



# TESINA DE LICENCIATURA

**Título:** Cloud Computing en aplicaciones Científicas. Arquitectura, Configuración y Análisis experimental de Costo/Performance.

**Autores:** A.P.U. José Enrique Pettoruti – A.P.U. Ismael Pablo Rodriguez

**Director:** Ing. Armando E. De Giusti

**Codirector:** Lic. Franco Chichizola

**Carrera:** Licenciatura en Informática (P03) – Licenciatura en Sistemas (P03)

## Resumen

El objetivo principal de esta Tesina ha sido desplegar un Cloud Privado con el propósito de analizar e investigar la arquitectura Cloud Computing para ambientes de cómputo científico, teniendo en cuenta el auge de esta arquitectura. Resulta de interés estudiar la evolución de las tecnologías de cómputo Cluster y Grid hacia Cloud. También interesa realizar el despliegue un Cloud Privado utilizando un gestor Open Source y llevar a cabo la ejecución de algunas aplicaciones paralelas de cómputo científico que utilizan pasaje de mensajes MPI. Como casos de prueba se han seleccionado las aplicaciones que resuelven el problema de las N-Reinas y los NAS Parallel Benchmarks. Cabe destacar que las aplicaciones antes mencionadas poseen diversas características con respecto al tiempo de procesamiento y de comunicación. Finalmente, analizar los tiempos de ejecución y calcular el overhead que introduce la arquitectura Cloud.

## Palabras Claves

Cloud, Cloud Computing, IaaS, PaaS, SaaS, Cloud Privado, Cloud Público, Virtualización, Hypervisor, KVM, Xen, Aplicaciones Paralelas, Algoritmos Científicos, Amazon EC2, Eucalyptus, Cluster, MPI, OpenMPI, N-Reinas, NAS Parallel Benchmarks, NPB, Euca2ools, UEC.

## Trabajos Realizados

Se ha investigado acerca de las tecnologías de virtualización y la arquitectura Cloud Computing. Luego, se ha desplegado un Cloud Privado utilizando el gestor Open Source Eucalyptus, con el propósito de ejecutar aplicaciones paralelas de cómputo científico. También, se ha creado una imagen del sistema operativo del nodo de un Cluster Virtual, donde se han ejecutado las aplicaciones N-Reinas y NAS Parallel Benchmarks. Se ha realizado un análisis de los tiempos de ejecución y calculado el overhead introducido por la arquitectura Cloud.

## Conclusiones

Finalizado el trabajo experimental, se puede concluir que Cloud se presenta como una alternativa viable para la ejecución de algunas clases de aplicaciones paralelas que se caracterizan por tener mayor tiempo de procesamiento que de comunicación. El overhead que introduce la arquitectura promedia el 3% sobre el tiempo de ejecución de este tipo de aplicaciones. En aplicaciones cuyo tiempo de comunicación es significativo sobre el tiempo de procesamiento, el overhead es excesivo.

## Trabajos Futuros

Analizar e investigar el despliegue de un Cloud Privado con Eucalyptus utilizando Xen o VMware, con el objetivo de comparar el rendimiento.  
Analizar e investigar esquemas de planificación predictiva en Cloud, en base a las características de las aplicaciones y el consumo de la arquitectura.  
Analizar e investigar la reducción del consumo energético en Cloud.  
Investigar la degradación de las comunicaciones en la red virtualizada y sus técnicas de virtualización.  
Estudiar la incidencia del tamaño del mensaje en algoritmos que se ejecutan sobre el Cloud y la factibilidad de optimización.

Tesina de Grado

**Cloud Computing en aplicaciones  
Científicas. Arquitectura, Configuración y  
Análisis experimental de Costo/Performance.**

Autores:

José Enrique Pettoruti

Ismael Pablo Rodriguez

Director: Ing. Armando E. De Giusti

Co-Director: Lic. Franco Chichizola



# Tabla de contenido

---

Índice de Figuras	vii
Índice de Tablas	ix
Agradecimientos	xi
Objetivos	xiii
<u>Capítulo 1 Evolución de las Tecnologías de Cómputo</u>	<u>1</u>
<u>Capítulo 2 Cloud Computing</u>	<u>7</u>
2.1 ¿Qué es Cloud Computing?	7
2.2 Clasificación en modelos de servicio	7
2.2.1 Infraestructura como Servicio (IaaS)	8
2.2.2 Plataforma como Servicio (PaaS)	8
2.2.3 Software como Servicio (SaaS)	8
2.3 Modelos de despliegue Cloud	9
2.3.1 Cloud público	9
2.3.2 Cloud privado	10
2.3.3 Cloud híbrido	10
2.4 Características y beneficios de un Cloud	11
2.5 Virtualización	11
2.5.1 Virtual Machine Monitor	12
2.5.2 Técnicas de Virtualización de CPU	13
2.6 Software de Virtualización	16
2.6.1 XEN	16
2.6.2 KVM (Kernel based Virtual Machine)	18
<u>Capítulo 3 Amazon Elastic Compute Cloud (EC2)</u>	<u>19</u>
3.1 Componentes de la infraestructura	19
3.1.1 Amazon Machine Images e Instancias	19
3.1.2 Tipos de instancias disponibles en EC2	20
3.1.3 Regiones y Zonas de Disponibilidad	22
3.1.4 Almacenamiento	22
3.1.5 Redes y seguridad	24
3.2 Funcionalidad	25
3.3 Beneficios	25

3.4	Costos	27
3.5	Acuerdo de nivel de servicio (SLA)	28

## Capítulo 4 Eucalyptus 29

4.1	Gestor de Cloud: Eucalyptus	29
4.2	Componentes de Eucalyptus	29
4.3	Seguridad	33
4.4	Redes	33
4.4.1	Características	34
4.4.2	Requerimientos	34
4.4.3	Modos de Red	35
4.5	Herramientas de administración	36
4.5.1	Euca2ools	36
4.5.2	HybridFox	37

## Capítulo 5 Despliegue de un Cloud Privado 39

5.1	Hardware	39
5.2	Software	40
5.3	Instalación	40
5.4	Publicación de una imagen	43
5.5	Configuración de red	44
5.6	Despliegue de un Cluster en el Cloud	45
5.6.1	Creación de una imagen para un nodo del Cluster	45
5.6.2	Modificación de los tipos de instancias	47
5.6.3	Instanciación del Cluster	48

## Capítulo 6 Ejecución de algoritmos en Cluster y Cloud 49

6.1	Problema N-Reinas	49
6.2	NAS Parallel Benchmarks	51
6.3	Escenarios de ejecución	53
6.3.1	Cluster Dedicado	53
6.3.2	Cluster Virtual	53
6.4	Ejecución de aplicaciones	54
6.5	Análisis de Resultados y Cálculo de Overhead	55
6.5.1	Overhead en N-Reinas con Tablero 18x18	55
6.5.2	Overhead en N-Reinas con Tablero 19x19	56
6.5.3	Overhead en N-Reinas con Tablero 20x20	57
6.5.4	Overhead en N-Reinas con Tablero 21x21	58
6.5.5	Resumen del Overhead en N-Reinas	59
6.5.6	Overhead en NPB ep.C	60

6.5.7	Overhead en NPB ft.C	61
6.5.8	Overhead en NPB lu.C	62
6.5.9	Resumen del Overhead en NPB	63
6.6	Pruebas de comunicación	63
6.6.1	Pruebas de red entre 2 nodos A	64
6.6.2	Pruebas de red entre 2 nodos B	66
6.6.3	Pruebas de red entre 2 nodos C	68
6.6.4	Resumen de la degradación de la red virtualizada	70
<b>Capítulo 7 Conclusiones</b>		<b>73</b>
<b>Capítulo 8 Líneas de trabajo futuro</b>		<b>75</b>
<b>Referencias Bibliográficas</b>		<b>77</b>
<b>Anexo A SLA de Amaxon EC2</b>		<b>81</b>
<b>Anexo B Tabla de Comandos Euca2ools en español</b>		<b>85</b>
<b>Anexo C Tiempos de Ejecución de N-Reinas</b>		<b>91</b>
C.1	Tiempos de ejecución en 4 Nodos	92
C.2	Tiempos de ejecución en 2 Nodos	93
C.2.1	Tiempos de ejecución en 2 Nodos A	93
C.2.2	Tiempos de ejecución en 2 Nodos B	94
C.2.3	Tiempos de ejecución en 2 Nodos C	95
C.3	Tiempos de ejecución en 1 Nodo	96
C.3.1	Tiempos de ejecución en 1 Nodo A	96
C.3.2	Tiempos de ejecución en 1 Nodo B	97
<b>Anexo D Tiempos de Ejecución de NPB</b>		<b>99</b>
D.1	Tiempos de ejecución en 4 Nodos	99
D.2	Tiempos de ejecución en 2 Nodos	100
D.2.1	Tiempos de ejecución en 2 Nodos A	100
D.2.2	Tiempos de ejecución en 2 Nodos B	100
D.2.3	Tiempos de ejecución en 2 Nodos C	101
D.3	Tiempos de ejecución en 1 Nodo	101
D.3.1	Tiempos de ejecución en 1 Nodo A	101
D.3.2	Tiempos de ejecución en 1 Nodo B	102
<b>Anexo E Pruebas de Comunicación</b>		<b>103</b>

E.1	Pruebas de comunicación entre 2 Nodos A	103
E.2	Pruebas de comunicación entre 2 Nodos B	104
E.3	Pruebas de comunicación entre 2 Nodos C	105

# Índice de Figuras

---

Figura 1.1: Mainframe con terminales interactivos	1
Figura 1.2: Arquitectura cliente/servidor	2
Figura 1.3: Cluster de Servidores	3
Figura 1.4: Arquitectura Grid	5
Figura 2.1: Modelos de servicio Cloud	8
Figura 2.2: Proveedores de servicios Cloud	9
Figura 2.3: Sistema virtualizado	12
Figura 2.4: Anillos de privilegios de la arquitectura x86	13
Figura 2.5: Virtualización Completa	14
Figura 2.6: Paravirtualización	15
Figura 2.7: Virtualización asistida por hardware	16
Figura 2.8: Entorno de virtualización Xen	16
Figura 2.9: Componentes de Xen	17
Figura 2.10: Arquitectura de KVM	18
Figura 3.1: AMIs e Instancias de EC2	19
Figura 3.2: Tipos de instancias de EC2	20
Figura 3.3: Regiones y zonas de disponibilidad de EC2	22
Figura 3.4: Volumen EBS asociado a una instancia de EC2	23
Figura 3.5: Creación de snapshot de un volumen EBS	24
Figura 3.6: Snapshot como punto de partida para un nuevo volumen	24
Figura 3.7: Cambio de instancia para un volumen EBS	24
Figura 4.1: Componentes de Eucalyptus	30
Figura 4.2: Interface de HybridFox	37
Figura 5.1: Topología física del hardware utilizado	40
Figura 5.2: Despliegue del Cloud Eucalyptus	41
Figura 6.1: Movimientos posibles de una reina en el tablero	49
Figura 6.2: Ejemplo de tablero válido (a) e inválido (b)	50
Figura 6.3: Fórmula para el cálculo del Tiempo Promedio de ejecución	54
Figura 6.4: Fórmula para el cálculo del Tiempo Promedio de ejecución	55
Figura 6.5: Gráfico del Overhead para N-Reinas Tablero 18x18	55
Figura 6.6: Gráfico del Overhead para N-Reinas Tablero 19x19	56
Figura 6.7: Gráfico del Overhead para N-Reinas Tablero 20x20	57
Figura 6.8: Gráfico del Overhead para N-Reinas Tablero 21x21	58
Figura 6.9: Gráfico del Overhead en N-Reinas	59
Figura 6.10: Gráfico del Overhead en NPB ep.C	60
Figura 6.11: Gráfico del Overhead en NPB ft.C	61
Figura 6.12: Gráfico del Overhead en NPB lu.C	62
Figura 6.13: Gráfico del overhead en NPB	63

Figura 6.14: Gráfico de las pruebas de red entre 2 nodos A	65
Figura 6.15: Gráfico de las pruebas de red entre 2 nodos B	67
Figura 6.16: Gráfico de las pruebas de red entre 2 nodos C	69
Figura 6.17: Gráfico resumen de la degradación de la red virtualizada	71

# Índice de Tablas

---

Tabla 1.1: Comparación de escalabilidad entre Mainframes y Clusters	4
Tabla 3.1: Tipos de instancias en Amazon EC2	21
Tabla 3.2: Costos de instancias de EC2 (Octubre 2011)	27
Tabla 4.1: Modos de Red de Eucalyptus	36
Tabla 5.1: Especificación de los servidores utilizados	39
Tabla 5.2: Especificación de los dispositivos de red utilizados	39
Tabla 5.3: Configuración de la arquitectura de Eucalyptus	40
Tabla 5.4: Zona de disponibilidad del Cloud	42
Tabla 5.5: Tipos de instancias por defecto de Eucalyptus	42
Tabla 5.6: Estado de Eucalyptus utilizando euca_conf	43
Tabla 5.7: Descarga de una imagen desde el repositorio de Ubuntu	43
Tabla 5.8: Publicación de una imagen en el Cloud con uec-publish-tarball	44
Tabla 5.9: Pruebas de VLAN Clean	45
Tabla 5.10: Configuración del CC en modo MANAGED en eucalyptus.local.conf	45
Tabla 5.11: Configuración de los NCs en modo MANAGED en eucalyptus.conf	45
Tabla 5.12: Creación y publicación de una imagen personalizada	47
Tabla 5.13: Tipos de instancias para el Cluster	47
Tabla 5.14: Zona de disponibilidad del Cloud con los nuevos tipos de instancias	48
Tabla 6.1: Cantidad de soluciones de N-Reinas para $1 < N < 26$	50
Tabla 6.2: Aplicaciones NAS Benchmarks[WAL08]	52
Tabla 6.3: Tamaños de problema de NPB	53
Tabla 6.4: Configuraciones de los escenarios de ejecución	54
Tabla 6.5: Overhead en N-Reinas con Tablero 18x18	55
Tabla 6.6: Overhead en N-Reinas con Tablero 19x19	56
Tabla 6.7: Overhead en N-Reinas con Tablero 20x20	57
Tabla 6.8: Overhead en N-Reinas con Tablero 21x21	58
Tabla 6.9: Resumen del overhead en N-Reinas	59
Tabla 6.10: Overhead en NPB ep.C	60
Tabla 6.11: Overhead en NPB ft.C	61
Tabla 6.12: Overhead en NPB lu.C	62
Tabla 6.13: Resumen del overhead en NPB	63
Tabla 6.14: Pruebas de red entre 2 nodos A	64
Tabla 6.15: Pruebas de red entre 2 nodos B	66
Tabla 6.16: Pruebas de red entre 2 nodos C	68
Tabla 6.17: Resumen de la degradación de la red virtualizada	70
Tabla A.1: SLA de Amazon EC2 (Octubre 2011)	84
Tabla B.1: Comandos de Euca2ools	90
Tabla C.1: Tiempos de ejecución de N-Reinas en 4 nodos físicos	92

Tabla C.2: Tiempos de ejecución de N-Reinas en 4 nodos virtuales	92
Tabla C.3: Tiempos de ejecución de N-Reinas en 2 nodos físicas (Dell)	93
Tabla C.4: Tiempos de ejecución de N-Reinas en 2 nodos virtuales (c1.xlarge)	93
Tabla C.5: Tiempos de ejecución de N-Reinas en 2 nodos físicos (Blade)	94
Tabla C.6: Tiempos de ejecución de N-Reinas en 2 nodos virtuales (m1.xlarge)	94
Tabla C.7: Tiempos de ejecución de N-Reinas en 2 nodos físicos (Dell + Blade)	95
Tabla C.8: Tiempos de ejecución de N-Reinas en 2 nodos virtuales (c1.xl + m1.xl)	95
Tabla C.9: Tiempos de ejecución de N-Reinas en 1 nodo físico (Dell)	96
Tabla C.10: Tiempos de ejecución de N-Reinas en 1 nodo virtual (c1.xlarge)	96
Tabla C.11: Tiempos de ejecución de N-Reinas en 1 nodo físico (Blade)	97
Tabla C.12: Tiempos de ejecución de N-Reinas en 1 nodo virtual (c1.xlarge)	97
Tabla D.1: Tiempos de ejecución de NAS en 4 nodos físicos	99
Tabla D.2: Tiempos de ejecución de NAS en 4 nodos virtuales	99
Tabla D.3: Tiempos de ejecución de NAS en 2 nodos físicos (Dell)	100
Tabla D.4: Tiempos de ejecución de NAS en 2 nodos virtuales (c1.xlarge)	100
Tabla D.5: Tiempos de ejecución de NAS en 2 nodos físicos (Blade)	100
Tabla D.6: Tiempos de ejecución de NAS en 2 nodos virtuales (m1.xlarge)	100
Tabla D.7: Tiempos de ejecución de NAS en 2 nodos físicos (Dell + Blade)	101
Tabla D.8: Tiempos de ejecución de NAS en 2 nodos virtuales (c1.xl + m1.xl)	101
Tabla D.9: Tiempos de ejecución de NAS en 1 nodo físico (Dell)	101
Tabla D.10: Tiempos de ejecución de NAS en 1 nodo virtual (c1.xlarge)	101
Tabla D.11: Tiempos de ejecución de NAS en 1 nodo físico (Blade)	102
Tabla D.12: Tiempos de ejecución de NAS en 1 nodo virtual (m1.xlarge)	102
Tabla E.1: Resultados de las pruebas de comunicación entre 2 nodos A	103
Tabla E.2: Resultados de las pruebas de comunicación entre 2 nodos B	104
Tabla E.3: Resultados de las pruebas de comunicación entre 2 nodos C	105

# Agradecimientos

---

A mis padres, por introducirme en la Informática en la época de las XT286 y por haberme otorgado la posibilidad de estudiar esta carrera y ofrecerme un apoyo invaluable en toda su duración.

A mis padres y hermanos, por su paciencia y apoyo incondicional en todo momento.

A mi abuelo, por su fanatismo y fe absoluta en mí y en mi carrera.

A Horacio Martinez del Pezzo, por sus consejos al inicio de mis estudios.

A Sabrina, por su paciencia, cariño, compañía y colaboración en todo momento.

A Ismael, quien sin su compañía, el desarrollo de esta Tesina hubiera sido más difícil.

José E.

A mi esposa Cristina, por su apoyo incondicional, paciencia y compañía para alcanzar esta meta.

A mi hijo Ignacio, por renovar mis fuerzas con sus sonrisas, miradas y afectos al culminar cada día.

A mis padres, por sus enseñanzas de como conducirme en esta vida.

A mis familiares, hermanos y suegros que me han apoyado y alentado en el transcurso de mi carrera.

A José, que ha sido un verdadero compañero y apoyo en este arduo trabajo.

Ismael

A nuestro director, Armando De Giusti, por el consejo y guía en nuestras carreras.

A nuestro codirector, Franco Chichizola, por el asesoramiento, dedicación y colaboración en el desarrollo de esta tesina.

A todos los miembros del III-LIDI que han aportado su granito de arena en nuestro trabajo y al instituto, por facilitarnos el acceso a los recursos necesarios para realizar todas las pruebas efectuadas.

A Adrián, por su colaboración, especialmente en las pruebas de comunicaciones.

A Fernando, con quién hemos debatido técnicamente y filosóficamente en temas de la Tesina.

José e Ismael



# Objetivos

---

Analizar las ventajas y desventajas de la utilización de Cloud Computing para ambientes de investigación en cómputo de altas prestaciones.

Estudiar la evolución de las arquitecturas en cómputo de altas prestaciones, incluyendo Cluster, Grid y Cloud.

Efectuar un análisis inicial de la relación costo/performance en la utilización de Cloud Computing en ambientes de investigación.



# Capítulo 1

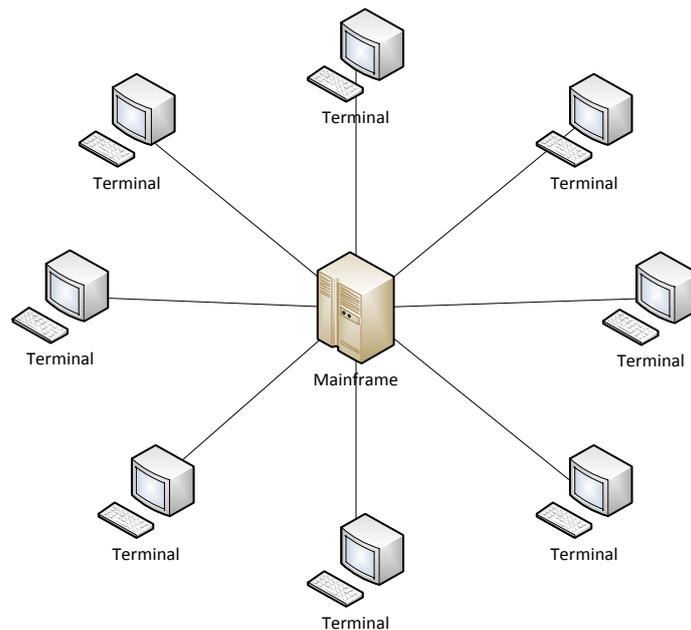
## Evolución de las Tecnologías de Cómputo

---

En este capítulo se presenta la evolución en el tiempo de las arquitecturas paralelas, iniciando con los grandes mainframes hasta llegar a la actualidad con Cloud Computing.

Desde la década de los 60, los mainframes han estado presentes en las empresas públicas y privadas, como en los organismos de gobierno, educación y ciencia. Los primeros mainframes, ejecutaban tareas o programas por medio de procesamiento por lotes: se introducían tarjetas perforadas, cintas de papel o cintas magnéticas que contenían el programa o la tarea a ejecutar; este sistema de procesamiento permitía la ejecución de un programa sin el control o supervisión del usuario. Esto fue evolucionando, al transcurrir las décadas, adicionando terminales interactivos permitiendo, en las ejecuciones de los programas, que el usuario introduzca datos o realice determinadas elecciones. También, por medio de la tecnología de virtualización se logró que soporten múltiples Sistemas Operativos (SO). [CAC10]

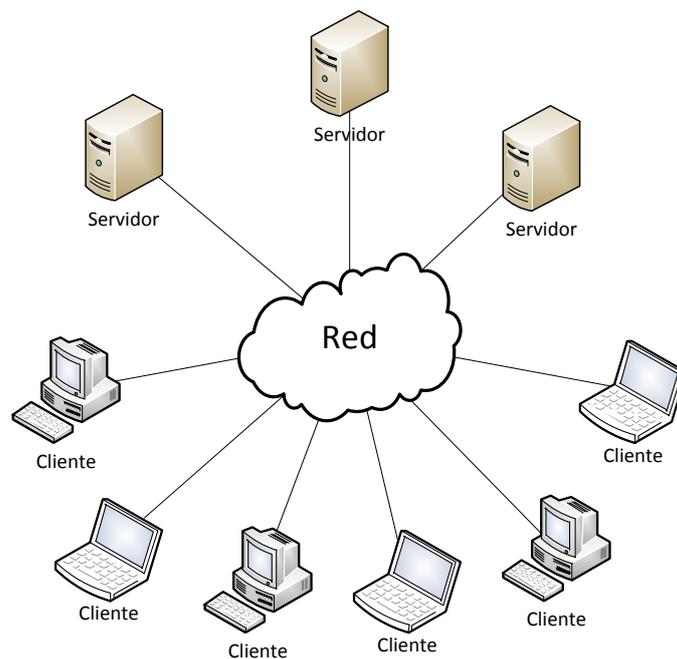
Los mainframes son computadoras centralizadas, diseñadas con el fin de controlar y procesar grandes cantidades de datos de forma rápida con una alta fiabilidad, disponibilidad y utilidad. Los mainframes escalan en forma vertical mediante la adición de más recursos de almacenamiento o de conectividad.



*Figura 1.1: Mainframe con terminales interactivos*

A fines de los años 80 y principio de los años 90, se comenzaron a implementar soluciones informáticas distribuidas/descentralizadas, en particular la arquitectura cliente/servidor, como una alternativa a los mainframes. En esta arquitectura, cada equipo (cliente/servidor) se compone de una unidad central de procesamiento con su propia memoria y se comunica por una red de datos. Un servidor proporciona un conjunto de funciones a uno o más clientes que invocan las solicitudes de dichas funciones. Los sistemas distribuidos presentan algunas ventajas con respecto a los mainframes:

- Reducción de costos: debido al alto costo que tenía un mainframe.
- Flexibilidad: los servidores, normalmente alojados en computadoras dedicadas se pueden desplegar, configurar y testear por separado del resto del sistema y conectarlos cuando estén preparados para su paso a producción.
- Reducción de latencia: los servidores pueden estar distribuidos entre diferentes datacenters para estar más cerca de los usuarios finales.
- Mayor escalabilidad: los mainframes presentan limitaciones a la hora de añadir nuevo hardware como recursos del sistema. En cambio, los sistemas distribuidos permiten aumentar la capacidad de todo el sistema añadiendo más servidores.



**Figura 1.2:** Arquitectura cliente/servidor

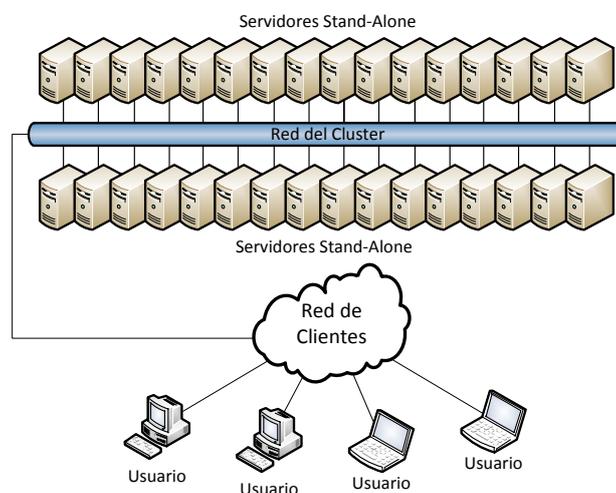
Los sistemas distribuidos pueden escalar en forma vertical (añadiendo más recursos a un nodo) y/u horizontal (añadiendo nuevos nodos). En el segundo caso, puede requerir el rediseño de la aplicación para incorporar paralelización y balance de carga, entre otros casos. Sin embargo, durante los años 90, los sistemas distribuidos presentaron algunas desventajas en comparación con los mainframes:

- Baja utilización de los servidores: por lo general los servidores están dedicados a una única aplicación, debido a los requerimientos de SO, librerías, componentes y otros.
- Mayores costos operativos: los sistemas distribuidos poseen mayor complejidad, por lo tanto requieren un mayor mantenimiento.
- Menor eficiencia energética: la utilización de servidores dedicados aumenta el consumo de energía tanto en hardware como en los sistemas de refrigeración.
- Mayor espacio físico requerido: aún agrupados en racks o Blades, usualmente requieren más espacio que los mainframes.
- Potencialmente, menor rendimiento de I/O: los sistemas distribuidos poseen una mayor latencia en el acceso a los datos.

Estas desventajas, a principios de la década del 2000, hicieron volver a utilizar las arquitecturas mainframe, lo cual permitió consolidar los recursos de hardware y al mismo tiempo ser capaces de ejecutar múltiples instancias de SO, dada la introducción de las tecnologías de virtualización y las particiones lógicas del sistema. Los mainframes evolucionaron, permitiendo reconfigurar dinámicamente los recursos asignados a una máquina virtual o partición lógica (procesadores, memoria, conexiones de dispositivos y otros).

En los sistemas distribuidos, la evolución de las tecnologías de virtualización y automatización dio origen a la arquitectura Cluster.

Un Cluster es una clase de sistema de procesamiento paralelo compuesto por un conjunto de computadoras stand-alone, interconectadas a través de algún tipo de red, trabajando en forma conjunta como un recurso de cómputo único e integrado. Stand-alone se refiere a una computadora independiente, con hardware y SO propios. [ROD07]



*Figura 1.3: Cluster de Servidores*

Los Clusters permiten mejorar el rendimiento y/o la disponibilidad, a un menor costo, que el de un único equipo de velocidad o disponibilidad similar.

Sin embargo, disponer de una gran cantidad de recursos físicos en un Cluster, a menudo resulta ser insuficiente para grandes demandas de cómputo requeridas por algunas aplicaciones. Se requirió una evolución del sistema para absorber estas demandas. La necesidad de unir recursos distribuidos y administrados independientemente dio lugar al concepto de Organizaciones Virtuales (VOs).

Las VOs son dominios administrativos independientes que establecen los medios para el uso de recursos no utilizados, pertenecientes a otras organizaciones de colaboración, como si estuvieran ubicados localmente.

Por otra parte, desde el punto de vista de la arquitectura de computadoras, la evolución de las mismas no distingue claramente entre servidores dedicados y mainframes, ya que los mainframes podrían considerarse como el modelo más alto de una familia de servidores. Por lo tanto, la cuestión clave es determinar si se debe escalar horizontalmente con servidores de bajo costo (Clusters o servidores pequeños) o verticalmente con grandes servidores de memoria compartida. Como es habitual, no hay ninguna solución milagrosa, y la decisión depende de las aplicaciones a ejecutar.

