

UNIVERSIDAD NACIONAL DE UCAYALI

FACULTAD DE INGENIERÍA DE SISTEMAS E INGENIERÍA CIVIL

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



**ANÁLISIS, DISEÑO Y DESPLIEGUE DE UN CLÚSTER DE
SERVIDORES PARA ALTO PROCESAMIENTO MATEMÁTICO**

Tesis para optar el título profesional de

INGENIERO DE SISTEMAS

SERGIO ORLANDO FUNG VARGAS

JOHN ANDRÉS WAGNER MENDOZA

**Pucallpa, Perú
2024**



UNIVERSIDAD NACIONAL DE UCAYALI
FACULTAD DE INGENIERÍA DE SISTEMAS E INGENIERÍA CIVIL
ESCUELA PROFESIONAL DE INGENIERÍA CIVIL
COMISIÓN DE GRADOS Y TÍTULOS



ACTA DE EVALUACIÓN DE SUSTENTACIÓN DE TESIS N° 008-2024

GRADUANDOS : FUNG VARGAS, SERGIO ORLANDO
WAGNER MENDOZA, JOHN ANDRÉS

TEMA : "ANALISIS, DISEÑO Y DESPLIEGUE DE UN CLUSTER DE SERVIDORES PARA ALTO PROCESAMIENTO MATEMATICO"

ASESOR : Dr. NILTON CESAR AYRA APAC

Table with 5 columns: CRITERIOS, PUNTAJE (3, 2, 1, 0). Rows include categories like I- PRESENTACION, II- DESARROLLO DE CONTENIDO, III- ABSOLUCION DE PREGUNTAS, and IV- APORTE CIENTIFICO Y TECNOLOGICO with specific criteria marked with 'X'.

EVALUACION

- EXCELENCIA
UNANIMIDAD
MAYORIA
DESAPROBADO

26 A 30 PUNTOS
21 A 25 PUNTOS
16 A 20 PUNTOS
15 A MENOS

CALIFICACION FINAL:

PUNTAJE : 16 / 30

CONCLUSIONES:

APROBADO POR : MAYORIA
DESAPROBADO POR :

JURADO EVALUADOR

Dr. JORGE LUIS HILARIO RIVAS
PRESIDENTE

M.SC. CLOTILDE RÍOS HIDALGO DE CERNA
MIEMBRO

Mg. RONALD ULLOA GALVEZ
MIEMBRO

M.SC. CLOTILDE RÍOS HIDALGO DE CERNA
SECRETARIA ACADÉMICA



UNIVERSIDAD NACIONAL DE UCAYALI
FACULTAD DE FACULTAD DE INGENIERIA DE SISTEMAS E INGENIERIA CIVIL
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS

Carretera Federico Basadre Km 6200 - Pucallpa - Ucayali



2022-121

ACTA DE APROBACION DE PROYECTO DE TESIS

En la Ciudad Universitaria, a los 31 días del mes AGOSTO del 2022 siendo horas 17:47:45. Los miembros del Jurado, declaran APROBADO POR UNANIMIDAD el PROYECTO DE INVESTIGACIÓN DE TESIS titulado:

ANÁLISIS, DISEÑO Y DESPLIEGUE DE UN CLÚSTER DE SERVIDORES PARA ALTO PROCESAMIENTO MATEMÁTICO

Presentado por los Bachilleres:

SERGIO ORLANDO FUNG VARGAS
JOHN ANDRÉS WAGNER MENDOZA

De la Escuela Profesional de:

INGENIERIA DE SISTEMAS

Siendo el Jurado Dictaminador, conformado por:

Presidente	: DR. JORGE LUIS HILARIO RIVAS
Primer Miembro	: MG. CLOTILDE RIOS HIDALGO DE CERNA
Segundo Miembro	: ING. RONALD HAROLD ULLOA GALVEZ
Asesor	: DR. NILTON CESAR AYRA APAC

Para dar fe de este proceso electrónico, el Vicerrectorado de Investigación de la Universidad Nacional de Ucayali - Pucallpa, mediante la Plataforma de Investigación se le asigna la presente acta y a partir de la presente fecha queda expedito para la ejecución de su PROYECTO DE INVESTIGACIÓN DE TESIS.

Pucallpa, AGOSTO de 2022



UNIVERSIDAD NACIONAL DE UCAYALI
 FACULTAD DE ING. DE SISTEMAS E ING. CIVIL
 DECANATURA

Dr. Nilton Cesar Ayra Apac
 DECANO(E)



UNIVERSIDAD NACIONAL DE UCAYALI

DIRECCIÓN DE PRODUCCIÓN INTELECTUAL

CONSTANCIA

ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN

N° V/0440-2023.

La Dirección de Producción Intelectual de la Universidad Nacional de Ucayali, hace constar por la presente, que el trabajo académico de investigación, titulado:

“ANÁLISIS, DISEÑO Y DESPLIEGUE DE UN CLÚSTER DE SERVIDORES PARA ALTO PROCESAMIENTO MATEMÁTICO”

Autor(es) : **FUNG VARGAS, SERGIO ORLANDO**
WAGNER MENDOZA JHON ANDRES

Facultad : **INGENIERÍA DE SISTEMAS E INGENIERÍA CIVIL**

Escuela : **Ing. Sistemas**

Asesor(a) : **Dr. Ayra Apac, Nilton César**

Presenta un **porcentaje de similitud de 10%**, verificado en el Sistema Antiplagio URKUND/OURIGINAL, De acuerdo a los criterios de porcentaje establecidos en el artículo 9 de la DIRECTIVA DE USO DEL SISTEMA ANTIPLAGIO, el cual indica que todo trabajo de investigación no debe superar el 10%. **En tal sentido, se declara, que el presente trabajo de investigación: SI Contiene un porcentaje aceptable de similitud**, procediéndose a emitir la presente Constancia de Originalidad de Trabajo de Investigación (COTI) a solicitud del asesor. En señal de conformidad se firma y sella el presente documento.

Fecha: 24/07/2023



Mg. JOSÉ MANUEL CÁRDENAS BERNAOLA
Director de Producción Intelectual



UNIVERSIDAD NACIONAL DE UCAYALI

DIRECCIÓN DE PRODUCCIÓN INTELECTUAL

AUTORIZACIÓN DE PUBLICACIÓN

Repositorio de la Universidad Nacional de Ucayali

Yo, John Andrés Wagner Mendoza

Autor de la tesis titulada: Análisis, diseño y despliegue de un clúster de servidores para alto procesamiento matemático

Sustentada el año 2024

Asesor(a): Dr. Cesar Ayra Apar

Facultad: Ingeniería de Sistemas y de Ingeniería Civil

Escuela Profesional: Ingeniería de Sistemas

Autorizo la publicación:

PARCIAL

TOTAL

De mi trabajo de investigación en el Repositorio Institucional de la Universidad Nacional de Ucayali (www.repositorio.unu.edu.pe), bajo los siguientes términos:

Primero: Otorgo a la Universidad Nacional de Ucayali licencia no exclusiva para reproducir, distribuir, comunicar, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público en general mi tesis (incluido el resumen) a través del Repositorio Institucional de la UNU, en formato digital sin modificar su contenido, en el Perú y en el extranjero; por el tiempo y las veces que considere necesario y libre de remuneraciones.

Segundo: Declaro que la tesis es una creación de mi autoría y exclusiva titularidad, por tanto me encuentro facultado a conceder la presente autorización, garantizando que la tesis no infringe derechos de autor de terceras personas, caso contrario, me hago único(a) responsable de investigaciones y observaciones futuras, de acuerdo a lo establecido en el estatuto de la Universidad Nacional de Ucayali, la Superintendencia Nacional de Educación Superior Universitaria y el Ministerio de Educación.

En señal de conformidad firmo la presente autorización.

Fecha: 03/07/24

Email: john.wagnerm@gmail.com Firma: [Firma]

Teléfono: 944494218 DNI: 71717711



UNIVERSIDAD NACIONAL DE UCAYALI

DIRECCIÓN DE PRODUCCIÓN INTELECTUAL

AUTORIZACIÓN DE PUBLICACIÓN

Repositorio de la Universidad Nacional de Ucayali

Yo, SERGIO ORLANDO FUNG VARGAS
 Autor de la tesis titulada: ANÁLISIS, DISEÑO Y DESPLIEGUE DE UN CLUSTER DE SERVIDORES PARA ALTO PROCESAMIENTO MATEMÁTICO
Sustentada el año 2024Asesor(a): DR. CESAR AYRA APACFacultad: INGENIERIA DE SISTEMAS E INGENIERIA CIVILEscuela Profesional: INGENIERIA DE SISTEMAS

Autorizo la publicación:

PARCIAL

TOTAL

 De mi trabajo de investigación en el Repositorio Institucional de la Universidad Nacional de Ucayali (www.repositorio.unu.edu.pe), bajo los siguientes términos:

Primero: Otorgo a la Universidad Nacional de Ucayali licencia no exclusiva para reproducir, distribuir, comunicar, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público en general mi tesis (incluido el resumen) a través del Repositorio Institucional de la UNU, en formato digital sin modificar su contenido, en el Perú y en el extranjero; por el tiempo y las veces que considere necesario y libre de remuneraciones.

Segundo: Declaro que la tesis es una creación de mi autoría y exclusiva titularidad, por tanto me encuentro facultado a conceder la presente autorización, garantizando que la tesis no infringe derechos de autor de terceras personas, caso contrario, me hago único(a) responsable de investigaciones y observaciones futuras, de acuerdo a lo establecido en el estatuto de la Universidad Nacional de Ucayali, la Superintendencia Nacional de Educación Superior Universitaria y el Ministerio de Educación.

En señal de conformidad firmo la presente autorización.

Fecha: 03 / 07 / 24Email: sergio fung.xfv@gmail.com Firma: Teléfono: 950564088 DNI: 71690156

DEDICATORIA

La presente tesis está dedicada a nuestros padres y familia, que nos brindan su apoyo para lograr nuestras metas y nos guían en la vida con valores y experiencias para ser mejores personas cada día.

AGRADECIMIENTO

Agradecemos a nuestros docentes los cuales nos han brindado sus conocimientos y experiencias para lograr ser los profesionales que aspiramos ser gracias a sus enseñanzas.

ÍNDICE GENERAL

DEDICATORIA	viii
AGRADECIMIENTO	viii
ÍNDICE GENERAL.....	ix
ÍNDICE DE TABLAS	xiii
ÍNDICE DE FIGURAS	xv
RESUMEN	xvii
ABSTRACT	xviii
INTRODUCCIÓN	xix
CAPÍTULO I	1
PLANTEAMIENTO DEL PROBLEMA	1
1.1. Planteamiento del Problema	1
1.2. Formulación del Problema	2
1.2.1. Problema General.....	2
1.2.2. Problemas Específicos	2
1.3. Objetivos de la Investigación.....	3
1.3.1. Objetivo General.....	3

1.3.2. Objetivos Específicos	3
1.4. Justificación e Importancia	3
1.4.1. Justificación	3
1.4.2. Importancia	4
1.5. Hipótesis y/o Sistemas de Hipótesis	4
1.5.1. Hipótesis General	4
1.5.2. Hipótesis Específicas.....	4
1.6. Variables	4
1.6.1. Variable Independiente	4
1.6.2. Variable dependiente.....	5
1.6.3. Operacionalización de las Variables.....	5
CAPÍTULO II.....	7
MARCO TEÓRICO	7
2.1. Antecedentes de la Investigación.....	7
2.2. Bases Teóricas.....	12
2.2.1. Clúster de servidores.....	12
2.2.2. Ventajas de los servidores en clúster	13

2.2.3. Computación de alto rendimiento	14
2.2.4. ¿Por qué es importante la computación de alto rendimiento?.....	14
2.2.5. ¿Cómo funciona HPC?	15
2.3. Definiciones de Términos Básicos	19
CAPÍTULO III	21
METODOLOGÍA DE LA INVESTIGACIÓN	21
3.1. Tipo y Nivel de Investigación.....	21
3.1.1. Tipo de Investigación.....	21
3.1.2. Nivel de Investigación.....	21
3.2. Población y Muestra.....	21
3.3. Procedimientos de Recolección de Datos	22
3.4. Técnicas e Instrumentos de Recolección de Datos.....	23
3.5. Procesamiento para la Recolección de Datos.....	23
3.6. Tratamiento de Datos.....	23
CAPÍTULO IV	24
RESULTADOS.....	24
CAPÍTULO V	49

DISCUSIÓN DE RESULTADOS 49

CONCLUSIONES 57

RECOMENDACIONES 59

REFERENCIAS BIBLIOGRÁFICAS 60

ANEXOS 62

ÍNDICE DE TABLAS

Tabla 1 Operacionalización de las variables	6
Tabla 2 Utilización de los recursos	22
Tabla 3 Especificaciones técnicas de las terminales	30
Tabla 4 Recursos en hardware.....	33
Tabla 5 Recursos en software	34
Tabla 6 Generación de 300000 números aleatorios vs número de núcleos	38
Tabla 7 Ordenamiento de 300000 números aleatorios, 16 núcleos vs método de ordenamiento	41
Tabla 8 Ordenamiento de 300000 números aleatorios, 12 núcleos vs método de ordenamiento	42
Tabla 9 Ordenamiento de 300000 números aleatorios, 8 núcleos vs método de ordenamiento	44
Tabla 10 Ordenamiento de 300000 números aleatorios, 4 núcleos vs método de ordenamiento	45
Tabla 11 Ordenamiento de 300000 números aleatorios, 1 núcleo vs método de ordenamiento	47
Tabla 12 Generación de 300.000 números en forma aleatoria.....	49
Tabla 13 Ordenamiento de 300.000 números método burbuja	51

Tabla 14 Ordenamiento de 300 000 números – método inserción52

Tabla 15 Ordenamiento de 300 000 números – método selección54

ÍNDICE DE FIGURAS

Figura 1 Rack de piso standard	26
Figura 2 Power Rack.....	26
Figura 3 Bandeja simple.....	27
Figura 4 Organizador horizontal.....	28
Figura 5 Switch 5 puertos Giga Ethernet para escritorio	28
Figura 6 Patch Panel Categoría 6	29
Figura 7 Patch Cord Categoría 6	30
Figura 8 Características de las cuatro PC's.....	31
Figura 9 Análisis y diseño del clúster	34
Figura 10 Generación de 300000 números aleatorios, tiempo vs número de núcleos.....	39
Figura 11 Ordenamiento de 300000 números aleatorios, 16 núcleos vs método de ordenamiento	41
Figura 12 Ordenamiento de 300000 números aleatorios, 12 núcleos vs método de ordenamiento	43
Figura 13 Ordenamiento de 300000 números aleatorios, 8 núcleos vs método de ordenamiento	44

Figura 14 Ordenamiento de 300000 números aleatorios, 4 núcleos vs método de ordenamiento	46
Figura 15 Ordenamiento de 300000 números aleatorios, 1 núcleo vs método de ordenamiento	47
Figura 16 Generación de 300 000 números aleatorios	50
Figura 17 Ordenamiento de 300 000 números método burbuja.....	51
Figura 18 Ordenamiento de 300 000 números método inserción	53
Figura 19 Ordenamiento de 300 000 números método selección.....	54

RESUMEN

En esta investigación se ha podido determinar en qué medida un Clúster de servidores influye en el requerimiento de alto procesamiento matemático, así mismo se ha logrado el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático. La investigación fue de corte experimental/aplicada, el instrumento de medición fue un formulario de recolección de datos, el cual se tomó el tiempo que demora en generar 300000 número en forma aleatoria, luego se procedió a desarrollar el ordenamiento por tres métodos, los cuales fueron: burbuja, inserción y selección la generación y ordenamiento se hizo aplicando el clúster construido aplicando desde un núcleo hasta 16 núcleos.

Palabras claves: Clúster, núcleo y procesamiento.

