clusters" de PCs. *Revista Ciencia, Docencia y Tecnología*, 13(25), 173-208. Recuperado de https://www.researchgate.net/publication/228590345_Supercomputadoras_basadas_en_clusters_de_PCs
- Ocampo, I., y Exequiel, L. (2017). *Introducción a la supercomputación en el Perú*. Lima: Instituto de Investigaciones de la Amazonía Peruana - IIAP
- Rebollo, M. (2013). Dispositivos de almacenamiento. *Facultad de Administración y Dirección de Empresas*, 91. Recuperado de https://www.researchgate.net/publication/50838253_Dispositivos_de_almacenamiento
- Samuelson, P. A., y Nordhaus, W. D. (2009). *Arquitectura de computadoras* (3ª ed.). Mexico: Pearson
- Sinisterra, M. M., Díaz Henao, T. M., y Ruiz López, E. G. (2012). Clúster de balanceo de carga y alta disponibilidad para servicios web y mail. *Informador Técnico*, 1(1), 93-102. Recuperado de <https://doi.org/10.23850/22565035.34>

Glosario

- **Big Data:** Big Data es en el sector de tecnologías de la información y la comunicación una referencia a los sistemas que manipulan grandes conjuntos de datos.
- **Paralelismo computacional:** La computación paralela es una forma de cómputo en la que muchas instrucciones se ejecutan simultáneamente, operando sobre el principio de que problemas grandes, a menudo se pueden dividir en unos más pequeños, que luego son resueltos simultáneamente.
- **Clúster:** Se aplica a los conjuntos o conglomerados de computadoras unidos entre sí normalmente por una red de alta velocidad y que se comportan como si fuesen una única computadora.
- **Dinámica de cuerpo:** La dinámica es la rama de la física que describe la evolución en el tiempo de un sistema físico en relación con las causas que provocan los cambios de estado físico y/o estado de movimiento.
- **Red computacional:** Una red de computadoras (también llamada red de ordenadores o red informática) es un conjunto equipos (computadoras y dispositivos), conectados por medio de cables, señales, ondas o cualquier otro método de transporte de datos, para compartir información (archivos), recursos (discos, impresoras, programas, etc.).
- **Open source:** También llamado “Código Abierto” es un término que se utiliza para denominar a cierto tipo de software que se distribuye mediante una licencia que le permite al usuario final, si tiene los conocimientos necesarios, utilizar el código fuente del programa para estudiarlo, modificarlo y realizar mejoras en el mismo.
- **Inteligencia artificial:** La inteligencia artificial es considerada una rama de la computación y relaciona un fenómeno natural con una analogía artificial a través de programas de computador.
- **Ciclos de maquina:** Cuando un microprocesador ejecuta una instrucción, éste realiza una serie de operaciones o tareas básicas llamadas ciclos de máquina. Estas tareas que involucran a las componentes internas del CPU, algunas veces a la memoria o a los puertos de entrada y salida, son

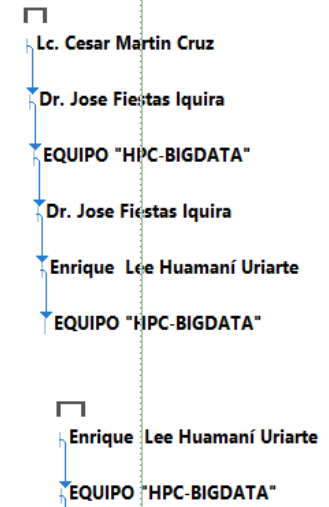
coordinadas con las señales de interface generadas en la unidad de control en una secuencia ordenada.

- **Criptoanálisis:** Es la parte de la cristología que se dedica al estudio de sistemas criptográficos con el fin de encontrar debilidades en los sistemas y romper su seguridad sin el conocimiento de información secreta. En el lenguaje no técnico, se conoce esta práctica como romper o forzar el código, aunque esta expresión tiene un significado específico dentro del argot técnico. A las personas que se dedican al criptoanálisis se llaman criptoanalistas.
- **Computaciones paralelas:** La computación paralela es una forma de cómputo en la que muchas instrucciones se ejecutan simultáneamente, 1 operando sobre el principio de que problemas grandes, a menudo se pueden dividir en unos más pequeños, que luego son resueltos simultáneamente (en paralelo).

ANEXOS

Anexo A: Planificación del proyecto.