De todos modos, mediante la introducción de la virtualización, la capa subyacente de hardware físico es transparente a los servidores, los cuales potencialmente pueden escalar verticalmente a la máxima capacidad de las máquinas físicas y horizontalmente a la capacidad máxima del datacenter.

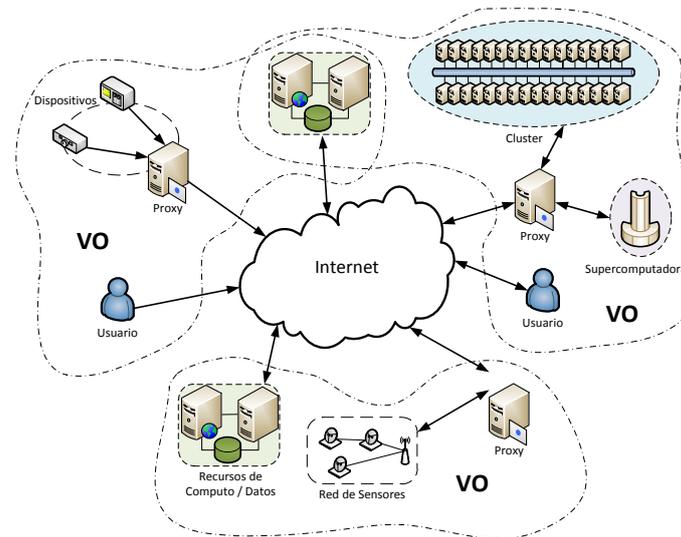
La Tabla 1 resume las soluciones de escalabilidad utilizada en los datacenters.

Escalabilidad	<i>Mainframes</i>	<i>Clusters</i>
<b>Vertical</b>	<ul style="list-style-type: none"> <li>- Incorporar más CPUs, memoria o disco</li> <li>- Actualizar a un modelo superior</li> <li>- Incorporar o mejorar la concurrencia de la aplicación</li> </ul>	<ul style="list-style-type: none"> <li>- Incorporar más CPUs, memoria o disco a los nodos</li> <li>- Incrementar el ancho de banda de la red</li> <li>- Actualizar los nodos a un modelo superior</li> <li>- Redistribuir los componentes de la aplicación</li> <li>- Incorporar o mejorar la concurrencia de la aplicación</li> </ul>
<b>Horizontal</b>	No aplicable	<ul style="list-style-type: none"> <li>- Agregar nuevos nodos</li> <li>- Incorporar o mejorar algoritmos de paralelización</li> </ul>

*Tabla 1.1: Comparación de escalabilidad entre Mainframes y Clusters*

Luego, se dio origen a la tecnología Grid Computing, la cual ha surgido con el objetivo de permitir el uso compartido de recursos entre organizaciones. Grid es también conocido como una Infraestructura Orientada a Servicios, cuyo objetivo es más general, ya que debe albergar diversas aplicaciones, pertenecientes a diferentes campos y tipos de organizaciones (investigación, gobierno o empresarial). De esta forma, se extiende la escala de los sistemas informáticos mediante la incorporación de recursos básicos pertenecientes a diferentes dominios administrativos dentro de una o más VO.

Grid es un tipo de sistema paralelo y distribuido que permite compartir, seleccionar y agregar recursos autónomos distribuidos geográficamente tales como computadoras, software, datos, bases de datos y dispositivos especiales. Esta configuración colaborativa depende de la disponibilidad, capacidad, costo y requerimientos del usuario. También, Grid puede definirse como un entorno virtual de procesamiento de información donde el usuario tiene la ilusión de un único y potente recurso de cómputo que en realidad se encuentra distribuido. [DEG07].



**Figura 1.4:** Arquitectura Grid

Formalmente, Grid se define como un "sistema que coordina recursos que no están sujetos a un control centralizado, utilizando protocolos e interfaces estándares, abiertas y de propósito general para ofrecer calidades de servicio no triviales" [FOS02].

Algunas características de un entorno Grid:

- Los recursos y servicios pueden vincularse o abandonar el Grid en forma dinámica.
- El Grid integra recursos (procesadores, instrumentos, bases de datos, etc) que son heterogéneos, geográficamente distribuidos y en general conectados por una WAN.

- Los recursos pueden ser accedidos bajo demanda por un conjunto de usuarios que conforman una comunidad virtual.
- El Grid se configura utilizando protocolos e infraestructura de propósito general, no necesariamente común a todos sus nodos.
- Los problemas clásicos de los sistemas distribuidos como la tolerancia a fallas, el balance de carga o la calidad de servicio aparecen con un grado mayor de complejidad en Grid.

Algunos autores consideran a Grid como un Cluster de Clusters, lo que resulta una definición algo restrictiva pero útil para cuando se habla de aplicaciones paralelas que evolucionan de Cluster a Grid.

Pueden mencionarse algunas similitudes y diferencias:

- Tanto Clusters como Grids se pueden basar en procesadores heterogéneos. Sin embargo en Grid, esta heterogeneidad se extiende a la red de comunicaciones y al tipo de componentes de cada nodo.
- El middleware necesario para Grid es más complejo que el de Clusters, dado que se requiere una etapa de identificación de recursos físicos y su ubicación. Además, en Grid es necesario monitorear la ejecución de tareas sobre múltiples recursos con usuarios de diferente nivel y con diversos privilegios de acceso a los mismos.
- Asimismo, las herramientas para el desarrollo de aplicaciones requieren un mayor nivel de abstracción en Grid, dada la complejidad y variedad de los múltiples usuarios que pueden utilizar la arquitectura.

Además, una aplicación determinada resulta difícil de escalar en Grid. Como tradicionalmente se concibe, Grid no ofrece a los desarrolladores mecanismos de soporte para escalar sus sistemas de acuerdo con los cambios en la demanda.

En este sentido, Grid no es diferente a los sistemas de cómputo anteriores: un administrador debe detectar sobrecargas en el servicio y escalar manualmente el sistema dependiendo de las métricas relevantes de rendimiento para el servicio en cuestión. Por otro lado, los recursos que se incluyen en una VO particular, pueden ser inferiores a la capacidad necesaria para llevar a cabo la tarea prevista.

Hacia fines del año 2006, las tecnologías Grid y Cluster en conjunto con Virtualización, comenzaron a evolucionar hacia un nuevo paradigma llamado Cloud Computing, el cual se describirá en el Capítulo 2.

# Capítulo 2

## Cloud Computing

---

En este capítulo se realiza una descripción del estado del arte, definiendo el concepto de “Cloud Computing”, su clasificación en capas y los distintos tipos de despliegues de un “Cloud”.

También se introduce el concepto de “Virtualización” y se presentan dos herramientas de software libre en Linux que brindan entornos de virtualización completa.

### 2.1 ¿Qué es Cloud Computing?

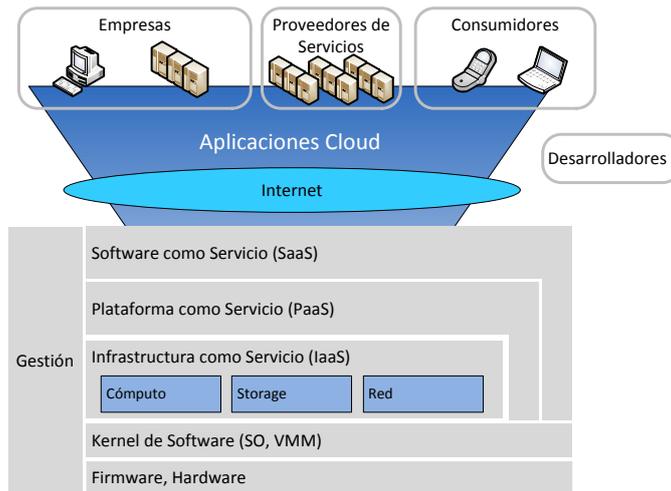
“Cloud Computing” es un paradigma informático de cómputo distribuido, que proporciona grandes conjuntos de recursos virtuales (como ser hardware, plataformas de desarrollo, almacenamiento y/o aplicaciones), fácilmente accesibles y utilizables por medio de una interfaz de administración Web. Estos recursos son proporcionados como servicios (del inglés, “as a service”) y pueden ser dinámicamente reconfigurados para adaptarse a una carga de trabajo variable (escalabilidad), permitiendo una mejor utilización de los mismos y evitando el sobredimensionamiento o sub dimensionamiento (elasticidad). Esta utilización se realiza bajo la demanda de los usuarios, en base a un modelo de autoservicio. [ROD11] [VAQ09]

El término “Cloud” o “Nube” hace referencia a la organización de la infraestructura física, la cual es transparente para el usuario final ya que el mismo desconoce la ubicación y organización real de la misma. [KEN10] [EUC11a]

Cloud Computing tiene su origen en el modelo de negocios conocido como Utility Computing, donde las principales empresas de tecnología de la información (IT), como ser Amazon, IBM, Google y Microsoft, brindan acceso a los recursos bajo una modalidad de pago por uso, garantizando la disponibilidad de los mismos por medio de acuerdos de nivel de servicio (SLAs). [CHE10]

### 2.2 Clasificación en modelos de servicio

Cloud Computing se divide en tres modelos de servicio principales de acuerdo a los recursos provistos: Infraestructura como Servicio (IaaS), Plataforma como Servicio (PaaS) y Software como Servicio (SaaS), como se observa en la Figura 2.1. [VEL09][CHE10][VAQ09]



**Figura 2.1:** Modelos de servicio Cloud [YOU08]

### 2.2.1 Infraestructura como Servicio (IaaS)

Este modelo ofrece, por medio de la tecnología de virtualización, recursos de infraestructura virtual (servidores, equipos, dispositivos de almacenamiento, dispositivos de red, etc.), flexibles y escalables; en donde múltiples usuarios coexisten compartiendo el mismo hardware físico de forma transparente, segura e independiente.

Amazon EC2 y GoGrid son ejemplos de algunos de los proveedores más importantes que ofrecen IaaS.

### 2.2.2 Plataforma como Servicio (PaaS)

Cuando los recursos proporcionados por el Cloud son entornos configurados con todas las herramientas de software necesarias para el desarrollo y/o despliegue de aplicaciones personalizadas, sin que el usuario se ocupe de la instalación y administración de la infraestructura subyacente, entonces se considera un modelo PaaS.

Estas plataformas ofrecen todo lo necesario para soportar los ciclos de vida completos de una aplicación y/o servicio WEB disponible en Internet.

Google App Engine (GAE) es un ejemplo de PaaS que permite el desarrollo y alojamiento de aplicaciones Web en servidores de Google. Los lenguajes de programación disponibles son Python y Java.

También, Windows Azure y Force.com brindan plataformas de desarrollo y/o despliegue en el Cloud.

### 2.2.3 Software como Servicio (SaaS)

Este modelo ofrece aplicaciones de software como recursos, las cuales pueden ser utilizadas sin necesidad de ser instaladas localmente en una computadora. Generalmente

se acceden a través de un navegador Web. El usuario no es responsable de la instalación y mantenimiento de las mismas, pues de esto se encarga el proveedor del servicio.

Google Docs, una de las herramientas ofrecidas por Google Apps, es un ejemplo de SaaS, la cual permite crear y editar documentos, hojas de cálculo, dibujos y presentaciones, por medio de un navegador Web.

Otros ejemplos son Salesforce, Windows Live y MobileMe.



Figura 2.2: Proveedores de servicios Cloud

## 2.3 Modelos de despliegue Cloud

Existen tres modelos de despliegue de un Cloud: Cloud público, Cloud privado y Cloud híbrido. [CHE10] [EUC11a] [NUR09]

### 2.3.1 Cloud público

Un Cloud público es aquel desplegado por un proveedor de IT, que ofrece servicios (IaaS, PaaS y/o SaaS) de acceso público desde Internet. Generalmente, este modelo de despliegue se relaciona con un contexto comercial, donde los usuarios son considerados clientes y pagan por el tiempo de uso de los servicios.

La principal ventaja de un Cloud público, es la posibilidad de acceder inmediatamente a un conjunto de recursos con una mínima inversión.

Las desventajas que presenta un Cloud público son el alto costo acumulado en el tiempo de uso del mismo y las garantías sobre la privacidad y seguridad de la información del usuario. Este último punto merece un análisis y discusión que se encuentra fuera del alcance de esta Tesina.

Dentro de los proveedores con más renombre en el mercado, se pueden citar Amazon Web Services (AWS), Google Apps, Microsoft Azure, Rackspace y Salesforce.com.

### 2.3.2 Cloud privado

Se considera un Cloud privado al que se encuentra desplegado en la intranet (en la red de datos detrás del firewall) de una organización, institución o empresa. El despliegue se efectúa sobre la infraestructura de hardware de la organización y los servicios proporcionados son de características similares al Cloud público, con la diferencia que el acceso está limitado a la red privada y no presenta carácter comercial.

Las principales ventajas de un Cloud privado es la capacidad de brindar mayor eficiencia en el uso de la infraestructura física, como también un alto nivel de seguridad sobre los datos sensibles de la organización. Con el objetivo de lograr mayor eficiencia en la utilización de los recursos del datacenter, éstos se consolidan a través de la virtualización. Por otro lado, dado que el Cloud privado se encuentra desplegado en la Intranet de la organización, se garantiza la seguridad de los datos por medio de las mismas políticas y medidas de seguridad del datacenter.

Un Cloud privado presenta como desventaja la limitación en la escalabilidad. Esta limitación se debe a la capacidad de los recursos físicos del datacenter durante grandes demandas de servicio. En este caso, un Cloud híbrido resuelve esta problemática.

En la actualidad existen diversas herramientas de software que permiten el despliegue de un Cloud privado, de las cuales se destacan Eucalyptus, OpenNebula, y OpenStack, entre otras.

### 2.3.3 Cloud híbrido

Se denomina Cloud híbrido al despliegue de un Cloud privado que utiliza de forma segura los recursos de un Cloud público. Este modelo es de gran utilidad cuando una organización requiere aumentar rápidamente sus recursos privados para satisfacer los picos de demanda de determinados servicios. Generalmente, la organización contrata servicios de un Cloud público para expandir los recursos de su Cloud privado. A esta modalidad, también se la conoce como *Cloudbursting*.

Un Cloud híbrido permite a las organizaciones obtener los beneficios combinados de un Cloud privado y uno público, logrando así el aumento de la eficiencia en el uso de la infraestructura física del datacenter, un mayor nivel de seguridad de los datos y una mayor disponibilidad de recursos para atender grandes demandas de servicio.

## 2.4 Características y beneficios de un Cloud

La utilización de un Cloud en una organización presenta las siguientes características y beneficios: [EUC11a]

### Recursos disponibles bajo demanda en el Cloud.

Al utilizar un Cloud público, las organizaciones no deben afrontar la instalación y el mantenimiento de un datacenter, sólo pagan por los servicios que utilizan bajo demanda.

Por otro lado, los Cloud privados e híbridos, permiten optimizar la infraestructura existente en el datacenter de la organización, por medio de la consolidación y la virtualización. Esto se traduce en un menor consumo de energía y una reducción de la infraestructura, incluyendo los gastos relacionados.

### Interfaces basadas en servicios Web

La utilización de interfaces basadas en servicios Web, permite a los usuarios acceder de manera sencilla a los recursos brindados por el Cloud.

### Escalabilidad y Elasticidad

En un Cloud, los recursos brindados son altamente escalables. Esto significa que una aplicación o un usuario pueden escalar (agregar) o reducir (liberar) dinámicamente sus recursos en respuesta a la variación en la carga de trabajo.

### Aprovisionamiento automático de recursos

En Cloud, la distribución de los recursos se realiza de forma inmediata y automática sin requerir la intervención de personal técnico y/o administrativo. De esta forma, el personal técnico se libera de tareas repetitivas de configuración y atención a demandas, mientras que el usuario accede rápidamente a los recursos.

### Autoservicio

El usuario de un Cloud puede solicitar directamente los recursos que necesita, sirviéndose de los mismos por medio de una interface Web. Esta interface le brinda control directo sobre el despliegue y la configuración de los recursos, evitando así las demoras en el acceso a los mismos.

Además, la combinación de los SLAs con las herramientas de auditoria y monitoreo, aseguran que el usuario recibe los recursos solicitados.

## 2.5 Virtualización

La virtualización es una tecnología que ha impactado significativamente en las infraestructuras de las organizaciones en los últimos años, brindando los beneficios de aprovechar, consolidar y reducir los costos de la infraestructura, como así también simplificar la gestión de los recursos y un mayor control de los mismos. [VMW11]

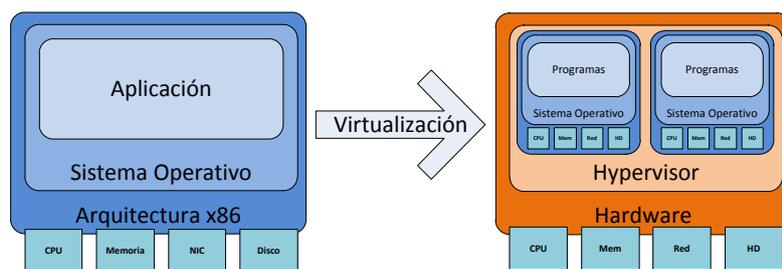
Esta tecnología es un componente fundamental del paradigma Cloud Computing.

La virtualización se inició durante la década de los 60, donde IBM fue determinante en la evolución de esta tecnología, que permitía la partición lógica de los grandes mainframes en distintas máquinas virtuales, permitiendo realizar varias tareas al mismo tiempo y aprovechar al máximo los recursos de hardware, pues estos eran muy costosos.

Esta tecnología fue dejada de lado durante la década de los 80 y 90, debido a los avances de las redes, los sistemas basados en la arquitectura x86, la gran adopción del S.O. Windows y la aparición de Linux como S.O. para servidores, se establecieron los equipos x86 como un estándar de la industria. Con el transcurrir del tiempo, comenzaron a aparecer los mismos problemas operacionales y de infraestructura que sufrieron los mainframes en la década del 60, como ser: la baja utilización de la infraestructura (las aplicaciones típicas en la arquitectura x86 logran una utilización media de entre 10% y 15% de los recursos del hardware), el incremento de los costos en la infraestructura física y la administración, y problemas en la recuperación y protección contra desastres.

A finales de los años 90 y principios del 2000, comenzó a resurgir esta tecnología, incluso para usuarios hogareños, incrementando su uso y popularidad hasta la fecha. Con la aparición de las nuevas tecnologías en virtualización asistida por hardware por parte de Intel (Intel VT) y AMD (AMD-V), se logró mejorar el rendimiento de las máquinas virtuales.

La virtualización introduce una capa de software sobre el hardware de arquitectura x86, permitiendo ejecutar concurrentemente varias instancias de máquinas virtuales sobre un único equipo, como muestra la figura 2.3.



**Figura 2.3:** Sistema virtualizado

### 2.5.1 Virtual Machine Monitor

El termino virtualización refiere a la abstracción de los recursos del hardware; para esto se crea una capa de software llamada Virtual Machine Monitor (VMM) que permite la abstracción entre el hardware, de la máquina física subyacente (*Host*), y el Sistema Operativo (S.O.), de la máquina virtual (*Guest*). [VMW07]

El VMM controla y gestiona cuatro de los recursos principales del hardware físico (CPU, Memoria, Red y Almacenamiento) permitiendo al usuario ejecutar concurrentemente múltiples instancias de S.O. en máquinas virtuales (VMs) sobre un único hardware físico. A cada una de las VM, le presenta una interfaz del hardware que sea compatible con el S.O. elegido. [GOL74]

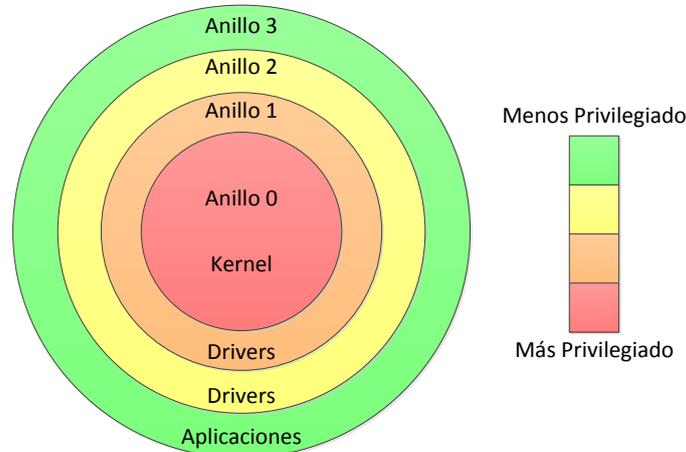
Existen dos enfoques de virtualización:

- Hypervisor (tipo I): se ejecuta directamente sobre el hardware físico, operando como una capa de software intermedia entre el hardware subyacente y los S.O. de las VMs.
- Hosted (tipo II): esta capa de software se ejecuta como un proceso sobre el S.O. de la máquina física.

Por otro lado, existen diversas estrategias para virtualizar los recursos de hardware de la arquitectura x86 (CPU, memoria, almacenamiento y dispositivos de E/S).

### 2.5.2 Técnicas de Virtualización de CPU

La arquitectura x86 convencional ofrece una protección basada en cuatro niveles de privilegio para ejecutar instrucciones sobre el procesador, como se muestra en la figura 2.4. Cada nivel es representado con un anillo. [CHE08]



**Figura 2.4:** Anillos de privilegios de la arquitectura x86

El anillo 0 es el nivel con mayor privilegio, denominado *nivel de kernel*, donde normalmente se ejecuta el S.O. controlando los recursos del hardware físico.

El anillo 3, es el de menor privilegio, donde se ejecutan las aplicaciones de los usuarios.

Cuando una aplicación de un usuario intenta ejecutar instrucciones privilegiadas, tales como acceder a la memoria o a dispositivos de E/S, se captura la petición y se eleva el nivel de privilegio de la ejecución, permitiendo ejecutar instrucciones sensibles del S.O. Luego, se retorna el control al nivel 3 para continuar con la ejecución de la aplicación.

En virtualización, es deseable que el VMM se ejecute en el anillo 0 para controlar las máquinas virtuales y los recursos de hardware, mientras que el SO Guest se ejecute en el anillo 3; de esta manera, cuando una VM intente ejecutar instrucciones privilegiadas, el VMM capture la petición y emule la ejecución del bloque de instrucciones de la VM. [GOL74] [VMW07]

Las técnicas de virtualización de CPU ofrecen soluciones a los mecanismos de protección de la arquitectura, los cuales impiden a los S.O. de las VMs ejecutar instrucciones directamente sobre el CPU en el nivel privilegiado 0, mientras que el VMM posee el control de la unidad de procesamiento.

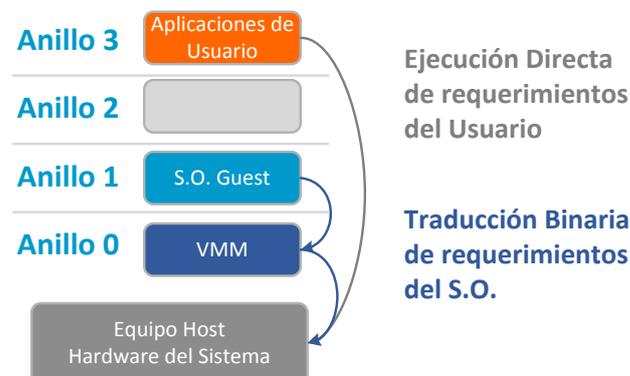
A continuación se describen tres técnicas de virtualización de CPU para controlar la ejecución de instrucciones sensibles y privilegiadas: [CHE08] [VMW07]

### Virtualización Completa (Full Virtualization)

Esta técnica combina la ejecución directa de instrucciones y la traducción binaria (binary translation) de instrucciones no virtualizables. El S.O. Host emula el hardware físico, permitiendo ejecutar varias instancias de máquinas virtuales, cada una con su propio S.O. Guest sin modificar.

Como se observa en la figura 2.5, la virtualización completa ubica al S.O. Guest en un anillo con mayores privilegios que el anillo 3, pero con menores privilegios que el anillo 0 donde se ejecuta el VMM. Cuando el S.O. Guest o una aplicación de usuario intentan ejecutar instrucciones privilegiadas, el VMM captura la llamada y realiza la traducción binaria del bloque de instrucciones privilegiadas a un bloque equivalente de instrucciones sin privilegios, las cuales se ejecutarán directamente sobre el CPU.

Por otro lado, las instrucciones sin privilegios se ejecutan directamente sobre el CPU, lo cual se conoce como ejecución directa de instrucciones.

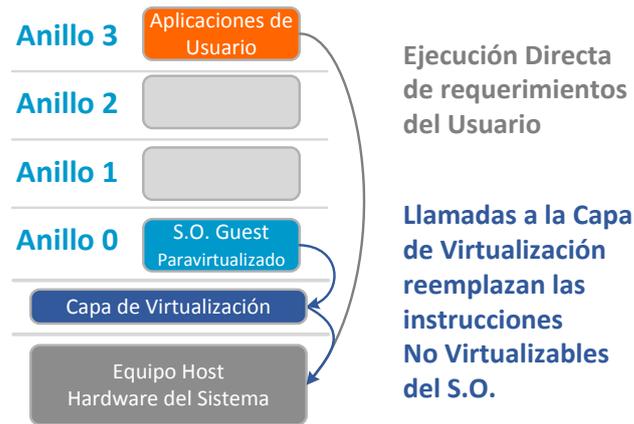


**Figura 2.5:** Virtualización Completa

## Paravirtualización

Esta técnica se diferencia de la virtualización completa pues incorpora una capa de virtualización en la arquitectura, entre el hardware físico y el anillo 0, ubicando al S.O. Guest en este último, como se presenta en la figura 2.6.

Se requiere la modificación del S.O. Guest, integrando al kernel un módulo que reemplace las llamadas a instrucciones privilegiadas con llamadas directas al hypervisor, quien se encarga de la comunicación con el hardware físico.



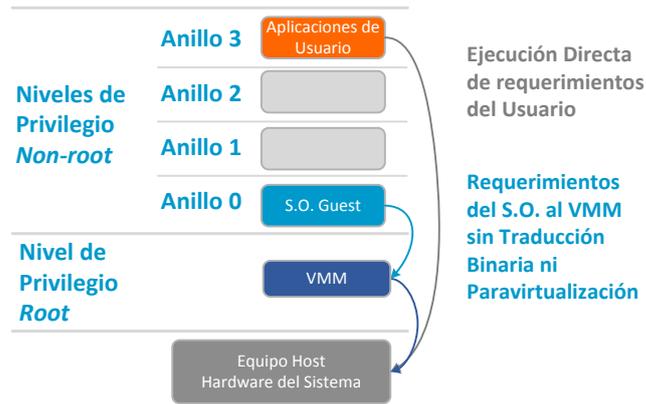
*Figura 2.6: Paravirtualización*

## Virtualización asistida por hardware

Esta técnica se distingue de la virtualización completa y la Paravirtualización, pues no requiere la traducción binaria de instrucciones privilegiadas ni la modificación del S.O. Guest.

Aprovecha el soporte de virtualización brindado por las extensiones del hardware provistas por los nuevos procesadores Intel (VT-x) y AMD (AMD-V), los cuales incorporan un nuevo nivel de privilegio en la arquitectura, llamado anillo -1 o modo root como se observa en la figura 2.7.

En esta técnica, el S.O. Guest (sin modificar) se ejecuta en el anillo 0 de los niveles de privilegio en modo Non-root, y el VMM en el nivel de privilegio en modo root, quien se encarga de acceder de forma directa a los recursos del hardware físico.



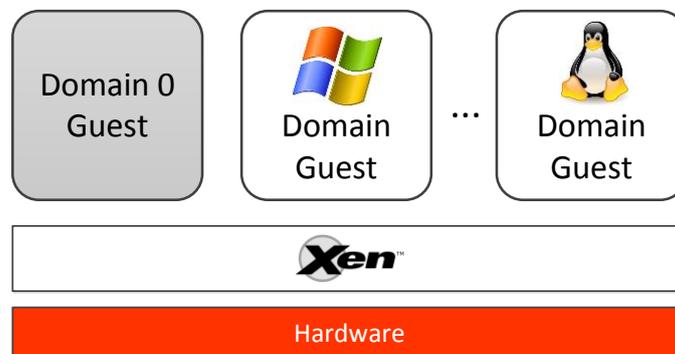
**Figura 2.7:** Virtualización asistida por hardware

## 2.6 Software de Virtualización

En la actualidad existen diversos software de virtualización, tanto propietarios como libres, de los cuales se ha optado por XEN y KVM, dos proyectos open source, ampliamente utilizados en diversos entornos.

### 2.6.1 XEN

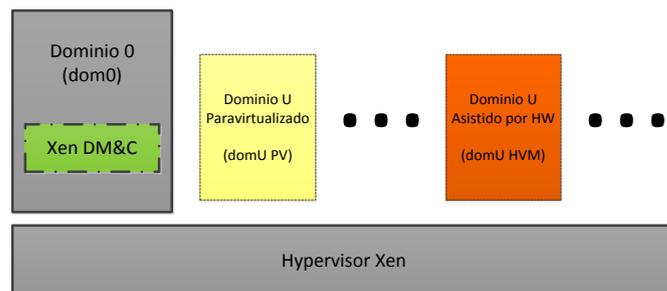
Se inició como un proyecto de investigación de Ian Pratt en la Universidad de Cambridge, publicando su primera versión en octubre de 2003. Luego, Pratt creó la empresa XenSource, la cual fue adquirida en 2007 por Citrix. En la actualidad se desarrollan dos versiones del proyecto, una forma de código abierto y otra orientada al uso comercial. [NUS09].