ABSTRACT

In this investigation it has been possible to determine to what extent a cluster of servers influences the requirement of high mathematical processing, likewise the analysis and architectural design of a cluster of servers for high mathematical processing has been achieved. The investigation was of an experimental/applied nature, the measurement instrument was a data collection form, which took the time it takes to generate 300,000 numbers randomly, then proceeded to develop the ordering by three methods, which They were: bubble, insertion, and selection. The generation and ordering was done by applying the cluster built by applying from one core to 16 cores.

Keywords: Cluster, core and processing.

INTRODUCCIÓN

En la era actual, caracterizada por una creciente dependencia de la tecnología avanzada y la computación de alto rendimiento, la necesidad de procesar grandes volúmenes de datos a velocidades extraordinarias se ha vuelto imperativa para diversas aplicaciones científicas y de ingeniería. En este contexto, el análisis, diseño y despliegue de un clúster de servidores especializados en alto procesamiento matemático representan un desafío crítico y una oportunidad para mejorar significativamente la capacidad de cálculo. Este tipo de infraestructura es esencial para realizar simulaciones complejas, análisis de datos a gran escala y operaciones de investigación que requieren un gran poder computacional, facilitando así avances en áreas como la física cuántica, la bioinformática y la inteligencia artificial.

El desarrollo de un clúster de servidores para alto procesamiento matemático implica no solo una cuidadosa selección de hardware capaz de soportar cargas de trabajo intensivas, sino también una planificación meticulosa en cuanto a su arquitectura de software y configuración de red. El diseño debe garantizar la escalabilidad, la eficiencia energética y la máxima utilización de recursos, mientras que el despliegue requiere una implementación estratégica que priorice la seguridad, la redundancia y la capacidad de recuperación ante fallos. Además, es fundamental integrar herramientas de monitoreo y gestión que permitan a los administradores optimizar el rendimiento del clúster y asegurar su operación continua sin interrupciones. A través de este proyecto, se busca proporcionar una solución robusta que no solo atienda las necesidades actuales de procesamiento, sino que también se adapte a las demandas futuras en un entorno tecnológico en constante evolución.

La estructura del informe, se conforma por cinco capítulos, los cuales son: En el Capítulo I se expone el planteamiento del problema, resaltando la creciente necesidad de sistemas de alto procesamiento matemático para aplicaciones como simulaciones, y se enfoca en analizar el impacto de este requerimiento en términos de análisis, diseño, rendimiento y eficiencia energética, además de evaluar las variables e hipótesis pertinentes. El Capítulo II aborda el marco teórico, destacando los antecedentes de investigaciones nacionales e internacionales y cómo estos contribuyen al desarrollo y distinción del estudio actual, incluyendo la definición de términos clave. En el Capítulo III se describe la metodología empleada, definiendo el tipo de estudio como explicativo y descriptivo, considerando la población y muestra a través de la recolección de datos y el uso de formularios en un clúster de servidores. El Capítulo IV detalla los resultados obtenidos, evaluando la ingeniería del proyecto, su motivación, objetivos e importancia, así como la influencia del uso de materiales en el procesamiento de datos. Finalmente, el Capítulo V discute los hallazgos, resaltando las diferencias en las capacidades de procesamiento de los servidores y concluyendo que el clúster tiene un impacto significativo en los requerimientos de procesamiento matemático, recomendando mejoras continuas para optimizar el procesamiento gráfico de datos.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

1.1. Planteamiento del Problema

En la era digital actual, la capacidad de procesar y analizar grandes volúmenes de datos de manera rápida se ha convertido en una ventaja competitiva esencial en diversos campos, tales como la investigación científica, la ingeniería, la medicina y la inteligencia artificial. El procesamiento de alto rendimiento (High-Performance Computing, HPC) facilita a organizaciones de todo el mundo la realización de simulaciones complejas, el modelado de fenómenos y análisis predictivos que serían inviables con sistemas computacionales tradicionales. Frente al crecimiento exponencial de los datos, es crucial expandir y optimizar los recursos computacionales mediante clústeres de servidores, capaces de procesar, analizar y almacenar información a una escala sin precedentes (Nugroho y Widiyanto, 2020).

Sin embargo, a pesar de los avances tecnológicos a nivel mundial, Perú enfrenta desafíos significativos en la adopción y desarrollo de infraestructuras de HPC. Las instituciones académicas y los sectores industriales están aún en etapas tempranas de implementación y de aprovechamiento de los beneficios del HPC. Los retos abarcan desde la adquisición de tecnología avanzada y costosa hasta la formación de talento humano que pueda desarrollar y mantener estas tecnologías. La necesidad de clústeres de servidores de alto rendimiento es evidente, especialmente para potenciar la investigación local y satisfacer las necesidades computacionales de industrias que requieren análisis rápidos y detallados, como la minería y la meteorología.

Además, hay una demanda creciente de sistemas capaces de realizar procesamiento matemático avanzado para simulaciones, cálculos matemáticos, búsquedas por similitud y tareas de computación gráfica. Aunque existen equipos especializados de alto procesamiento computacional, su elevado costo limita su disponibilidad en Perú, donde solo existen nueve de estos equipos y ninguno en nuestra región. Una alternativa viable es desarrollar un clúster de servidores que combine la capacidad de procesamiento de cada servidor para formar una máquina capaz de realizar estos cálculos complejos. Sin embargo, aún no contamos con la experiencia necesaria para establecer un clúster en nuestra región que pueda satisfacer esta demanda.

1.2. Formulación del Problema

1.2.1. Problema General

- ¿En qué medida un Clúster de servidores influye en el requerimiento de alto procesamiento matemático?

1.2.2. Problemas Específicos

- ¿Es posible determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático?
- ¿En qué medida el rendimiento computacional y la eficiencia del consumo de energía varía en un clúster de servidores para alto procesamiento matemático?

1.3. Objetivos de la Investigación

1.3.1. Objetivo General

- Determinar en qué medida un Clúster de servidores influye en el requerimiento de alto procesamiento matemático.

1.3.2. Objetivos Específicos

- Determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático.
- Determinar en qué medida el rendimiento computacional y la eficiencia del consumo de energía varía en un clúster de servidores para alto procesamiento matemático.

1.4. Justificación e Importancia

1.4.1. Justificación

En nuestra universidad, así como en la región, se carece de un equipo de procesamiento de alta capacidad especializado en matemáticas, esencial para el procesamiento eficiente de datos en investigaciones, así como en simulaciones y procesamiento gráfico. Esta deficiencia subraya la urgencia de analizar, diseñar y construir un clúster que cumpla con estos requisitos técnicos específicos. Además, es fundamental seleccionar adecuadamente el sistema operativo, los compiladores y los lenguajes de programación que se emplearán en este equipo, para maximizar su eficiencia y rendimiento en las tareas de investigación avanzada.

1.4.2. Importancia

Esta investigación establece un fundamento crucial para futuros estudios que requieran alta capacidad de procesamiento. No se limita exclusivamente al ámbito matemático, sino que también abre la posibilidad de explorar la aplicación de procesamiento gráfico en investigaciones posteriores.

1.5. Hipótesis y/o Sistemas de Hipótesis

1.5.1. Hipótesis General

- El uso de un Clúster de servidores influye significativamente en el requerimiento de alto procesamiento matemático.

1.5.2. Hipótesis Específicas

- Es posible determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático.
- El rendimiento computacional y la eficiencia del consumo de energía afecta significativamente un clúster de servidores para alto procesamiento matemático.

1.6. Variables

1.6.1. Variable Independiente

- Clúster de servidores.

Conformado por 4 servidores HP Gen8(Intel Xeon Silver, (10 Cores, 20 Hilos Windows y Linux reconoce 20 CPUs internos, 8 Gb RAM y SSD 240 GB.

1.6.2. Variable dependiente

- Alto procesamiento matemático.

Son algoritmos que requieran gran capacidad de procesamiento, ordenamiento de bases de datos, búsquedas, simulaciones, se considerarán la complejidad, tiempo y velocidad de procesamiento.

1.6.3. Operacionalización de las Variables

Variable independiente: Clúster de servidores.

- Dimensión 1: Análisis.
- Dimensión 2: Diseño.
- Dimensión 3: Despliegue.

Variable dependiente: Alto procesamiento matemático.

- Dimensión 1: Complejidad.
- Dimensión 2: Tiempo.
- Dimensión 3: Velocidad de procesamiento.

Tabla 1*Operacionalización de las variables*

Variables	Dimensiones	Indicadores
	Análisis	▪ Metodología
Clúster de servidores	Diseño	▪ Cantidad de Servidores
	Despliegue	▪ Configuración de los servidores
	Complejidad	▪ Tipo de procesamiento
Alto procesamiento matemático	Tiempo	▪ Tiempo de procesamiento (CPUs)
	Velocidad de procesamiento	▪ Número de ciclos del CPU (GHz)

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes de la Investigación

Acero et al. (2016), “Clúster de Alta Disponibilidad para Virtualizar los Servidores de Adquisición y Procesamiento de Datos del Instituto Geofísico”, “Un clúster de alta disponibilidad permite garantizar que los servicios que un sistema informático brinda estén disponibles, aunque un fallo de hardware o software suceda. Este documento presenta de forma resumida el diseño e implementación de una solución de alta disponibilidad basada en Pacemaker para brindar alta disponibilidad a los sistemas de adquisición y procesamiento del Instituto Geofísico: SeisComp3, Earthworm y ShakeMap. La solución está formada por dos clusters: el primero al que se denomina clusterwa es un clúster de alta disponibilidad del tipo activo/activo formado por tres servidores físicos. Este clúster administra tres servidores virtuales KVM y los módulos de los sistemas de adquisición y procesamiento que fueron instalados y configurados en los servidores virtuales. El segundo es un clúster de almacenamiento del tipo activo/pasivo al que se denomina clusteralmawa formado por dos servidores físicos. Este clúster administra un dispositivo de almacenamiento replicado que utiliza tecnología DRBD; este dispositivo contiene los discos duros de los tres servidores virtuales KVM. Pacemaker monitoriza y controla, mediante agentes de recursos, el funcionamiento de los servidores físicos, de los servidores virtuales y de los módulos que forman los sistemas mencionados y en caso de detectar alguna falla de hardware o software

Pacemaker realiza las acciones necesarias, en base a políticas y reglas configuradas por el administrador, para que los sistemas operen sin interrupción”.

Sebastián y Fernández (2014) “Computación paralela y clústeres de cálculo (Parallel computing and computer clusters)”, “Las principales razones para el uso de máquinas de cómputo paralelo son: disminuir el tiempo total de ejecución de una aplicación, conseguir resolver problemas más complejos, de grandes dimensiones y permitir la ejecución simultánea de tareas concurrentes. Existen además motivos económicos y que se han alcanzado unos límites físicos muy difíciles de mejorar tecnológicamente en transmisiones de datos y velocidad de CPU de los procesadores actuales. Por ello, la única forma de seguir aumentando el rendimiento de los ordenadores es mediante el cambio del paradigma de computación secuencial al paralelo, con múltiples procesadores, núcleos e hilos (ver Subsección 1.2.1). La ley de Moore postula que aproximadamente cada dos años se duplica el número de transistores que se pueden meter en un circuito integrado de tamaño estándar. Esta ley es empírica, formulada por el cofundador de Intel, Gordon E. Moore, el 19 de abril de 1965, cuyo cumplimiento se ha podido constatar hasta hoy. Las habilidades de muchos dispositivos electrónicos digitales están ligadas a esta ley: velocidad y capacidad de procesamiento, capacidad de memoria, resolución de sensores, etc. Como se puede comprobar en la Figura 1.1, la ley de Moore ligada a la capacidad de procesamiento se mantiene vigente gracias a la tecnología multinúcleo (múltiples núcleos en un único procesador), dado que desde al año 2006 existen muy pocos avances respecto a la velocidad de las CPUs”.

Cáceres (2012), “Estrategia de implementación de un clúster de alta disponibilidad de n nodos sobre Linux usando software, libre”. “Un clúster de alta

disponibilidad permite el acceso continuo a los servicios y aplicaciones más importantes dentro de una organización. El siguiente trabajo presenta una estrategia para la implementación de un clúster de alta disponibilidad de N nodos sobre Linux usando software libre. Este documento menciona los componentes más importantes del clúster y expone los conceptos más importantes. También explica la función que cumple cada uno de los componentes y cómo deben ser integrados dentro del clúster. Por último, muestra cómo instalar, configurar y probar un clúster de alta disponibilidad con tres nodos para los servicios de transferencia de archivos, resolución de nombres, y servidor de páginas Web”.

Chuquiguanca y Malla (2015) Universidad Nacional de Loja, Ecuador, “Arquitectura clúster de alto rendimiento utilizando herramientas de software libre”, “En los últimos años la ciencia y la tecnología han tenido grandes avances en lo que respecta a la computación de alto rendimiento, enmarcados en arquitecturas diseñadas especialmente para su funcionamiento y caracterizadas por su elevado coste, siendo inalcanzables por centros de investigación e Instituciones de Educación Superior. La informática para investigaciones, usa potentes herramientas y métodos para procesar datos en investigaciones académicas avanzadas. Con un clúster construido por hardware convencional y herramientas de software libre, la Carrera de Ingeniería en Sistemas de la Universidad Nacional Loja, puede obtener una arquitectura con la velocidad y potencia de una supercomputadora, a una fracción del costo mínimo. La finalidad de este proyecto es aportar e incentivar a la investigación académica y científica de la UNL, optimizando el tiempo de procesamiento en tareas como: simulación de procesos, aprendizaje de redes neuronales, renderizado de imagen y video, reconocimiento de patrones, análisis

de grandes volúmenes de datos Big Data, cifrado de códigos, evaluación de algoritmos, etc. La implementación de la arquitectura clúster de alto rendimiento permite realizar millones de operaciones por segundo, logrando avances importantes en la indagación académica, en cualquier campo de la investigación dentro del ámbito universitario. La implementación de la arquitectura clúster se realizó con el sistema operativo GNU/Linux Debian “Wheezy” por su estabilidad y su funcionamiento como servidor de alto rendimiento, usado como estándar para el desarrollo del proyecto. Para el middleware se utilizó Mosix como gestor de procesos para obtener una imagen única de sistema orientada a la computación distribuida. Para la programación paralela se implementó MPICH como una de las aplicaciones más importantes del estándar Message Passing Interface. Para el proceso de renderizado se usó Blender, para análisis de Big Data se empleó el framework Apache Hadoop. Dentro de las configuraciones realizadas tenemos: el acceso remoto utilizando claves públicas a través del protocolo Secure Shell, la compartición del sistema de ficheros en red a través del protocolo Network File System, además del diseño de la topología de red con direccionamiento IP estático y la configuración del sistema de monitorización Ganglia Monitoring System”.

Salazar et al. (2012), Universidad de Guanajuato, México, “Diseño e implementación de un clúster de cómputo de alto rendimiento”, “Este trabajo presenta el diseño, construcción y configuración de un sistema de computadoras agrupadas en un arreglo tipo clúster, el cual tiene un bajo costo y es comúnmente utilizado para resolver problemas que demandan grandes capacidades de cómputo, tanto de memoria como de procesamiento. En general estas tecnologías son costosas, pero en este trabajo se presenta una alternativa económica. Estos

problemas no pueden ser resueltos en computadoras aisladas debido a la gran cantidad de datos que involucran y a la cantidad de operaciones numéricas que se requieren realizar. En este trabajo se muestran resultados en aplicaciones numéricas que tienen gran utilidad práctica (Mecánica de Sólidos, Mecánica de Fluidos, Problemas Térmicos, etc.), en donde se refleja un buen desempeño y un adecuado uso de los recursos computacionales. El clúster mencionado en este trabajo ofrece una excelente capacidad de cómputo a un bajo costo”.