		PROYECTO: IMPLEMENTACION Y CONSTRUCCION DE UN CLUSTER DE ALTO RENDIMIENTO PARA LA SIMULACION DE SISTEMA VEHICULAR EN EL CENTRO DE COMUNICACIONES E INFORMACION DE LA UNIVERSIDAD NACIONAL DE INGENIERIA	240 días	vie 4/12/15	jue 3/11/16			
		FASE I	12 días	vie 4/12/15	lun 21/12/15			
			Entrevista personal con el co-investigador del proyecto	1 día	vie 4/12/15	vie 4/12/15	Lc. Cesar Martin Cruz	
			Entrevista personal con el Investigador del proyecto	1 día	lun 7/12/15	lun 7/12/15	3	Dr. Jose Fiestas Iquira
			Primera reunion y presentacion con el equipo de trabajo "HPC-BIG DATA"	1 día	vie 11/12/15	vie 11/12/15	4	EQUIPO "HPC-BIGDATA"
			Plan de actividades a realizar los tesisas "HPC-BIG DATA "	1 día	lun 14/12/15	lun 14/12/15	5	Dr. Jose Fiestas Iquira
			Entrega de primera a desarrollar el HPC a los co-investigadores e investigaciones	1 día	vie 18/12/15	vie 18/12/15	6	Enrique Lee Huamaní Uriarte
			Segunda reunion y ultima del año para saludar al equipo de trabajo y resaltar punto a avanzar	1 día	lun 21/12/15	lun 21/12/15	7	EQUIPO "HPC-BIGDATA"
		FASE II	18 días	lun 4/01/16	mié 27/01/16			
			Presentacion del proyecto de Tesis en el Encuentro Cientifico Internacional	1 día	lun 4/01/16	lun 4/01/16		Enrique Lee Huamaní Uriarte
			Pimera reunion con el equipo de trabajo	1 día	mar 5/01/16	mar 5/01/16	10	EQUIPO



Anexo B: Prototipo de clúster de alto rendimiento.

Antes de su implementación en CTIC-UNI se realizó un prototipo de red computacion tipo cluster para ver su escalabilidad , el resuelto fue un éxito y fue expuesto en Chile



Anexo C: Modelo de Investigación.

Antes de su implementación en CTIC-UNI se realizó una encuesta al equipo HPC-BIG DATA, en la imagen del Anexo C visualizó la encuesta dirigida a nuestro jefe investigador Jose Fiesta Iqira.

ENCUESTAS REALIZADA A: EQUIPO HPC-BIG DATA

ENCUESTA		
NOMBRE DE LA EMPRESA Y /O INSTITUCIÓN: CTIC-UNI		
AREA DE INVESTIGACION CTIC-UNI	NOMBRE DE ENCUESTADO: Jose Fiesta Iqira (Jefe Investigador)	REALIZADO POR: Enrique Lee Huamani Uriarte
LUGAR: UNI	FECHA: 05/04/2018	HORA: 3 pm

La presente encuesta tiene como objetivo recoger información para determinar los requerimientos del usuario y la información que maneja actualmente en su área de trabajo.

INSTRUCCIONES: Marcar con un aspa (X) según la opción que sea conveniente para usted y conteste verazmente las preguntas.

1. ¿Actualmente institución cuenta con un Hardware de alto rendimiento?
 Si No No se probablemente

2. ¿Se requiere de un Hardware de alto rendimiento para resolver problemas de alta complejidad?
 Si No No se probablemente

3. ¿Se reduciría tiempo en resultados, si se construye un Hardware de mayor cantidad de núcleos?
 Si No No se probablemente

4. ¿Las computadoras en la institución son adecuada para hacer simulaciones de alta complejidad?
 Si No No se probablemente

5. ¿Sería necesario comprar Cluster de alto rendimiento en vez de construir uno?
 Si No No se probablemente

6. ¿Se lograría los objetivos propuestos, si se construyese un cluster de alto rendimiento?
 Si No No se probablemente

7. ¿Utilizaría un sistema operativo licenciado para resolver sus actividades?

Si No No se probablemente

8. ¿Serviría para dar otros tipos de servicios a nivel de empresa?

Si No No se probablemente

9. ¿Existen otras áreas que tenga una Cluster de alto rendimiento?

Si No No se probablemente

ENCUESTADO

ENCUESTADOR

Anexo D: Exposición del prototipo del clúster de alto rendimiento Chile 2016 antes de su implementación en CTIC-UNI

Diapositiva sobre clúster de alto rendimiento en latinoameria , el protitipo de clúster fue nombrado en esa cumbre en Santiago de Chile

Astrophysical Particle Simulations with large custom GPU clusters on three continents (Spurzem, Fiestas et al..PRACE award ISC'11, Hamburg)