**Figura 2.8:** Entorno de virtualización Xen

El entorno de virtualización Xen está compuesto por: [XEN08]

- El hypervisor Xen
- Dominio 0 (dom0)
- Dominio de Administración y Control (Xen DM&C)
- Dominios U con soporte de paravirtualización (domU PV)
- Dominios U con virtualización asistida por hardware (domU HVM)



*Figura 2.9: Componentes de Xen*

### Hypervisor Xen

Este componente es la capa básica de abstracción que se encuentra ubicada directamente sobre el hardware. Es el responsable de planificar la ejecución sobre el CPU y administrar la memoria de las VMs. [XEN08]

El hypervisor no sólo abstrae el hardware de las máquinas virtuales sino que también controla la ejecución de las mismas, las cuales comparten un entorno de procesamiento común. Este no tiene conocimiento de los dispositivos de E/S.

### Dominio 0 (dom0)

Este componente es una VM con un S.O. Linux con kernel modificado que se ejecuta sobre el hypervisor, con privilegios de acceso a los recursos del hardware físico para realizar E/S, como también interactuar con las otras VMs (domU) que se ejecutan en el entorno. Es requisito indispensable que el dom0 se encuentre en ejecución antes de iniciar otras máquinas virtuales. [XEN08]

### Dominios U (domU)

Este componente se clasifica en domU PV y domU HVM, los cuales refieren a las máquinas virtuales ejecutándose en el entorno. [XEN08]

Se denomina Dominio U PV a aquellas VMs ejecutando un sistema operativo modificado, entre los cuales se soportan Linux, Solaris, FreeBSD y UNIX. Por otro lado, a las VMs que ejecutan Windows u otro sistema operativo sin modificaciones, se las denomina Dominio U HVM.

El domU PV está al tanto de que no tiene acceso directo al hardware y reconoce que otras VMs se ejecutan en la misma máquina. En contraste con el domU HVM, el cual no posee esa información.

Cuando un domU requiere utilizar dispositivos de hardware como ser la red o un dispositivo de bloques, los datos deben dirigirse al dom0 el cual se encarga de transmitir los requerimientos al dispositivo físico.

### Dominio de Administración y Control (Xen DM&C)

El Xen DM&C consiste en una serie de servicios de Linux que se ejecutan en el dom0 realizando tareas de administración y control del entorno de virtualización. [XEN08]

### 2.6.2 KVM (Kernel based Virtual Machine)

KVM es un proyecto open source de virtualización, basado en las tecnología de virtualización asistida por hardware y se encuentra incorporado en el kernel de Linux. En febrero de 2007 se introduce la primera versión en el kernel 2.6.20. [KVM09] [NUS09]

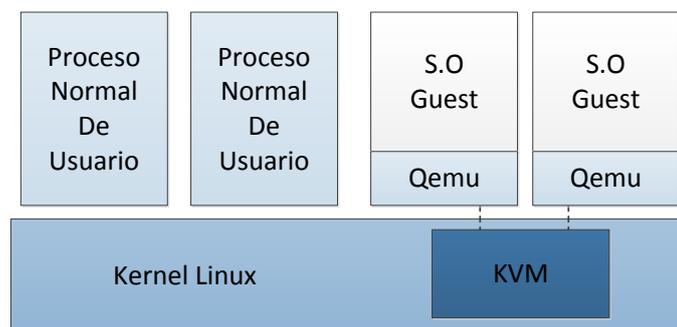
El desarrollo se inició por la empresa Qumranet, siendo adquirida luego por Red Hat. El objetivo principal fue utilizar el kernel de Linux como parte del hypervisor, sin necesidad de crear un nuevo kernel de virtualización. De esta manera, KVM se encuentra implementado como un módulo del kernel, convirtiendo al mismo en un hypervisor tipo I.

Dado que KVM fue diseñado para utilizar la virtualización asistida por hardware, requiere que el CPU soporte las extensiones Intel VT-X o AMD-V y que las mismas se encuentren habilitadas. Esto brinda una solución optimizada de virtualización en la que no se requiere modificar el S.O. de las VMs ni dar soporte a hardware heredado.

### Arquitectura de KVM

KVM, en su arquitectura, implementa una VM como un proceso normal de Linux, planificado por el scheduler estándar del S.O., donde cada CPU virtual es un thread. Esto permite a KVM beneficiarse de todas las características del kernel. [KVM09]

Los dispositivos virtuales son emulados por una versión modificada de QEMU, que provee emulación de BIOS, bus PCI, bus USB, controladores de disco IDE/SCSI, interfaces de red, entre otros.



*Figura 2.10: Arquitectura de KVM*

KVM soporta la *virtualización híbrida*, en la cual se utilizan drivers paravirtualizados instalados en S.O. Guest con el fin de permitir a las VMs un uso optimizado de los dispositivos de E/S sin necesidad de emularlos.

# Capítulo 3

## Amazon Elastic Compute Cloud (EC2)

---

En este capítulo se realiza una introducción a Amazon EC2, un proveedor público de IaaS, que se ha convertido en el estándar de-facto de Cloud Computing.

Amazon EC2 es un servicio web que proporciona recursos de infraestructura virtual, flexibles y escalables, bajo la modalidad comercial de infraestructura como servicio. Esencialmente, los recursos ofrecidos son instancias de VMs y almacenamiento para las mismas.

EC2 brinda una interfaz de servicios Web la cual permite obtener y configurar recursos de manera simple, teniendo el control total de los mismos.

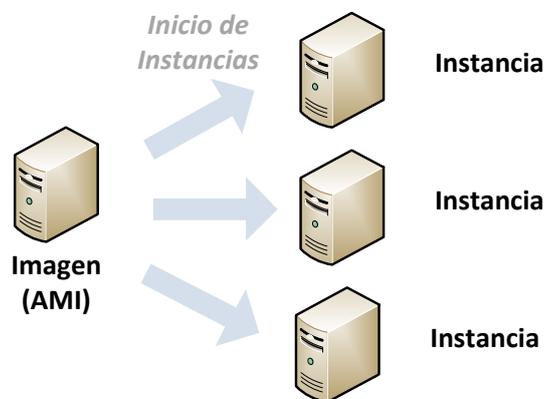
La utilización del servicio permite reducir a minutos el tiempo necesario para obtener e iniciar nuevas instancias de VMs, y posibilita escalar dinámicamente la capacidad de la infraestructura en respuesta a las variaciones en la carga de trabajo.

Amazon EC2 impacta en la economía de las organizaciones, permitiendo pagar sólo por el tiempo de utilización de los recursos.

### 3.1 Componentes de la infraestructura

#### 3.1.1 Amazon Machine Images e Instancias

Una Amazon Machine Image (AMI) es una imagen que contiene una cierta configuración de software (S.O. y aplicaciones), a partir de la cual se inicia la ejecución de una instancia de VM. La instancia es una copia en ejecución de la imagen. Pueden existir varias instancias de una misma AMI. [EC211a]



*Figura 3.1: AMIs e Instancias de EC2*

Las instancias tienen un tiempo de vida determinado desde el inicio de la ejecución hasta que las mismas se detienen, tanto por intervención del usuario como por una falla. Si una instancia falla, la misma puede ser iniciada nuevamente a partir de la AMI.

Amazon publica varias AMIs con configuraciones estándar de software para uso público. Además, los miembros de la comunidad de desarrolladores de Amazon Web Services (AWS) publican AMIs personalizadas. También, el usuario puede crear sus propias AMIs personalizadas en base a sus requerimientos, para luego ser utilizadas al iniciar nuevas instancias.

### 3.1.2 Tipos de instancias disponibles en EC2

Existen varios tipos de instancias, los cuales representan diferentes configuraciones de hardware, particularmente cantidad de procesadores, memoria y almacenamiento.

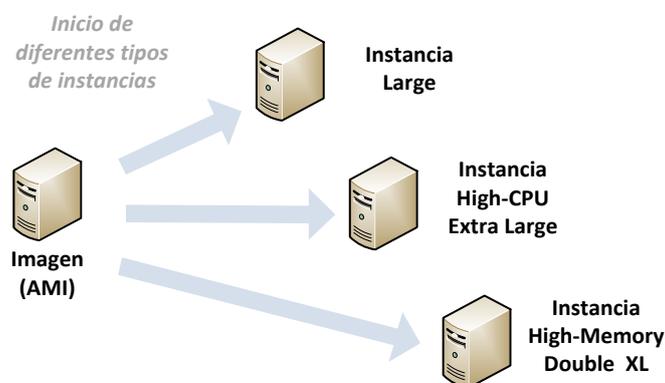


Figura 3.2: Tipos de instancias de EC2

En la tabla 3.1 se presentan los diferentes tipos de instancias disponibles en Amazon EC2.

Tipo	CPU	Memoria	Almacenamiento	E/S	Nombre
<b>Small</b>	1 Unidad de Cómputo de EC2 (1 core virtual con 1 Unidad de Cómputo)	1.7 GB	160 GB	Moderada	m1.small
<b>Large</b>	4 Unidad de Cómputo de EC2 (2 cores virtuales con 2 Unidades de Cómputo c/u)	7.5 GB	850 GB	Alta	m1.large
<b>Extra Large</b>	8 Unidad de Cómputo de EC2 (4 cores virtuales con 2 Unidades de Cómputo c/u)	15 GB	1690 GB	Alta	m1.xlarge
<b>Micro</b>	Hasta 2 Unidades de Cómputo de EC2 (para ráfagas cortas periódicas)	613 MB	Ninguno (se debe adquirir separado)	Baja por	t1.micro

<b>High-CPU Medium</b>	5 Unidad de Cómputo de EC2 (2 cores virtuales con 2.5 Unidades de Cómputo c/u)	1.7 GB	350 GB	Moderada	c1.medium
<b>High-CPU Extra Large</b>	20 Unidad de Cómputo de EC2 (8 cores virtuales con 2.5 Unidad de Cómputo de EC2s c/u)	7 GB	1690 GB	Alta	c1.xlarge
<b>High-Memory Extra Large</b>	6.5 Unidad de Cómputo de EC2 (2 cores virtuales con 3.25 Unidades de Cómputo c/u)	17.1 GB	420 GB	Moderada	m2.xlarge
<b>High-Memory Double Extra Large</b>	13 Unidad de Cómputo de EC2 (4 cores virtuales con 3.25 Unidades de Cómputo c/u)	34.2 GB	850 GB	Alta	m2.2xlarge
<b>High-Memory Quadruple Extra Large</b>	26 Unidad de Cómputo de EC2 (8 cores virtuales con 3.25 Unidades de Cómputo c/u)	68.4 GB	1690 GB	Alta	m2.4xlarge
<b>Cluster Compute</b>	33.5 Unidades de Cómputo de EC2 (2 x Intel Xeon X5570, quad-core "Nehalem" architecture)	23 GB	1690 GB	Muy Alta (10 Gbps Ethernet)	cc1.4xlarge
<b>Cluster GPU</b>	33.5 Unidades de Cómputo de EC2 (2 x Intel Xeon X5570, quad-core "Nehalem" architecture), y 2 GPUs NVIDIA Tesla M2050 "Fermi"	22 GB	1690 GB	Muy Alta (10 Gbps Ethernet)	cg1.4xlarge

*Tabla 3.1: Tipos de instancias en Amazon EC2*

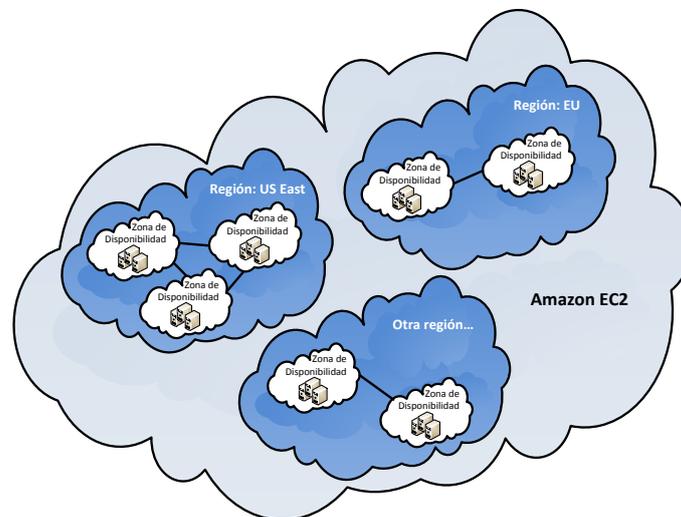
**Nota:** La capacidad de una unidad de computo de EC2 es equivalente a un CPU de 1.0-1.2 GHz tipo Opteron o Xeon del año 2007.

Es importante destacar, de la tabla citada anteriormente, las instancias tipo cc1.4xlarge y cg1.4xlarge, las cuales pueden ser de gran utilidad en entornos de cómputo científico.

### 3.1.3 Regiones y Zonas de Disponibilidad

Amazon cuenta con varios datacenters en diferentes partes del mundo (como ser, en Norteamérica, Europa, Asia). De esta manera, EC2 se encuentra disponible para su uso en diferentes regiones, brindando al usuario la posibilidad de seleccionar la región más conveniente.

Cada región está integrada por varios datacenters distribuidos geográficamente en la misma región. Estos datacenters conforman lo que se denominan Zonas de Disponibilidad. Cada zona se encuentra diseñada para tolerar los fallos de otras zonas. De este modo, un fallo en una zona no afecta a las otras zonas de la misma región. El usuario puede aprovechar esta arquitectura al enviar instancias a diferentes zonas, evitando un único punto de falla.



**Figura 3.3:** Regiones y zonas de disponibilidad de EC2

### 3.1.4 Almacenamiento

Amazon proporciona dos tipos de almacenamiento, que pueden ser utilizados con EC2: Amazon S3 y EBS, los cuales se mencionan a continuación.

#### Amazon Simple Storage Service (S3)

S3 es un servicio de almacenamiento de objetos que permite almacenar y recuperar información desde cualquier parte de la Web. [AS311]

Los objetos se encuentran organizados en depósitos o *buckets*, similares a directorios. No se permite la creación de sub-buckets. Se pueden almacenar objetos con un tamaño máximo de 5 GB, de forma encriptada. Además, la carga o descarga de los mismos se realiza en forma segura.

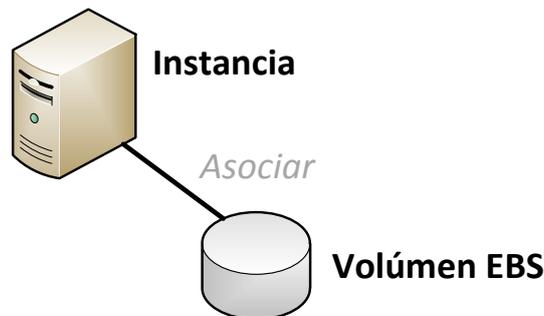
Los buckets se almacenan en una única región, usualmente seleccionada por cercanía, para optimizar la latencia y minimizar los costos. Los mismos incluyen mecanismos de autenticación diseñados para garantizar que los datos se mantienen seguros frente a

accesos no autorizados. Se permite definir permisos de acceso (público o privado) para los objetos.

### Amazon Elastic Block Store (EBS)

EBS proporciona volúmenes de almacenamiento a nivel de bloque para ser utilizados por las instancias de EC2. Estos volúmenes brindan almacenamiento de información fuera de la instancia, los cuales persisten independientemente del tiempo de vida de la misma.

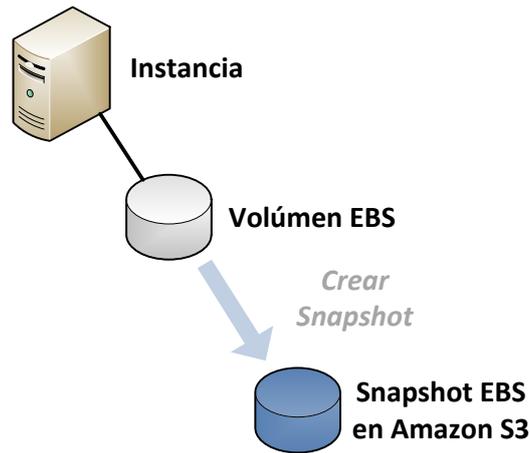
EBS ofrece volúmenes de almacenamientos altamente disponibles y confiables, los cuales pueden ser asociados a una instancia de EC2 en ejecución, para ser utilizados como un dispositivo de almacenamiento por bloques estándar dentro de la instancia. Esencialmente, los volúmenes EBS se comportan como discos rígidos virtuales conectados a una instancia, la cual puede soportar la conexión de múltiples volúmenes EBS.



*Figura 3.4: Volumen EBS asociado a una instancia de EC2*

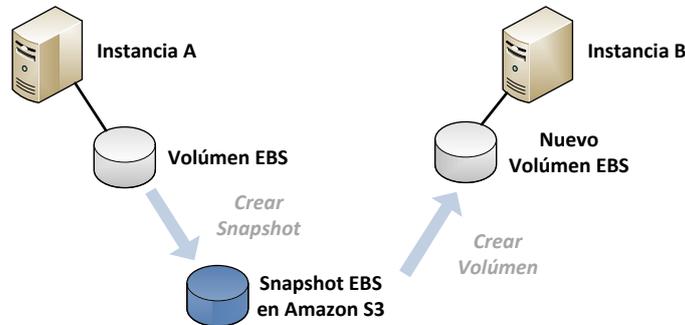
Por otro lado, los volúmenes EBS pueden ser utilizados como partición de arranque de instancias de EC2, permitiendo detener y reiniciar las mismas sin perder la información de las aplicaciones, pagando únicamente por los recursos de almacenamiento que se utilizan.

Los volúmenes EBS se encuentran replicados automáticamente dentro de la misma zona de disponibilidad para evitar una posible pérdida de información debido a un fallo. Además, EBS permite realizar snapshots (instantáneas) de los volúmenes en un punto en el tiempo, las cuales son almacenadas en S3. Estas snapshots son utilizadas como backups o como punto de partida para un nuevo volumen.



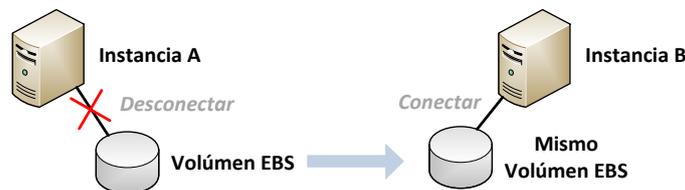
**Figura 3.5:** Creación de snapshot de un volumen EBS

En la figura 3.6 se ilustra la utilización de una snapshot como punto de partida para un nuevo volumen EBS conectado a una nueva instancia.



**Figura 3.6:** Snapshot como punto de partida para un nuevo volumen

Por otro lado, un volumen EBS se puede desconectar de una instancia y conectarse en otra instancia como se ilustra en la figura 3.7.



**Figura 3.7:** Cambio de instancia para un volumen EBS

### 3.1.5 Redes y seguridad

Al iniciar una instancia en EC2, a la misma se le asigna una dirección IP pública en forma dinámica. Las instancias pueden detenerse por intervención del usuario o por algún tipo de fallo. En ambos casos, al iniciar una nueva instancia se le asignará una nueva dirección IP distinta a la anterior. Aquí, el usuario puede requerir que la IP de la

instancia sea estática; para esto, Amazon ofrece la utilización de IPs estáticas a través del servicio Amazon Elastic IPs.

Con el objetivo de controlar el acceso a las instancias, EC2 proporciona los *grupos de seguridad*, los cuales permiten especificar los protocolos, puertos y rangos de IP a los cuales se les permite conectar a las instancias; estos grupos son análogos al funcionamiento de un firewall de red.

Es posible crear varios grupos de seguridad y asignar diferentes reglas a cada uno de ellos. Luego, se permite asignar cada instancia a uno o más grupos de seguridad y utilizar las reglas para determinar el tráfico permitido hacia la misma.

## 3.2 Funcionalidad

Amazon EC2 presenta un entorno virtual de cómputo, que permite utilizar servicios web para iniciar instancias de VMs con distintos sistemas operativos, configurar aplicaciones, gestionar permisos de acceso a la red y crear una imagen de un sistema personalizado que puede ser instanciado las veces que sean necesarias. [EC211b]

La utilización de Amazon EC2 requiere:

- Seleccionar una imagen de una VM pre configurada (AMI - Amazon Machine Image) la cual permite un rápido despliegue. O bien crear una imagen personalizada que contenga el SO, las aplicaciones, bibliotecas, datos y valores de configuración.
- Configurar la seguridad y los permisos de acceso a la red de la instancia.
- Seleccionar el tipo de instancia, y a continuación, iniciar, finalizar y monitorear tantas instancias de la AMI como sea necesario, a través de servicios web o las herramientas de gestión proporcionadas.
- Determinar la zona de ejecución de las instancias, definir la utilización de IPs estáticas y/o el tipo de almacenamiento a utilizar.
- Pagar sólo por los recursos utilizados, en base al tiempo de uso de las instancias o el volumen de datos transferido/utilizado.

## 3.3 Beneficios

A continuación se detallan los beneficios sobresalientes de Amazon EC2: [EC211b]

### Elástico

Amazon EC2 permite aumentar o reducir la capacidad de la infraestructura en cuestión de minutos, sin demorar horas ni días. Se pueden solicitar o finalizar una, cientos o

incluso miles de instancias simultáneamente. Esto se realiza a través de servicios Web o las herramientas de administración.

### Control total

El usuario posee el control total sobre sus instancias de VMs, accediendo a las mismas como un usuario privilegiado. Además, tiene el poder de detener y reiniciar sus instancias desde los servicios web y las herramientas administrativas.

### Flexible

Permite al usuario seleccionar entre varios tipos de instancias de VMs, sistemas operativos y paquetes de software. De esta manera, Amazon EC2 permite seleccionar la configuración de memoria, CPU y almacenamiento de las instancias. A modo de ejemplo, entre las opciones de sistemas operativos se incluyen varias distribuciones de Linux y Microsoft Windows Server.

### Integración con otros servicios de Amazon

Amazon EC2 es compatible con Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), Amazon SimpleDB y Amazon Simple Queue Service (Amazon SQS) con el objetivo de brindar una solución completa para una gran gama de aplicaciones.

### Confiable

El contrato de nivel de servicio (SLA) de Amazon EC2 asegura una disponibilidad del 99,95% en cada región de EC2.

### Seguro

Amazon EC2 ofrece diversos mecanismos de protección con el objetivo de que el usuario configure la seguridad de su infraestructura virtual. Entre estos se destacan el firewall que controla el acceso a la red de las instancias de VMs, y también la utilización de VPNs para establecer una conexión segura.

### Económico

El usuario sólo paga por el tiempo de uso de los recursos virtuales que utiliza, bajo las siguientes modalidades:

#### Instancias On-Demand

El usuario paga por la capacidad de la infraestructura por hora, sin compromisos a largo plazo, evitando así los costos y la complejidad de la planificación, la compra y el mantenimiento del hardware, transformando lo que normalmente son grandes costos fijos en costos variables mucho más pequeños.

## Instancias Reservadas

El usuario tiene la opción de realizar un pago único reducido para cada instancia que desee reservar y, a cambio, obtiene un importante descuento en la tarifa por uso de dicha instancia. Tras el pago por una instancia, la misma queda reservada.

## Instancias Spot

El usuario oferta sobre la capacidad sin utilizar de Amazon EC2 y puede ejecutar instancias mientras su oferta supere el precio spot actual. El precio spot cambia periódicamente según la oferta y la demanda, y los clientes cuyas ofertas lo igualen o lo superen obtienen acceso a las Instancias Spot disponibles.

## 3.4 Costos

En la tabla 3.2 se indican los costos de uso por hora, de las instancias bajo demanda de EC2, tanto para Linux como para Windows, de la región US East (Virginia). [EC211c]

	Linux	Windows
<b>Instancias Estándar</b>		
Small (m1.small)	\$0.085 por hora	\$0.12 por hora
Large (m1.large)	\$0.34 por hora	\$0.48 por hora
Extra Large (m1.xlarge)	\$0.68 por hora	\$0.96 por hora
<b>Instancias Micro</b>		
Micro (t1.micro)	\$0.02 por hora	\$0.03 por hora
<b>Instancias Hi-Memory</b>		
Extra Large (m2.xlarge)	\$0.50 por hora	\$0.62 por hora
Double Extra Large (m2.2xlarge)	\$1.00 por hora	\$1.24 por hora
Quadruple Extra Large (m2.4xlarge)	\$2.00 por hora	\$2.48 por hora
<b>Instancias Hi-CPU</b>		
Medium (c1.medium)	\$0.17 por hora	\$0.29 por hora
Extra Large (c1.xlarge)	\$0.68 por hora	\$1.16 por hora
<b>Instancias de Cluster de Computo</b>		
Quadruple Extra Large (cc1.4xlarge)	\$1.60 por hora	\$1.98 por hora
<b>Instancias de Cluster de GPU</b>		
Quadruple Extra Large (cg1.4xlarge)	\$2.10 por hora	\$2.60 por hora

*Tabla 3.2: Costos de instancias de EC2 (Octubre 2011)*

### **3.5 Acuerdo de nivel de servicio (SLA)**

El acuerdo de nivel de servicio o Service Level Agreement (SLA), es un contrato escrito entre Amazon y el usuario, con el objeto de fijar el nivel acordado para la calidad del servicio. El SLA es una herramienta que ayuda a ambas partes a llegar a un consenso en término de la disponibilidad del servicio, basándose en el porcentaje del tiempo de actividad anual. Amazon asegura un 99,95%, siendo 4,38 horas el tiempo de inactividad sobre el cual se puede exigir un resarcimiento económico del 10%. [EC211d]

En el Anexo A se transcribe el acuerdo de nivel de servicio (SLA) de EC2, el cual debe ser aceptado por el usuario previo al uso del servicio.

# Capítulo 4

## Eucalyptus

---

Este capítulo presenta Eucalyptus, un software que permite la gestión de un Cloud Privado. El mismo ofrece características y funcionalidades similares a Amazon EC2.

Por otro lado, se describen las herramientas Euca2ools e Hybridfox, las cuales brindan una interface al usuario para administrar un Cloud.

### 4.1 Gestor de Cloud: Eucalyptus

Eucalyptus es un proyecto de software Open Source, bajo la licencia GPL, que permite implementar, administrar y ofrecer Infraestructura como Servicio (IaaS) sobre arquitecturas Clouds privadas e híbridas, desplegadas en la infraestructura IT existente en una organización [NUR08][NUR09].

Eucalyptus surgió como una alternativa a Amazon EC2, brindando servicios Web compatibles con la API de Amazon; esto permite que las herramientas de gestión se utilicen en ambos entornos, ofreciendo compatibilidad y capacidad de migración entre Clouds.

Eucalyptus se presenta formalmente como un software para el despliegue y gestión de Clouds. El mismo puede ser desplegado sobre las principales distribuciones de Linux, incluyendo Ubuntu, Red Hat Enterprise Linux, CentOS y Debian, sin la necesidad de alterar las instalaciones existentes del SO. Además, permite la utilización de diversos Hypervisors de virtualización, como ser KVM, XEN, entre otros.

### 4.2 Componentes de Eucalyptus

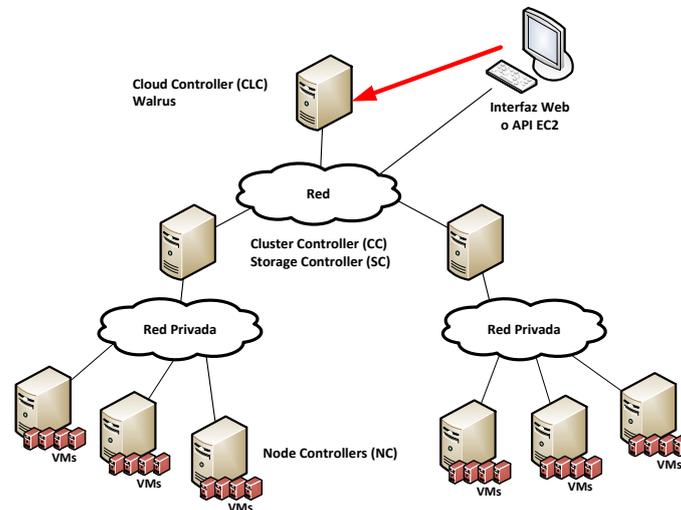
Eucalyptus se encuentra desarrollado en forma modular, presentando cinco principales componentes de alto nivel: [NUR08][SHA10]

- Cloud Controller (CLC)
- Cluster Controller (CC)
- Node Controller (NC)
- Storage Controller (SC)
- Walrus

Estos componentes se encuentran implementados como servicios web stand-alone. Esta disposición presenta las siguientes ventajas:

- En primer lugar, cada servicio Web expone una API bien definida, independiente del lenguaje, en la forma de un documento WSDL el cual contiene tanto las operaciones que puede realizar el servicio como así también las estructuras de entrada/salida de datos.
- Por otro lado, de esta forma se aprovechan las características de los servicios Web, tales como las políticas de seguridad (WSS) para la comunicación segura entre los componentes y la base de paquetes de software de servicios web disponibles en la industria.