Sale y Rodríguez (2019), Universidad Tecnológica Nacional, Tucumán, Argentina, “implementación y Operación de un Cluster HPC Utilizando Laboratorios de Computadoras en Horarios de Inactividad”. “Este trabajo presenta un reporte preliminar de un método que logra facilitar a alumnos, docentes e investigadores el acceso a clusters HPC a costos accesibles, utilizando recursos existentes en las instituciones donde desempeñan sus actividades. Los laboratorios de informática de las universidades constituyen un reservorio de potencia de procesamiento. Estos pueden ser utilizados para conformar un clúster de cálculo de alto desempeño, en los horarios que no se encuentran ocupados. De esta manera se puede al mismo tiempo reducir los costos de poseer un clúster HPC, ampliando el abanico de servicios y posibilidades que la institución brinda a sus miembros y aumentar la utilización de los laboratorios, mejorando su amortización y por ende, justificando mejor la inversión”.

2.2. Bases Teóricas

2.2.1. Clúster de servidores

Borges (2019), un servidor en clúster, también conocido como clúster de servidores, se trata de la unión de varios sistemas informáticos (servidores) que funcionan como si fueran uno solo.

Cuando hablamos de «unir» nos referimos a que comparten recursos de hardware y software, funcionando, así como si fueran un solo sistema unificado. Esta unión de recursos se realiza con diversos fines, aunque la razón más popular es ofrecer velocidad y por sobre todo alta disponibilidad ante fallos.

Proveedores pioneros en el uso de servidores en clúster fueron el proyecto ARPANET en 1969, así como Sun Microsystems o IBM. Luego ya en la década de los 90 y posteriores años se popularizó el concepto gracias al uso masivo de servicios de empresas como Microsoft con productos como el viejo MSN, el servicio de servidores de correo Hotmail, Yahoo con Yahoo Mail, o los cientos de servicios de Google que conocemos hoy en día.

En palabras claras, un clúster de servidores se puede definir como un grupo de equipos informáticos que comparten recursos balanceados y están conectados a una red de alta velocidad.

2.2.2. Ventajas de los servidores en clúster

Existen múltiples beneficios y ventajas del uso de servidores en clúster:

Alta disponibilidad: este es uno de los grandes beneficios de la tecnología de clustering, es decir, ofrecer un mejor uptime de servidores gracias a que si falla uno, el resto tomará el trabajo ofreciendo así una alta disponibilidad del servicio.

Alta velocidad de despacho: se obtiene gracias al equilibrio de las cargas, haciendo que los servidores que reciben las solicitudes de los clientes puedan despachar los datos de forma inmediata, sin delays ni retrasos de ningún tipo.

Balanceo de carga: se establecen diferentes métodos de balanceo de carga, haciendo que una o varias peticiones se deriven a determinados hosts, mientras que otras irán destinadas a otro grupo de servidores bajo el clúster. Esto hace que ningún servidor se vea saturado.

Escalabilidad: la mayoría de los clústeres de servidores soportan escalabilidad de forma horizontal, permitiendo agregar más servidores de forma masiva según se necesite debido a una posible creciente demanda de parte de los usuarios.

Resistencia ante ataques DDOS: tener un clúster de servidores ofrece un beneficio adicional de soportar mejor las cargas de sistema que se originan ante ataques DDOS masivos, donde se suelen inundar los servidores con peticiones no deseadas. El uso de múltiples servidores puede ayudar a resistir mejores diferentes tipos de ataque destinados a agotar los recursos de sistema o red (Abarca, 2020).

2.2.3. Computación de alto rendimiento

HPC (2022), la computación de alto rendimiento (HPC) representa la capacidad de procesar datos y realizar cálculos complejos a velocidades muy altas. Para ponerlo en perspectiva, un equipo portátil o de sobremesa con un procesador de 3GHz puede realizar unos 3.000 millones de cálculos por segundo. Aunque esto es mucho más rápido de lo que puede lograr cualquier humano, palidece en comparación con las soluciones HPC que pueden realizar cuatrillones de cálculos por segundo.

Uno de los tipos de soluciones HPC más conocidos es el superordenador. Un superordenador contiene miles de nodos de computación que trabajan juntos para completar una o varias tareas. Esto se denomina procesamiento paralelo. Es similar a tener miles de equipos conectados en red, combinando la potencia computacional para completar tareas más rápidamente.

2.2.4. ¿Por qué es importante la computación de alto rendimiento?

Es a través de los datos que se hacen descubrimientos científicos innovadores, se alimentan las innovaciones que cambian los juegos y se mejora la calidad de vida de miles de millones de personas en todo el mundo. HPC es la base de los avances científicos, industriales y sociales.

A medida que evolucionan tecnologías como el Internet de las cosas (IoT), la inteligencia artificial (AI) y las imágenes 3D, el tamaño y la cantidad de datos con los que las organizaciones deben trabajar crece exponencialmente. Para muchos fines, como la transmisión de un evento deportivo en directo, el seguimiento de una

tormenta en desarrollo, la prueba de nuevos productos o el análisis de tendencias de stock, la capacidad de procesar datos en tiempo real es crucial.

Para seguir un paso por delante de la competencia, las organizaciones necesitan una infraestructura de TI muy fiable y rápida para procesar, almacenar y analizar cantidades masivas de datos.

2.2.5. ¿Cómo funciona HPC?

Las soluciones HPC tienen tres componentes principales:

- Computación.
- Red.
- Almacenamiento.

Para crear una arquitectura de computación de alto rendimiento, los servidores de computación se conectan en un clúster. Los programas de software y los algoritmos se ejecutan simultáneamente en los servidores del clúster. El clúster está conectado en red al almacenamiento de datos para capturar la salida. Juntos, estos componentes funcionan sin problemas para completar un conjunto diverso de tareas.

Para funcionar al máximo rendimiento, cada componente debe seguir el ritmo de los demás. Por ejemplo, el componente de almacenamiento debe poder alimentar e ingerir datos hacia y desde los servidores informáticos tan rápidamente como se procesa. Asimismo, los componentes de red deben ser capaces de admitir el transporte de datos de alta velocidad entre los servidores de computación y el

almacenamiento de datos. Si un componente no puede mantener el ritmo del resto, el rendimiento de toda la infraestructura HPC se resiente.”

Prototipado3D (2022), “Una simulación computacional es el estudio de fenómenos físicos cuyas ecuaciones gobernantes se resuelven con el uso de métodos numéricos y la capacidad de cálculo de las computadoras. Las simulaciones se dividen en dos grandes ramas de análisis: el análisis termo estructural, basado en el método del elemento finito (MEF) y el análisis de fluidos, basado en la dinámica de fluidos computacional (CFD).

El proceso mediante el cual se realiza una simulación computacional suele constar de tres partes fundamentales: a) preprocesamiento, b) solución de ecuaciones (o procesamiento) y c) post procesamiento (ver figura 1). A continuación, se hace una breve descripción de estas etapas:

En el preprocesamiento se introducen los datos geométricos, las condiciones de borde, las excitaciones del sistema, el mallado y el tipo de análisis a realizar. Para poder llevarlo a cabo se necesita conocer de antemano las condiciones de trabajo iniciales, lo que implica muchas veces tomar datos en campo o suposiciones sobre el comportamiento preliminar del componente en estudio, y de aquí el discernimiento sobre qué análisis se debe hacer para obtener la respuesta buscada.

El procesamiento, o solución de las ecuaciones, se logra por medio del uso de un computador y algoritmos especializados en el cálculo numérico. Aunque este paso lo realiza la computadora de forma independiente en base a la configuración previa, la calidad de la información inicial suministrada determinará a su vez la validez y precisión de los resultados a obtener.

Finalmente se realiza el post procesamiento, en donde se visualizan y analizan con más detalle los resultados. De ser necesario se solicitan resultados adicionales que complementen el estudio que se está llevando a cabo, todos en base al archivo de solución resuelto en la etapa anterior. En caso de que los resultados no sean los esperados, o se observen incongruencias en los mismos, se decide si se debe volver a la etapa de preprocesamiento o si se está conforme con la solución.

Mediante la simulación computacional se pueden estudiar una gran variedad de fenómenos físicos, lo que permite al diseñador estudiar diversos escenarios sin necesidad de llevar a cabo numerosos experimentos de laboratorio. Usar este tipo de tecnología para el desarrollo de productos permite alcanzar soluciones de forma rápida y precisa. Desde la ingeniería conceptual hasta la ingeniería de detalles se podrá tener una amplia comprensión de la física involucrada en el problema y de las diferentes respuestas de los sistemas diseñados.

Otra característica importante es que permiten el estudio de productos, equipos y/o procesos existentes, lo que conlleva a las empresas a mejorar sus actividades al aplicar esta tecnología en distintos ámbitos, tales como: mantenimiento, procesos de fabricación, análisis de fallas, investigación, optimización y mejora de productos.

En el caso de la dinámica de fluidos computacional, se puede estudiar el comportamiento de un fluido incluyendo todas sus características físico-químicas reales, permitiendo a los ingenieros evaluar el comportamiento de sus diseños antes de ser construidos, y sin la necesidad de costosos experimentos. Por mencionar algunos ejemplos, con apoyo del CFD se pueden realizar estudios aerodinámicos

para evaluar el comportamiento del flujo externo sobre vehículos y aeronaves (ver figura 3). El estudio proporciona valores como coeficientes de arrastre y sustentación, perfiles de presiones, perfiles de velocidad, etc. De igual forma en otro ramo industrial, en la simulación de turbomáquinas se puede evaluar la eficiencia de consumo de estos equipos y sus curvas de operación

En el campo de los análisis estructurales por el método del elemento finito, hacen posible obtener valores de esfuerzo – deformación en cualquier elemento de una estructura compleja. Otros análisis importantes obtenidos con este método son: estudios térmicos, análisis de vibraciones, evaluaciones de daño acumulado y vida a fatiga, optimización estructural, por mencionar sólo algunos. Con el uso de simulaciones computacionales se estudia el comportamiento de mecanismos complejos y de la interacción entre sus partes. Por ejemplo, luego de un análisis dinámico, es posible obtener valores de desplazamientos, velocidades y aceleraciones en un ensamblaje representando partes de los componentes de la suspensión de un vehículo.

Como hemos visto, todas las simulaciones computacionales se pueden llevar a cabo si se ejecutan de forma adecuada cada una de sus tres etapas fundamentales, tanto en aplicaciones FEA o CFD. Existen una gama de estudios muy variados en otras físicas que no fueron mencionadas, como por ejemplo estudios electromagnéticos de alta y baja frecuencia. En todo caso, esta publicación busca dar información adicional a todos aquellos técnicos y especialistas en áreas de diseño sobre los beneficios técnicos, económicos y de tiempo cuando se implementan las herramientas de simulación en los ciclos convencionales de diseño.”.

2.3. Definiciones de Términos Básicos

- **Análisis:** En el contexto de un clúster de servidores, el análisis implica la evaluación y estudio de las necesidades computacionales y de procesamiento específicas que se requieren. Esto incluye determinar la cantidad de procesamiento matemático necesario, identificar los problemas de cuello de botella, la cantidad de datos a procesar, y los requerimientos de tiempo real o batch. También involucra la evaluación de software y hardware existentes, presupuesto disponible, y recursos (Aliaga et al., 2016).
- **Diseño:** El diseño de un clúster de servidores implica la planificación de la arquitectura del clúster, incluyendo la selección de hardware (servidores, CPUs, GPUs, memoria, almacenamiento, redes) y software (sistemas operativos, herramientas de gestión de clústeres, aplicaciones de procesamiento matemático). Se toman decisiones sobre la configuración de la red, la escalabilidad, la tolerancia a fallos, y la eficiencia energética. Se establece cómo los nodos del clúster interactuarán, cómo se distribuirán las tareas y cómo se manejará la carga de trabajo (Aliaga et al., 2016).
- **Despliegue:** El despliegue del clúster de servidores consiste en la instalación física del hardware, la configuración del software y la puesta en marcha del sistema. Esto incluye la configuración de los sistemas operativos, la instalación y configuración de software específico para el clúster, la prueba de rendimiento y la optimización. El despliegue también

implica la implementación de medidas de seguridad y la configuración de respaldos y recuperación ante desastres (Aliaga et al., 2016).

- **Clúster de servidores:** Un clúster de servidores es un grupo de servidores que trabajan juntos para realizar tareas que requieren alto procesamiento. Estos servidores están conectados a través de una red de alta velocidad y operan como una sola entidad. Un clúster permite el procesamiento paralelo y distribuido de tareas, lo que lo hace ideal para aplicaciones de procesamiento matemático intenso, como simulaciones, cálculos científicos y análisis de datos grandes (Aliaga et al., 2016).
- **Alto procesamiento matemático:** Este término se refiere a la capacidad de realizar cálculos matemáticos complejos y extensivos de manera eficiente. En un clúster, esto se logra mediante la distribución del trabajo entre varios nodos, lo que permite manejar cálculos que serían demasiado pesados o lentos para un solo servidor (Aliaga et al., 2016).

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

3.1. Tipo y Nivel de Investigación

3.1.1. Tipo de Investigación

El tipo de estudio fue explicativo porque estuvo dirigido a responder las causas de los eventos. Su interés se centra en explicar porque ocurre un fenómeno y en qué condiciones se da éste, o porqué dos o más variables están relacionadas, su finalidad es poder explicar el comportamiento de una variable en función de otra(s), con relación de causa – efecto. Requiere de control tanto metodológico como estadístico” (Hernández et al., 2010).

3.1.2. Nivel de Investigación

Según Hernández et al. (2010), en metodología de la investigación mencionó: Una investigación es Descriptiva cuando nos permite describir situaciones y eventos, es decir cómo son y cómo se comportan determinados fenómenos Los estudios descriptivos buscan especificar las propiedades importantes de personas, grupos, comunidades o cualquier otro fenómeno que sea sometido a análisis.

3.2. Población y Muestra

3.2.1. Población

Fue un Clúster de cuatro servidores, el cual se construyó y donde se instaló y programó para las pruebas de cálculo matemático.

3.2.2. Muestra

Por ser una unidad de investigación, la muestra será igual a la población.

3.3. Procedimientos de Recolección de Datos

Plan de recolección de datos

La técnica aplicada en la investigación fue un formulario para la recolección de datos, tiempo que se demora en la generación de números aleatorios, así como el tiempo de ordenamiento por los tres métodos seleccionados.

Utilización de los recursos

El resultado de la muestra de la investigación se aplica al clúster construido con los 4 servidores, al cual se le aplicó la generación de 300000 números aleatorios y su posterior ordenamiento por los tres métodos de ordenamiento.

Tabla 2

Utilización de los recursos

Técnica de Ordenamiento	Número de núcleos
Burbuja	
Inserción	1 al 16
Selección	

Recursos logísticos: Para la ejecución del plan de recolección de información se utilizó 10 formularios por técnica de ordenamiento y por generación de números en forma aleatoria, además de lapiceros y tableros.

3.4. Técnicas e Instrumentos de Recolección de Datos

El formulario fue el instrumento seleccionado para la recolección de información en este estudio.

3.5. Procesamiento para la Recolección de Datos

- La recolección de datos se dio en dos etapas, la primera en la generación de números aleatorios y la segunda cuando se aplicó los métodos de ordenamiento.
- La recolección de datos se dio en el clúster construido dentro de la investigación.
- Para obtener información se desarrolló en los ambientes del centro de investigación en Inteligencia Artificial de la Facultad de Ingeniería de Sistemas e Ingeniería Civil.
- La entrega de información fue de forma ordenada, según los tiempos de generación de los números aleatorios y de acuerdo a la finalización de los procesos de ordenamiento, para que sea más fácil el manejo y se desarrolló de una manera más eficiente el procesamiento de los datos.

3.6. Tratamiento de Datos

Para el tratamiento de la información se utilizó Software Excel, donde se construyeron tablas y figuras.

CAPÍTULO IV

RESULTADOS

INGENIERÍA DEL PROYECTO

Motivación del proyecto

Los clústeres en los sistemas computacionales son empleados para mejorar el rendimiento y la disponibilidad por encima de lo que una computadora personal, siendo más económico que un sistema de alta capacidad de procesamiento, es asequible por el precio, disponibilidad y rapidez de procesamiento.