DIRAC (GPU) Cluster:
40 nodos (GPUs)
UC Berkeley, CA

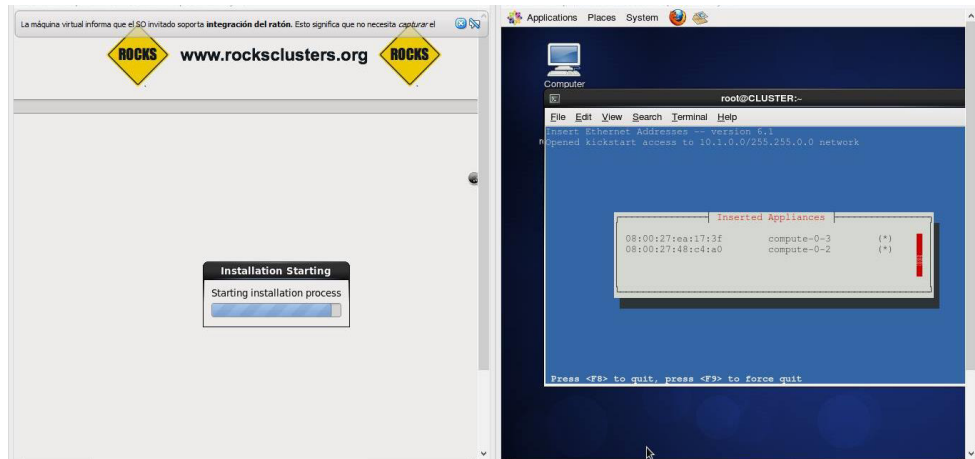
CESSAR (CPU) Cluster:
10 nodos Lima, Perú

Kepler (GPU) Cluster:
85 nodos (GPUs)
(Heidelberg, Germany)

LAOHU (GPU) Cluster:
170 nodos (GPUs)
(Beijing, China)

Anexo E: Configuración gráfica nodo maestro

Configuración automática del nodo maestro, para la comunicación con los nodos esclavos , esta configuración es opcional si no funcionara la manual explicada en los puntos anteriores



Anexo F: Características de los componentes del clúster

Los componentes que se están utilizando en cada nodo del clúster son los siguientes :

- Placa madre : Z270-A
- Memoria ram : HyperX FURY DDR4 GB
- CPU: CORE i7-7920HQ



ANEXO G: Componente switch TP-Link TL-SG1024D

El siguiente componente es un switch, en la cual nos servirá para que tenga una comunicación nuestros nodos esclavos con el nodo maestro de clúster.

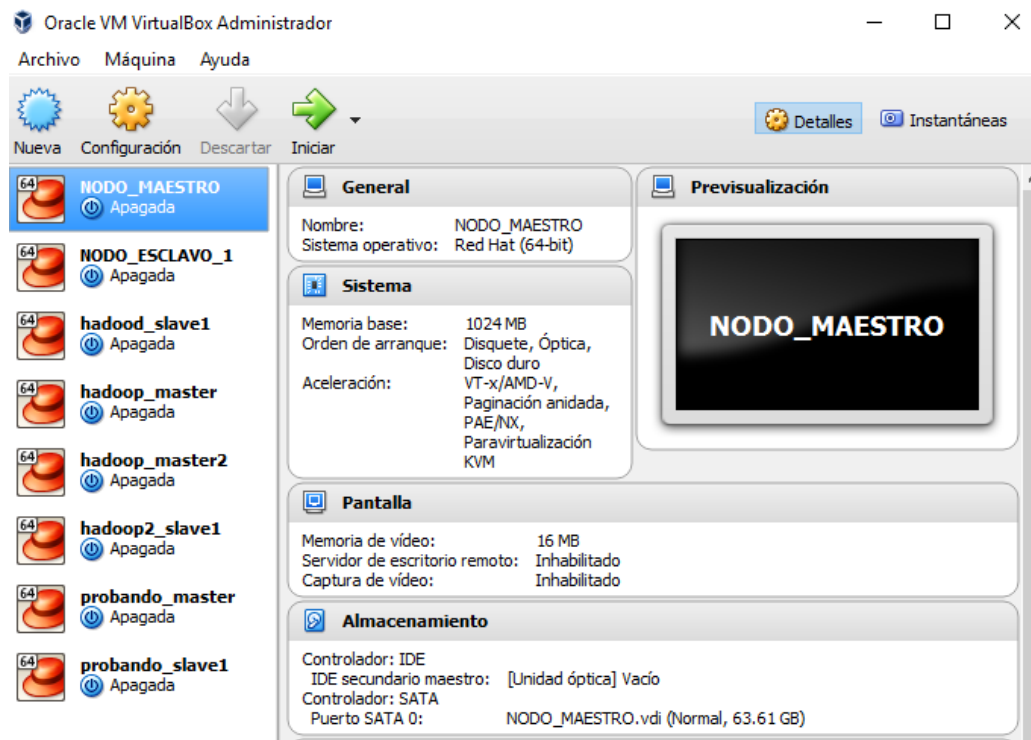


Anexo H: Característica de sistema operativo Open source

Se utilizará un software libre "OPEN SOURCE" debido a que son sistemas operativos de libre modificación en su código, esta es una ventaja considerable en utilizar un HPC, en este debemos hacer diversas modificaciones en S.O, darle características especiales a la red, por ello se consideró utilizar el ROCKS CLÚSTER SAND BOA



Anexo I: Simulación del clúster de alto rendimiento con virtual box



Anexo J: Entorno del sistema operativo del nodo master