En la figura 4.1 se muestra un esquema de la arquitectura de un sistema Eucalyptus y sus componentes clave y a continuación se realiza una descripción de los mismos.



*Figura 4.1: Componentes de Eucalyptus*

### Cloud Controller (CLC)

Este componente se presenta como el front-end del Cloud, proporcionando gestión de alto nivel de los recursos del mismo.

El CLC es responsable de recibir y procesar las solicitudes de los usuarios, que se conectan con el objetivo de instanciar, configurar y administrar VMs, redes virtuales, almacenamiento. Además, se encarga de controlar la autenticación de los usuarios y la seguridad del sistema.

También tiene la responsabilidad de consultar a los CCs sobre la información de los recursos con el fin tomar decisiones de scheduling de alto nivel sobre los mismos.

En esencia, el CLC es responsable de presentar y administrar los recursos virtualizados, a través de una API y una interfaz web.

## Cluster Controller (CC)

Este componente funciona como puente entre el CLC y los NCs. Se presenta como el front-end de un Cluster de Node Controllers, proporcionando la gestión de máquinas y redes virtuales. Cada CC representa una zona de disponibilidad de recursos del Cloud, de forma similar a las presentes en Amazon EC2.

El CC es responsable de recolectar la información del estado de los NCs. También se encarga de recibir las solicitudes de instanciación de VMs desde el CLC y planificar su ejecución sobre los nodos. Además, administra la configuración de las redes, tanto pública como privada, de las instancias.

El CC presenta una colección de operaciones, de las cuales es importante destacar las siguientes: `runInstances`, `describeInstances`, `terminateInstances`, `describeResources`.

Las tres primeras operaciones citadas realizan la ejecución, descripción y terminación de instancias, respectivamente. La última operación realiza la descripción de los recursos disponibles en el Cluster.

Las operaciones de descripción (`describeInstances`) y terminación (`terminateInstances`) de instancias son reenviadas al NC correspondiente para ser procesadas.

La operación de ejecución de una instancia (`runInstances`), implica la tarea de planificación para determinar el NC sobre el cual se ejecutará la VM. Dicha tarea consiste en consultar la disponibilidad de recursos de cada nodo, por medio de la operación `describeResource`; para luego seleccionar el primer nodo que posea suficientes recursos para la ejecución de la VM.

Por otro lado, la operación `describeResources`, es utilizada para determinar el número de instancias que pueden ser ejecutadas sobre los NCs.

Es importante destacar que este componente debe pertenecer al mismo dominio de broadcast Ethernet de los nodos que administra.

## Node Controller (NC)

El NC es responsable de iniciar, inspeccionar y terminar las instancias de las VMs a través del Hypervisor.

Generalmente existen varios NCs en un Cloud. Cada NC se ejecuta sobre una única máquina física, dado que un nodo puede administrar múltiples instancias de máquinas virtuales sobre un mismo equipo.

El NC presenta también una colección de operaciones, de las cuales se mencionan las más significativas: `runInstance`, `describeInstance`, `terminateInstance` y `describeResource`.

Las tres primeras operaciones citadas realizan la ejecución, descripción y terminación de una instancia, respectivamente. Estas acciones son llevadas a cabo a través de llamadas al Hypervisor. Es importante destacar que la operación runInstance realiza la carga de las imágenes del S.O. que será ejecutado en las instancias de VMs. Estas imágenes se denominan Eucalyptus Machine Image (EMI) y son similares a las AMIs provistas en Amazon EC2.

Por otro lado, la operación describeResource reporta a su CC los recursos físicos del equipo (cores de computo, memoria y capacidad de disco).

### Storage Controller (SC)

Este componente proporciona un manejo de volúmenes virtuales de disco para las instancias de VMs. Estos recursos de almacenamiento se presentan como discos rígidos virtuales, con acceso por bloques, que pueden ser formateados y utilizados como un disco físico tradicional. Estos volúmenes pueden ser asociados y disociados en forma dinámica a una instancia.

Por otro lado, sobre los volúmenes se permite la creación de snapshots (instantáneas) y su almacenamiento en Walrus. Los snapshots son copias del estado actual de un volumen, las cuales pueden ser utilizadas como backups o como punto de partida para un nuevo volumen.

El Storage Controller (SC) provee una funcionalidad similar a la de Amazon Elastic Block Storage (EBS) y es capaz de comunicarse con diversos sistemas de almacenamiento (NFS, iSCSI, etc.).

### Walrus

Este componente permite a los usuarios almacenar información persistente, organizada en forma de buckets (cubetas) y objetos. Se brinda la posibilidad de crear, eliminar y listar buckets, subir, bajar o eliminar objetos y configurar políticas y permisos de acceso.

Los buckets son contenedores de almacenamiento, los cuales permiten guardar y obtener objetos, por medio de los comandos Put y Get.

Generalmente, Walrus es utilizado por Eucalyptus para el almacenamiento de las imágenes de los SO de VMs y datos del usuario.

La interface de Walrus es compatible con Amazon S3.

## 4.3 Seguridad

La seguridad de Eucalyptus se divide en tres capas básicas, las cuales se relacionan entre ellas: [WAR09]

- Autenticación y Autorización.
- Aislamiento de la red.
- Aislamiento de las instancias de VMs.

### Autenticación y Autorización

Eucalyptus provee autenticación y autorización para los siguientes casos:

- Los usuarios del Cloud, los cuales poseen diversos privilegios.
- Los componentes del sistema (NC, CLC, CC), que requieren una relación de confianza para interactuar.

En forma general, la autenticación y autorización de usuarios y componentes se realiza utilizando certificados X.509 generados de forma local, pares de llaves criptográficas y comunicaciones seguras. Estas tareas son realizadas por el CLC, quien contiene el repositorio de privilegios de cada usuario.

## 4.4 Redes

Eucalyptus proporciona la configuración de red necesaria para permitir el acceso desde y hacia las instancias de VMs. Es recomendable que las máquinas virtuales residan en subredes virtuales configuradas para ser independientes y distintas de la red física de los componentes de Eucalyptus. Estas subredes virtuales son definidas al momento de la creación de las instancias y se liberan cuando la última instancia en una subred virtual termina. [EUC11]

La configuración de la red es un aspecto muy importante en la seguridad de Eucalyptus, pues se debe evitar la interceptación del tráfico de red de las instancias por un usuario malicioso [WAR09].

Eucalyptus ofrece cuatro diferentes modos de red (MANAGED, MANAGED-NOVLAN, SYSTEM y STATIC) los cuales presentan diferentes características y requerimientos.

#### 4.4.1 Características

La elección de un modo de red depende de las características deseadas y el grado de control que se permite sobre la red física subyacente. A continuación se presenta una lista de características de los modos de red de Eucalyptus.

- **Conectividad:** Esta característica permite que una VM sea accedida a través de una dirección IP y un puerto específico, siempre y cuando la dirección IP sea pública y se permita el tráfico mediante una regla en el grupo de seguridad.
- **Control de Direcciones IP:** Esta característica tiene como objetivo la asignación de direcciones IPs a las VMs en forma dinámica y controlada por Eucalyptus.
- **Grupos de seguridad:** definen un conjunto de reglas para permitir determinado tráfico de datos en la red. Estas reglas se aplican a todas las instancias de VMs asociadas a un grupo.
- **IPs Elásticas:** esta característica permite al usuario controlar la asignación de direcciones IP públicas, a partir de un pool de direcciones disponibles.
- **Aislamiento de VMs:** Eucalyptus permite el aislamiento del tráfico de red entre máquinas virtuales de distintos grupos de seguridad y/o usuarios. El aislamiento es implementado por medio del uso de VLANs taggeadas por grupo de seguridad. Esto evita que usuarios mal intencionados puedan interceptar el tráfico de otro usuario.

#### 4.4.2 Requerimientos

Al momento de decidir el modo de red a utilizar, también se deben tener en cuenta los requerimientos del mismo, los cuales se describen a continuación.

- **VLAN Clean:** Con el objetivo de lograr el aislamiento del tráfico de red entre grupos de seguridad y/o usuarios, Eucalyptus define VLANs para cada uno de los mismos. Se requiere que los dispositivos de red físicos permitan el reenvío de paquetes de VLANs taggeados, sin remover ni modificar el tag.
- **Direcciones IP disponibles:** generalmente, cada instancia de VM posee dos direcciones IP asociadas a la misma: una privada (perteneciente a la subred virtual del grupo de seguridad al que pertenece) y otra pública para ser accedida desde el exterior. De esta última, se deben disponer tantas direcciones IP públicas como VMs se requieran.
- **Servidor DHCP:** para lograr la asignación de direcciones IP a las instancias, se requiere un servicio DHCP ejecutándose en el CC.

### 4.4.3 Modos de Red

Eucalyptus, dependiendo del nivel de seguridad que una instalación puede requerir, permite los siguientes modos de red:

#### Modo MANAGED

Este modo gestiona la red de datos de las instancias de VM brindando todas las características antes mencionadas, incluyendo grupos de seguridad, aislamiento de la red, e IPs Elásticas. El aislamiento es realizado a través de VLANs, lo cual requiere que los dispositivos de red físicos soporten VLAN Clean. Por lo tanto, se requiere definir un segmento privado de red, el cual se fragmentará en subredes para la asignación de direcciones IP a las instancias de VM. Esta asignación es realizada por Eucalyptus, quien gestiona un servidor de DHCP.

Por otro lado, el modo MANAGED permite a los usuarios crear grupos de seguridad para definir reglas de acceso a las instancias de VM, como permitir el tráfico ICMP y/o el acceso vía SSH, entre otros. Estos grupos son similares a los provistos por Amazon EC2.

Además, este modo brinda la posibilidad de especificar un conjunto de direcciones IP públicas las cuales pueden ser asignadas por los usuarios a las instancias de VM, ya sea en el arranque o dinámicamente en tiempo de ejecución. Esta característica es similar a ElasticIPs de Amazon EC2.

#### Modo MANAGED-NOVLAN

Este modo es similar al modo MANAGED, proporcionando las características de IPs elásticas y grupos de seguridad. Sin embargo, no proporciona aislamiento de red a las instancias de VM, sólo brinda subredes distintas a cada grupo de seguridad (aislamiento en Capa 3 del modelo OSI).

El modo MANAGED-NOVLAN es frecuentemente utilizado cuando los dispositivos físicos de red no soportan VLAN Clean.

#### Modo STATIC

Este modo sólo permite administrar la asignación de direcciones IP a las instancias de VM, por medio de entradas estáticas de pares MAC-IP en el servidor DHCP controlado por Eucalyptus.

A distinción de los modos anteriores, el modo STATIC no brinda IPs Elásticas, grupos de seguridad, ni tampoco aislamiento de red a las instancias de VM.

Es requerimiento de este modo que no exista en la red un servidor de DHCP externo a Eucalyptus.

## Modo SYSTEM

Este modo ofrece el menor número de características, proporcionando solo conectividad. Requiere de un servidor de DHCP externo a Eucalyptus en la red.

Al igual que el modo STATIC, no brinda IPs Elásticas, grupos de seguridad, ni tampoco aislamiento de red a las instancias de VM.

En la tabla 4.1 se presenta un resumen de características y requerimientos de los modos de red de Eucalyptus.

Modo de Red	Requerimientos			Características				
	VLAN Clean	Direcciones IP disponibles	Servidor DHCP	Conectividad	Control de direcciones IP	Grupos de Seguridad	IPs Elásticas	Aislamiento de VMs
MANAGED	✓	✓		✓	✓	✓	✓	✓
MANAGED-NOVLAN		✓	✗	✓	✓	✓	✓	✓*
STATIC		✓	✗	✓	✓			
SYSTEM			✓	✓				

✗ - Un servidor de DHCP externo a Eucalyptus puede interferir con la asignación de direcciones IP.  
\* - Sólo provee aislamiento a las VMs en la capa 3 del modelo OSI.

*Tabla 4.1: Modos de Red de Eucalyptus*

## 4.5 Herramientas de administración

Entre las herramientas para la gestión de un Cloud, se presentan Euca2ools e HybridFox.

### 4.5.1 Euca2ools

Euca2ools es un proyecto de código abierto, implementado en lenguaje Python. Su objetivo es brindar herramientas para interactuar con los servicios Web de un Cloud, por medio de la línea de comandos. Estas herramientas están basadas en las provistas por Amazon (api-tools y ami-tools), permitiendo gestionar un Cloud desplegado en Amazon EC2 como en Eucalyptus [EUC11a].

Euca2ools permite realizar las siguientes acciones sobre el Cloud:

- Gestión de imágenes.
- Administración de instancias.
- Administración de volúmenes e instantáneas.
- Gestión de direcciones IP.
- Administración de claves SSH.
- Gestión de grupos de seguridad.
- Consulta de las zonas de disponibilidad.

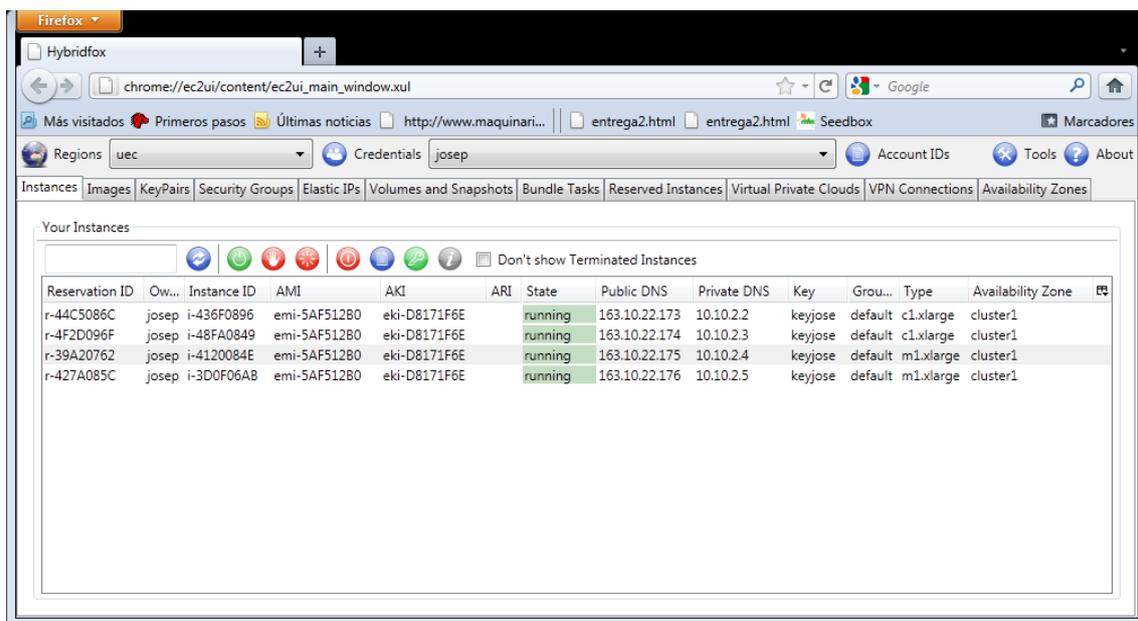
En el Anexo B se presenta la descripción y sintaxis de cada comando de Euca2ools.

## 4.5.2 HybridFox

HybridFox es una interface gráfica que permite gestionar un Cloud desplegado en Amazon EC2 y Eucalyptus. Funciona como una extensión del navegador Web Mozilla Firefox [HYB11].

HybridFox surge como una evolución de ElasticFox hacia Clouds privados e híbridos, permitiendo la administración de imágenes, instancias, IPs elásticas, grupos de seguridad, claves SSH y almacenamiento.

En la figura 4.2 se presenta una captura de pantalla de la interface de HybridFox.



*Figura 4.2: Interface de HybridFox*



# Capítulo 5

## Despliegue de un Cloud Privado

En este capítulo se detalla el despliegue de un Cloud Privado utilizando herramientas de software Open Source, con el objetivo de estudiar la arquitectura de Cloud Computing y realizar la ejecución de aplicaciones científicas sobre la misma.

Se ha realizado el despliegue de un Cloud Privado, utilizando servidores de altas prestaciones con la distribución de Linux Ubuntu Enterprise Cloud (UEC) y Eucalyptus como herramienta de gestión del Cloud, del cual se ha documentado el proceso de instalación y configuración del mismo.

Una vez completado el despliegue, se ha realizado la personalización de una imagen básica de Ubuntu Linux, con el objetivo de iniciar instancias de VMs que conformen los nodos del Cluster para aplicaciones científicas.

### 5.1 Hardware

Para el despliegue de la arquitectura se utilizaron 5 servidores y 2 switches, facilitados por el Instituto de Investigación en Informática LIDI (III-LIDI). En la tabla 5.1 se describe la especificación del hardware de los servidores.

Cantidad	Equipo	Formato	Procesadores	Memoria RAM	Disco Rígido	Red
2	Dell PowerEdge R610	Rack	2 x Intel Quad-Core Xeon E5620 2.4 GHz	48 GB	500 GB	Gigabit Ethernet
2	HP BL260c	Blade	2 x Intel Quad-Core Xeon E5405 2.0 GHz	10 GB	250 GB	Gigabit Ethernet
1	Dell PowerEdge R710	Rack	2 x Intel Quad-Core Xeon E5520 2.27 GHz	24 GB	4 x 500 GB	Gigabit Ethernet

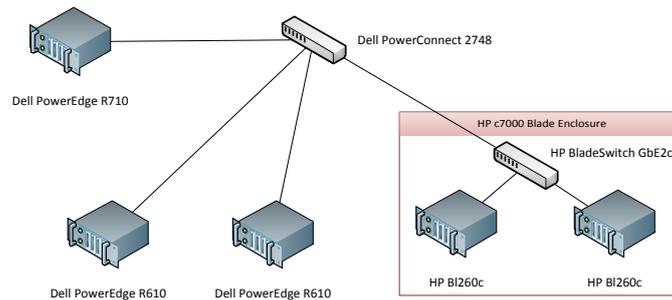
*Tabla 5.1: Especificación de los servidores utilizados*

En la tabla 5.2 se sintetiza la especificación de los switches utilizados.

Cantidad	Equipo	Formato	Puertos	Soporte
1	Dell PowerConnect 2748	Rack	48 Gigabit	VLAN Administración
1	HP BladeSwitch GbE2c	Blade	21 Gigabit	VLAN Administración

*Tabla 5.2: Especificación de los dispositivos de red utilizados*

En la figura 5.1 se presenta la topología física del hardware descrito anteriormente.



**Figura 5.1:** Topología física del hardware utilizado

## 5.2 Software

A partir del estudio del software Eucalyptus, presentado en el Capítulo 4, se ha optado por utilizarlo como herramienta de gestión del Cloud Privado dada su compatibilidad y similitud de uso con Amazon EC2.

Además, se ha utilizado la distribución de Linux Ubuntu Enterprise Cloud (UEC) en su versión 10.10 (Maverick), la cual integra Eucalyptus versión 2.0.0. Esta distribución brinda la posibilidad de desplegar un Cloud Privado del tipo Infraestructura como Servicio (IaaS), compatible con Amazon EC2. [WAR09].

Con respecto a la virtualización, se ha empleado Kernel based Virtual Machine (QEMU-KVM) en su versión 0.12.5, también integrado en UEC. [KVM09].

Como herramientas de administración se han utilizado Euca2ools versión 1.2, euca\_conf e Hybridfox versión 1.7.

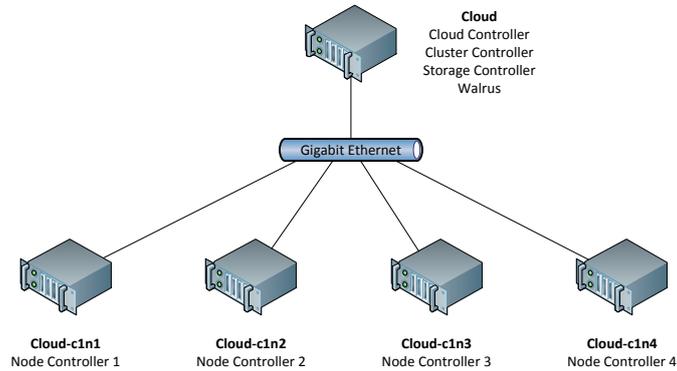
## 5.3 Instalación

Se ha diseñado la configuración de la arquitectura de Eucalyptus, ubicando cada componente en el hardware disponible, tal como se describe en la tabla 5.3.

Componente	Equipo	Hostname	Dirección IP
<b>Cloud Controller (CLC)</b>	VM en Dell PowerEdge R710	Cloud	163.10.22.167
<b>Cluster Controller (CC)</b>			
<b>Storage Controller (SC)</b>			
<b>Walrus</b>			
<b>Node Controller 1</b>	Dell PowerEdge R610	Cloud-c1n1	163.10.22.169
<b>Node Controller 2</b>	Dell PowerEdge R610	Cloud-c1n2	163.10.22.170
<b>Node Controller 3</b>	HP BI260c	Cloud-c1n3	163.10.22.171
<b>Node Controller 4</b>	HP BI260c	Cloud-c1n4	163.10.22.172

**Tabla 5.3:** Configuración de la arquitectura de Eucalyptus

En la figura 5.2, se ilustra la arquitectura de Eucalyptus que se ha utilizado para el despliegue del Cloud Privado.



*Figura 5.2: Despliegue del Cloud Eucalyptus*

A continuación, se ha procedido a realizar la instalación del S.O. UEC sobre los equipos, seleccionando la instalación de los componentes de Eucalyptus asignados a cada uno.

Una vez finalizada la instalación de los componentes, se ha utilizado la interface web de administración, accesible a través del protocolo HTTPS en el puerto TCP 8443 del CLC (<https://cloud:8443>), donde se ha creado un usuario del Cloud y se le han otorgado privilegios de administrador. A continuación, se han descargado las credenciales de seguridad del usuario para autenticar su acceso al Cloud.

Las credenciales del usuario mencionadas anteriormente, han sido copiadas al CLC, por medio de la herramienta SCP. Luego, las mismas son descomprimidas en el subdirectorio `.euca`, del directorio raíz de un usuario local del CLC (`/home/user/.euca`). De esta forma, el usuario local obtiene los privilegios de acceso al Cloud y se encuentra en condiciones de operar con el Cloud por medio de las herramientas `Euca2ools`.

Por otro lado, con el fin de comprobar el funcionamiento del Cloud, se han verificado las zonas de disponibilidad por defecto de la instalación de Eucalyptus. Para esto, se ha ejecutado el comando `euca-describe-availability-zones`, del cual se puede obtener mayor información en el Anexo B.

En la tabla 5.4 se presenta la salida obtenida tras la ejecución del comando, detallando las zonas y los tipos de instancias disponibles.

```

user@cloud:~$ euca-describe-availability-zones verbose
AVAILABILITYZONE      cluster1      163.10.22.167
AVAILABILITYZONE      |- vm types   free / max   cpu   ram   disk
AVAILABILITYZONE      |- m1.small   0032 / 0032  1     192   2
AVAILABILITYZONE      |- c1.medium  0032 / 0032  1     256   5
AVAILABILITYZONE      |- m1.large   0016 / 0016  2     512  10
AVAILABILITYZONE      |- m1.xlarge  0016 / 0016  2    1024  20
AVAILABILITYZONE      |- c1.xlarge  0008 / 0008  4    2048  20

```

*Tabla 5.4: Zona de disponibilidad del Cloud*

Tras la ejecución del comando anterior, se puede extraer la configuración por defecto de los tipos de instancias que pueden ser creadas en el Cloud Eucalyptus. La tabla 5.5 resume dicha configuración.

Nombre	CPU	Memoria	Almacenamiento
<b>m1.small</b>	1	192 MB	2 GB
<b>c1.medium</b>	1	256 MB	5 GB
<b>m1.large</b>	2	512 MB	10 GB
<b>m1.xlarge</b>	2	1 GB	20 GB
<b>c1.xlarge</b>	4	2 GB	20 GB

*Tabla 5.5: Tipos de instancias por defecto de Eucalyptus*

Además, en la tabla 5.4 se indica la cantidad de instancias de cada tipo que pueden ser creadas, conforme a la capacidad de los recursos físicos disponibles en los NCs. Estos recursos se encuentran conformados por 32 cores físicos de CPU y 116 GB de memoria RAM, como ha sido descrito anteriormente en la sección 5.1. A modo de ejemplo, los usuarios del Cloud pueden crear hasta 8 instancias del tipo c1.xlarge, las cuales poseen 4 CPUs, 2 GB de memoria y 20 GB de almacenamiento, cada una.

Por otro lado, se ha utilizado la herramienta euca\_conf para verificar el estado de la infraestructura Cloud.

En la tabla 5.6 se presentan los resultados de la ejecución de dicha herramienta.

```

user@cloud:~$ sudo euca_conf --list-clusters
registered clusters:
cluster1 163.10.22.167

user@cloud:~$ sudo euca_conf --list-nodes
registered nodes:
163.10.22.169 cluster1
163.10.22.170 cluster1
163.10.22.171 cluster1
163.10.22.172 cluster1

```

```

user@cloud:~$ sudo euca_conf --list-scs
registered storage controllers:
cluster1 163.10.22.167

user@cloud:~$ sudo euca_conf --list-walruses
registeredwalruses:
walrus 163.10.22.167

```

*Tabla 5.6: Estado de Eucalyptus utilizando euca\_conf*

Luego, se ha testado el funcionamiento del Cloud, realizando la publicación de una imagen de S.O., disponible en el repositorio de Ubuntu, y creando una instancia de la misma.

## 5.4 Publicación de una imagen

La publicación es una tarea fundamental, que requiere ser realizada con el fin de disponer de una imagen e iniciar una instancia de la misma.

Esta tarea se inicia con la descarga de una imagen desde un repositorio de imágenes, en este caso se ha utilizado el repositorio de Ubuntu, situado en <http://uec-images.ubuntu.com>.

De dicho repositorio, se ha descargado la imagen de Ubuntu Server versión 10.10 Natty, por medio del comando `wget` en la consola del CLC. En la tabla 5.7 se presenta la salida de la ejecución del comando `wget` para descargar la imagen correspondiente.

```

user@cloud:~$ wget http://uec-
images.ubuntu.com/natty/current/natty-server-uec-amd64.tar.gz
--2011-04-28 16:14:43-- http://uec-
images.ubuntu.com/natty/current/natty-server-uec-amd64.tar.gz
Resolving uec-images.ubuntu.com... 91.189.88.35
Connecting to uec-images.ubuntu.com|91.189.88.35|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 182364877 (174M) [application/x-gzip]
Saving to: `natty-server-uec-amd64.tar.gz.1'

```

*Tabla 5.7: Descarga de una imagen desde el repositorio de Ubuntu*

A continuación, se ha efectuado la publicación de la misma utilizando el comando `uec-publish-tarball`, el cual es provisto como herramienta de UEC. La ejecución de este comando debe ser llevada a cabo por el usuario al cual se le han otorgado los privilegios de administrador en el Cloud. En la tabla 5.8, se puede visualizar el resultado de dicha ejecución. [UEC10].

```
user@cloud:~/natty-server-uec-amd64$ uec-publish-tarball natty-server-uec-amd64.tar.gz natty-server-uec-amd64
Thu Apr 28 16:24:18 ART 2011: ===== extracting image =====
Warning: no ramdisk found, assuming '--ramdisk none'
kernel : natty-server-uec-amd64-vmlinuz-virtual
ramdisk: none
image  : natty-server-uec-amd64.img
Thu Apr 28 16:24:27 ART 2011: ===== bundle/upload kernel =====
Thu Apr 28 16:24:29 ART 2011: ===== bundle/upload image =====
Thu Apr 28 16:25:10 ART 2011: ===== done =====
emi="emi-13851989"; eri="none"; eki="eki-8EEB1E98";
```

*Tabla 5.8: Publicación de una imagen en el Cloud con uec-publish-tarball*

Una vez finalizada la tarea de publicación, por medio de la herramienta HybridFox, se ha comprobado que la imagen se encuentra visible en el Cloud. Además, se ha creado una instancia de la imagen, verificando el correcto funcionamiento de la misma.

### 5.5 Configuración de red

La instalación de Eucalyptus establece por defecto la configuración de red en modo MANAGED-NOVLAN, el cual no provee aislamiento entre las instancias de VMs de diferentes usuarios. Por este motivo, se ha decidido modificar la configuración de red a modo MANAGED, el cual provee aislamiento de las instancias de diferentes usuarios, asignando las mismas en VLANs distintas. [EUC11b].

El modo MANAGED requiere que los dispositivos de red, que integran la arquitectura, soporten VLAN Clean. Esto implica que los mismos permitan el paso de paquetes etiquetados con VLANs, dejando intactas sus etiquetas sin removerlas ni modificarlas.

Con el propósito de modificar el modo de red de Eucalyptus, se han configurado los dispositivos de red (Switches) para soportar VLAN Clean.