Objetivo del proyecto

El desarrollo de clúster para procesamiento matemático, tiene un alto impacto en nuestra universidad porque tendremos la capacidad de gran capacidad de procesamiento para aquellas investigaciones que lo requieran como es simulaciones, analítica de datos, inteligencia artificial, búsquedas por similitud, etc. Por lo que nuestro objetivo primordial será construir un Clúster de cuatro equipos computacionales, para tal fin, con la capacidad de que sea incremental.

Importancia un sistema de alto procesamiento computacional

Diaz et al. (2007), "Los supercomputadores, por su propia definición, son las máquinas más costosas que se puedan encontrar en el mercado debido a que usan la tecnología más avanzada disponible. Para mantener un computador dentro de la definición de supercomputador se requiere de una inversión considerable en investigación y desarrollo, cosa que solo es posible dentro de grandes compañías

sólidas. Es precisamente esta alta inversión que hace que los supercomputadores sean onerosos y estén fuera del alcance de la gran mayoría. Proyectos con altos requerimientos de ciclos de CPU, usualmente solicitan tiempo en centros de supercomputación o se corren en máquinas más lentas, lo que lleva a esperar por semanas y hasta meses por los resultados”.

Materiales

- **Sistemas de Cableado Estructurado (SCE)**

Rack de piso standard

Los Racks de piso SATRA de 38 RUs, están diseñados en base a las normas internacionales, con materiales de la mejor calidad para asegurar su resistencia y duración, permitiendo organizar y dar pronto mantenimiento a los equipos y puntos de red instalados. Son de fácil armado y práctica instalación, incluyen tornillos para poder armar la estructura y una base para poder anclar al piso. El anclaje se realiza a través de los orificios de las bases los cuales le brindaran un mayor soporte y estabilidad a la estructura. Están fabricados al 100% con acero LAF, ofreciendo una mayor resistencia al impacto y soporte de la estructura.

Figura 1

Rack de piso standard



Power Rack

El Power Rack SATRA, permite la conexión de los equipos activos a la red eléctrica de manera segura. Diseñado según el estándar EIA, normalizada a 19 pulgadas, para que sea rackeado en un rack o gabinete ocupando solo 1RU. Presenta un Circuit Breaker protector de 15A, un switch con indicador de luz para la activación de los tomacorrientes que cumplen con el estándar NEMA 5-15R.

Figura 2

Power Rack



Bandeja simple

Las bandejas simples para montaje en racks y gabinetes SATRA son diseñadas en las medidas de 1RU, 2RU y 3RU con una profundidad de hasta 39 cm de alta capacidad, cumplen con el estándar EIA para el ancho de 19", con opción de montaje frontal. Pintadas en polvo electrostático que proporciona cinco veces mayor resistencia a corrosión y ralladuras. Solución para instalar aquellos equipos no rackeables.

Figura 3

Bandeja simple



Organizador horizontal

El organizador horizontal de cables SATRA presenta un diseño basado en la norma EIA 310 – D de 19", rackeable en 2 y 1 RU. De base metálica con cuerpo de plástico para mayor duración. Cuenta con divisiones para la correcta presentación y separación de los cables UTP. Diseño ideal para utilizarlo en gabinetes y racks.

Figura 4

Organizador horizontal



Switch 5 puertos Giga Ethernet para escritorio

Switch no administrable de 10/100/1000 Mbps para escritorio que proporciona 5 u 8 puertos auto-negociables, todos soportando características MDI/MDIX. Idealmente diseñado para pequeñas oficinas o el hogar (SOHO) permitiendo interconectar pequeñas redes o grupos de trabajo. De simple y fácil instalación, eliminando cuellos de botella brindando a los usuarios flexibilidad y escalabilidad.

Figura 5

Switch 5 puertos Giga Ethernet para escritorio



Patch Panel Categoría 6

Los patch panel SATRA están diseñados para cumplir y exceder las especificaciones de rendimiento exigidas por la norma ANSI/TIA-568 C.2, tanto para las categorías 5e, 6 y 6A. Asegure el máximo desempeño y logre una instalación sencilla con una terminación estandarizada tipo T568A/B en cumplimiento con la norma y realice el etiquetado de los puntos de red para una mejor administración de cableado.

Figura 6

Patch Panel Categoría 6



Patch Cord Categoría 6

Los patch cords SATRA ofrecen una solución única para entornos de alta densidad de cableado. Su diseño inyectado mejora el control y protección del cable. Son ideales para la conexión de servidores, switches, Patch Panel, o cualquier equipo de distribución con alta concentración de puntos con salidas RJ-45. Están fabricados con cable multifilar, brindando mayor flexibilidad, cumple y supera todas las especificaciones de la norma según sus categorías ya sea 5e, 6 o 6^a

Figura 7*Patch Cord Categoría 6*

Nodos – PC (Terminales)

Un nodo es una unidad de procesamiento, lo cual puede ser una computadora personal común o un servidor. Es por ello que los nodos forman una de las partes fundamentales del clúster, quienes son responsables de todas las actividades en lo que se refiere a la ejecución del software, desde el sistema operativo, lenguaje de programación y almacenamiento de los resultados.

Tabla 3*Especificaciones técnicas de las terminales*

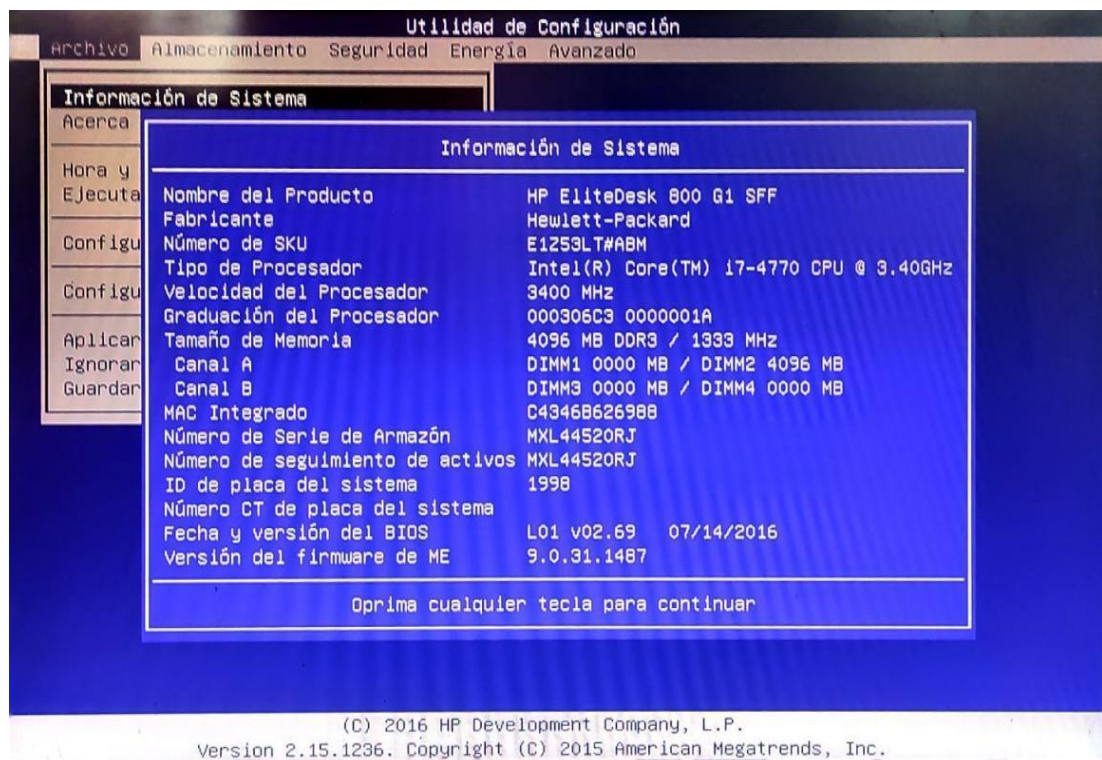
Modelo	EliteDesk 800 G1 SFF (C8N26AV)
Procesador	Intel® Core™ i7-4790 con Intel HD Graphics 4600 (3,6 GHz, 8 MB de caché, 4 núcleos, hasta 4.0GHz con Turbo Boost)
Placa	HP Intel® Q87 Express
Socket	Socket 1150
Chipset	Intel Q87
Memoria RAM	4/8 GB DDR3 ampliable a 32GB
Disco Duro	500GB SATA de 7200 rpm
Case	Factor de forma reducido SFF
Fuente de poder	240W, 92 % de eficiencia, PFC activo
S.O.	N/A
Monitor	No
Unidad óptica	No
Tarjeta de video	Intel HD Graphics 4600
Conectividad	Conexión de red Gigabit Intel® I217LM
Multimedia	Audio HD con códec Realtek ALC221 Tecnología de gestión de audio DTS Studio Sound

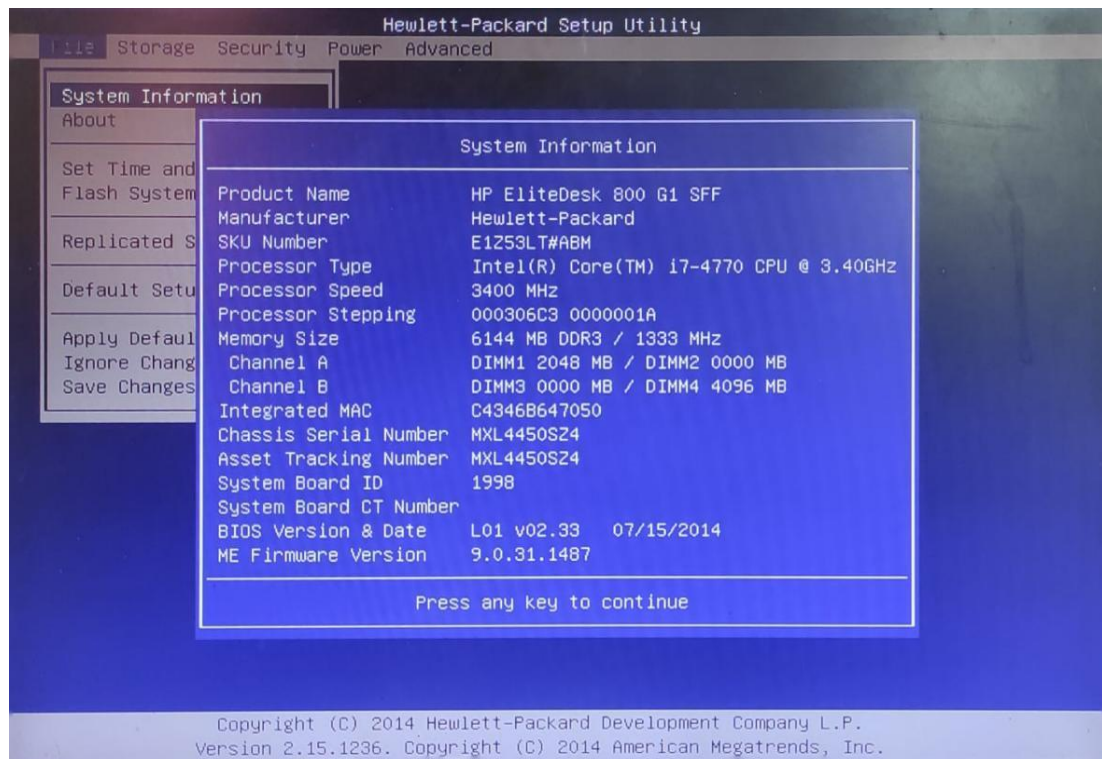
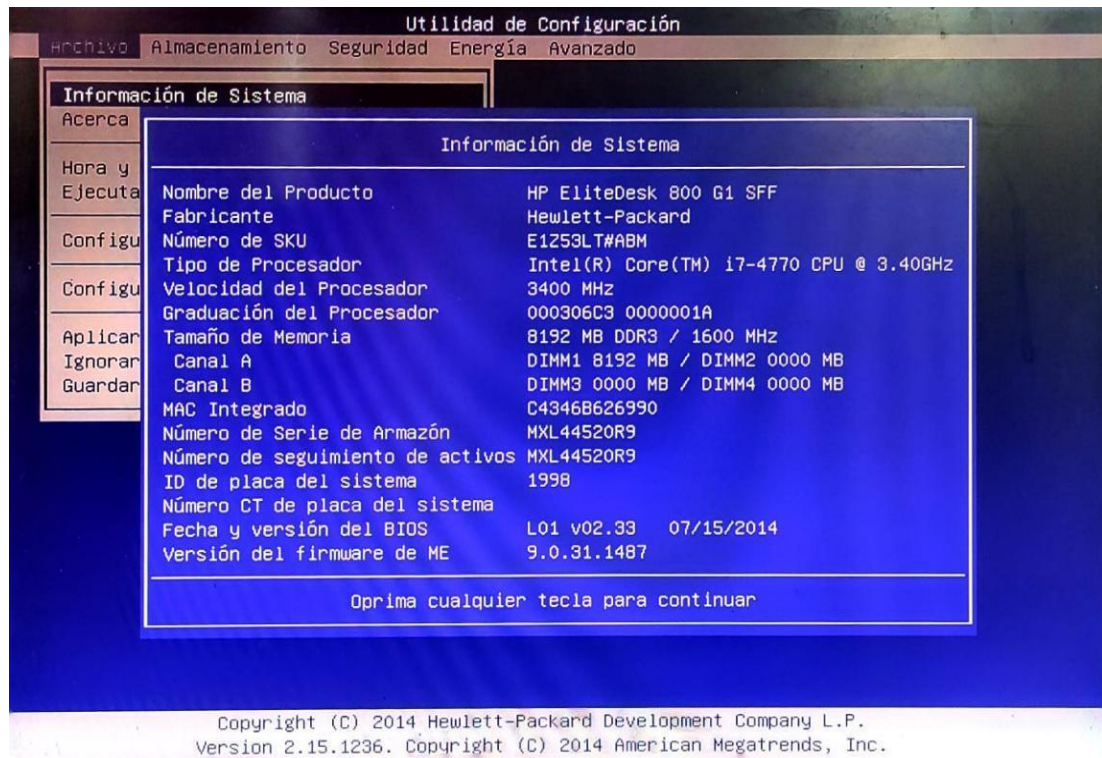
Puertos y Ranuras	• 4 USB 3.0
	• 6 USB 2.0
	• 1 serie
	• 2 PS/2
	• 1 VGA
	• 2 DisplayPort
	• 1 entrada de audio
	• 1 salida de audio
	• 1 RJ-45
	• 1 auricular
	• 1 micrófono

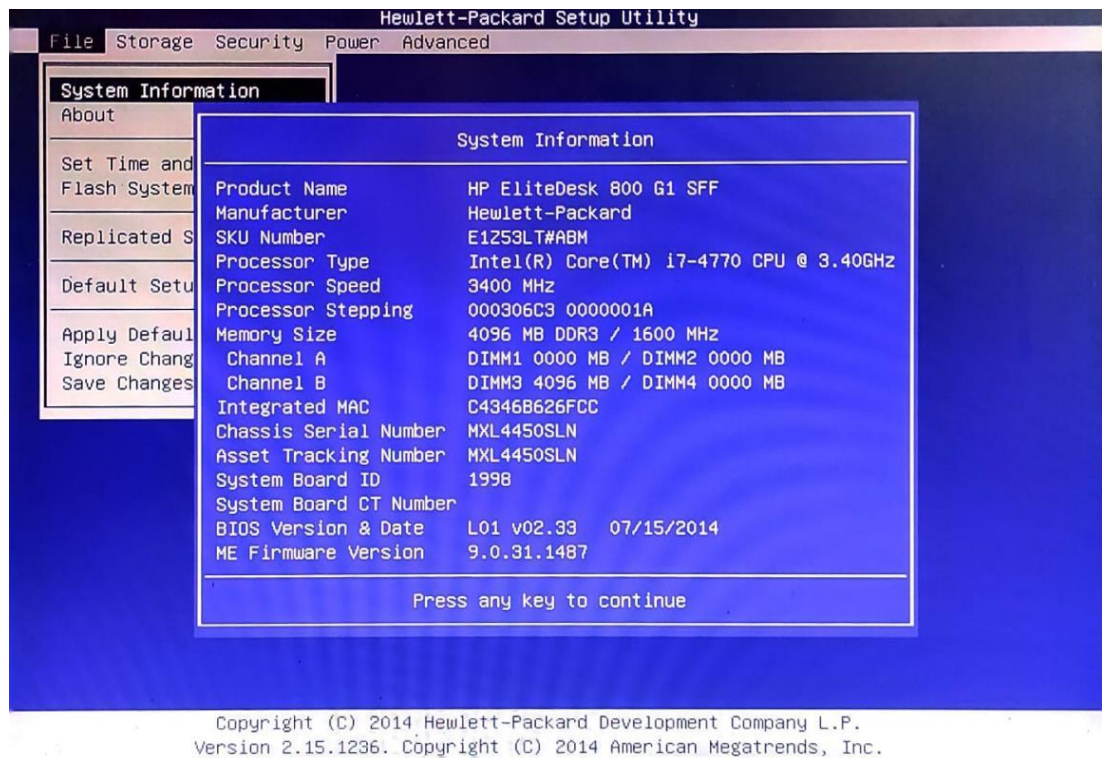
Las características de las cuatro PC's que se ha utilizado en la presente investigación fueron:

Figura 8

Características de las cuatro PC's







Siendo el requerimiento en hardware:

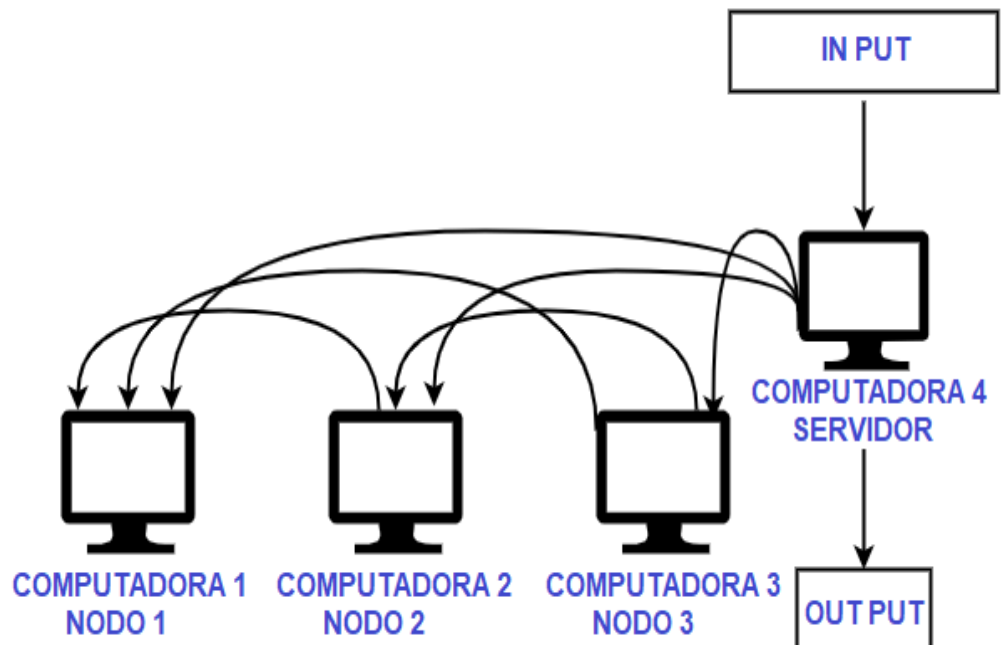
Tabla 4

Recursos en hardware

Descripción	Cantidad
Rack de piso de 38RU	1
Power Rack de 6 tomas	1
Bandeja Fija de 2 RU	1
Patch Panel de 12U	1
Patch Cord Cat 5E- 1 Mt	4
Switch Giga de 5 Puertos	1
CPU Core i7 8 Gb- HD -500G 240W Consumo	4

Tabla 5*Recursos en software*

	DESCRIPCIÓN
Sistema operativo	DEBIAN 11
Programas usados	MPI, PYTHON, MPY4PY
Lenguaje de programación	PYTHON V 3.9, Y C++
Librerías	MPI, MPI4PY, NUMPY
Algoritmos	BURBUJA, INSERCIÓN, SELECCIÓN
Ide	Visual Studio Code

Análisis y diseño del clúster**Figura 9***Análisis y diseño del clúster*

Despliegue del clúster

Construcción de los Patch Cord



Armado de Rack/Patch Panel / Bandejas



Configuración del Clúster



La determinación los criterios del análisis y diseño de un clúster de servidores para alto procesamiento matemático, en base al análisis y diseño propuesto se plantea como un modelo incremental, el cual podrá variar de acuerdo a la necesidad de procesamiento o sistema a ejecutarse (simulaciones, programas de aprendizaje máquina, etc.) enmarcado en una red estructurada con una topología estrella para facilitar la transferencia de información, y todo lo que conlleva (velocidad, consistencia, fiabilidad, seguridad, etc.).

RESULTADOS

Luego de haber determinado la validez del instrumento para la recolección de los datos que fue un formulario, dichos resultados son presentados a continuación en cuadros estadísticos, tablas de distribución de frecuencias y figuras, los mismos que facilitaron el análisis y la interpretación correspondiente.

No debe confundirse el procesamiento distribuido con el procesamiento paralelo, un sistema de procesamiento paralelo tiende a estar físicamente muy cerca, mientras que un sistema distribuido no necesariamente.

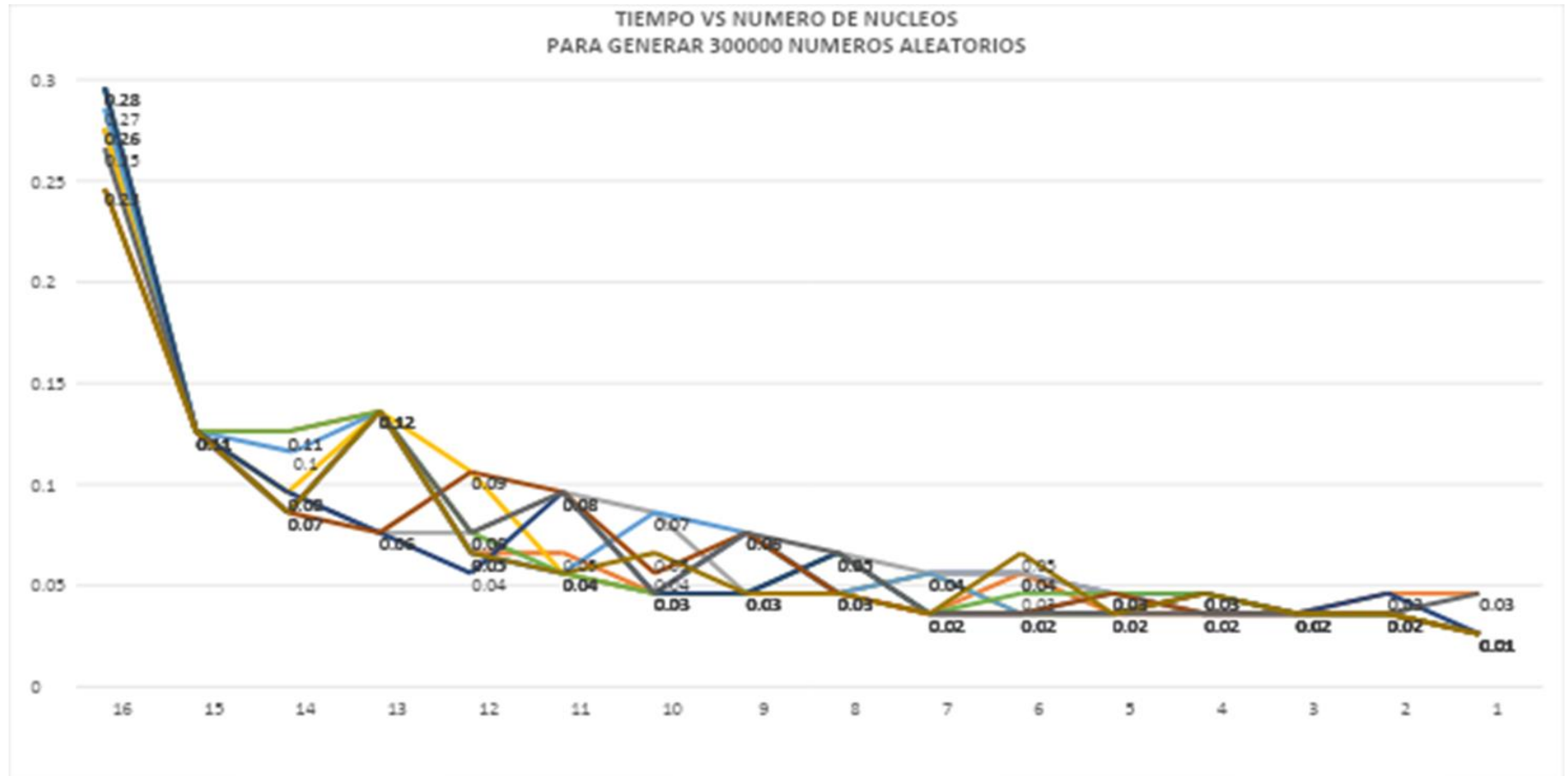
Para poder demostrar cómo influye un clúster de servidores en el requerimiento de alto procesamiento matemático, se ha aplicado en la generación de números aleatorios y su posterior ordenamiento, en nuestro caso aplicamos su generación desde un núcleo hasta 16 núcleos, distribuyendo la carga entre ellos, tal como se ha podido demostrar en las tablas.

Tabla 6*Generación de 300000 números aleatorios vs número de núcleos*

NÚMERO DE GENERACIONES	NUMERO DE NUCLEOS															
	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
1	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.03	0.06	0.03	0.04	0.05	0.12	0.07	0.11	0.28
2	0.03	0.03	0.02	0.02	0.02	0.04	0.02	0.03	0.03	0.03	0.05	0.05	0.12	0.11	0.11	0.26
3	0.01	0.02	0.02	0.02	0.03	0.04	0.04	0.05	0.03	0.07	0.08	0.06	0.06	0.08	0.11	0.26
4	0.01	0.02	0.02	0.03	0.02	0.02	0.04	0.03	0.03	0.03	0.04	0.09	0.12	0.08	0.11	0.26
5	0.01	0.02	0.02	0.02	0.03	0.02	0.04	0.03	0.06	0.07	0.04	0.05	0.12	0.1	0.11	0.27
6	0.01	0.02	0.02	0.03	0.03	0.03	0.02	0.05	0.03	0.03	0.04	0.06	0.12	0.11	0.11	0.28
7	0.01	0.03	0.02	0.03	0.02	0.02	0.02	0.05	0.03	0.03	0.08	0.04	0.06	0.08	0.11	0.28
8	0.01	0.02	0.02	0.02	0.03	0.02	0.02	0.03	0.06	0.04	0.08	0.09	0.06	0.07	0.11	0.23
9	0.03	0.02	0.02	0.02	0.02	0.02	0.02	0.05	0.06	0.03	0.08	0.06	0.12	0.07	0.11	0.25
10	0.01	0.02	0.02	0.03	0.02	0.05	0.02	0.03	0.03	0.05	0.04	0.05	0.12	0.07	0.11	0.23

Figura 10

Generación de 300000 números aleatorios, tiempo vs número de núcleos



Interpretación parcial

Se ha procedido a generar los 3 000 000 número aleatorios, haciendo uso desde un procesador hasta los 16 procesadores del clúster, es decir un total de 160 veces. Al usar un solo procesador el tiempo de generación tiene un tiempo mínimo de 23 segundos y un máximo de 28 segundos, al ampliar en 02 procesadores el tiempo se reduce a más de la, en los 10 procesos el tiempo fue de 11 segundos: a partir del uso del 03 procesador hasta cuando se usan los 08 procesador la disminución del tiempo de procesamiento disminuye un segundo en promedio entre uno y otro; mientras que al usar 09 procesadores; hasta 11 procesadores la diferencia es menor a un segundo, finalmente en al usar los Por lo que se ha podido apreciar que la diferencia entre el uso de 12 y 16 procesadores, la diferencia es de milésimas de segundo.

En el caso del consumo de energía eléctrica el sistema habilito progresivamente el uso de los terminales de acuerdo al requerimiento de procesamiento, cada vez que se requiere el procesamiento de 4 procesadores, se inicia el funcionamiento de un nodo, por lo que podemos afirmar que el consumo de energía es eficiente, iniciando con 240W. por el primer nodo, llegando a consumir un máximo de 960 W. cuando el requerimiento está en su máxima capacidad.

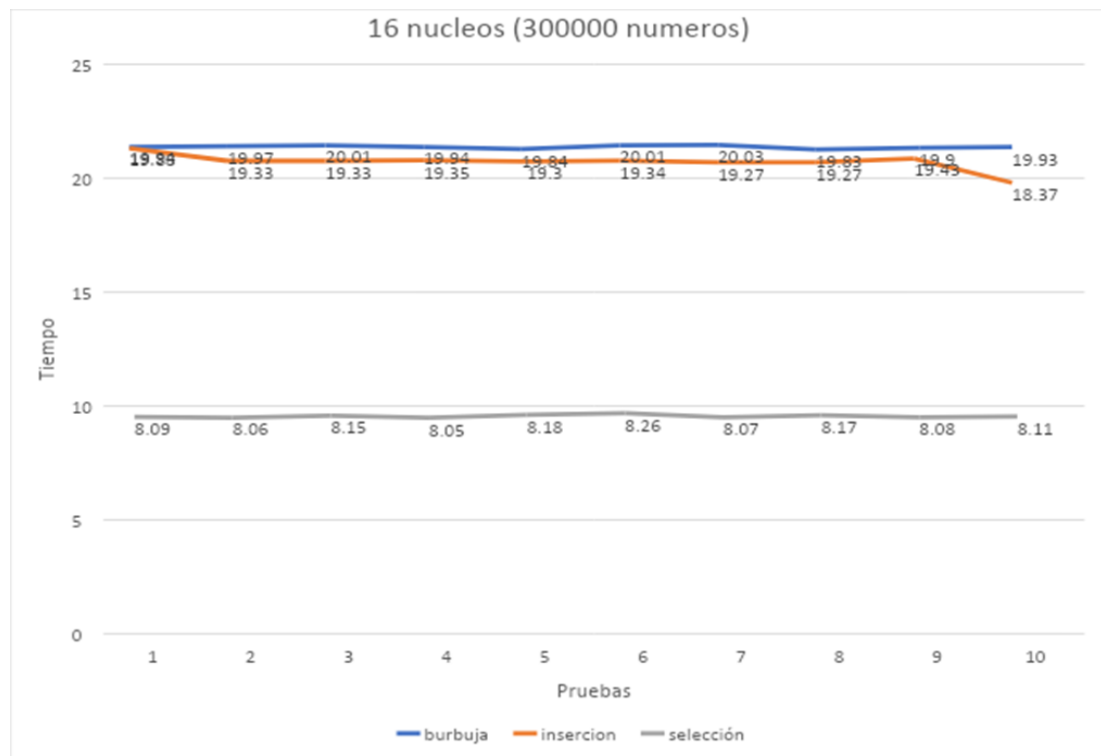
Tabla 7

Ordenamiento de 300000 números aleatorios, 16 núcleos vs método de ordenamiento

16 núcleos (300000 números)			
Nº de prueba	Burbuja	Inserción	Selección
1	19.94	19.88	8.09
2	19.97	19.33	8.06
3	20.01	19.33	8.15
4	19.94	19.35	8.05
5	19.84	19.3	8.18
6	20.01	19.34	8.26
7	20.03	19.27	8.07
8	19.83	19.27	8.17
9	19.9	19.43	8.08
10	19.93	18.37	8.11

Figura 11

Ordenamiento de 300000 números aleatorios, 16 núcleos vs método de ordenamiento



Interpretación parcial

En el ordenamiento de 300.000 números aleatorios, usando 16 núcleos, se ha tenido los siguientes resultados, el método de selección es el más eficiente, debido a que el tiempo promedio del ordenamiento en las 10 corridas ha sido 8.12 segundos, el método de inserción en las 10 corridas a alcanzado un tiempo promedio de 19,29 segundos, finalmente el método burbuja a alcanzado un tiempo promedio de 19.94 segundos, por lo que sostenemos que no solo dependerá de la capacidad de cálculo que se tenga en un clúster, sino el algoritmo a emplear para que sea eficiente el trabajo a realizar.

