

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

“COMPARATIVA DE EFICIENCIA ENTRE VIRTUALIZACIÓN COMPLETA Y PARAVIRTUALIZACIÓN USANDO DIFERENTES HIPERVISORES DE DISTRIBUCIÓN LIBRE Y PROPIETARIA SOBRE HARDWARE HETEROGÉNEO EN CONDICIONES DE LABORATORIO EN FUNCIÓN DEL RENDIMIENTO Y EL CONSUMO DE RECURSOS”

INFORME DE PROYECTO DE GRADUACIÓN

Previa a la obtención del título de:

INGENIERO EN TELEMÁTICA

PRESENTADA POR:

**CASTRO REYES DAVID ALEXANDER
VILLACÍS GUILLEN PATRICIO JOSÉ**

GUAYAQUIL - ECUADOR

2013

AGRADECIMIENTO

Agradecemos:

En primer lugar a Dios por habernos permitido culminar con éxito este trabajo.

A nuestro director Ing. Ignacio Marín, por su guía en el proceso para realizar nuestra tesis.

A la Ing. Patricia Chávez, por haber colaborado en el proceso del mismo.

DEDICATORIA

A mi familia, mis amigos, mis profesores

David Castro Reyes

A mis padres por su apoyo y estímulo
para continuar con mis estudios y a todas
las personas que han creído en mí.

Patricio Villacís Guillén

TRIBUNAL DE SUSTENTACIÓN

Dr. Boris Vintimilla Burgos Ph. D
PRESIDENTE

Ignacio Marín-García MSIS
DIRECTOR DE PROYECTO

Ing. Rayner Durango MSIG
MIEMBRO PRINCIPAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Informe, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”

(Reglamento de Graduación de la ESPOL)

David A. Castro Reyes

Patricio J. Villacís Guillén

RESUMEN

En el presente proyecto de investigación se realizó una comparativa de eficiencia entre las dos técnicas de virtualización más usadas en el mercado: virtualización y paravirtualización, en función del rendimiento y el consumo de recursos, utilizando las plataformas más conocidas: VMware vSphere, Citrix XenServer y Xen sobre tres tipos diferentes de hardware: un servidor Sun Microsystems, un computador de escritorio y un servidor Hewlett-Packard. La hipótesis planteada tuvo dos niveles, el primer nivel consistió en determinar cuál de las técnicas de virtualización presentó un mejor rendimiento para cada escenario; y el segundo nivel permitió determinar la independencia del rendimiento de las técnicas de virtualización sobre el hardware usado. Para llevar a cabo la comparativa se realizaron diferentes pruebas consistentes en medir el consumo de memoria RAM y procesador, usando los diferentes softwares de medición para cada sistema operativo, Linux y Windows. Se crearon los escenarios para las pruebas planteadas, las mismas que se

ejecutaron sobre los ambientes de trabajo individual y simultáneo, utilizando carga de uso de recursos y de tráfico con solicitudes web; siendo 200 el número de repeticiones para cada una de ellas. De los resultados obtenidos se realizó un análisis estadístico para determinar si la hipótesis planteada en cada caso fue cumplida.

ÍNDICE GENERAL

RESUMEN	V
ÍNDICE GENERAL	VII
GLOSARIO	X
ABREVIATURAS	XII
ÍNDICE DE FIGURAS	XIII
ÍNDICE DE TABLAS	XV
INTRODUCCIÓN	XXII
1. GENERALIDADES	1
1.1 Antecedentes	2
1.2 Justificación	4
1.3 Objetivos	6
1.3.1 Objetivo General	6
1.3.2 Objetivos Específicos	7
1.4 Alcances y Restricciones	8
1.5 Descripción del Proyecto	8
1.6 Metodología	11

1.7 Planteamiento de la Hipótesis	13
2. TECNOLOGÍAS DE VIRTUALIZACIÓN	15
2.1 Virtualización	16
2.2 Virtualización de Plataforma	18
2.3 Virtualización de Recursos	32
2.4 Virtualización de Aplicaciones	34
2.5 Virtualización de Escritorio	36
2.6 Comparativa entre la técnica de Virtualización y Paravirtualización	37
3. SOFTWARE DE VIRTUALIZACIÓN Y PARAVIRTUALIZACIÓN	42
3.1 Software de Virtualización	43
3.2 Software de Paravirtualización	48
3.3 Comparativa	53
4. SOFTWARE DE MEDICIÓN DE RENDIMIENTO	57
4.1 Evaluación de Memoria RAM	59
4.2 Evaluación de CPU	65
5. ESCENARIOS	69
5.1 Arquitecturas para las Pruebas	70
5.2 Descripción de las Pruebas	75

5.3 Escenarios para las Pruebas	81
6. PRUEBAS Y RESULTADOS	83
6.1 Requerimientos mínimos	83
6.2 Pruebas de Rendimiento	84
6.2.1 Memoria RAM	84
6.2.2 CPU	85
6.3 Resultados	85
6.4 Análisis de Resultados	92
CONCLUSIONES	101
RECOMENDACIONES	104
ANEXOS	106
ANEXO A: ESTUDIO PRELIMINAR	107
ANEXO B: LISTA DE ESCENARIOS PARA LAS PRUEBAS	112
ANEXO C: SCRIPTS PERSONALIZADOS	115
ANEXO D: RESULTADOS DE LAS PRUEBAS DE ESTUDIO ...	121
ANEXO E: COMPARATIVA DE ESTUDIO	139
BIBLIOGRAFÍA	157

GLOSARIO

.Net. Es un framework de programación de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

Abstracción. Propiedad y/o técnica de programación que oculta los detalles de la implementación.