A continuación, se ha verificado el funcionamiento de los dispositivos de red, por medio de la creación de subinterfaces virtuales en el CC y los NCs, y se han configurado las mismas, con un segmento de red privado. Luego, se han realizado pruebas de conectividad, por medio del comando ping, corroborando el correcto funcionamiento de la interconexión. En la tabla 5.9 se muestra el procedimiento realizado.

```
//Configuración de VLAN Clean en el CC:
root@cloud# vconfig add eth0 10
root@cloud# ifconfig eth0.10 192.168.1.1 up

//Configuración de VLAN Clean en cada NC:
root@cloud-cln1# vconfig add eth0 10
root@cloud-cln1# ifconfig eth0.10 192.168.1.2 up
```

```

//Test de ping desde cada NC al CC:
root@cloud-cln1# ping 192.168.1.1 --> OK

//Test de ping desde el CC a cada NC:
root@cloud-cln1# ping 192.168.1.2 --> OK

//Eliminación de la configuración VLAN Clean en cada equipo:
# vconfig rem eth0.10

```

**Tabla 5.9:** Pruebas de VLAN Clean

Efectuadas las pruebas de conectividad, se han modificado los archivos de configuración de Eucalyptus en el CC y los NCs, como se observan en las tablas 5.10 y 5.11, respectivamente.

```

# eucalyptus.local.conf en el CC
VNET_MODE="MANAGED"
VNET_PUBINTERFACE="eth0"
VNET_PRIVINTERFACE="eth0"
VNET_SUBNET="10.10.0.0"
VNET_NETMASK="255.255.0.0"
VNET_DNS="163.10.22.88"
VNET_ADDRSPERNET="64"
VNET_PUBLICIPS="163.10.22.173-163.10.22.189"

```

**Tabla 5.10:** Configuración del CC en modo MANAGED en eucalyptus.local.conf

```

# eucalyptus.conf en cada NC
VNET_PUBINTERFACE="eth0"
VNET_PRIVINTERFACE="eth0"
VNET_MODE="MANAGED"

```

**Tabla 5.11:** Configuración de los NCs en modo MANAGED en eucalyptus.conf

Una vez completada la configuración, se ha verificado nuevamente el funcionamiento de las instancias y se ha corroborado la correcta conectividad de la red.

## 5.6 Despliegue de un Cluster en el Cloud

Una vez desplegado el Cloud Privado, se ha creado un entorno de cómputo paralelo, para la ejecución de aplicaciones científicas, las cuales utilizan las librerías de pasaje de mensajes MPI.

Dicho entorno, consiste de un Cluster cuyos nodos son instancias de VMs iniciadas en el Cloud. Para esto, se ha creado una imagen que contenga las librerías y las configuraciones necesarias.

### 5.6.1 Creación de una imagen para un nodo del Cluster

La tarea de creación de una imagen en el Cloud, implica la personalización de una imagen existente y la publicación de la misma.

A partir de una imagen con un S.O. Linux Ubuntu, la cual ha sido descargada del repositorio oficial, se ha realizado la instalación de la librería OpenMPI versión 1.4.1 y se ha efectuado la configuración de permisos para la ejecución de aplicaciones en el Cluster. [MPI11].

Como primer paso, se ha creado una instancia a partir de la imagen ya mencionada, a la cual se le ha adjuntado un nuevo volumen de almacenamiento de 10GB. En este volumen se ha creado una partición con su correspondiente filesystem, la cual se ha montado con el propósito de crear la imagen sobre la misma.

Luego, se ha instalado OpenMPI en la instancia y se ha configurado el servicio SSH para permitir el acceso del usuario root sin requerir contraseña.

Por último, se ha realizado la creación, upload y registro de la imagen, por medio de los comandos de Euca2ools. Los resultados de la ejecución de dichos comandos se presentan en la tabla 5.12. [EUC11c].

```
//Creamos la imagen a partir de la VM actual

root@ubuntu# euca-bundle-vol -s 2048 -d /mnt/ -e /root/.euca --kernel
eki*** -r x86_64 -p ubuntu-mpi

1+0 records in
1+0 records out
1048576 bytes (1.0 MB) copied, 0.00250326 s, 419 MB/s
mke2fs 1.41.12 (17-May-2010)
Filesystem label=uec-rootfs
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
131072 inodes, 524288 blocks
26214 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=536870912
16 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 27 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.

Checking image
Tarring image
```

```

Encrypting image
Splitting image...
Part: ubuntu-mpi.part.0
Part: ubuntu-mpi.part.1
Part: ubuntu-mpi.part.2
. . .
Part: ubuntu-mpi.part.24
Part: ubuntu-mpi.part.25
Generating manifest /mnt/ubuntu-mpi.manifest.xml

//Con el manifest .xml que genera el bundle, debemos subirla al walrus:

root@ubuntu# euca-upload-bundle -b ubuntu-mpi -m /mnt/ubuntu-mpi.manifest.xml

Checking bucket: ubuntu-mpi
Creating bucket: ubuntu-mpi
Uploading manifest file
Uploading part: ubuntu-mpi.part.0
Uploading part: ubuntu-mpi.part.1
Uploading part: ubuntu-mpi.part.2
. . .
Uploading part: ubuntu-mpi.part.24
Uploading part: ubuntu-mpi.part.25
Uploaded image as ubuntu-mpi/ubuntu-mpi.manifest.xml

//Por último, registramos la imagen en el cloud:

root@ubuntu# euca-register name-bukket/name-prefix-bundle.manifest.xml
IMAGE    emi-ABE81004
//Este último comando informa el ID de la imagen con que fue registrado.

```

*Tabla 5.12: Creación y publicación de una imagen personalizada*

### 5.6.2 Modificación de los tipos de instancias

Con el objetivo de obtener una configuración de Cluster que emule el hardware existente, se han modificado los tipos de instancias del Cloud de acuerdo a la descripción que se realiza en la tabla 5.13.

Nombre	CPU	Memoria	Almacenamiento
<b>m1.small</b>	1	192 MB	2 GB
<b>c1.medium</b>	1	1 GB	5 GB
<b>m1.large</b>	2	2 GB	10 GB
<b>m1.xlarge</b>	8	8 GB	20 GB
<b>c1.xlarge</b>	8	45 GB	20 GB

*Tabla 5.13: Tipos de instancias para el Cluster*

Los cambios de configuración se han llevado a cabo por medio del portal de administración de Eucalyptus, en la solapa de configuración. A continuación, se corrobora la aplicación de dichos cambios, por medio de la ejecución del comando `euca-describe-availability-zones`, cuya salida se presenta en la tabla 5.14.

```

user@cloud:~$ euca-describe-availability-zones verbose
AVAILABILITYZONE      cluster1      163.10.22.167
AVAILABILITYZONE      |- vm types   free / max    cpu    ram    disk
AVAILABILITYZONE      |- m1.small   0032 / 0032   1      192    2
AVAILABILITYZONE      |- c1.medium  0032 / 0032   1      1024   5
AVAILABILITYZONE      |- m1.large   0016 / 0016   2      2048   10
AVAILABILITYZONE      |- m1.xlarge  0004 / 0004   8      8192   20
AVAILABILITYZONE      |- c1.xlarge  0002 / 0002   8      46080  20

```

*Tabla 5.14: Zona de disponibilidad del Cloud con los nuevos tipos de instancias*

### 5.6.3 Instanciación del Cluster

Una vez finalizada la creación de la imagen para los nodos del Cluster y efectuadas las modificaciones en los tipos de instancias, se ha desplegado el Cluster iniciando cuatro instancias: dos de tipo `c1.xlarge` y dos de tipo `m1.xlarge`.

A continuación, se ha dado inicio a la planificación y ejecución de las aplicaciones, tal como se describe en el capítulo 6.

# Capítulo 6

## Ejecución de algoritmos en Cluster y Cloud

---

En este capítulo se describe la ejecución de aplicaciones científicas sobre dos escenarios de ejecución: un Cluster Dedicado y un Cluster Virtual desplegado sobre el Cloud Privado.

Las aplicaciones científicas que han sido seleccionadas para su ejecución son: una solución paralela al problema de N-Reinas y algunas aplicaciones de la suite de benchmarks de sistemas paralelos NAS.

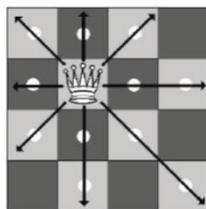
A partir de la ejecución de estas aplicaciones, se ha analizado el tiempo de ejecución de las mismas y se ha calculado el overhead introducido por la arquitectura Cloud.

Luego, se han realizado pruebas de comunicación de red y se han analizado los resultados obtenidos.

A continuación, se realiza una breve descripción de las aplicaciones mencionadas, se detallan los escenarios de ejecución y se exhiben los resultados del análisis que se ha llevado a cabo.

### 6.1 Problema N-Reinas

El problema de las *8-reinas* es un problema combinatorio clásico, el cual consiste en distribuir 8 reinas sobre un tablero de ajedrez clásico de 8x8 casillas, de manera tal que las mismas no se ataquen entre sí. Una reina puede moverse en cualquier dirección (vertical, horizontal y diagonal), sobre la cantidad deseada de casillas; esto se ve representado en la figura 6.1. No es válido ubicar dos reinas en la misma fila, columna o diagonal, pues las mismas se atacarían entre sí.



*Figura 6.1: Movimientos posibles de una reina en el tablero*

El problema de las *N-reinas* es una generalización del problema original, el cual consiste en ubicar *N* reinas en un tablero de ajedrez de *NxN* casillas (para  $N > 0$ ) [BRU97]

[DEG03]. En la figura 6.2 se ilustran dos ejemplos de tableros para  $N = 5$ , uno válido (a) y otro inválido, (b).

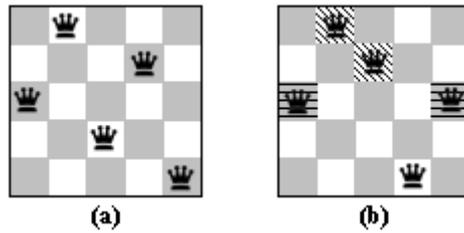


Figura 6.2: Ejemplo de tablero válido (a) e inválido (b)

Una solución inicial al problema de las N-Reinas consiste en comprobar todas las combinaciones posibles de ubicación de las reinas en el tablero, descartando aquellas combinaciones inválidas. En la tabla 6.1 se presenta la cantidad de combinaciones válidas conforme al tamaño del tablero.

Tamaño del Tablero (N)	Cantidad de Combinaciones Válidas
1	1
2	0
3	0
4	2
5	10
6	4
7	40
8	92
9	352
10	724
11	2.680
12	14.200
13	73.712
14	365.596
15	2.279.184
16	14.772.512
17	95.815.104
18	666.090.624
19	4.968.057.848
20	39.029.188.884
21	314.666.222.712
22	2.691.008.701.644
23	24.233.937.684.440
24	227.514.171.973.736
25	2.207.893.435.808.352
26	22.317.699.616.364.044

Tabla 6.1: Cantidad de soluciones de N-Reinas para  $1 < N < 26$

A partir de los resultados presentados en la tabla 6.1, se observa un incremento del tiempo de ejecución en forma exponencial, dada la cantidad de combinaciones posibles a medida que aumenta el tamaño del tablero. Por ende, es complejo trabajar con tableros donde  $N$  tiende a valores altos, por lo tanto, se introduce naturalmente la paralelización como herramienta para su resolución.

La solución paralela ubica, en forma fija, la reina en una o más filas y luego obtiene todas las posibles soluciones para tal combinación inicial. Cada procesador se encarga de resolver el problema para un subconjunto de estas combinaciones iniciales, de manera tal que el sistema completo pueda encontrar todas las posibles soluciones para el tablero utilizado. Se deben tener en cuenta dos aspectos fundamentales: la construcción de las combinaciones y la forma de distribuirlas entre los nodos del Cluster. [ROD07]

Luego de establecer las combinaciones iniciales, se utiliza el paradigma de programación Master/Worker, con el fin de distribuirlas entre los procesos, de forma tal que en conjunto encuentren todas las soluciones del tablero utilizado.

Debido a la irregularidad en la cantidad de trabajo que se debe realizar con cada combinación inicial, la distribución estática de las tareas no logra un buen balance del trabajo entre los procesadores. Esto lleva a distribuir las combinaciones en forma dinámica.

En resumen, se ha utilizado el paradigma Master/Worker con Distribución Dinámica bajo Demanda, implementado en lenguaje C y utilizando la librería de pasaje de mensajes OpenMPI.

Es importante destacar, que esta aplicación pertenece a una clase de problemas en los que el tiempo de comunicación entre procesos  $T_c$  no es significativo frente al tiempo de procesamiento  $T_p$  ( $T_p \gg T_c$ ).

## 6.2 NAS Parallel Benchmarks

Los NAS Parallel Benchmark (NPB) son un conjunto de pruebas que ha sido desarrollado, en el año 1991, por el Ames Research Center de la NASA, con el objetivo de disponer de una herramienta para evaluar el rendimiento de sistemas paralelos de cómputo de altas prestaciones (HPC).

Este conjunto de pruebas se basa en las características de cómputo y comunicación de aplicaciones que resuelven problemas de cálculos de dinámica de fluidos. NPB se encuentra conformado por ocho aplicaciones: block tridiagonal (BT), lower-upper diagonal (LU), scalar pentadiagonal (SP), conjugate gradient (CG), embarrassingly parallel (EP), 3-D fast-Fourier transform (FT), integer sort (IS) y multigrid (MG). La tabla 6.2 presenta una descripción de cada aplicación. [BAI95]

Aplicación	Descripción
<b>Embarrassingly Parallel (EP)</b>	Realiza la acumulación de estadísticas bidimensionales, a partir de la generación de gran cantidad de números aleatorios de Gauss, creados a partir de un esquema particular.
<b>MultiGrid (MG)</b>	Resuelve una ecuación de Poisson de 3 dimensiones. Realiza cuatro iteraciones de un algoritmo para obtener una solución aproximada a la ecuación de Poisson de 3 dimensiones con límites periódicos.
<b>Conjugate gradient (CG)</b>	Calcula una aproximación del menor valor propio de una matriz simétrica, definida positiva, utilizando el método del gradiente conjugado.
<b>3-D Fourier Transform (FT)</b>	Resuelve una ecuación de derivadas parciales en 3 dimensiones, utilizando la transformada rápida de Fourier directa e inversa.
<b>Integer sort (IS)</b>	Realiza la ordenación de números enteros pequeños, utilizando un algoritmo de generación de llaves secuenciales y ordenando las mismas en paralelo.
<b>LU solver (LU)</b>	Resuelve ecuaciones de diferencias finitas Navier-Stokes de 3 dimensiones, a partir de una aproximación de la factorización de un bloque triangular inferior en bloques triangular superior del esquema original.
<b>Pentadiagonal solver (SP)</b>	Resuelve múltiples sistemas independientes de ecuaciones escalares ( $N \times N \times N$ ), pentadiagonal.
<b>Block Tridiagonal (BT)</b>	Resuelve múltiples sistemas de ecuaciones ( $N \times N \times N$ ) de matrices tridiagonales con un tamaño de bloque (5x5).

*Tabla 6.2: Aplicaciones NAS Benchmarks[WAL08]*

Las aplicaciones citadas se encuentran implementadas en los lenguajes Fortran 90 y C, utilizando las librerías de comunicación basada en pasaje de mensajes MPI.

Además, este conjunto de pruebas implementa soluciones para distintos tamaños de problemas, los cuales se clasifican en clases como se presenta en la tabla 6.3.

Aplicación	Clase A	Clase B	Clase C
<b>Embarrassingly parallel (EP)</b>	$2^{28}$	$2^{30}$	$2^{32}$
<b>Multigrid (MG)</b>	$256^3$	$256^3$	$512^3$
<b>Conjugate gradient (CG)</b>	14000	75000	150000
<b>3-D FFT PDE (FT)</b>	$256^2 \times 128$	$512 \times 256^2$	$512^3$
<b>Integer sort (IS)</b>	$2^{23}$	$2^{25}$	$2^{27}$
<b>LU solver (LU)</b>	$64^3$	$102^3$	$162^3$
<b>Pentadiagonal solver (SP)</b>	$64^3$	$102^3$	$162^3$
<b>Block tridiagonal solver (BT)</b>	$64^3$	$102^3$	$162^3$

*Tabla 6.3: Tamaños de problema de NPB*

También, existen dos clases adicionales, small (S) y workstation (w), las cuales resuelven problemas de tamaño muy pequeño que se ejecutan en un único equipo.

## 6.3 Escenarios de ejecución

Con el objetivo de analizar la posibilidad de ejecución de algoritmos paralelos sobre Cloud y calcular el overhead introducido por la arquitectura, se han diseñado 2 escenarios de ejecución para aplicaciones científicas. Estos escenarios se conforman por un Cluster Dedicado y un Cluster Virtual, éste último desplegado sobre el Cloud Privado presentado en el Capítulo 5. [ROD11]

### 6.3.1 Cluster Dedicado

Este escenario presenta un Cluster heterogéneo, integrado por 4 equipos multicore e interconectados por una red Gigabit Ethernet. La especificación del hardware de estos equipos se describe en la sección 5.1.

Cada nodo del Cluster es un equipo multicore al que se le han instalado y configurado las librerías de pasaje de mensajes OpenMPI, con el fin de llevar a cabo la ejecución de las aplicaciones N-Reinas y NPB. [MPI11]

### 6.3.2 Cluster Virtual

Este escenario presenta un Cluster Virtual desplegado sobre los recursos virtuales que proporciona el Cloud Privado, como se ha presentado en el Capítulo 5.

El Cluster Virtual se encuentra conformado por cuatro instancias de VMs las cuales ejecutan una imagen de un nodo Cluster. Esta imagen fue creada y configurada con las librerías de pasaje de mensajes OpenMPI, como se ha detallado en la Sección 5.6.1.

Los nodos del Cluster Virtual corresponden a 4 instancias de VMs del Cloud (2 m1.xlarge y 2 c1.xlarge), las cuales poseen una especificación similar a los nodos del Cluster Dedicado. [ROD11]

## 6.4 Ejecución de aplicaciones

Se ha realizado la ejecución de las aplicaciones N-Reinas y NPB en ambos escenarios de ejecución, determinando el tiempo de ejecución de las mismas.

Para la ejecución de N-Reinas, se han tenido en cuenta tableros de tamaño 18x18, 19x19, 20x20 y 21x21.

En el caso de NPB, se han seleccionado de la clase C las aplicaciones embarrassingly parallel (ep.C), 3-D fast-Fourier transform (ft.C) y lower-upper diagonal (lu.C), dadas sus características de tiempo de procesamiento y comunicación.

Las aplicaciones EP, FT y LU, se caracterizan por tener una gran demanda de cómputo, requiriendo un volumen mínimo, medio y alto de comunicación, respectivamente.

Es importante destacar, que la aplicación EP es similar a N-Reinas en cuanto a que el tiempo de comunicación entre procesos no es significativo frente al tiempo de procesamiento.

En todos los casos, se ha repetido 10 veces la ejecución de las aplicaciones, obteniendo el tiempo de ejecución en cada repetición ( $T_k$ ) y calculando un promedio de las mismas, cuya fórmula se presenta en la figura 6.3.

$$T_{promedio} = \frac{\sum_{k=1}^n T_k}{n} \quad \text{para } n = 10$$

*Figura 6.3: Fórmula para el cálculo del Tiempo Promedio de ejecución*

Además, la ejecución de las aplicaciones se ha llevado a cabo en cada escenario considerando las configuraciones que presenta la tabla 6.4.

Configuración	Nº Nodos	Cluster Dedicado	Cluster Virtual
<b>4 nodos</b>	4	2 x Dell + 2 x Blade	2 x c1.xlarge + 2 x m1.xlarge
<b>2 nodos A</b>	2	2 x Dell	2 x c1.xlarge
<b>2 nodos B</b>	2	2 x Blade	2 x m1.xlarge
<b>2 nodos C</b>	2	Dell + Blade	c1.xlarge + m1.xlarge
<b>1 nodo A</b>	1	Dell	c1.xlarge
<b>1 nodo B</b>	1	Blade	m1.xlarge

*Tabla 6.4: Configuraciones de los escenarios de ejecución*

En los Anexos C y D se presentan las tablas con los tiempos de ejecución de N-Reinas y NPB, respectivamente.

## 6.5 Análisis de Resultados y Cálculo de Overhead

Una vez realizadas las ejecuciones de las aplicaciones, se ha analizado el tiempo promedio de ejecución y se ha calculado el overhead que introduce la arquitectura Cloud en cada configuración mencionada en la tabla 6.4. Además, se han representado gráficamente los resultados obtenidos.

En la figura 6.4 se describe la fórmula utilizada para determinar el porcentaje de overhead introducido por Cloud.

$$\% \text{ Overhead} = \frac{T_{\text{Promedio C.Virtual}} - T_{\text{Promedio C.Dedicado}}}{T_{\text{Promedio C.Dedicado}}} \times 100$$

Figura 6.4: Fórmula para el cálculo del Tiempo Promedio de ejecución

### 6.5.1 Overhead en N-Reinas con Tablero 18x18

N-Reinas - Tablero 18 x 18				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	42,937	43,3739	0,437	1,02%
1 nodo B	52,573	53,2817	0,709	1,35%
2 nodos A	20,111	20,8075	0,697	3,47%
2 nodos B	24,620	25,4973	0,878	3,57%
2 nodos C	22,263	22,916	0,653	2,93%
4 nodos	11,087	11,782	0,694	6,26%

Tabla 6.5: Overhead en N-Reinas con Tablero 18x18

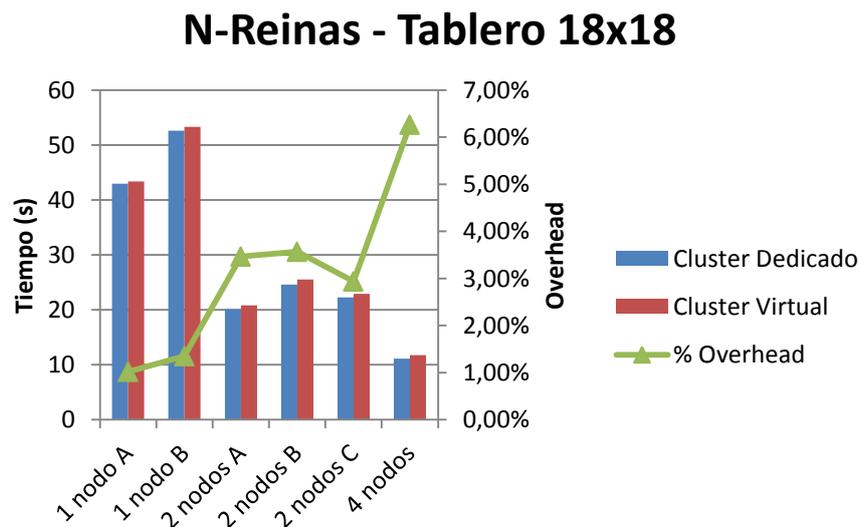


Figura 6.5: Gráfico del Overhead para N-Reinas Tablero 18x18

## 6.5.2 Overhead en N-Reinas con Tablero 19x19

N-Reinas - Tablero 19 x 19				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	329,848	332,130	2,281	0,69%
1 nodo B	403,621	407,869	4,248	1,05%
2 nodos A	153,981	155,743	1,762	1,14%
2 nodos B	188,327	191,539	3,212	1,71%
2 nodos C	170,594	172,615	2,021	1,18%
4 nodos	82,259	83,684	1,424	1,73%

Tabla 6.6: Overhead en N-Reinas con Tablero 19x19

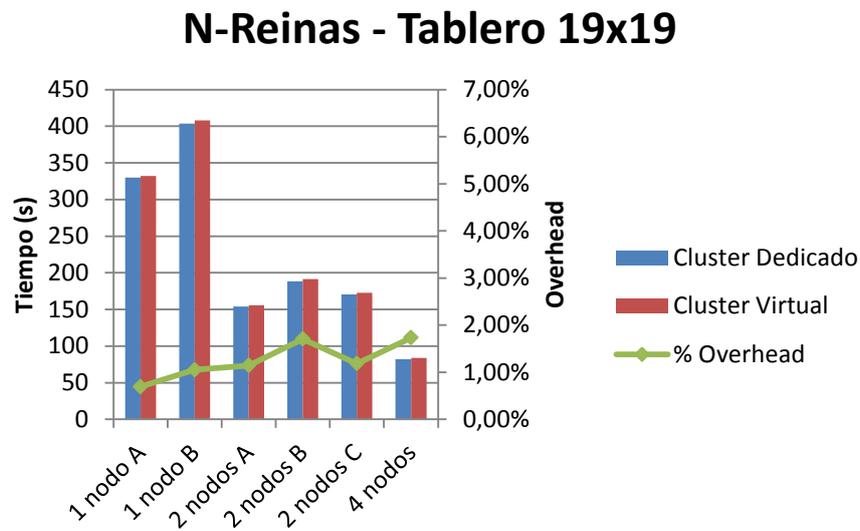


Figura 6.6: Gráfico del Overhead para N-Reinas Tablero 19x19

### 6.5.3 Overhead en N-Reinas con Tablero 20x20

N-Reinas - Tablero 20 x 20				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	2644,782	2663,091	18,309	0,69%
1 nodo B	3241,920	3269,054	27,133	0,84%
2 nodos A	1235,086	1244,541	9,456	0,77%
2 nodos B	1513,167	1527,579	14,413	0,95%
2 nodos C	1367,860	1379,689	11,828	0,86%
4 nodos	662,381	667,269	4,888	0,74%

Tabla 6.7: Overhead en N-Reinas con Tablero 20x20

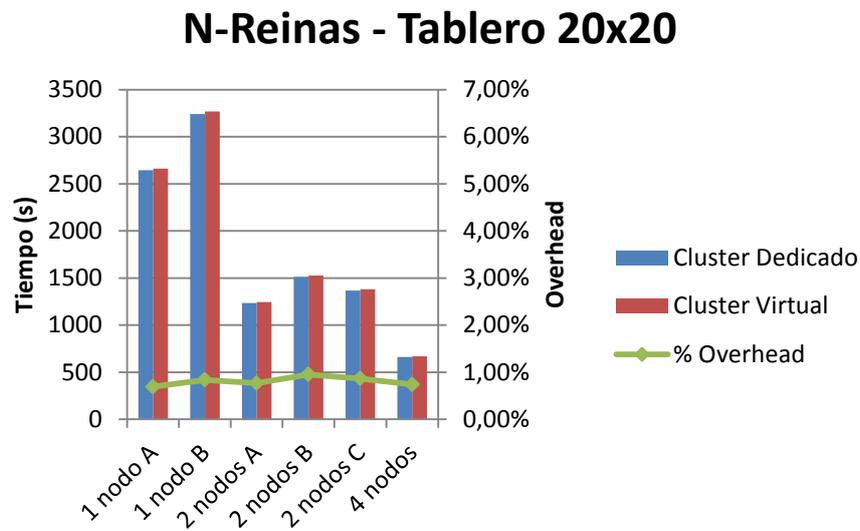


Figura 6.7: Gráfico del Overhead para N-Reinas Tablero 20x20

### 6.5.4 Overhead en N-Reinas con Tablero 21x21

N-Reinas - Tablero 21 x 21				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	22313,862	22502,196	188,334	0,84%
1 nodo B	27373,016	27568,582	195,566	0,71%
2 nodos A	10416,551	10488,593	72,042	0,69%
2 nodos B	12756,442	12868,899	112,457	0,88%
2 nodos C	11532,724	11645,106	112,382	0,97%
4 nodos	5565,153	5610,145	44,992	0,81%

Tabla 6.8: Overhead en N-Reinas con Tablero 21x21

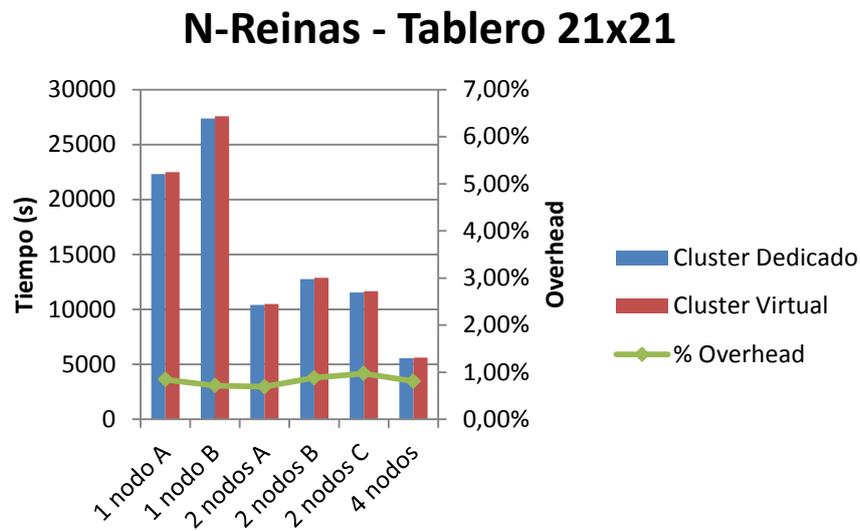


Figura 6.8: Gráfico del Overhead para N-Reinas Tablero 21x21

### 6.5.5 Resumen del Overhead en N-Reinas

Resumen del Overhead de N-Reinas						
Tablero	4 nodos	2 nodos A	2 nodos B	2 nodos C	1 nodo A	1 nodo B
18x18	6,26%	3,47%	3,57%	2,93%	1,02%	1,35%
19x19	1,73%	1,14%	1,71%	1,18%	0,69%	1,05%
20x20	0,74%	0,77%	0,95%	0,86%	0,69%	0,84%
21x21	0,81%	0,69%	0,88%	0,97%	0,84%	0,71%