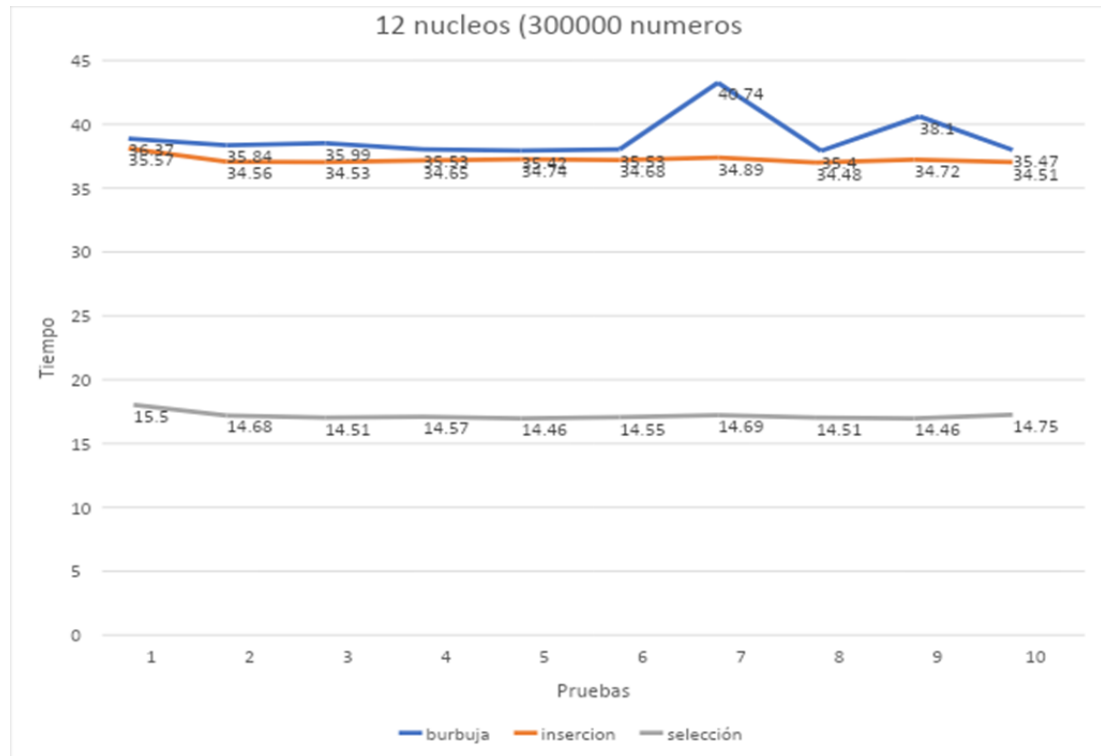
Tabla 8

Ordenamiento de 300000 números aleatorios, 12 núcleos vs método de ordenamiento

12 núcleos (300000 números)			
N° de prueba	Burbuja	Inserción	Selección
1	36.37	35.57	15.5
2	35.84	34.56	14.68
3	35.99	34.53	14.51
4	35.53	34.65	14.57
5	35.42	34.74	14.46
6	35.53	34.68	14.55
7	40.74	34.89	14.69
8	35.4	34.48	14.51
9	38.1	34.72	14.46
10	35.47	34.51	14.75

Figura 12

Ordenamiento de 300000 números aleatorios, 12 núcleos vs método de ordenamiento



Interpretación parcial

En el ordenamiento de 300.000 números aleatorios, usando 14 núcleos, se ha podido apreciar que: el método de selección es el más eficiente, con un tiempo promedio del ordenamiento en las 10 corridas de 14.67 segundos con una diferencia de casi 6 segundos frente al uso de 16 procesadores, mientras que el método de inserción en las 10 corridas se ha tenido un tiempo promedio de 34.73 segundos, demorando casi un 45% más de tiempo frente al uso de los 16 procesadores, finalmente el método burbuja alcanzado un tiempo promedio de 36.44 segundos, con una diferencia de casi 47% frente al uso de 16 procesadores.

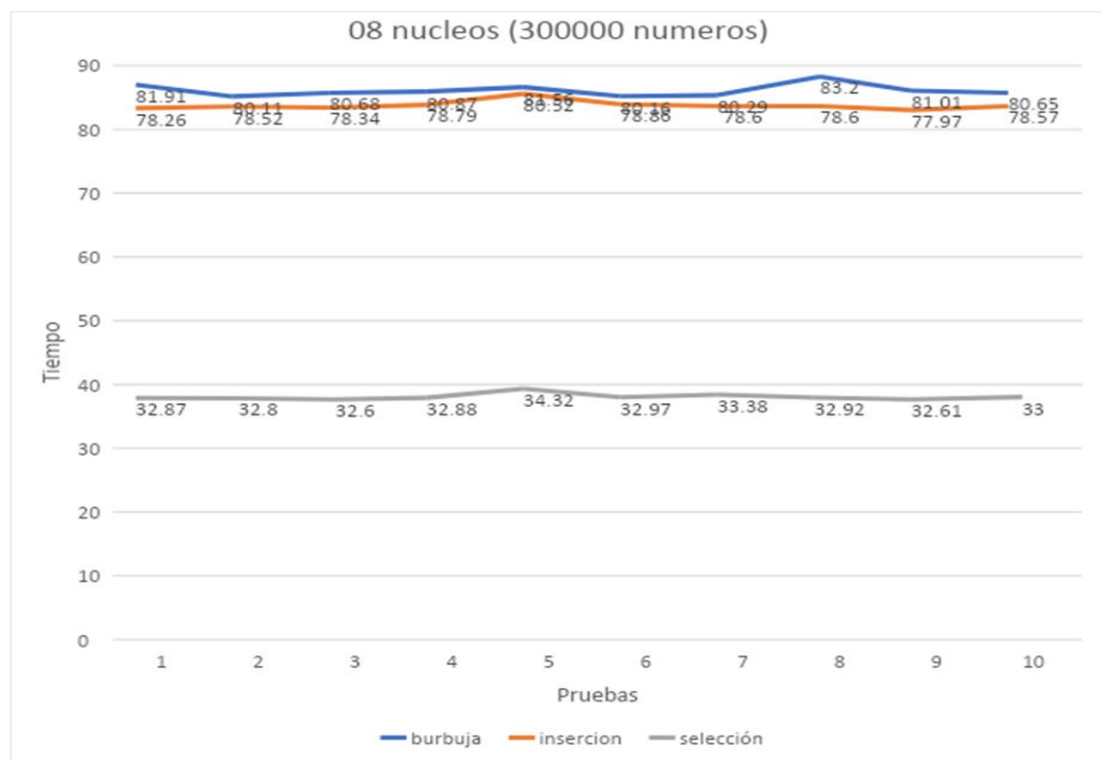
Tabla 9

Ordenamiento de 300000 números aleatorios, 8 núcleos vs método de ordenamiento

8 núcleos (300000 números)			
N° de prueba	Burbuja	Inserción	Selección
1	81.91	78.26	32.87
2	80.11	78.52	32.8
3	80.68	78.34	32.6
4	80.87	78.79	32.88
5	81.56	80.52	34.32
6	80.16	78.86	32.97
7	80.29	78.6	33.38
8	83.2	78.6	32.92
9	81.01	77.97	32.61
10	80.65	78.57	33

Figura 13

Ordenamiento de 300000 números aleatorios, 8 núcleos vs método de ordenamiento



Interpretación parcial

En el ordenamiento de 300.000 números aleatorios, usando 8 núcleos, se ha podido apreciar que: el método de selección es el más eficiente, con un tiempo promedio del ordenamiento en las 10 corridas ha alcanzado un promedio de 33.04 segundos con una diferencia de casi 18.17 segundos frente al uso de 14 procesadores, mientras que el método de inserción en las 10 corridas se ha tenido un tiempo promedio de 78.70 segundos, demorando 43.97 segundos más de tiempo frente al uso de los 14 procesadores, finalmente el método burbuja a alcanzado un tiempo promedio de 81.04 segundos, con una diferencia de 44.60 segundos frente al uso de 14 procesadores.

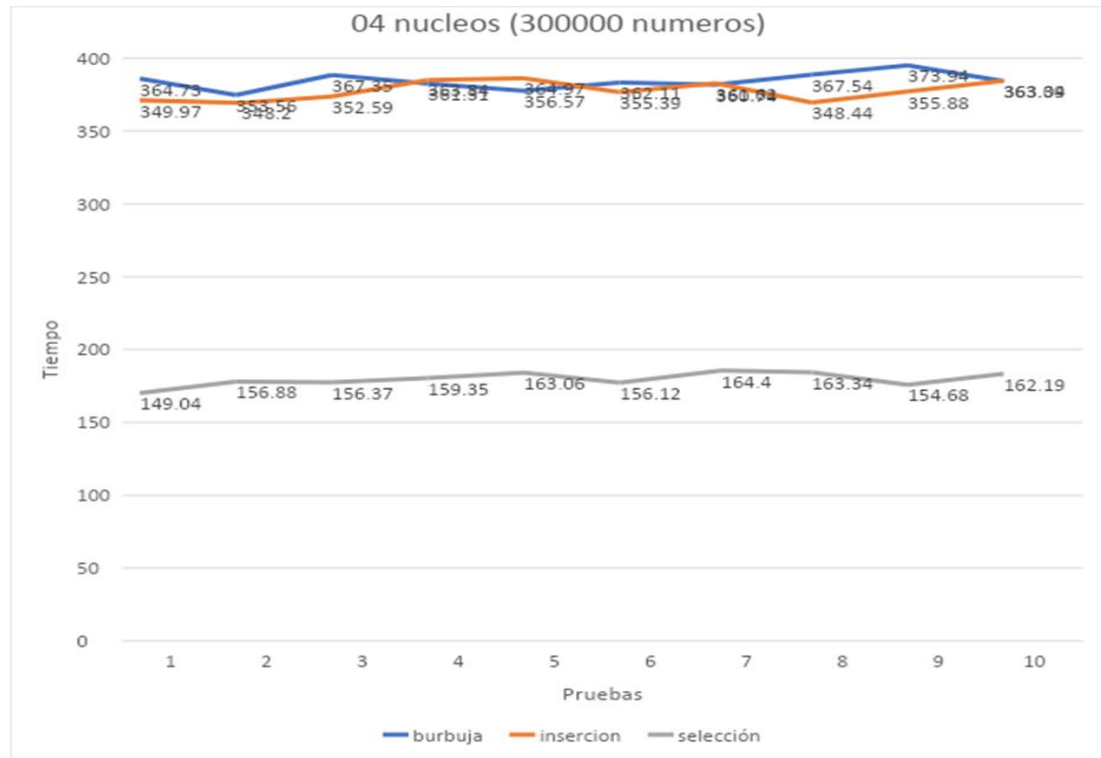
Tabla 10

Ordenamiento de 300000 números aleatorios, 4 núcleos vs método de ordenamiento

4 núcleos (300000 números)			
Nº de prueba	Burbuja	Inserción	Selección
1	364.73	349.97	149.04
2	353.56	348.2	156.88
3	367.35	352.59	156.37
4	361.31	363.84	159.35
5	356.57	364.97	163.06
6	362.11	355.39	156.12
7	360.74	361.62	164.4
8	367.54	348.44	163.34
9	373.94	355.88	154.68
10	363.04	363.39	162.19

Figura 14

Ordenamiento de 300000 números aleatorios, 4 núcleos vs método de ordenamiento



Interpretación parcial

En el ordenamiento de 300.000 números aleatorios, usando 4 núcleos, se ha podido apreciar que: el método de selección sigue siendo el más eficiente por el algoritmo con que ha sido diseñado, con un tiempo promedio del ordenamiento en las 10 corridas de 158.54 segundos con una diferencia de 125.54 segundos frente al uso de 8 procesadores, mientras que el método de inserción en las 10 corridas se ha tenido un tiempo promedio de 356.43 segundos, demorando 277.73 segundos más de tiempo frente al uso de los 14 procesadores, finalmente el método burbuja a alcanzado un tiempo promedio de 363.09 segundos, con una diferencia de 282.05 segundos frente al uso de 8 procesadores.

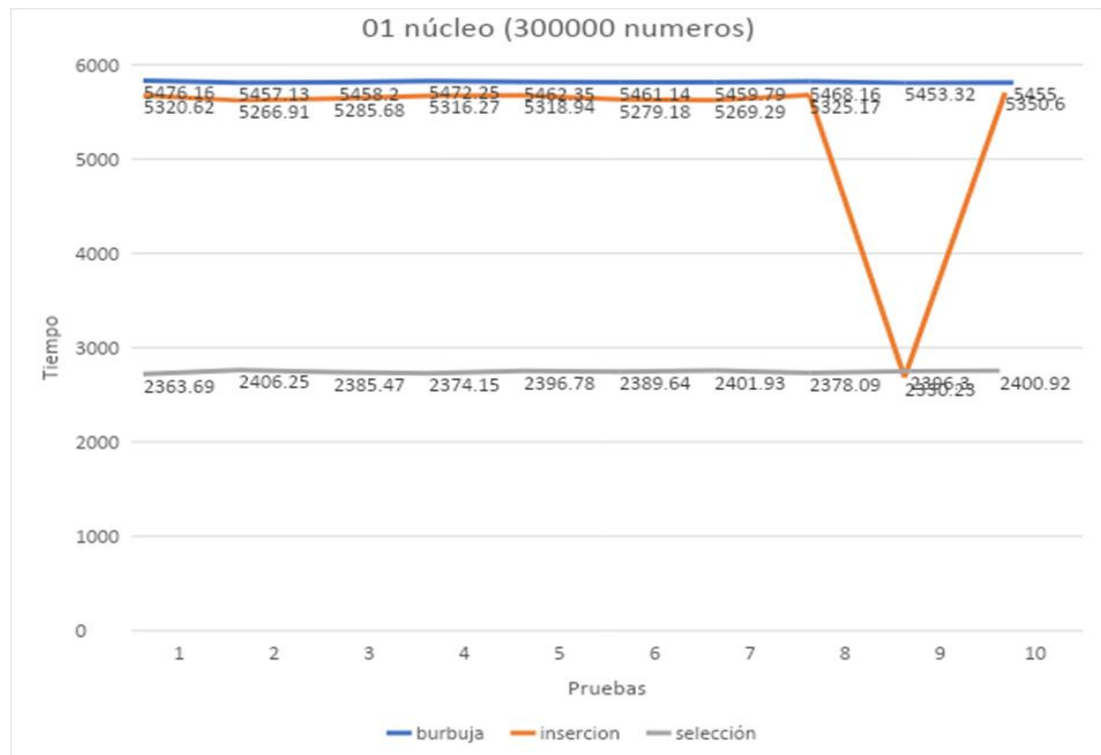
Tabla 11

Ordenamiento de 300000 números aleatorios, 1 núcleo vs método de ordenamiento

1 núcleo (300000 números)			
Nº de prueba	burbuja	inserción	selección
1	5476.16	5320.62	2363.69
2	5457.13	5266.91	2406.25
3	5458.2	5285.68	2385.47
4	5472.25	5316.27	2374.15
5	5462.35	5318.94	2396.78
6	5461.14	5279.18	2389.64
7	5459.79	5269.29	2401.93
8	5468.16	5325.17	2378.09
9	5453.32	2330.23	2396.3
10	5455	5350.6	2400.92

Figura 15

Ordenamiento de 300000 números aleatorios, 1 núcleo vs método de ordenamiento



Interpretación parcial

En el ordenamiento de 300.000 números aleatorios, usando 1 núcleo, se ha podido apreciar que: el método de selección sigue siendo el más eficiente por el algoritmo con que ha sido diseñado, con un tiempo promedio del ordenamiento en las 10 corridas de 2,389.32 segundos, mientras que el método de inserción en las 10 corridas se ha tenido un tiempo promedio de 5,006.29 segundos, finalmente el método burbuja a alcanzado un tiempo promedio de 5,462.35 segundos. Aquí podemos apreciar que se hace más pesado el ordenamiento con un solo procesador; por lo que el uso de un clúster para la solución de generación de números aleatorios y su posterior ordenamiento, tiene un valor diferenciado.

CAPÍTULO V

DISCUSIÓN DE RESULTADOS

A continuación, mostrando tablas de datos y figuras más importantes de la investigación, apreciando los valores diferenciados en el uso de un solo procesador y el uso de 16 procesadores.

En las tablas 12 al 15, podemos apreciar:

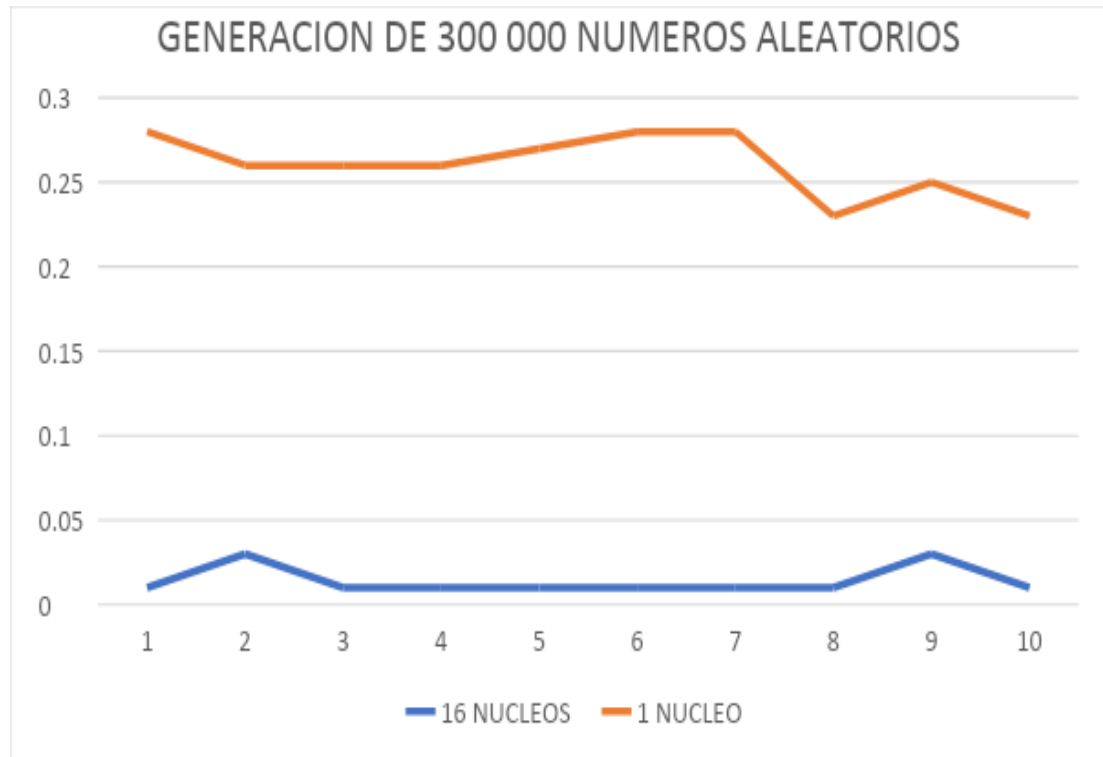
Tabla 12

Generación de 300.000 números en forma aleatoria

Número de generaciones	Numero de núcleos	
	16 núcleos	1 núcleo
1	0.01	0.28
2	0.03	0.26
3	0.01	0.26
4	0.01	0.26
5	0.01	0.27
6	0.01	0.28
7	0.01	0.28
8	0.01	0.23
9	0.03	0.25
10	0.01	0.23

Figura 16

Generación de 300 000 números aleatorios



Interpretación parcial

En la generación de números aleatorios, un procedimiento sencillo, se puede apreciar la diferencia de usar un clúster, al usar un solo núcleo, la generación de 300 mil números aleatorios ha tomado 26 segundos; mientras que al usar los 16 núcleos el tiempo se ha reducido a 0.014 segundo; reduciendo el tiempo en forma exponencial.

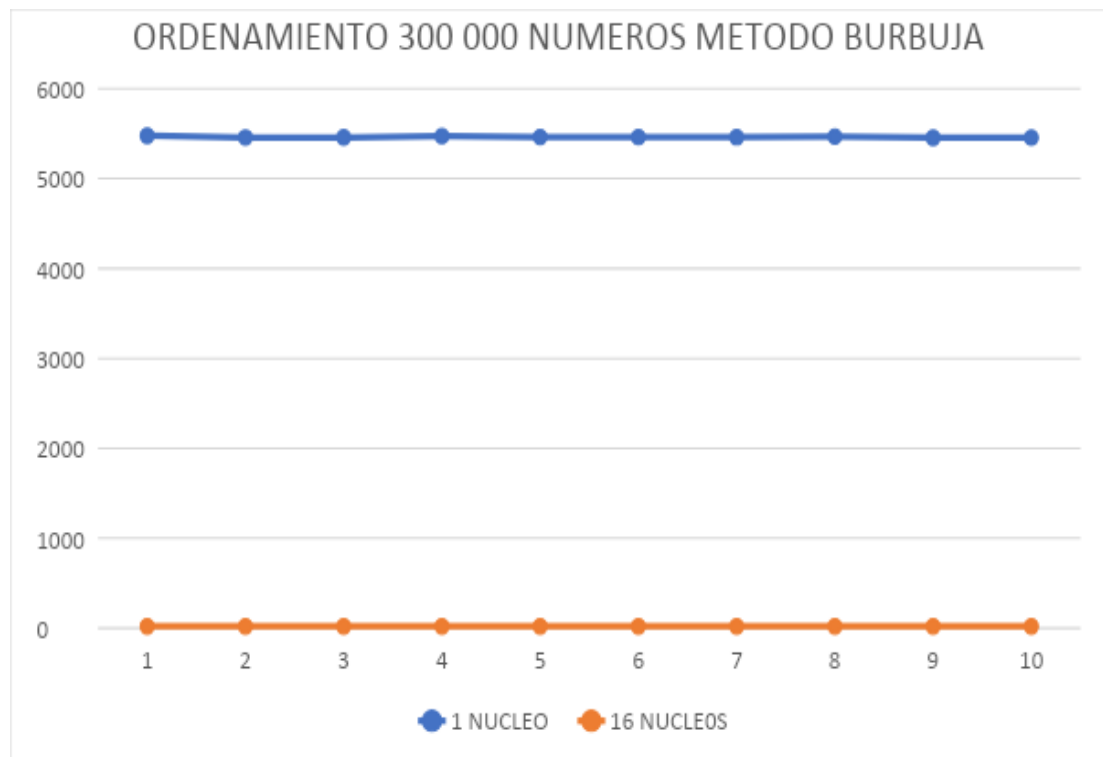
Tabla 13

Ordenamiento de 300.000 números método burbuja

N° de prueba	Burbuja	
	1 núcleo	16 núcleos
1	5476.16	19.94
2	5457.13	19.97
3	5458.2	20.01
4	5472.25	19.94
5	5462.35	19.84
6	5461.14	20.01
7	5459.79	20.03
8	5468.16	19.83
9	5453.32	19.9
10	5455	19.93

Figura 17

Ordenamiento de 300 000 números método burbuja



Interpretación parcial:

En el caso de este método de ordenamiento, que es el más básico en su algoritmo podemos apreciar que en las 10 corridas con bases de datos diferentes de 300.000 números aleatorios, al usar un núcleo el tiempo de ordenamiento es de 5,462.35 segundos (91.04 minutos); pero al usar los 16 núcleos del clúster se reduce a 19.94 segundos (0.33 minutos), esto se debe a la capacidad de desarrollar un procesamiento paralelo y una programación por hebras.

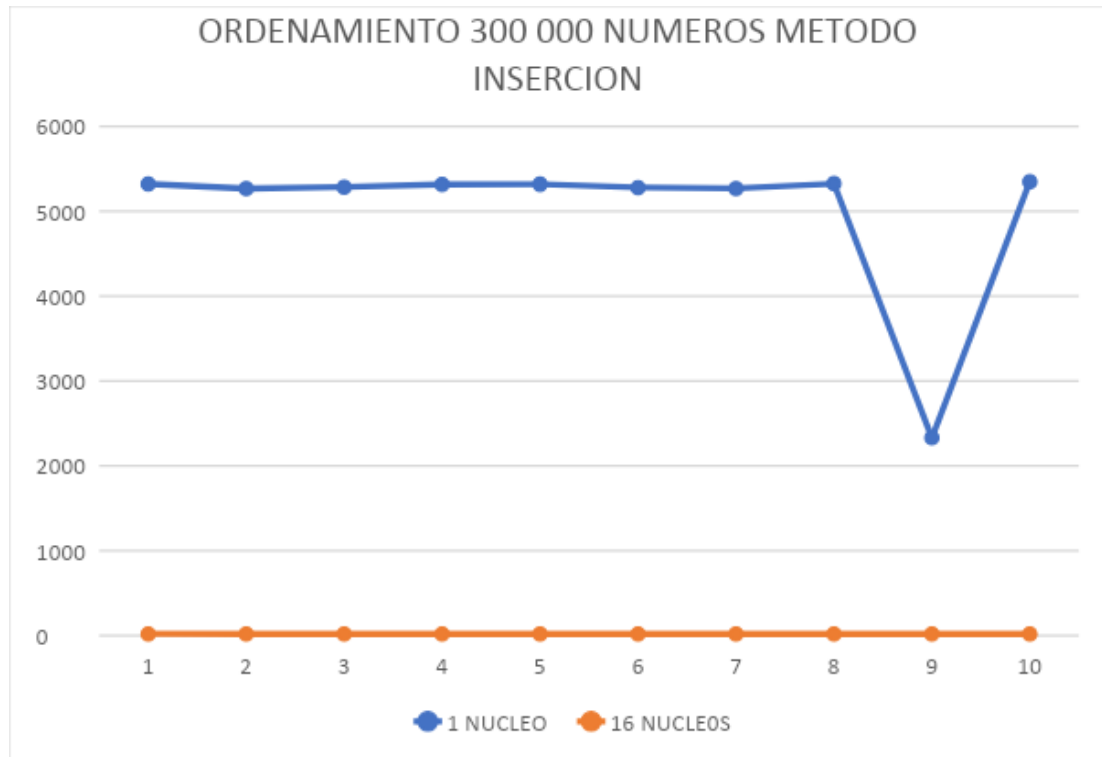
Tabla 14

Ordenamiento de 300 000 números – método inserción

N° de prueba	Inserción	
	1 núcleo	16 núcleos
1	5320.62	19.88
2	5266.91	19.33
3	5285.68	19.33
4	5316.27	19.35
5	5318.94	19.3
6	5279.18	19.34
7	5269.29	19.27
8	5325.17	19.27
9	2330.23	19.43
10	5350.6	18.37

Figura 18

Ordenamiento de 300 000 números método inserción

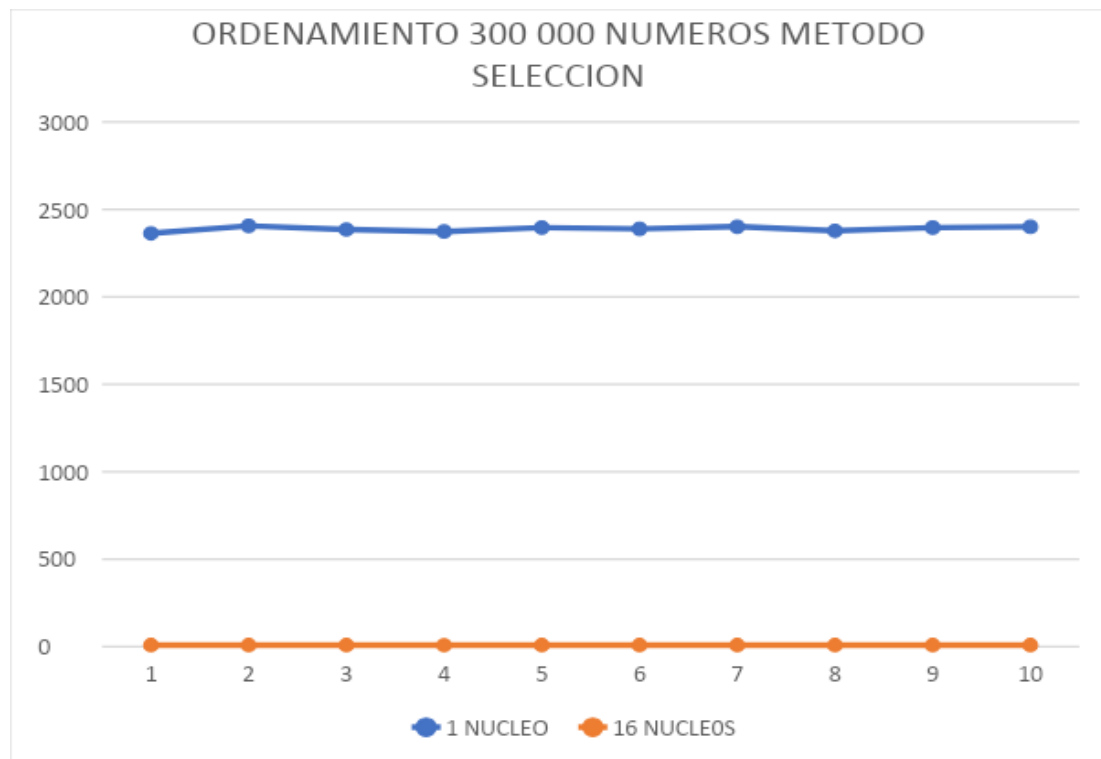


Interpretación parcial:

Al usar esta metodología en el ordenamiento de los 300.000 números aleatorios en las 10 corridas, al usar un núcleo se ha tenido un tiempo promedio de 5,006.29 segundos (83.44 minutos); mientras que, al usar los 16 núcleos del clúster, se puede apreciar que baja a un promedio de 19.29 segundos (0.32 minutos), apreciándose la eficiencia de utilizar un clúster y las ventajas en el cálculo matemático.

Tabla 15*Ordenamiento de 300 000 números – método selección*

N° de prueba	Selección	
	1 núcleo	16 núcleos
1	2363.69	8.09
2	2406.25	8.06
3	2385.47	8.15
4	2374.15	8.05
5	2396.78	8.18
6	2389.64	8.26
7	2401.93	8.07
8	2378.09	8.17
9	2396.3	8.08
10	2400.92	8.11

Figura 19*Ordenamiento de 300 000 números método selección*

Interpretación parcial:

Después de las corridas en el clúster para el ordenamiento de los 300 000 números aleatorios, se ha tenido que al usar un solo núcleo el tiempo de ordenamiento fue en promedio de 2,389.32 segundos (39.82 minutos), mientras que al usar los 16 núcleos se ha tenido un tiempo de ordenamiento promedio de 8.12 segundos (0.14 minutos) siendo este método el más eficiente. Aquí podemos sostener que no solo dependerá del aporte de los tiempos de procesamiento tener una alta capacidad de procesamiento sino un buen algoritmo, la combinación de ambos permitirá que el costo de procesamiento computacional sea el adecuado.

Verificación de hipótesis

Se puede observar el comportamiento de los indicadores de cómo influye en el requerimiento de procesamiento matemático, por lo que podemos aceptar la hipótesis general, tal como se ha podido apreciar, donde la generación de 300 mil números aleatorios ha tomado 26 segundos; mientras que al usar los 16 núcleos el tiempo se ha reducido a 0.014 segundo, reduciéndose los tiempos en forma exponencial.