Tabla 6.9: Resumen del overhead en N-Reinas

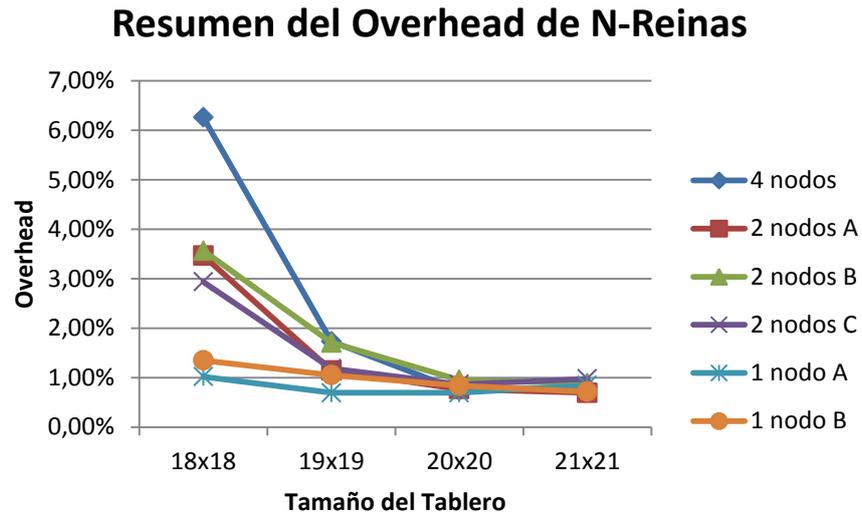


Figura 6.9: Gráfico del Overhead en N-Reinas

### 6.5.6 Overhead en NPB ep.C

Embarassingly Parallel (EP) - Clase C				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	47,271	48,274	1,003	2,12%
1 nodo B	61,805	63,917	2,112	3,42%
2 nodos A	23,728	24,511	0,783	3,30%
2 nodos B	31,091	32,726	1,635	5,26%
2 nodos C	30,819	32,22	1,401	4,55%
4 nodos	15,520	16,335	0,815	5,25%

Tabla 6.10: Overhead en NPB ep.C

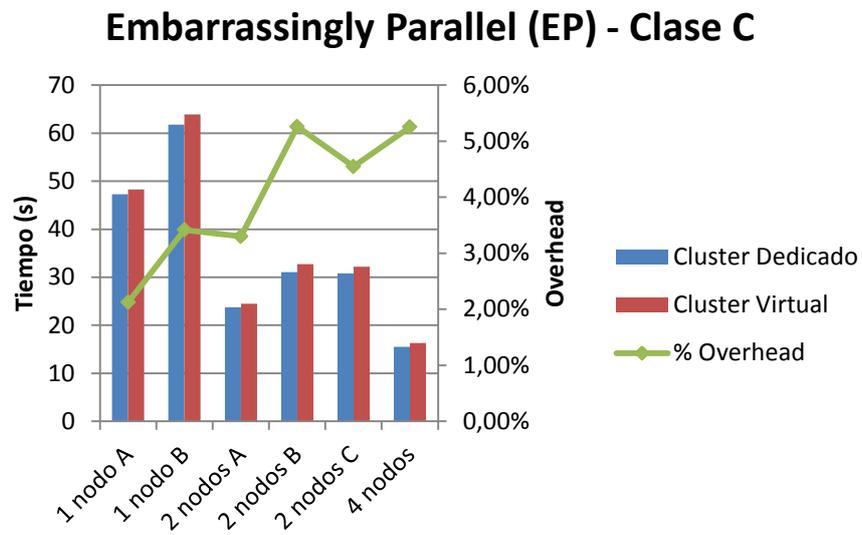


Figura 6.10: Gráfico del Overhead en NPB ep.C

### 6.5.7 Overhead en NPB ft.C

3-D fast-Fourier Transform (FT) - Clase C				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	66,950	70,601	3,651	5,45%
1 nodo B	162,004	162,159	0,155	0,10%
2 nodos A	137,328	221,100	83,772	61,00%
2 nodos B	170,365	371,395	201,030	118,00%
2 nodos C	166,024	319,865	153,841	92,66%
4 nodos	215,251	289,818	74,567	34,64%

Tabla 6.11: Overhead en NPB ft.C

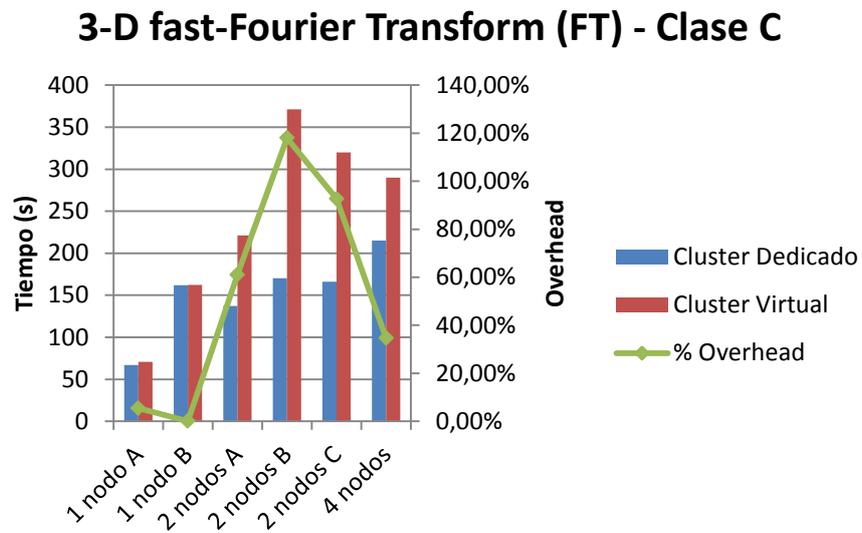


Figura 6.11: Gráfico del Overhead en NPB ft.C

### 6.5.8 Overhead en NPB lu.C

Lower-Upper diagonal (LU) - Clase C				
Configuración	Cluster Dedicado	Cluster Virtual	Diferencia	% Overhead
1 nodo A	196,268	200,288	4,020	2,05%
1 nodo B	481,481	482,675	1,194	0,25%
2 nodos A	103,461	146,977	43,516	42,06%
2 nodos B	203,441	297,833	94,392	46,40%
2 nodos C	208,851	296,364	87,513	41,90%
4 nodos	157,816	346,109	188,293	119,31%

Tabla 6.12: Overhead en NPB lu.C

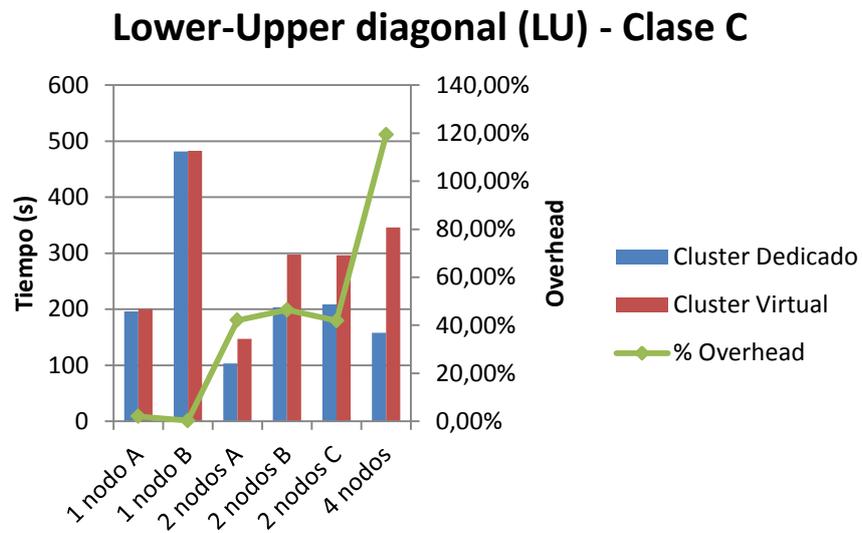


Figura 6.12: Gráfico del Overhead en NPB lu.C

### 6.5.9 Resumen del Overhead en NPB

Resumen del Overhead en NAS Parallel Benchmarks						
	4 nodos	2 nodos A	2 nodos B	2 nodos C	1 nodo A	1 nodo B
<b>ep.C</b>	5,25%	3,30%	5,26%	4,55%	2,12%	3,42%
<b>ft.C</b>	34,64%	61,00%	118,00%	92,66%	5,45%	0,10%
<b>lu.C</b>	119,31%	42,06%	46,40%	41,90%	2,05%	0,25%

Tabla 6.13: Resumen del overhead en NPB

### Resumen del Overhead NAS Parallel Benchmarks

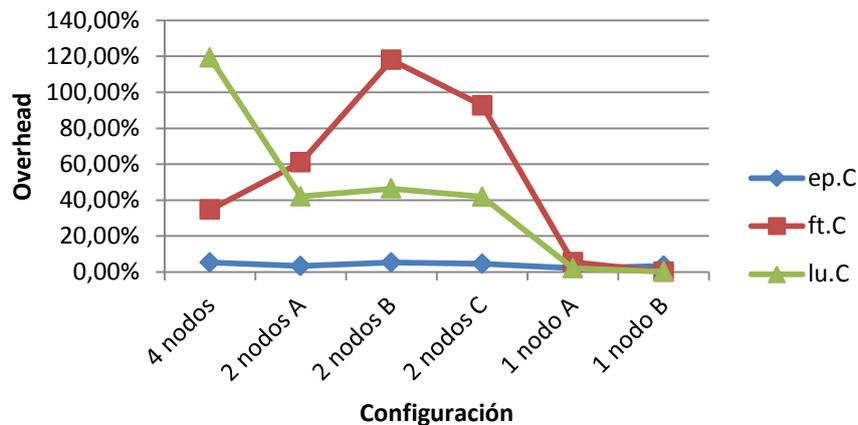


Figura 6.13: Gráfico del overhead en NPB

## 6.6 Pruebas de comunicación

A partir del análisis del overhead obtenido en la ejecución de aplicaciones que poseen una alta demanda de comunicación de red, se han efectuado pruebas de comunicación utilizando una aplicación que hace uso de las librerías de pasajes de mensajes OpenMPI, con el objetivo de determinar si existe una degradación en la red virtualizada entre las instancias del Cloud, con respecto a la red física subyacente.

Se ha ejecutado la aplicación mencionada con diferentes tamaños de mensajes, desde 1 Byte a 1 GigaByte; se ha repetido dicha ejecución 200 veces con el fin de evitar inconsistencias. Los escenarios seleccionados han sido: 2 Nodos A, 2 Nodos B y 2 Nodos C.

### 6.6.1 Pruebas de red entre 2 nodos A

Pruebas de red entre 2 nodos A					
Tamaño del Mensaje	Físicas (Mbps)	Virtuales (Mbps)	Diferencia (Mbps)	Degradación	
1.00 byte	0,3	0,1	0,2	76,54%	
2.00 bytes	0,5	0,1	0,4	76,54%	
4.00 bytes	1,0	0,2	0,7	76,15%	
8.00 bytes	2,0	0,5	1,5	76,36%	
16.00 bytes	3,9	0,9	3,0	76,15%	
32.00 bytes	7,9	1,9	6,0	76,43%	
64.00 bytes	15,5	3,7	11,8	76,05%	
128.00 bytes	29,2	7,4	21,7	74,52%	
256.00 bytes	55,0	14,7	40,3	73,21%	
512.00 bytes	95,3	28,2	67,1	70,40%	
1.00 Kbyte	156,3	52,3	104,0	66,56%	
2.00 Kbytes	226,4	64,3	162,1	71,60%	
4.00 Kbytes	330,7	110,8	219,9	66,49%	
8.00 Kbytes	459,6	184,4	275,2	59,88%	
16.00 Kbytes	497,0	284,7	212,3	42,71%	
32.00 Kbytes	697,4	452,9	244,5	35,05%	
64.00 Kbytes	711,7	431,0	280,7	39,44%	
128.00 Kbytes	795,2	605,5	189,7	23,86%	
256.00 Kbytes	834,7	729,4	105,3	12,62%	
512.00 Kbytes	865,3	801,5	63,8	7,37%	
1.00 Mbyte	882,4	846,1	36,3	4,11%	
2.00 Mbytes	890,5	866,1	24,4	2,74%	
4.00 Mbytes	894,0	880,8	13,2	1,48%	
8.00 Mbytes	895,8	887,6	8,2	0,91%	
16.00 Mbytes	896,7	890,6	6,1	0,68%	
32.00 Mbytes	897,2	888,3	8,9	0,99%	
64.00 Mbytes	897,5	886,5	11,0	1,22%	
128.00 Mbytes	897,4	881,6	15,9	1,77%	
256.00 Mbytes	897,6	888,1	9,4	1,05%	
512.00 Mbytes	897,7	892,1	5,5	0,62%	
1.00 Gbyte	897,7	892,5	5,2	0,58%	

*Tabla 6.14: Pruebas de red entre 2 nodos A*

### Pruebas de red entre 2 nodos A

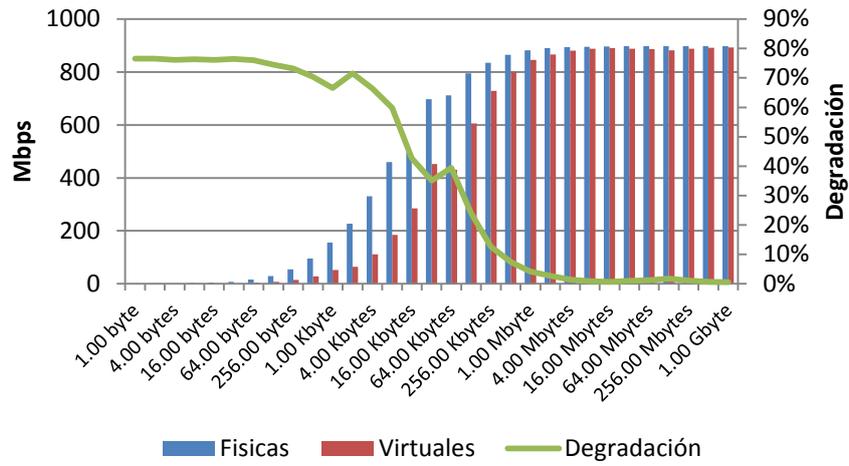


Figura 6.14: Gráfico de las pruebas de red entre 2 nodos A

## 6.6.2 Pruebas de red entre 2 nodos B

Pruebas de red entre 2 nodos B					
Tamaño del Mensaje	Físicas (Mbps)	Virtuales (Mbps)	Diferencia (Mbps)	Degradación	
1.00 byte	0,1	0,0	0,1	86,64%	
2.00 bytes	0,3	0,0	0,3	86,68%	
4.00 bytes	0,6	0,1	0,5	86,45%	
8.00 bytes	1,2	0,2	1,0	86,40%	
16.00 bytes	2,4	0,3	2,1	86,64%	
32.00 bytes	4,8	0,6	4,2	87,48%	
64.00 bytes	9,5	1,3	8,2	86,73%	
128.00 bytes	18,4	2,5	15,9	86,45%	
256.00 bytes	35,8	4,9	30,9	86,20%	
512.00 bytes	57,4	9,8	47,7	82,96%	
1.00 Kbyte	98,3	18,9	79,4	80,75%	
2.00 Kbytes	153,2	35,2	118,0	77,05%	
4.00 Kbytes	251,0	66,2	184,8	73,62%	
8.00 Kbytes	404,5	124,3	280,3	69,28%	
16.00 Kbytes	519,8	223,2	296,5	57,05%	
32.00 Kbytes	623,4	340,8	282,6	45,33%	
64.00 Kbytes	627,4	221,9	405,5	64,63%	
128.00 Kbytes	741,3	349,6	391,7	52,84%	
256.00 Kbytes	807,3	474,8	332,5	41,19%	
512.00 Kbytes	849,4	584,2	265,2	31,22%	
1.00 Mbyte	874,5	607,3	267,1	30,55%	
2.00 Mbytes	884,9	674,7	210,2	23,76%	
4.00 Mbytes	891,0	679,2	211,8	23,77%	
8.00 Mbytes	894,5	689,6	205,0	22,91%	
16.00 Mbytes	896,2	686,1	210,1	23,44%	
32.00 Mbytes	897,0	684,6	212,4	23,68%	
64.00 Mbytes	897,4	680,7	216,7	24,15%	
128.00 Mbytes	897,2	685,8	211,4	23,56%	
256.00 Mbytes	897,5	683,5	213,9	23,84%	
512.00 Mbytes	897,7	534,8	362,9	40,43%	
1.00 Gbyte	897,7	519,6	378,1	42,12%	

*Tabla 6.15: Pruebas de red entre 2 nodos B*

## Pruebas de red entre 2 nodos B

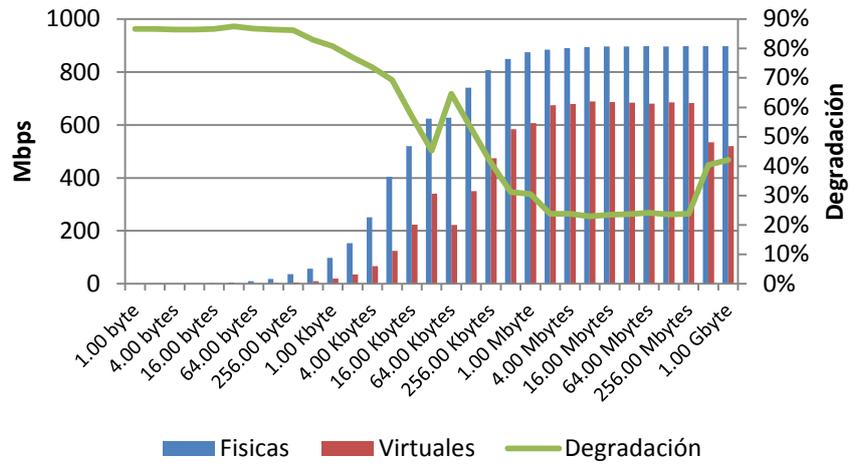
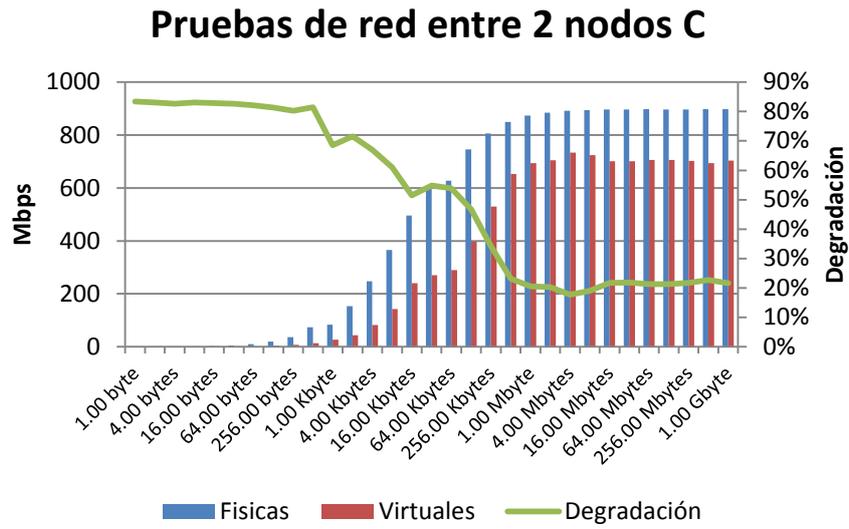


Figura 6.15: Gráfico de las pruebas de red entre 2 nodos B

### 6.6.3 Pruebas de red entre 2 nodos C

Pruebas de red entre 2 nodos C					
Tamaño del Mensaje	Físicas (Mbps)	Virtuales (Mbps)	Diferencia (Mbps)	Degradación	
1.00 byte	0,2	0,0	0,1	83,36%	
2.00 bytes	0,3	0,1	0,3	83,12%	
4.00 bytes	0,6	0,1	0,5	82,61%	
8.00 bytes	1,3	0,2	1,1	83,06%	
16.00 bytes	2,6	0,4	2,2	82,82%	
32.00 bytes	5,1	0,9	4,2	82,63%	
64.00 bytes	10,0	1,8	8,2	82,05%	
128.00 bytes	19,1	3,6	15,6	81,32%	
256.00 bytes	36,5	7,2	29,3	80,22%	
512.00 bytes	73,7	13,7	60,0	81,44%	
1.00 Kbyte	84,0	26,4	57,6	68,58%	
2.00 Kbytes	153,2	43,5	109,7	71,59%	
4.00 Kbytes	248,0	81,8	166,2	67,02%	
8.00 Kbytes	365,5	142,5	223,0	61,00%	
16.00 Kbytes	496,0	240,8	255,2	51,45%	
32.00 Kbytes	599,5	270,4	329,1	54,89%	
64.00 Kbytes	627,4	289,3	338,1	53,89%	
128.00 Kbytes	745,7	397,2	348,5	46,73%	
256.00 Kbytes	806,3	530,4	275,9	34,22%	
512.00 Kbytes	849,5	653,1	196,4	23,12%	
1.00 Mbyte	874,0	694,1	180,0	20,59%	
2.00 Mbytes	884,9	705,0	179,9	20,33%	
4.00 Mbytes	891,2	733,2	158,0	17,73%	
8.00 Mbytes	894,4	724,4	170,0	19,01%	
16.00 Mbytes	896,1	701,4	194,7	21,73%	
32.00 Mbytes	896,9	700,8	196,2	21,87%	
64.00 Mbytes	897,4	705,8	191,6	21,35%	
128.00 Mbytes	896,4	705,4	191,0	21,31%	
256.00 Mbytes	897,1	702,6	194,5	21,68%	
512.00 Mbytes	897,4	693,8	203,7	22,69%	
1.00 Gbyte	897,6	703,9	193,7	21,58%	

*Tabla 6.16: Pruebas de red entre 2 nodos C*



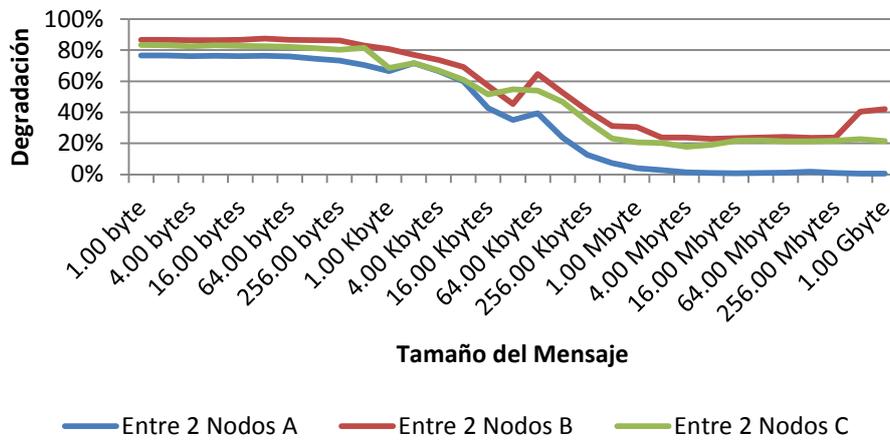
**Figura 6.16:** Gráfico de las pruebas de red entre 2 nodos C

#### 6.6.4 Resumen de la degradación de la red virtualizada

Tamaño del Mensaje	Entre 2 Nodos A	Entre 2 Nodos B	Entre 2 Nodos C
1.00 byte	77%	87%	83%
2.00 bytes	77%	87%	83%
4.00 bytes	76%	86%	83%
8.00 bytes	76%	86%	83%
16.00 bytes	76%	87%	83%
32.00 bytes	76%	87%	83%
64.00 bytes	76%	87%	82%
128.00 bytes	75%	86%	81%
256.00 bytes	73%	86%	80%
512.00 bytes	70%	83%	81%
1.00 Kbyte	67%	81%	69%
2.00 Kbytes	72%	77%	72%
4.00 Kbytes	66%	74%	67%
8.00 Kbytes	60%	69%	61%
16.00 Kbytes	43%	57%	51%
32.00 Kbytes	35%	45%	55%
64.00 Kbytes	39%	65%	54%
128.00 Kbytes	24%	53%	47%
256.00 Kbytes	13%	41%	34%
512.00 Kbytes	7%	31%	23%
1.00 Mbyte	4%	31%	21%
2.00 Mbytes	3%	24%	20%
4.00 Mbytes	1%	24%	18%
8.00 Mbytes	1%	23%	19%
16.00 Mbytes	1%	23%	22%
32.00 Mbytes	1%	24%	22%
64.00 Mbytes	1%	24%	21%
128.00 Mbytes	2%	24%	21%
256.00 Mbytes	1%	24%	22%
512.00 Mbytes	1%	40%	23%
1.00 Gbyte	1%	42%	22%

*Tabla 6.17: Resumen de la degradación de la red virtualizada*

## Resumen de la degradación de la Red Virtualizada



**Figura 6.17:** Gráfico resumen de la degradación de la red virtualizada



# Capítulo 7

## Conclusiones

---

El objetivo principal de esta Tesina ha sido llevar a cabo el despliegue de un Cloud Privado con el propósito de analizar e investigar la arquitectura Cloud Computing para ambientes de cómputo científico.

Para alcanzar tal fin, se ha desplegado un Cloud Privado utilizando el gestor Eucalyptus. Además, se ha confeccionado una imagen que permite instanciar nodos de un Cluster Virtual sobre el que se han llevado a cabo diversas ejecuciones de aplicaciones científicas.

Luego de efectuar el análisis de los resultados obtenidos en las ejecuciones de cada aplicación, se puede concluir que la arquitectura Cloud Computing puede ser utilizada para la ejecución de algunas clases de aplicaciones paralelas de cómputo científico. Además, brinda los beneficios de asignación dinámica de recursos (Clusters Virtuales bajo demanda), ágil administración, ambientes personalizados para aplicaciones y seguridad.

Cuando las aplicaciones se caracterizan por tener mayor tiempo de procesamiento que de comunicación, Cloud se presenta como una alternativa viable para la ejecución de este tipo de soluciones. Se puede destacar que el overhead introducido por la arquitectura, posee una incidencia promedio del 3% sobre el tiempo de ejecución.

Teniendo en cuenta lo antes mencionado, se puede concluir que el overhead introducido por la arquitectura, en este tipo de soluciones, es tolerable cuando no se disponen los recursos para la ejecución en un Cluster Dedicado. También es importante destacar, que en entornos de Cómputo de Altas Prestaciones, un overhead de esta característica puede hacer no viable la utilización de esta arquitectura.

Para el caso de las aplicaciones que se caracterizan por tener un tiempo de comunicación significativo sobre el tiempo de procesamiento, el overhead que introduce la arquitectura es excesivo. Tal incremento puede derivar de la degradación que sufre la red virtualizada con respecto a la red física subyacente.

Por otro lado, de las pruebas de comunicación de red que se han efectuado, es importante destacar la variabilidad del ancho de banda disponible entre instancias de acuerdo al tamaño del mensaje utilizado y el hardware subyacente. Es conveniente contemplar estos factores al momento de utilizar Cloud Públicos, como ser Amazon EC2, los cuales realizan la asignación de instancias en equipos heterogéneos que a su vez pueden encontrarse en diversos segmentos de red. De la misma forma, en los casos en que se desea realizar el despliegue de un Cloud Privado es necesario contemplarlo al momento de seleccionar el hardware a utilizar.

## Capítulo 8

# Líneas de trabajo futuro

---

Analizar e investigar el despliegue de un Cloud Privado con Eucalyptus, utilizando diversos software de virtualización, como ser Xen o VMware. Sobre dicho despliegue efectuar nuevamente las pruebas de rendimiento.

Investigar los orígenes de la degradación de la red virtualizada y las diversas técnicas para la virtualización de la misma.

Estudiar la incidencia del tamaño del mensaje en algoritmos que se ejecutan sobre el Cloud y la factibilidad de optimización.

Analizar e investigar esquemas de planificación predictiva en Cloud, en base al conocimiento previo de las características de las aplicaciones y el consumo de la arquitectura.

Analizar e investigar la reducción del consumo energético en Cloud, apagando los nodos ociosos, y encendiendo los mismos a medida que son requeridos.