Al aplicar los métodos de ordenamiento de los 300 mil números aleatorios se ha tenido los siguientes resultados: En el caso del este método de ordenamiento por burbuja, que es el algoritmo básico, podemos apreciar, al usar un núcleo el tiempo de ordenamiento es de 91.04 minutos en promedio; pero al usar los 16 núcleos del clúster se reduce a 0.33 minutos, esto se debe a la capacidad de desarrollar un procesamiento paralelo y una programación por hebras; mientras que en las 10 corridas por el método de inserción, al usar un núcleo se ha tenido un

tiempo promedio de 83.44 minutos; mientras que al usar los 16 núcleos del clúster, se puede apreciar que baja a un promedio de 0.32 minutos, finalmente Al aplicar el método de ordenamiento por selección, al usar un solo núcleo el tiempo de ordenamiento fue en promedio de 39.82 minutos, mientras que al usar los 16 núcleos se ha tenido un tiempo de ordenamiento promedio de 0.14 minutos, siendo este método el más eficiente.

Por todo lo demostrado, estamos en las condiciones de poder confirmar la hipótesis general, así como las específicas.

CONCLUSIONES

- Se ha podido determinar qué un clúster de servidores fluye significativamente en el requerimiento de procesamiento matemático, tal como se ha podido demostrar en las tablas 9 al 12, donde se aprecia que tanto en la generación de números como en su ordenamiento al usar todos los nodos del clúster los tiempos de procesamiento se reducen exponencialmente de horas a solo segundos.
- Se ha podido determinar los criterios del análisis y diseño de un clúster de servidores para alto procesamiento matemático, tal como se aprecia en el Análisis y Diseño del Clúster, donde sí tiene que definir un servidor y luego los nodos, los cuales se interconectan a través de una red de topología estrella; con una capacidad incremental, de poder incrementando más nodos de acuerdo a la necesidad del usuario (sistema a procesar).
- Se ha podido determinar en qué medida el rendimiento computacional y la eficiencia del consumo de energía varía en un clúster de servidores para alto procesamiento matemático, por el tiempo que se demora en ejecutar un conjunto de instrucciones, en nuestro caso la generación de 300 000 números aleatorios y su posterior ordenamiento; en el caso de la generación de números aleatorios el rendimiento computacional llega a superar los 1,857.14% en promedio en las 10 corridas; mientras que en el ordenamiento se llega a superar el 27,155,72% cuando se utiliza el 100% de la capacidad del clúster, la eficiencia del consumo de energía se traslada al tiempo que se requiere que el clúster debe estar en funcionamiento para terminar de ejecutar

el programa, por lo que la eficiencia es directamente proporcional al uso progresivo de nodo, teniendo un consumo máximo de 240 Watts, por equipo; consumiendo un máximo de 960 Watts, cuando se usan los 16 nodos.

RECOMENDACIONES

- Mejorar el clúster que se ha desarrollado incrementando más terminales, para incrementar la capacidad de procesamiento, y pueda ser aplicado en requerimientos de alta capacidad de procesamiento matemático y simulaciones.
- Esta investigación sirva de base para poder construir un clúster para procesamiento gráfico; donde se pueda aplicar en requerimiento de búsquedas por similitud, Inteligencia Artificial, Etc.

REFERENCIAS BIBLIOGRÁFICAS

- Abarca, F. (2020). Análisis de rendimiento de un clúster heterogéneo formado por placas Raspberry Pi. Universidad Complutense de Madrid.
- Acero, W., Aguilar, J., & Mejía, D. (2016). Clúster de alta disponibilidad para virtualizar los servidores de adquisición y procesamiento de datos del Instituto Geofísico.
- Aliaga, E., Román, W., & Pinedo, R. (2016). Sistema automatizado para el control y monitoreo del comportamiento de alevinos de paiche en cautiverio. Universidad Nacional de Ucayali.
- Borges, E. (2019). Servidor en clúster. Recuperado 15 de diciembre de 2022, de infranetworking website: <https://blog.infranetworking.com/servidor-en-cluster/>
- Cáceres, G. (2012). Estrategia de implementación de un clúster de alta disponibilidad de N nodos sobre Linux usando software libre.
- Chuquiguanca, L. y Malla, E. (2015). Arquitectura clúster de alto rendimiento utilizando herramientas de software libre (Tesis de pregrado, Universidad). Universidad Nacional de Loja, Ecuador.
- Díaz, G., Hamar, V., Isea, R., Rojas, F., Ruíz, N., Torrens, R., Uzcátegui, M., Flórez-López, J., Hoeger, H., Núñez, L., & Mendoza, C. (2007). E-science initiatives in Venezuela. In J. Casado, R. Mayo, & R. Muñoz (Eds.), Proceedings of the Spanish Conference on e-Science Grid Computing (pp. 45-52). CIEMAT.

Hernández, R., Fernández, C., y Baptista, P. (2010). Metodología de la investigación. Mc Graw-Hill 5ta Edición.

HPC (2022). Recuperado 16 de diciembre de 2022, de NetApp website:

[https://www.netapp.com/es/data-storage/high-performance-computing/what-is-hpc/#:~:text=La%20computaci%C3%B3n%20de%20alto%20rendimiento%20\(HPC\)%20representa%20la%20capacidad%20de,millones%20de%20c%C3%A1culos%20por%20segundo](https://www.netapp.com/es/data-storage/high-performance-computing/what-is-hpc/#:~:text=La%20computaci%C3%B3n%20de%20alto%20rendimiento%20(HPC)%20representa%20la%20capacidad%20de,millones%20de%20c%C3%A1culos%20por%20segundo)

Nugroho, S., & Widiyanto, A. (2020). Designing parallel computing using raspberry pi clusters for IoT servers on apache Hadoop. Journal of Physics: Conference Series.

Prototicad3D. (2022). Recuperado 15 de diciembre de 2022, de SIMULACION COMPUTACIONAL website:

<https://www.prototicad3d.com/2022/03/simulacion-computacional-una.html>

Salazar, J., Rocha, Q., Botello, S. (2012). Diseño e implementación de un clúster de cómputo de alto rendimiento. Acta Universitaria, 21.

Sale, E., y Rodríguez, S. (2019). Implementación y Operación de un Clúster HPC utilizando Laboratorios de Computadoras en Horarios de Inactividad. Revista Tecnología Y Ciencia, (27), 112–126.

Sebastián, A., & Fernández, J. (2014). Computación paralela y clústeres de cálculo (Parallel computing and computer clusters) (pp. 157-182).

ANEXOS

Anexo 1. Matriz de consistencia

FORMULACIÓN DEL PROBLEMA	OBJETIVOS	HIPÓTESIS	METODOLOGÍA DE LA INVESTIGACIÓN
¿En qué medida un Clúster de servidores influye en el requerimiento de alto procesamiento matemático?	Determinar en qué medida un Clúster de servidores influye en el requerimiento de alto procesamiento matemático	El uso de un Clúster de servidores influye significativamente en el requerimiento de alto procesamiento matemático.	<p>TIPO DE INVESTIGACIÓN Explicativo.</p> <p>NIVEL DE INVESTIGACIÓN Descriptiva.</p> <p>POBLACIÓN Fue un Clúster de cuatro servidores, el cual se construyó y donde se instaló y programó para las pruebas de cálculo matemático.</p> <p>MUESTRA Por ser una unidad de investigación, la muestra será igual a la población.</p>
<p>a) ¿Es posible determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático?</p> <p>b) ¿En qué medida el rendimiento computacional y la eficiencia del consumo de energía varía en un clúster de servidores para alto procesamiento matemático?</p>	<p>a) Determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático.</p> <p>b) Determinar en qué medida el rendimiento computacional y la eficiencia del consumo de energía varía en un clúster de servidores para alto procesamiento matemático</p>	<p>a) Es posible determinar el análisis y diseño arquitectónico de un clúster de servidores para alto procesamiento matemático.</p> <p>b) El rendimiento computacional y la eficiencia del consumo de energía afecta significativamente un clúster de servidores para alto procesamiento matemático</p>	

Anexo 2. Instrumento estadístico

UNIVERSIDAD NACIONAL DE UCAYALI

FACULTAD DE INGENIERÍA DE SISTEMAS E INGENIERÍA CIVIL

FORMULARIO DE TOMA DE DATOS

01 NUCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.28	5476.16	5320.62	2363.69
0.26	5457.13	5266.91	2406.25
0.26	5458.2	5285.68	2385.47
0.26	5472.25	5316.27	2374.15
0.27	5462.35	5318.94	2396.78
0.28	5461.14	5279.18	2389.64
0.28	5459.79	5269.29	2401.93
0.23	5468.16	5325.17	2378.09
0.25	5453.32	2330.23	2396.3
0.23	5455	5350.6	2400.92

02 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.11	1373.6	1291.84	585.72
0.11	1418.83	1341.65	611.62
0.11	1426.18	1335.57	621.33
0.11	1409.16	1326.42	610.25
0.11	1394.37	1329.33	619.38
0.11	1415.9	1340.19	605.29
0.11	1390.32	1298.51	604.26
0.11	1379.48	1338.3	609.42
0.11	1420.52	1331.1	609.5
0.11	1424.2	1333.23	608.79

03 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.07	626.65	627.05	285.55
0.11	631.84	612.6	284.26
0.08	644.44	619.87	282.29
0.08	629.52	621.6	285.65
0.1	638.16	614.25	283.19
0.11	635.79	619.63	282.43
0.08	631.58	625.59	285.29
0.07	636.47	613.2	281.61
0.07	641.34	618.76	284.23
0.07	637.14	620.64	280.93

04 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.12	364.73	349.97	149.04
0.12	353.56	348.2	156.88
0.06	367.35	352.59	156.37
0.12	361.31	363.84	159.35
0.12	356.57	364.97	163.06
0.12	362.11	355.39	156.12
0.06	360.74	361.62	164.4
0.06	367.54	348.44	163.34
0.12	373.94	355.88	154.68
0.12	363.04	363.39	162.19

05 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.05	219.39	208.67	90.62
0.05	220.87	216.01	96.71
0.06	220.27	2015.89	87.38
0.09	218.46	217.51	94.85
0.05	219.5	217.62	92.41
0.06	221.15	211.13	94.04
0.04	217.84	210.69	93.08
0.09	221.54	209.88	92.47
0.06	214.68	214.22	94.58
0.05	215.81	208.94	94.14

06 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.04	145.44	142.66	61.62
0.05	156.59	143.33	64.57
0.08	146.8	141.16	59.31
0.04	145.07	140.78	58.93
0.04	149.81	140.29	63.12
0.04	145.82	140.11	59.27
0.08	148.62	141.9	61.47
0.08	146.68	145.13	59.26
0.08	145.69	140.9	60.06
0.04	150.48	141.25	59.27

07 NUCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.03	105.81	102.08	43.52
0.03	105.79	103.43	43.92
0.07	105.38	102.59	44.42
0.03	105.83	103.57	44.55
0.07	105.76	102.46	43.49
0.03	106.41	101.98	43.67
0.03	104.6	103.7	43.31
0.04	105.02	106.6	45.31
0.03	105.38	104.13	44.53
0.05	105.53	104.74	43.11

08 NUCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.06	81.91	78.26	32.87
0.03	80.11	78.52	32.8
0.03	80.68	78.34	32.6
0.03	80.87	78.79	32.88
0.06	81.56	80.52	34.32
0.03	80.16	78.86	32.97
0.03	80.29	78.6	33.38
0.06	83.2	78.6	32.92
0.06	81.01	77.97	32.61
0.03	80.65	78.57	33

09 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.03	63.79	61.78	26.77
0.03	64.91	62.02	26.07
0.05	63.54	61.61	25.9
0.03	63.28	62.17	26.45
0.03	63.32	61.95	26.12
0.05	65.77	61.8	25.71
0.05	63.97	62.02	26.38
0.03	64.68	61.43	25.77
0.05	63.43	61.75	26
0.03	63.29	61.75	25.8

10 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.04	53.52	50.04	22.51
0.02	51.69	50	20.96
0.04	51.35	50.39	21.35
0.04	51.22	49.78	21.01
0.04	51.34	50.13	21.08
0.02	51.28	49.97	21.38
0.02	51.58	49.81	20.81
0.02	51.57	50.75	21.54
0.02	51.12	49.73	20.95
0.02	52.28	50.21	21.34

11 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.04	42.36	41.19	17.21
0.04	42.42	41.09	17.2
0.04	42.35	41.26	17.42
0.02	42.37	41.36	17.31
0.02	42.29	41.07	17.47
0.03	42.22	41.18	17.21
0.02	42.34	41.32	18.01
0.02	42.54	41.18	17.21
0.02	42.28	41.17	17.28
0.05	42.37	42.33	19.31

12 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.03	36.37	35.57	15.5
0.02	35.84	34.56	14.68
0.03	35.99	34.53	14.51
0.02	35.53	34.65	14.57
0.03	35.42	34.74	14.46
0.03	35.53	34.68	14.55
0.02	40.74	34.89	14.69
0.03	35.4	34.48	14.51
0.02	38.1	34.72	14.46
0.02	35.47	34.51	14.75

13 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.03	30.26	29.39	12.44
0.02	30.25	29.44	12.33
0.02	30.28	29.6	12.32
0.03	30.25	29.48	12.36
0.02	30.23	29.51	12.88
0.03	30.37	29.5	13.17
0.03	30.27	29.4	12.45
0.02	30.27	29.38	12.31
0.02	30.26	29.43	12.31
0.03	30.15	30.28	12.41

14 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.02	25.96	25.25	10.67
0.02	26.09	25.45	10.76
0.02	26.04	25.54	10.65
0.02	26.05	25.33	10.75
0.02	26.13	25.76	10.77
0.02	25.93	25.31	10.64
0.02	26.05	25.98	10.55
0.02	25.97	25.19	10.61
0.02	26.09	25.52	10.62
0.02	26.09	25.49	10.63

15 NÚCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.02	22.61	22.41	9.23
0.03	23.06	22.09	9.19
0.02	22.83	22.48	10.16
0.02	22.61	21.97	9.56
0.02	22.61	22.05	9.6
0.02	22.74	22.42	9.69
0.03	22.69	22.08	9.2
0.02	22.72	22.1	9.46
0.02	22.57	22.02	9.26
0.02	22.84	23.78	9.91

16 NUCLEOS (300000 NÚMEROS)			
GENERAR	BURBUJA	INSERCIÓN	SELECCIÓN
0.01	19.94	19.88	8.09
0.03	19.97	19.33	8.06
0.01	20.01	19.33	8.15
0.01	19.94	19.35	8.05
0.01	19.84	19.3	8.18
0.01	20.01	19.34	8.26
0.01	20.03	19.27	8.07
0.01	19.83	19.27	8.17
0.03	19.9	19.43	8.08
0.01	19.93	18.37	8.11

Anexo 3. Equipo utilizado



Anexo 4. Código de ejecución del programa

```
from mpi4py import MPI

import numpy as np

import time

import sys

import random

comm = MPI.COMM_WORLD

rank = comm.Get_rank()

size = comm.Get_size()

def calculo(numero):

pi=0.0

golpes=0

for i in range(numero):

x=random.random()
```

```
y=random.random()

if((x*x)+(y*y)<1):

golpes+=1

mypi=1 * np.arange(1,dtype=np.float64)

mypi[0]=(golpes*1.0)/numero

total=np.empty(mypi.shape, dtype=mypi.dtype)

comm.Allreduce(mypi,total,op=MPI.SUM)

pi=(4.0/size) * total[0]

return pi

if __name__ == "__main__":

timeStart=MPI.Wtime()

numero=int(1000000000/size)

pi=calculo(numero)

timeEnd=MPI.Wtime()

if rank==0:

print("valor computacional de pi en:", size,"procesos es: ",pi, " demora en
",(timeEnd-timeStar
```