# Referencias Bibliográficas

---

- [CAC10] Cáceres, J., Vaquero, L. M., Rodero-Merino, L., Polo, Á., Hierro, J. J.: Service Scalability Over the Cloud. In: Furht, B., Escalante, A. (eds.), Handbook of Cloud Computing, pp. 357—377. Springer, USA. (2010)
- [ROD07] Rodriguez, I. P., Pousa, A., Pettoruti, J.E., Chichizola, F., Naiouf, M., De Giusti, L., De Giusti, A.: Estudio del overhead en la migración de algoritmos paralelos de cluster y multicluster a GRID. In: XIII° Congreso Argentino de Ciencias de la Computación (CACIC 2007) Proceedings. Argentina. (2007)
- [DEG07] De Giusti, A., Naiouf, M., Pettoruti, J. E., Pousa, A., Rodríguez, I. P., De Giusti, L., Chichizola, F., Ardenghi, J., Bertogna, L., Printista, M.: Parallel Algorithms on Multi-Cluster Architectures using GRID Middleware. Experiences in Argentine Universities”. In: Garcia Tobio, J., Doallo Biempica, R., López Cabido, I., Martín Santamaría, M. J., Gómez Tato, A., Touriño Dominguez, J. (eds.), 1st Iberian Grid Infrastructure Conference (IBERGRID) Proceedings, pp. 322—332. Spain. (2007)
- [FOS02] Foster, I.: What is the Grid? A Three Point Checklist. In: Grid Today, Volume 1, Issue 6, pp. 22--25. USA. (2002)
- [ROD11] Rodriguez, I., Pettoruti, J.E., Chichizola, F., De Giusti, A.: Despliegue de un Cloud Privado para entornos de cómputo científico. In: XI° Workshop de Procesamiento Paralelo y Distribuido (WPPD), XVII° Congreso Argentino de Ciencias de la Computación (CACIC 2011) Proceedings. La Plata, Argentina. (2011)
- [VAQ09] Vaquero, L. M., Rodero-Merino, L., Cáceres, J., Lindner, M.: A Break in the Clouds: Towards a Cloud Definition. In: ACM SIGCOMM Computer Communication Review, Volume 39, Issue 1, pp. 50--55. USA. (2009)
- [VEL09] Velte, A.T., Velte, T.J., Elsenpeter, R.: Cloud Computing: A Practical Approach, McGraw Hill Professional. (2009)
- [KEN10] Kena, E. K., Guerrero, A. C., Bernal, I. M.: Implementación de un prototipo de Cloud Computing de modelo privado para ofrecer Infraestructura como Servicio (IaaS). Ecuador. In: XXIII Jornadas en Ingeniería Eléctrica y Electrónica. Ecuador. (2010)
- [CHE10] Chen, X., Wills, G. B., Gilbert, L., Bacigalupo, D.: TecRes Report: Using Cloud for Research: a Technical Review. In: Computing, pp. 1--52. UK. (2010)

- [YOU08] Youseff, L., Butrico, M., Da Silva, D.: Toward a Unified Ontology of Cloud Computing. In: Grid Computing Environments Workshop 2008, pp. 1--10. USA. (2008)
- [VMW11] History of Virtualization. <http://www.vmware.com>. Agosto 2011.
- [VMW07] VMware: Understanding full virtualization, paravirtualization and hardware assist. Technical Report. <http://www.vmware.com/resources/techresources/1008>. USA. (2007)
- [GOL74] Popek, G.J., Goldberg, R.P.: Formal Requirements for Virtualizable Third Generation Architectures. In: Communications in the ACM, Volume 17, Number 7, pp. 412--421. USA. (1974)
- [CHE08] Chen, W., Lu, H., Shen, L., Wang, Z., Xiao, N., Chen, D.: A Novel Hardware Assisted Full Virtualization Technique. In: Wang, G. (eds.), IEEE ICYCS The 9<sup>th</sup> International Conference for Young Computer Scientists Proceedings, pp. 1292--1297. China. (2008)
- [NUS09] Nussbaum, L., Anhalt, F., Olivier, M., Gelas, J.: Linux-based virtualization for HPC clusters. In: Montreal Linux Symposium (2009), pp. 221—234. Canada. (2009)
- [XEN08] Xen Architecture Overview. Version 1.2. <http://www.xen.org>. (2008)
- [KVM09] KVM – Kernel Based Virtual Machine. Red Hat. White Paper. <http://www.redhat.com>. (2009)
- [EC211a] Amazon Elastic Compute Cloud User Guide. <http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide>. Octubre 2011.
- [AS311] Amazon Simple Storage Service (Amazon S3). <http://aws.amazon.com/es/s3/>. Octubre 2011.
- [EC211b] Amazon Elastic Compute Cloud (Amazon EC2). <http://aws.amazon.com/ec2/>. Octubre 2011.
- [EC211c] Amazon EC2 Pricing. <http://aws.amazon.com/ec2/pricing/>. Octubre 2011.
- [EC211d] Amazon EC2 Service Level Agreement. <http://aws.amazon.com/es/ec2-sla/>. Octubre 2011.
- [NUR08] Nurmi, D., Wolski, R., Grzegorzczak, C., Obertelli, G., Soman, S., Youseff, L., Zagorodnov, D.: Eucalyptus: A Technical Report on an Elastic Utility Computing Architecture Linking Your Programs to Useful Systems. Technical Report, UCSB Computer Science Technical Report Number 2008-10. Santa Barbara, USA. (2008).

- [NUR09] Nurmi, D., Wolski, R., Grzegorzczak, C., Obertelli, G., Soman, S., Youseff, L., Zagorodnov, D.: The Eucalyptus Open-Source Cloud-Computing System. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID '09), pp. 124-131, IEEE Computer Society, Washington (2009).
- [SHA10] Shafer, J.: I/O Virtualization Bottlenecks in Cloud Computing Today. In: 2<sup>nd</sup> Workshop on I/O Virtualization. Pittsburgh, USA. (2010)
- [WAR09] Wardley, S., Goyer, E., Barcet, N.: Ubuntu Enterprise Cloud Architecture. Technical Report, Canonical. (2009)
- [EUC11] Eucalyptus Administrator's Guide (2.0). <http://open.eucalyptus.com>. Agosto 2011.
- [EUC11a] Eucalyptus User's Guide (2.0). <http://open.eucalyptus.com>. Agosto 2011.
- [HYB11] HybridFox sitio del proyecto. <http://code.google.com/p/hybridfox/>. Agosto 2011.
- [UEC10] Johnson, D., Kiran, M., Murthy, R., Suseendran, R.B., Yogesh, G.: Eucalyptus Beginner's Guide – UEC Edition. CSS Corp. Diciembre 2010.
- [EUC11b] Eucalyptus Network Configuration (2.0). <http://open.eucalyptus.com>. Agosto 2011.
- [MPI11] OpenMPI Web Site. <http://www.open-mpi.org>. Agosto 2011.
- [EUC11c] Managing Eucalyptus Images (2.0). <http://open.eucalyptus.com>. Agosto 2011.
- [BRU75] Bruen, A., Dixon, R.: The n-queens problem. In: Discrete mathematics, volume 12, pp. 393--395. (1975)
- [DEG03] De Giusti, L., Novarini, P., Naiouf, M., De Giusti, A.: Parallelization of the N-queens problem. Load unbalance analysis. In: IV<sup>o</sup> Workshop de Procesamiento Paralelo y Distribuido (WPPD), IX<sup>o</sup> Congreso Argentino de Ciencias de la Computación (CACIC 2003) Proceedings. La Plata, Argentina. (2003)
- [BAI95] Bailey, D., Harris, T., Saphir, W., van der Wijngaart, R., Woo, A., Yarrow, M.: The NAS Parallel Benchmarks 2.0. Technical Report NAS-95-020. USA. (1995)
- [WAL08] Walker, E.: Benchmarking Amazon EC2 for high-performance scientific computing. In: ;login: The USENIX Magazine, Vol. 33, No. 5, pp. 18--23. USA. (2008)



# Anexo A

## SLA de Amazon EC2

---

En este anexo se transcribe el acuerdo de nivel de servicio (SLA) de Amazon EC2, el cual debe ser aceptado por el usuario previo al uso del servicio.

**Fecha de entrada en vigor: 23 de octubre de 2008**

Este Contrato de nivel de servicio de Amazon EC2 ("SLA") es una política que rige el uso de Amazon Elastic Compute Cloud ("Amazon EC2") en virtud de los términos y condiciones del Contrato de cliente de Amazon Web Services (el "Contrato de AWS") entre Amazon Web Services, LLC ("AWS", "a nosotros" o "nosotros") y los usuarios de los servicios de AWS ("usted"). Este SLA se aplica de forma independiente a cada cuenta que utilice Amazon EC2. A menos que se indique lo contrario en el presente documento, este SLA está sujeto a los términos y condiciones del Contrato de AWS, y los términos en mayúsculas tendrán el significado indicado en el Contrato de AWS. Nos reservamos el derecho a cambiar los términos de este SLA de acuerdo con el Contrato de AWS.

**Compromiso de servicio**

AWS utilizará los esfuerzos comerciales razonables para que Amazon EC2 esté disponible con un Porcentaje de tiempo de actividad anual (definido a continuación) mínimo del 99,95% durante el Año de servicio. En caso de que Amazon EC2 no cumpla con el compromiso de Porcentaje de tiempo de actividad anual, tendrá derecho a recibir un Crédito de servicio tal y como se define a continuación.

**Definiciones**

"Año de servicio" son los 365 días anteriores a partir de la fecha de una reclamación de SLA.

El "Porcentaje de tiempo de actividad anual" se calcula restando a 100% el porcentaje de periodos de 5 minutos durante el Año de servicio en el que Amazon EC2 se encontró en el estado de "Región no disponible". Si ha estado utilizando Amazon EC2 durante un periodo de tiempo inferior a 365 días, su Año de servicio seguirán siendo los 365 días anteriores, pero se considerará que los días anteriores a su uso del servicio han tenido una Disponibilidad de región del 100%. Todo tiempo de inactividad que tenga lugar antes de una reclamación de Crédito de

servicio con éxito no podrá utilizarse para futuras reclamaciones. Las mediciones de Porcentaje de tiempo de actividad anual excluyen los tiempos de inactividad derivados de forma directa o indirecta de cualquier Exclusión de SLA de Amazon EC2 SLA (definido a continuación).

“Región no disponible” y “No disponibilidad de la región” significan que más de una Zona de disponibilidad en la que está ejecutando una instancia, dentro de la misma Región, está “No disponible” para usted.

“No disponible” significa que todas sus instancias en ejecución no tienen conectividad externa durante un periodo de cinco minutos, y que no podrá ejecutar instancias de sustitución.

El “Periodo de crédito apto” es un único mes, y hace referencia al ciclo de facturación mensual en el que el evento de Región no disponible más reciente incluido en la reclamación de SLA tuvo lugar.

Un “Crédito de servicio” es un crédito en dólares, calculado tal y como se establece a continuación, que podríamos abonar a una cuenta Amazon EC2 que cumpliera determinadas condiciones.

### **Compromisos de servicio y Créditos de servicio**

Si el Porcentaje de tiempo de actividad anual de un cliente cae por debajo del 99,95% durante el Año de servicio, dicho cliente optará a recibir un Crédito de servicio equivalente al 10% de su factura (excluyendo los pagos únicos realizados por Instancias reservadas) durante el Periodo de crédito apto. Para realizar una reclamación, el cliente no tiene por qué esperar 365 días a partir del día en el que empezaron a utilizar el servicio ni 365 días desde su última reclamación realizada correctamente. El cliente podrá realizar una reclamación en cualquier momento en el que su Porcentaje de tiempo de actividad anual durante los 365 días en curso sea inferior al 99,95%.

Aplicaremos los Créditos de servicio únicamente a pagos futuros a Amazon EC2 que tenga que realizarnos, siempre que podamos emitir el Crédito de servicio a la tarjeta de crédito que utilizó para pagar el servicio Amazon EC2 durante el ciclo de facturación en el que se produjo el error. Los Créditos de servicio no le otorgarán derecho a reembolsos ni ningún otro tipo de pagos por parte de AWS. Un Crédito de servicio será aplicable y se emitirá únicamente si la cantidad de crédito del ciclo

de facturación mensual correspondiente es superior a un dólar (1 \$ USD). Los Créditos de servicio no podrán transferirse ni aplicarse a ninguna otra cuenta. A menos que se indique lo contrario en el Contrato de AWS, su único remedio exclusivo de no disponibilidad o no funcionamiento de Amazon EC2 o cualquier otro fallo a la hora de ofrecerle el servicio Amazon EC2 es la recepción de un Crédito de servicio (si cumple ciertas condiciones), de acuerdo con los términos y condiciones de este SLA o la finalización de su uso de Amazon EC2.

#### **Solicitud de crédito y procedimientos de pago**

Para recibir un Crédito de servicio deberá enviar una solicitud a través de correo electrónico a `aws-sla-request @ amazon.com`. Para poder optar al Crédito de servicio, la solicitud de crédito deberá (i) incluir su número de cuenta en el asunto del mensaje de correo electrónico (el número de cuenta se encuentra en la parte superior de la página Actividad de la cuenta de AWS); (ii) incluir, en el cuerpo del correo electrónico, las fechas y horas en las que tuvo lugar cada uno de los incidentes de Región no disponible que afirma ha sufrido, incluyendo los identificadores de las instancias que estaban en ejecución y que se vieron afectadas durante el momento en el que se produjo cada incidente; (iii) incluir sus registros de solicitudes a servidor que documentan los errores y que corroboran la interrupción que reclama (toda información confidencial o delicada de estos registros deberá eliminarse o sustituirse por asteriscos); y (iv) nosotros debemos recibirla dentro de los treinta (30) días laborales del último incidente del que se informa en la reclamación de SLA. Si el Porcentaje de tiempo de actividad anual de dicha solicitud es confirmado por nosotros y es inferior al 99,95% en lo referente al Año de servicio, le emitiremos el Crédito de servicio durante el ciclo de facturación posterior al mes en el que se produjo el error. Si no nos proporciona la solicitud y otra información, tal y como se establece anteriormente, no optará a recibir un Crédito de servicio.

#### **Exclusiones del SLA de Amazon EC2**

El Compromiso de servicio no se aplicará a ninguna falta de disponibilidad, suspensión o finalización de Amazon EC2, ni a ningún otro problema de rendimiento de Amazon EC2: (i) derivados de las Suspensiones de servicio descritas en la Sección 7.1 del Contrato de AWS; (ii) provocadas por factores ajenos a nuestro control razonable, incluyendo eventos de fuerza mayor, problemas de acceso a Internet o cualquier otro problema relacionado que se encuentre fuera del punto de demarcación de Amazon EC2; (iii) derivado de acciones u omisiones por su parte o de cualquier otra tercera parte; (iv) derivados de sus equipos,

software o cualquier otra tecnología y/o equipos, software u otra tecnología de terceros (que no sean equipos de terceros bajo nuestro control directo); (v) derivados de fallos de instancias individuales no atribuibles a No disponibilidad de la región; o (vi) derivados de nuestra suspensión y finalización de sus derechos de uso de Amazon EC2 de acuerdo con el Contrato de AWS (denominados, de forma colectiva, las “Exclusiones de SLA de Amazon EC2”). Si la disponibilidad se ve afectada por otros factores diferentes a los que se mencionan explícitamente en este contrato, emitiremos un Crédito de servicio considerando tales factores según nuestro único criterio.

*Tabla A.1: SLA de Amazon EC2 (Octubre 2011)*

# Anexo B

## Tabla de Comandos

### Euca2ools en español

---

Dada la escasa documentación existente en español, acerca de las herramientas que conforman Euca2ools, se ha confeccionado una tabla que resume la funcionalidad y sintaxis de cada comando. La misma se presenta a continuación.

Comando	Descripción	Sintaxis
<b>euca-add-group</b>	Crea un nuevo grupo de seguridad.	<code>euca-add-group -d, --description group_description [-h, --help] [--version] [--debug] group_name</code>
<b>euca-add-keypair</b>	Crea un nuevo par de llaves para su uso con instancias.	<code>euca-add-keypair [-h, --help] [--version] [--debug] keypair_name</code>
<b>euca-allocate-address</b>	Asigna una dirección IP pública.	<code>euca-allocate-address [-h, --help] [--version] [--debug]</code>
<b>euca-associate-address</b>	Asocia una instancia con una dirección IP pública.	<code>euca-associate-address -i, --instance instance_id [-h, --help] [--version] [--debug] ip</code>
<b>euca-attach-volume</b>	Adjunta un volumen a una instancia.	<code>euca-attach-volume -i, --instance instance_id -d, --device device [-h, --help] [--version] [--debug] volume_id</code>
<b>euca-authorize</b>	Agrega una nueva regla a un grupo de seguridad.	<code>euca-authorize [-P, --protocol protocol] [-p, --port-range port_range] [-t, --icmp-type-code type:code] [-o, --source-group source_group] [-u, --source-group-user source_group_user] [-s, --source-subnet source_subnet] [-h, --help] [--version] [--debug] group_name</code>

<b>euca-bundle-image</b>	Empaqueta una imagen para su uso con Eucalyptus o Amazon EC2.	euca-bundle-image -i, --image image_path -u, --user user [-c, --cert cert_path] [-k, --privatekeyprivate_key_path] [-p, --prefix prefix] [--kernel kernel_id] [--ramdiskramdisk_id] [-B, --block-device-mapping mapping] [-d, --destination destination_path] [--ec2cert ec2cert_path] [-r, --arch target_architecture] [--batch] [-h, --help] [--version] [--debug]
<b>euca-bundle-vol</b>	Empaqueta como imagen el sistema de archivos local de una instancia que se está ejecutando.	euca-bundle-vol -u, --user user -s, --size size_in_MB [-c, --cert cert_path] [-k, --privatekeyprivate_key_path] [-a, --all] [-e, --exclude dir1, dir2,...dirN] [-p, --prefix prefix] [--[no-]inherit] [-v, --volume volume_path] [--fstabfstab_path] [--generatefstab] [--kernel kernel_id] [--ramdiskramdisk_id] [-B, --block-device-mapping mapping] [-d, --destination destination_path] [--ec2cert ec2cert_path] [-r, --arch target_architecture] [--batch] [--version]
<b>euca-confirm-product-instance</b>	Confirma si una instancia se encuentra corriendo con código de producto adjunto.	euca-confirm-product-instance -i, --instance instance_id [-h, --help] [--version] [--debug] product_code
<b>euca-create-snapshot</b>	Crea una instantánea de un volumen existente.	euca-create-snapshot [-h, --help] [--version] [--debug] volume_id
<b>euca-create-volume</b>	Crea un volumen en la zona de disponibilidad especificada.	euca-create-volume -S, --size size   --snapshot snapshot_id -z zone [-h, --help] [--version] [--debug]

<b>euca-delete-bundle</b>	Elimina un paquete previamente subido.	<code>euca-delete-bundle -b, --bucket bucket [-m, --manifest manifest_path] [-p, --prefix prefix] [--clear] [-h, --help] [--version] [--debug]</code>
<b>euca-delete-group</b>	Elimina un grupo de seguridad.	<code>euca-delete-group [-h, --help] [--version] [--debug] group_name</code>
<b>euca-delete-keypair</b>	Elimina un par de llaves.	<code>euca-delete-keypair [-h, --help] [--version] [--debug] keypair_name</code>
<b>euca-delete-snapshot</b>	Elimina un snapshot.	<code>euca-delete-snapshot [-h, --help] [--version] [--debug] snapshot_id</code>
<b>euca-delete-volume</b>	Elimina un volumen.	<code>euca-delete-volume [-h, --help] [--version] [--debug] volume_id</code>
<b>euca-deregister</b>	Des-Registra una imagen del Cloud.	<code>euca-deregister [-h, --help] [--version] [--debug] image_id</code>
<b>euca-describe-addresses</b>	Muestra información sobre las direcciones IP.	<code>euca-describe-addresses [-h, --help] [--version] [--debug] [ip1 ip2...ipN]</code>
<b>euca-describe-availability-zones</b>	Muestra información acerca de las zonas de disponibilidad.	<code>euca-describe-availability-zones [--regionregion] [-h, --help] [--version] [--debug] [zone1 zone2...zoneN] [verbose]</code>
<b>euca-describe-groups</b>	Muestra información sobre grupos de seguridad.	<code>euca-describe-groups [-h, --help] [--version] [--debug] [group1 group2...groupN]</code>
<b>euca-describe-image-attribute</b>	Muestra los atributos de una imagen.	<code>euca-describe-image-attribute [-l, --launch-permission] [-p, --product-code] [-B, --block-device-mapping] [--kernel] [--ramdisk] [-h, --help] [--version] [--debug] image_id</code>

<b>euca-describe-images</b>	Muestra información sobre las imagines.	<code>euca-describe-images [-a] [-o owner] [-x user] [-h, --help] [--version] [--debug] [image1 image2 ... imageN]</code>
<b>euca-describe-instances</b>	Muestra información sobre las instancias.	<code>euca-describe-instances [-h, --help] [--version] [--debug] [instance1... instanceN]</code>
<b>euca-describe-keypairs</b>	Muestra información sobre los pares de llaves.	<code>euca-describe-keypairs [-h, --help] [--version] [--debug] [keypair1 keypair2...keypairN]</code>
<b>euca-describe-regions</b>	Muestra información sobre las regiones.	<code>euca-describe-regions [-h, --help] [--version] [--debug] [region1 region2...regionN]</code>
<b>euca-describe-snapshots</b>	Muestra información sobre las instantáneas.	<code>euca-describe-snapshots [-h, --help] [--version] [--debug] [snapshot1 snapshot2...snapshotN]</code>
<b>euca-describe-volumes</b>	Muestra información sobre los volúmenes.	<code>euca-describe-volumes [-h, --help] [--version] [--debug] [volume1 volume2...volumeN]</code>
<b>euca-detach-volume</b>	Separa un volumen de una instancia.	<code>euca-detach-volume [-h, --help] [--version] [--debug] volume_id</code>
<b>euca-disassociate-address</b>	Disocia una dirección IP pública de una instancia.	<code>euca-disassociate-address [-h, --help] [--version] [--debug] ip</code>
<b>euca-download-bundle</b>	Descarga una imagen empaquetada desde un bucket.	<code>euca-download-bundle -b, --bucket bucket [-m, --manifest manifest_path] [-d, --directory directory] [-h, --help] [--version] [--debug]</code>
<b>euca-get-console-output</b>	Imprime la salida de consola de una instancia que se encuentra corriendo.	<code>euca-get-console-output [-h, --help] [--version] [--debug] instance_id</code>

<b>euca-modify-image-attribute</b>	Modifica los atributos de una imagen.	<code>euca-modify-image-attribute [-l, --launch-permission] [-p, --product-code] [-a, --add entity ] [-r, --remove entity] [-h, --help] [--version] [--debug] image_id</code>
<b>euca-reboot-instances</b>	Reinicia las instancias especificadas.	<code>euca-reboot-instances [-h, --help] [--version] [--debug] [instance1... instanceN]</code>
<b>euca-register</b>	Registra el manifiesto de una imagen para su uso en el Cloud.	<code>euca-register [-h, --help] [--version] [--debug] image_location</code>
<b>euca-release-address</b>	Librea una dirección IP pública.	<code>euca-release-address [-h, --help] [--version] [--debug] ip</code>
<b>euca-reset-image-attribute</b>	Restablece los atributos de una imagen.	<code>euca-reset-image-attribute -l, --launch-permission [-h, --help] [--version] [--debug] image_id</code>
<b>euca-revoke</b>	Revoca una regla de un grupo de seguridad.	<code>euca-revoke [-P   --protocol protocol] [-p   --port-range port_range] [-t   --icmp-type-code type:code] [-o   --source-group source_group] [-u   --source-group-user source_group_user] [-s   --source-subnet source_subnet] [-h, --help] [--version] [--debug] group_name</code>
<b>euca-run-instances</b>	Crea una nueva instancia.	<code>euca-run-instances [-n, --instance-count count] [-g, --group group_name] [-k, --key keyname] [--addressing addressing] [-t, --instance-type instance_type] [-z, --availability-zone zone] [--kernel kernel_id] [--ramdiskramdisk_id] [-h, --help] [--version] [--debug] image_id</code>

<b>euca-terminate- instances</b>	Detiene las instancias especificadas.	<code>euca-terminate-instances [-h, --help] [--version] [--debug] [instance1... instanceN]</code>
<b>euca-unbundle</b>	Desempaqueta una imagen previamente empaquetada.	<code>euca-unbundle-image -m, --manifest manifest_path [-k, --privatekeyprivate_key_path] [-s, --source source_directory] [-d, --destination destination_directory] [-h, --help] [--version] [--debug]</code>
<b>euca-upload- bundle</b>	Sube una imagen previamente empaquetada al Cloud.	<code>euca-upload-bundle -b, --bucket bucket -m, --manifest manifest_path [-a, --access-key access_key_id] [-s, --secret-key secret_key] [--aclcanned_acl] [--ec2cert ec2cert_path] [-d, --directory bundle_path] [--part part] [--urlurl] [--skipmanifest] [--debug] [-h, --help] [--version]</code>
<b>euca-version</b>	Muestra la versión de Euca2ools.	<code>euca-version [-h, --help] [--debug]</code>

*Tabla B.1: Comandos de Euca2ools*

# Anexo C

## Tiempos de Ejecución de N-Reinas

---

En este anexo se presentan los tiempos de todas las ejecuciones del algoritmo N-Reinas, en las diferentes configuraciones de la arquitectura, y se calculan los promedios de los mismos.

## C.1 Tiempos de ejecución en 4 Nodos

N-Reinas en 4 Nodos Físicos (2 Dell + 2 Blade)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	13,766	10,801	10,815	10,792	10,767	10,784	10,764	10,807	10,787	10,791	11,087
19	82,174	82,216	82,201	82,253	82,400	82,286	82,221	82,350	82,186	82,306	82,259
20	663,476	662,652	659,451	662,159	663,073	660,339	662,654	663,490	663,929	662,583	662,381
21	5563,404	5556,867	5570,507	5563,942	5570,617	5564,819	5562,946	5565,640	5558,242	5574,546	5565,153
<b>Total</b>	<b>6322,820</b>	<b>6312,536</b>	<b>6322,974</b>	<b>6319,146</b>	<b>6326,857</b>	<b>6318,228</b>	<b>6318,585</b>	<b>6322,287</b>	<b>6315,144</b>	<b>6330,226</b>	<b>6320,880</b>

*Tabla C.1: Tiempos de ejecución de N-Reinas en 4 nodos físicos*

N-Reinas en 4 Nodos Virtuales (2 c1.xlarge + 2 ml.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	11,377	11,867	11,472	11,379	12,510	11,388	12,603	11,383	11,465	12,373	11,782
19	83,750	83,618	83,576	83,975	83,731	83,540	83,665	83,776	83,719	83,487	83,684
20	665,684	668,168	668,495	666,164	666,580	666,519	669,467	669,930	666,321	665,361	667,269
21	5601,572	5603,377	5610,500	5616,577	5618,858	5607,837	5606,930	5606,358	5609,143	5620,301	5610,145
<b>Total</b>	<b>6362,383</b>	<b>6367,030</b>	<b>6374,043</b>	<b>6378,095</b>	<b>6381,679</b>	<b>6369,284</b>	<b>6372,665</b>	<b>6371,447</b>	<b>6370,648</b>	<b>6381,522</b>	<b>6372,880</b>

*Tabla C.2: Tiempos de ejecución de N-Reinas en 4 nodos virtuales*

## C.2 Tiempos de ejecución en 2 Nodos

### C.2.1 Tiempos de ejecución en 2 Nodos A

N-Reinas en 2 Nodos Físicos (2 Dell)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	20,112	20,096	20,111	20,100	20,106	20,123	20,085	20,114	20,125	20,134	20,111
19	154,075	154,081	154,043	153,902	153,912	153,886	154,011	153,940	154,040	153,924	153,981
20	1237,050	1234,043	1234,108	1237,335	1234,180	1234,170	1235,596	1235,260	1234,371	1234,744	1235,086
21	10422,959	10414,742	10412,581	10445,306	10418,197	10412,872	10409,418	10408,292	10413,502	10407,639	10416,551
<b>Total</b>	<b>11834,196</b>	<b>11822,962</b>	<b>11820,843</b>	<b>11856,643</b>	<b>11826,395</b>	<b>11821,051</b>	<b>11819,110</b>	<b>11817,606</b>	<b>11822,038</b>	<b>11816,441</b>	<b>11825,729</b>

**Tabla C.3:** Tiempos de ejecución de N-Reinas en 2 nodos físicas (Dell)

N-Reinas en 2 Nodos Virtuales (2 c1.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	20,834	20,783	21,041	20,784	20,700	20,635	20,834	21,131	20,652	20,681	20,808
19	155,662	155,663	156,092	155,717	155,691	155,696	155,755	155,954	155,638	155,563	155,743
20	1242,396	1245,237	1242,756	1245,074	1244,426	1245,774	1243,905	1246,976	1243,603	1245,265	1244,541
21	10493,473	10488,358	10518,051	10469,019	10489,986	10477,426	10477,038	10489,632	10499,295	10483,654	10488,593
<b>Total</b>	<b>11912,365</b>	<b>11910,041</b>	<b>11937,940</b>	<b>11890,594</b>	<b>11910,803</b>	<b>11899,531</b>	<b>11897,532</b>	<b>11913,693</b>	<b>11919,188</b>	<b>11905,163</b>	<b>11909,685</b>

**Tabla C.4:** Tiempos de ejecución de N-Reinas en 2 nodos virtuales (c1.xlarge)

## C.2.2 Tiempos de ejecución en 2 Nodos B

N-Reinas en 2 Nodos Físicos (2 Blade)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	24,633	24,614	24,549	24,610	24,592	24,651	24,615	24,579	24,650	24,702	24,620
19	187,945	188,471	188,079	188,713	188,804	187,869	188,112	188,889	188,319	188,065	188,327
20	1514,685	1515,286	1515,867	1513,885	1515,743	1507,605	1512,761	1511,813	1511,171	1512,850	1513,167
21	12740,184	12756,421	12776,499	12748,811	12775,183	12768,492	12719,101	12724,721	12788,749	12766,262	12756,442
<b>Total</b>	<b>14467,447</b>	<b>14484,792</b>	<b>14504,994</b>	<b>14476,019</b>	<b>14504,322</b>	<b>14488,617</b>	<b>14444,589</b>	<b>14450,002</b>	<b>14512,889</b>	<b>14491,879</b>	<b>14482,555</b>

**Tabla C.5:** Tiempos de ejecución de N-Reinas en 2 nodos físicos (Blade)

N-Reinas en 2 Nodos Virtuales (2 mI.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	25,693	25,406	25,437	25,752	25,679	25,327	25,500	25,375	25,361	25,443	25,497
19	190,890	192,176	192,230	191,120	191,716	191,813	191,048	190,785	191,430	192,177	191,539
20	1528,976	1532,456	1524,100	1531,902	1527,242	1529,646	1522,734	1529,129	1525,155	1524,451	1527,579
21	12898,478	12850,331	12869,174	12865,263	12859,555	12868,236	12896,673	12860,803	12855,029	12865,451	12868,899
<b>Total</b>	<b>14644,037</b>	<b>14600,369</b>	<b>14610,941</b>	<b>14614,037</b>	<b>14604,192</b>	<b>14615,022</b>	<b>14635,955</b>	<b>14606,092</b>	<b>14596,975</b>	<b>14607,522</b>	<b>14613,514</b>

**Tabla C.6:** Tiempos de ejecución de N-Reinas en 2 nodos virtuales (mI.xlarge)

### C.2.3 Tiempos de ejecución en 2 Nodos C

N-Reinas en 2 Nodos Físicos (1 Dell + 1 Blade)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	22,303	22,274	22,247	22,264	22,226	22,229	22,294	22,275	22,246	22,268	22,263
19	170,485	170,830	170,935	170,608	170,943	170,718	170,299	170,253	170,620	170,246	170,594
20	1370,063	1366,288	1367,185	1365,786	1367,922	1367,068	1367,288	1365,540	1372,417	1369,045	1367,860
21	11522,238	11523,245	11521,162	11524,968	11530,579	11531,941	11563,325	11520,926	11529,432	11559,424	11532,724
<b>Total</b>	<b>13085,089</b>	<b>13082,637</b>	<b>13081,529</b>	<b>13083,626</b>	<b>13091,670</b>	<b>13091,956</b>	<b>13123,206</b>	<b>13078,994</b>	<b>13094,715</b>	<b>13120,983</b>	<b>13093,441</b>

**Tabla C.7:** Tiempos de ejecución de N-Reinas en 2 nodos físicos (Dell + Blade)

N-Reinas en 2 Nodos Virtuales (1 cl.xlarge + ml.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	23,183	22,996	22,806	22,914	22,934	22,861	22,971	22,860	22,771	22,864	22,916
19	172,373	172,562	172,766	172,795	173,048	172,693	172,677	172,310	172,491	172,433	172,615
20	1378,980	1378,776	1378,908	1377,789	1382,397	1376,372	1381,107	1380,388	1381,733	1380,436	1379,689
21	11623,903	11620,665	11625,145	11648,362	11623,626	11631,768	11659,310	11643,511	11657,009	11717,756	11645,106
<b>Total</b>	<b>13198,439</b>	<b>13194,999</b>	<b>13199,625</b>	<b>13221,860</b>	<b>13202,005</b>	<b>13203,694</b>	<b>13236,065</b>	<b>13219,069</b>	<b>13234,004</b>	<b>13293,489</b>	<b>13220,325</b>

**Tabla C.8:** Tiempos de ejecución de N-Reinas en 2 nodos virtuales (cl.xl + ml.xl)

## C.3 Tiempos de ejecución en 1 Nodo

### C.3.1 Tiempos de ejecución en 1 Nodo A

N-Reinas en 1 Nodo Físico (1 Dell)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	42,991	43,052	42,936	42,891	42,894	42,924	42,896	42,946	42,895	42,941	42,937
19	329,620	329,594	329,675	330,060	329,585	330,686	330,242	329,453	329,467	330,102	329,848
20	2644,449	2644,032	2644,373	2644,496	2646,506	2645,717	2643,828	2644,146	2645,385	2644,884	2644,782
21	22330,696	22295,257	22300,115	22337,676	22305,087	22313,518	22303,740	22310,998	22309,689	22331,845	22313,862
<b>Total</b>	<b>25347,756</b>	<b>25311,935</b>	<b>25317,099</b>	<b>25355,123</b>	<b>25324,072</b>	<b>25332,845</b>	<b>25320,706</b>	<b>25327,543</b>	<b>25327,436</b>	<b>25349,772</b>	<b>25331,429</b>

*Tabla C.9: Tiempos de ejecución de N-Reinas en 1 nodo físico (Dell)*

N-Reinas en 1 Nodo Virtual (1 c1.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	43,378	43,296	43,400	43,137	43,510	43,444	43,315	43,558	43,379	43,322	43,374
19	331,897	331,321	332,324	331,449	332,264	332,850	333,287	332,202	332,362	331,342	332,130
20	2660,308	2663,916	2665,548	2659,833	2661,323	2673,246	2660,603	2655,135	2666,883	2664,112	2663,091
21	22488,700	22510,027	22460,255	22465,510	22558,069	22513,370	22442,605	22451,031	22579,760	22552,635	22502,196
<b>Total</b>	<b>25524,283</b>	<b>25548,560</b>	<b>25501,527</b>	<b>25499,929</b>	<b>25595,166</b>	<b>25562,910</b>	<b>25479,810</b>	<b>25481,926</b>	<b>25622,384</b>	<b>25591,411</b>	<b>25540,791</b>

*Tabla C.10: Tiempos de ejecución de N-Reinas en 1 nodo virtual (c1.xlarge)*

### C.3.2 Tiempos de ejecución en 1 Nodo B

N-Reinas en 1 Nodo Físico (1 Dell)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	52,522	52,598	52,454	52,465	52,647	52,787	52,660	52,594	52,612	52,388	52,573
19	402,278	403,518	405,924	404,405	404,165	402,958	403,346	402,262	403,188	404,170	403,621
20	3249,010	3241,348	3241,669	3260,274	3229,337	3233,207	3229,099	3239,617	3243,633	3252,009	3241,920
21	27405,106	27347,079	27516,328	27266,564	27384,593	27525,605	27326,921	27435,801	27256,917	27265,245	27373,016
<b>Total</b>	<b>31108,916</b>	<b>31044,543</b>	<b>31216,375</b>	<b>30983,708</b>	<b>31070,742</b>	<b>31214,557</b>	<b>31012,026</b>	<b>31130,274</b>	<b>30956,350</b>	<b>30973,812</b>	<b>31071,130</b>

*Tabla C.11: Tiempos de ejecución de N-Reinas en 1 nodo físico (Blade)*

N-Reinas en 1 Nodo Virtual (1 m1.xlarge)											
Run											
Tablero	1	2	3	4	5	6	7	8	9	10	Promedio
18	53,665	53,054	54,162	52,957	53,172	53,274	53,242	53,004	53,100	53,187	53,282
19	408,298	408,456	411,529	408,952	404,852	406,376	406,715	408,275	407,641	407,595	407,869
20	3261,015	3273,932	3265,792	3263,739	3273,975	3260,886	3260,147	3258,383	3291,076	3281,591	3269,054
21	27562,769	27535,833	27568,562	27572,447	27512,529	27547,261	27562,213	27700,341	27584,018	27539,847	27568,582
<b>Total</b>	<b>31285,747</b>	<b>31271,275</b>	<b>31300,045</b>	<b>31298,095</b>	<b>31244,528</b>	<b>31267,797</b>	<b>31282,317</b>	<b>31420,003</b>	<b>31335,835</b>	<b>31282,220</b>	<b>31298,786</b>

*Tabla C.12: Tiempos de ejecución de N-Reinas en 1 nodo virtual (c1.xlarge)*



# Anexo D

## Tiempos de Ejecución de NPB

En este anexo se presentan los tiempos de todas las ejecuciones de las aplicaciones seleccionadas de los NAS Parallel Benchmarks, en las diferentes configuraciones de la arquitectura, y se calculan los promedios de los mismos.

### D.1 Tiempos de ejecución en 4 Nodos

NPB en 4 Nodos Físicos (2 Dell + 2 Blade)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	15,29	15,28	15,33	15,54	15,92	15,54	15,22	15,30	16,50	15,28	<b>15,52</b>
<b>ft.C</b>	213,73	215,05	214,03	215,15	213,16	217,99	216,99	220,55	214,38	211,48	<b>215,25</b>
<b>lu.C</b>	152,19	157,05	162,85	156,76	153,74	160,59	154,82	153,17	172,29	154,70	<b>157,82</b>
<b>Total</b>	<b>381,21</b>	<b>387,38</b>	<b>392,21</b>	<b>387,45</b>	<b>382,82</b>	<b>394,12</b>	<b>387,03</b>	<b>389,02</b>	<b>403,17</b>	<b>381,46</b>	<b>388,59</b>

*Tabla D.1: Tiempos de ejecución de NAS en 4 nodos físicos*

NPB en 4 Nodos Virtuales (2 c1.xlarge + 2 m1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	16,67	16,04	16,29	16,53	16,07	16,43	16,47	16,22	16,32	16,31	<b>16,34</b>
<b>ft.C</b>	283,91	292,82	287,95	287,10	291,16	289,31	298,28	289,56	286,99	291,1	<b>289,82</b>
<b>lu.C</b>	330,33	351,96	346,78	333,58	334,42	351,68	337,79	371,12	353,08	350,35	<b>346,11</b>
<b>Total</b>	<b>630,91</b>	<b>660,82</b>	<b>651,02</b>	<b>637,21</b>	<b>641,65</b>	<b>657,42</b>	<b>652,54</b>	<b>676,90</b>	<b>656,39</b>	<b>657,76</b>	<b>652,26</b>

*Tabla D.2: Tiempos de ejecución de NAS en 4 nodos virtuales*

## D.2 Tiempos de ejecución en 2 Nodos

### D.2.1 Tiempos de ejecución en 2 Nodos A

NPB en 2 Nodos Físicos (2 Dell)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	23,52	23,56	23,91	23,58	23,55	23,91	23,58	24,23	23,55	23,89	23,73
<b>ft.C</b>	138,3	138,19	133,09	135,65	138,61	134,31	139,71	138,42	138,67	138,33	137,33
<b>lu.C</b>	104,77	102,09	104,15	102,44	106,44	104,3	102,66	104	101,55	102,21	103,46
<b>Total</b>	266,59	263,84	261,15	261,67	268,60	262,52	265,95	266,65	263,77	264,43	264,52

*Tabla D.3: Tiempos de ejecución de NAS en 2 nodos físicos (Dell)*

NPB en 2 Nodos Virtuales (2 c1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	24,03	23,99	24,13	24,13	24,13	23,95	26,21	26,15	24,15	24,24	24,51
<b>ft.C</b>	218,67	221,12	222,37	222,95	222,76	222,6	220	216,5	222,02	222,01	221,10
<b>lu.C</b>	146	144,46	152,91	142,26	152,33	146,06	144,16	145,43	147,62	148,54	146,98
<b>Total</b>	388,70	389,57	399,41	389,34	399,22	392,61	390,37	388,08	393,79	394,79	392,59

*Tabla D.4: Tiempos de ejecución de NAS en 2 nodos virtuales (c1.xlarge)*

### D.2.2 Tiempos de ejecución en 2 Nodos B

NPB en 2 Nodos Físicos (2 Blade)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	31,8	30,74	31,02	31,01	30,61	31,82	31,28	30,59	31,37	30,67	<b>31,09</b>
<b>ft.C</b>	169,5	170,71	171,77	170,01	168,38	171,93	171,85	171,35	169,58	168,57	<b>170,37</b>
<b>lu.C</b>	202,99	203,76	207,97	203,22	199,65	203,91	201,28	203,2	202,8	205,63	<b>203,44</b>
<b>Total</b>	404,29	405,21	410,76	404,24	398,64	407,66	404,41	405,14	403,75	404,87	<b>404,90</b>

*Tabla D.5: Tiempos de ejecución de NAS en 2 nodos físicos (Blade)*

NPB en 2 Nodos Virtuales (2 m1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	32,21	32,13	35,06	32,06	31,47	32,25	33,44	32,45	33,41	32,78	<b>32,73</b>
<b>ft.C</b>	370,68	373,36	373,53	371,64	371,7	365,8	373,03	373,05	371,25	369,91	<b>371,40</b>
<b>lu.C</b>	276,94	292,76	314,38	296,59	299,28	314,9	299,49	287,85	298,18	297,96	<b>297,83</b>
<b>Total</b>	679,83	698,25	722,97	700,29	702,45	712,95	705,96	693,35	702,84	700,65	<b>701,95</b>

*Tabla D.6: Tiempos de ejecución de NAS en 2 nodos virtuales (m1.xlarge)*

## D.2.3 Tiempos de ejecución en 2 Nodos C

NPB en 2 Nodos Físicos (Dell + Blade)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
ep.C	32,09	30,72	30,49	30,48	31,34	30,47	30,58	30,61	30,55	30,86	<b>30,82</b>
ft.C	163,82	166,94	167,42	168,23	164,14	170,83	164,92	166,57	164,2	163,17	<b>166,02</b>
lu.C	211,03	204,77	212,82	206,08	212,56	207,75	208,94	207,37	205,81	211,38	<b>208,85</b>
<b>Total</b>	<b>406,94</b>	<b>402,43</b>	<b>410,73</b>	<b>404,79</b>	<b>408,04</b>	<b>409,05</b>	<b>404,44</b>	<b>404,55</b>	<b>400,56</b>	<b>405,41</b>	<b>405,69</b>

*Tabla D.7: Tiempos de ejecución de NAS en 2 nodos físicos (Dell + Blade)*

NPB en 2 Nodos Virtuales (c1.xlarge + m1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
ep.C	31,53	32,03	32,40	32,05	32,51	32,44	32,16	32,10	32,73	32,25	<b>32,22</b>
ft.C	330,12	301,25	325,94	305,94	317,34	341,20	316,57	311,54	321,32	327,43	<b>319,87</b>
lu.C	299,34	291,06	296,05	304,39	296,90	292,42	293,84	298,40	295,05	296,19	<b>296,36</b>
<b>Total</b>	<b>660,99</b>	<b>624,34</b>	<b>654,39</b>	<b>642,38</b>	<b>646,75</b>	<b>666,06</b>	<b>642,57</b>	<b>642,04</b>	<b>649,10</b>	<b>655,87</b>	<b>648,45</b>

*Tabla D.8: Tiempos de ejecución de NAS en 2 nodos virtuales (c1.xl + m1.xl)*

## D.3 Tiempos de ejecución en 1 Nodo

### D.3.1 Tiempos de ejecución en 1 Nodo A

NPB en 1 Nodo Físico (Dell)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
ep.C	46,75	47,14	46,74	47,08	47,21	47,16	47,17	46,87	46,74	49,85	<b>47,27</b>
ft.C	67,12	66,83	66,86	66,89	67,03	67,05	66,84	67,06	66,73	67,09	<b>66,95</b>
lu.C	194,95	196,07	198,36	196,49	197,97	198,62	195,33	191,24	196,24	197,41	<b>196,27</b>
<b>Total</b>	<b>308,82</b>	<b>310,04</b>	<b>311,96</b>	<b>310,46</b>	<b>312,21</b>	<b>312,83</b>	<b>309,34</b>	<b>305,17</b>	<b>309,71</b>	<b>314,35</b>	<b>310,49</b>

*Tabla D.9: Tiempos de ejecución de NAS en 1 nodo físico (Dell)*

NPB en 1 Nodo Virtual (c1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
ep.C	47,76	47,56	48,56	49,91	48,43	47,79	48,17	48,5	47,74	48,32	<b>48,27</b>
ft.C	68,46	70,6	71,01	70,75	71,09	70,74	71,52	70,4	70,54	70,9	<b>70,60</b>
lu.C	199,22	200,62	199,01	199,07	199,96	199,86	198,72	205,44	201,02	199,96	<b>200,29</b>
<b>Total</b>	<b>315,44</b>	<b>318,78</b>	<b>318,58</b>	<b>319,73</b>	<b>319,48</b>	<b>318,39</b>	<b>318,41</b>	<b>324,34</b>	<b>319,30</b>	<b>319,18</b>	<b>319,16</b>

*Tabla D.10: Tiempos de ejecución de NAS en 1 nodo virtual (c1.xlarge)*

### D.3.2 Tiempos de ejecución en 1 Nodo B

NPB en 1 Nodo Físico (Blade)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	62,74	60,98	61,92	60,96	63,22	61,05	61,25	63,57	61,38	60,98	<b>61,81</b>
<b>ft.C</b>	161,48	161,72	161,47	162,06	162,94	161,8	162,37	162,46	161,88	161,86	<b>162,00</b>
<b>lu.C</b>	489,62	472,17	480,23	481,95	474,4	472,38	478,48	484,87	495,09	485,62	<b>481,48</b>
<b>Total</b>	<b>713,84</b>	<b>694,87</b>	<b>703,62</b>	<b>704,97</b>	<b>700,56</b>	<b>695,23</b>	<b>702,10</b>	<b>710,90</b>	<b>718,35</b>	<b>708,46</b>	<b>705,29</b>

*Tabla D.11: Tiempos de ejecución de NAS en 1 nodo físico (Blade)*

NPB en 1 Nodo Virtual (m1.xlarge)											
Run											
	1	2	3	4	5	6	7	8	9	10	Promedio
<b>ep.C</b>	63,39	62,23	63,09	63,53	67,25	65,05	63,87	63,42	63,99	63,35	<b>63,92</b>
<b>ft.C</b>	165,93	161,82	161,44	161,33	162,51	161,89	162,12	161,81	161,15	161,59	<b>162,16</b>
<b>lu.C</b>	477,54	480,71	478,62	491,95	481,69	475,44	481,81	483,47	486,72	488,8	<b>482,68</b>
<b>Total</b>	<b>706,86</b>	<b>704,76</b>	<b>703,15</b>	<b>716,81</b>	<b>711,45</b>	<b>702,38</b>	<b>707,80</b>	<b>708,70</b>	<b>711,86</b>	<b>713,74</b>	<b>708,75</b>

*Tabla D.12: Tiempos de ejecución de NAS en 1 nodo virtual (m1.xlarge)*

# Anexo E

## Pruebas de Comunicación

En este anexo se presentan los resultados obtenidos tras efectuar las pruebas de comunicación de red en las diferentes configuraciones de la arquitectura.

### E.1 Pruebas de comunicación entre 2 Nodos A

Pruebas de comunicación entre 2 Nodos A						
Tamaño mensaje	Entre 2 Dell		Entre 2 c1.xlarge		Degradación	
	Tiempo	Mbps	Tiempo	Mbps	Diferencia Mbps	Porcentaje
<b>1.00 B</b>	0,000061	0,25	0,00026	0,06	0,19	76,54%
<b>2.00 B</b>	0,000061	0,50	0,00026	0,12	0,38	76,54%
<b>4.00 B</b>	0,000062	0,98	0,00026	0,23	0,75	76,15%
<b>8.00 B</b>	0,000061	2,00	0,000258	0,47	1,53	76,36%
<b>16.00 B</b>	0,000062	3,94	0,00026	0,94	3,00	76,15%
<b>32.00 B</b>	0,000062	7,88	0,000263	1,86	6,02	76,43%
<b>64.00 B</b>	0,000063	15,50	0,000263	3,71	11,79	76,05%
<b>128.00 B</b>	0,000067	29,15	0,000263	7,43	21,72	74,52%
<b>256.00 B</b>	0,000071	55,02	0,000265	14,74	40,28	73,21%
<b>512.00 B</b>	0,000082	95,27	0,000277	28,20	67,07	70,40%
<b>1.00 KB</b>	0,0001	156,25	0,000299	52,26	103,99	66,56%
<b>2.00 KB</b>	0,000138	226,45	0,000486	64,30	162,15	71,60%
<b>4.00 KB</b>	0,000189	330,69	0,000564	110,82	219,87	66,49%
<b>8.00 KB</b>	0,000272	459,56	0,000678	184,37	275,19	59,88%
<b>16.00 KB</b>	0,000503	497,02	0,000878	284,74	212,28	42,71%
<b>32.00 KB</b>	0,000717	697,35	0,001104	452,90	244,45	35,05%
<b>64.00 KB</b>	0,001405	711,74	0,00232	431,03	280,71	39,44%
<b>128.00 KB</b>	0,002515	795,23	0,003303	605,51	189,72	23,86%
<b>256.00 KB</b>	0,004792	834,72	0,005484	729,39	105,33	12,62%
<b>512.00 KB</b>	0,009245	865,33	0,009981	801,52	63,81	7,37%
<b>1.00 MB</b>	0,018132	882,42	0,01891	846,11	36,30	4,11%
<b>2.00 MB</b>	0,035935	890,50	0,036946	866,13	24,37	2,74%
<b>4.00 MB</b>	0,071592	893,95	0,072665	880,75	13,20	1,48%
<b>8.00 MB</b>	0,142895	895,76	0,14421	887,59	8,17	0,91%
<b>16.00 MB</b>	0,28549	896,70	0,287451	890,59	6,12	0,68%
<b>32.00 MB</b>	0,570638	897,24	0,576371	888,32	8,92	0,99%
<b>64.00 MB</b>	1,14092	897,52	1,155054	886,54	10,98	1,22%
<b>128.00 MB</b>	2,282128	897,41	2,323165	881,56	15,85	1,77%
<b>256.00 MB</b>	4,563349	897,59	4,611862	888,14	9,44	1,05%
<b>512.00 MB</b>	9,125607	897,69	9,182318	892,15	5,54	0,62%
<b>1.00 GB</b>	18,2501	897,75	18,356936	892,52	5,22	0,58%

Tabla E.1: Resultados de las pruebas de comunicación entre 2 nodos A

## E.2 Pruebas de comunicación entre 2 Nodos B

Pruebas de comunicación entre 2 Nodos B						
Tamaño mensaje	Entre 2 Blade		Entre 2 m1.xlarge		Degradación	
	Tiempo	Mbps	Tiempo	Mbps	Diferencia Mbps	Porcentaje
<b>1.00 B</b>	0,000103	0,15	0,000771	0,02	0,13	86,64%
<b>2.00 B</b>	0,000103	0,30	0,000773	0,04	0,26	86,68%
<b>4.00 B</b>	0,000105	0,58	0,000775	0,08	0,50	86,45%
<b>8.00 B</b>	0,000105	1,16	0,000772	0,16	1,00	86,40%
<b>16.00 B</b>	0,000103	2,37	0,000771	0,32	2,05	86,64%
<b>32.00 B</b>	0,000102	4,79	0,000815	0,60	4,19	87,48%
<b>64.00 B</b>	0,000103	9,48	0,000776	1,26	8,22	86,73%
<b>128.00 B</b>	0,000106	18,43	0,000782	2,50	15,93	86,45%
<b>256.00 B</b>	0,000109	35,84	0,00079	4,94	30,89	86,20%
<b>512.00 B</b>	0,000136	57,44	0,000798	9,79	47,65	82,96%
<b>1.00 KB</b>	0,000159	98,27	0,000826	18,92	79,35	80,75%
<b>2.00 KB</b>	0,000204	153,19	0,000889	35,15	118,03	77,05%
<b>4.00 KB</b>	0,000249	251,00	0,000944	66,21	184,80	73,62%
<b>8.00 KB</b>	0,000309	404,53	0,001006	124,25	280,28	69,28%
<b>16.00 KB</b>	0,000481	519,75	0,00112	223,21	296,54	57,05%
<b>32.00 KB</b>	0,000802	623,44	0,001467	340,83	282,61	45,33%
<b>64.00 KB</b>	0,001594	627,35	0,004507	221,88	405,48	64,63%
<b>128.00 KB</b>	0,002698	741,29	0,005721	349,59	391,70	52,84%
<b>256.00 KB</b>	0,004955	807,27	0,008425	474,78	332,49	41,19%
<b>512.00 KB</b>	0,009418	849,44	0,013693	584,24	265,20	31,22%
<b>1.00 MB</b>	0,018297	874,46	0,026344	607,35	267,11	30,55%
<b>2.00 MB</b>	0,036162	884,91	0,047431	674,66	210,24	23,76%
<b>4.00 MB</b>	0,071829	891,01	0,094232	679,17	211,83	23,77%
<b>8.00 MB</b>	0,143094	894,52	0,185628	689,55	204,97	22,91%
<b>16.00 MB</b>	0,285658	896,18	0,373124	686,10	210,08	23,44%
<b>32.00 MB</b>	0,570794	897,00	0,747874	684,61	212,39	23,68%
<b>64.00 MB</b>	1,141055	897,42	1,504286	680,72	216,69	24,15%
<b>128.00 MB</b>	2,282742	897,17	2,986379	685,78	211,39	23,56%
<b>256.00 MB</b>	4,564015	897,46	5,992596	683,51	213,95	23,84%
<b>512.00 MB</b>	9,126027	897,65	15,318775	534,77	362,88	40,43%
<b>1.00 GB</b>	18,250128	897,75	31,530797	519,62	378,13	42,12%

*Tabla E.2: Resultados de las pruebas de comunicación entre 2 nodos B*

### E.3 Pruebas de comunicación entre 2 Nodos C

Pruebas de comunicación entre 2 Nodos C						
Tamaño mensaje	Entre Dell y Blade		Entre c1.xl y m1.xl		Degradación	
	Tiempo	Mbps	Tiempo	Mbps	Diferencia Mbps	Porcentaje
<b>1.00 B</b>	0,000093	0,16	0,000559	0,03	0,14	83,36%
<b>2.00 B</b>	0,000093	0,33	0,000551	0,06	0,27	83,12%
<b>4.00 B</b>	0,000096	0,64	0,000552	0,11	0,53	82,61%
<b>8.00 B</b>	0,000093	1,31	0,000549	0,22	1,09	83,06%
<b>16.00 B</b>	0,000094	2,60	0,000547	0,45	2,15	82,82%
<b>32.00 B</b>	0,000095	5,14	0,000547	0,89	4,25	82,63%
<b>64.00 B</b>	0,000098	9,96	0,000546	1,79	8,18	82,05%
<b>128.00 B</b>	0,000102	19,15	0,000546	3,58	15,57	81,32%
<b>256.00 B</b>	0,000107	36,51	0,000541	7,22	29,29	80,22%
<b>512.00 B</b>	0,000106	73,70	0,000571	13,68	60,02	81,44%
<b>1.00 KB</b>	0,000186	84,01	0,000592	26,39	57,61	68,58%
<b>2.00 KB</b>	0,000204	153,19	0,000718	43,52	109,66	71,59%
<b>4.00 KB</b>	0,000252	248,02	0,000764	81,81	166,21	67,02%
<b>8.00 KB</b>	0,000342	365,50	0,000877	142,53	222,97	61,00%
<b>16.00 KB</b>	0,000504	496,03	0,001038	240,85	255,18	51,45%
<b>32.00 KB</b>	0,000834	599,52	0,001849	270,42	329,10	54,89%
<b>64.00 KB</b>	0,001594	627,35	0,003457	289,27	338,08	53,89%
<b>128.00 KB</b>	0,002682	745,71	0,005035	397,22	348,49	46,73%
<b>256.00 KB</b>	0,004961	806,29	0,007542	530,36	275,93	34,22%
<b>512.00 KB</b>	0,009417	849,53	0,012249	653,11	196,41	23,12%
<b>1.00 MB</b>	0,018306	874,03	0,023053	694,05	179,98	20,59%
<b>2.00 MB</b>	0,036163	884,88	0,045392	704,97	179,91	20,33%
<b>4.00 MB</b>	0,071811	891,23	0,087286	733,22	158,01	17,73%
<b>8.00 MB</b>	0,143108	894,43	0,176694	724,42	170,01	19,01%
<b>16.00 MB</b>	0,285689	896,08	0,364984	701,40	194,68	21,73%
<b>32.00 MB</b>	0,57084	896,92	0,730624	700,77	196,15	21,87%
<b>64.00 MB</b>	1,141131	897,36	1,450906	705,77	191,59	21,35%
<b>128.00 MB</b>	2,284575	896,45	2,903228	705,42	191,03	21,31%
<b>256.00 MB</b>	4,565927	897,08	5,830165	702,55	194,53	21,68%
<b>512.00 MB</b>	9,128337	897,43	11,808035	693,76	203,66	22,69%
<b>1.00 GB</b>	18,252995	897,61	23,275224	703,92	193,68	21,58%

*Tabla E.3: Resultados de las pruebas de comunicación entre 2 nodos C*