AMD64. Es una arquitectura desarrollada por AMD a partir de la arquitectura x86 permitiendo que memorias de 4GB sean direccionadas directamente por una CPU de 32 bits.

Anillo. Conocido también como anillo de protección, es el mecanismo para proteger datos y funcionalidad de los fallos y comportamiento malicioso.

AMD-V. Extensiones de virtualización para los procesadores AMD

Bare-metal. El término bare-metal (vacío) se refiere a la arquitectura física subyacente de una computadora.

Benchmark. Programa informático que tiene el objetivo de medir el rendimiento de un elemento del sistema y poder comparar los resultados con máquinas similares.

Chroot. Es una llamada al sistema en UNIX que permite configurar un directorio como "raíz" del sistema de ficheros para un proceso y sus hijos

Framework. Es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado.

Fluent. Es un software que contiene amplias capacidades de modelado físico necesario para modelar el flujo, la turbulencia, la transferencia de calor, y las reacciones de las aplicaciones industriales

Intel VT. Componentes de la tecnología Intel que son compatibles con la virtualización

Java. Es un lenguaje de programación originalmente desarrollado por James Gosling de Sun Microsystems que deriva mucho de su sintaxis de C y C++.

PowerFLOW. Es una aplicación de análisis de fluidos para simular los fenómenos de fluidos más complejos.

Preboot eXecution Environment (PXE) o entorno de ejecución de prearranque, es un entorno para arrancar e instalar el sistema operativo en ordenadores a través de una red.

Script. Archivo que incluye un conjunto de comandos que son ejecutados de forma secuencial, desde la primera línea hasta la última.

Spiceworks. Red social de negocios con más de 1.9 millones de profesionales de TI de pequeñas y medianas empresas en todo el mundo

Swapping. Técnica mediante la cual se intercambia un proceso que está en memoria por otro que no lo está. Para esto se hace uso de un área de memoria conocida como de intercambio (swap).

Swap. Conocido como espacio de intercambio, es una zona del disco que se usa para guardar las imágenes de los procesos que no han de mantenerse en memoria física.

Traducción binaria. Convertir el código de la CPU origen a código de la CPU destino.

Vanille. Conocido en español núcleo “vanilla”, es un núcleo de linux que no ha sido modificado por nadie.

VMBus. Es un canal lógico que permite la comunicación entre particiones en Microsoft Hyper-V

Vmkernel. Es el núcleo utilizado por vSphere. Proporciona servicios básicos del sistema operativo necesarios para soportar la virtualización: abstracción de hardware, controladores de hardware, planificador, asignador de memoria, sistema de archivos (VMFS) y monitor de máquina virtual (VMM).

VM/370. Es un sistema operativo que proporciona a múltiples usuarios sistemas de cómputo IBM System/370 separadas e independientes.

ABREVIATURAS

CLR	Entorno en tiempo de ejecución de lenguaje común
COS	Sistema operativo de consola
CPU	Unidad central de procesamiento
HTTP	Protocolo de transferencia de hipertexto
JVM	Máquina virtual Java
LAN	Red de área local
LVM	Administrador de volúmenes lógicos
NAS	Almacenamiento conectado a red
NIST	Instituto Nacional de Estándares y Tecnología de los Estados Unidos de América
PXE	Entorno de ejecución de pre arranque
PYMES	Pequeñas y Medianas Empresas
RAID	Conjunto redundante de discos independientes
RDP	Protocolo de escritorio remoto
SAN	Red de área de almacenamiento
USB	Bus universal en serie
VDI	Infraestructura de escritorio virtual
VM / MV	Máquina Virtual
VMI	Interfaz de máquinas virtuales
VMM	Monitor de máquinas virtuales
VPN	Red privada virtual
VPS	Servidor privado virtual
DM&C	Dominio de Administración y Control
PVM	Máquina con paravirtualización
HVM	Máquina con virtualización asistida por hardware

ÍNDICE DE FIGURAS

Figura 1.1	Entorno de Trabajo del Proyecto de Graduación	10
Figura 1.2	Metodología del Proyecto de Graduación	13
Figura 2.1	Arquitectura general de la técnica de Emulación	20
Figura 2.2	Arquitectura general de la técnica de Virtualización ...	22
Figura 2.3	Arquitectura general de la técnica de Paravirtualización	24
Figura 2.4	Arquitectura general de la técnica Virtualización a nivel de sistema operativo	27
Figura 2.5	Arquitectura básica de virtualización de plataforma	28
Figura 2.6	Tipos de Hipervisores	30
Figura 2.7	Arquitectura del Hipervisor Tipo I	31
Figura 2.8	Arquitectura del Hipervisor Tipo II	32
Figura 2.9	Entorno Virtual SystemGuard de Microsoft	36
Figura 3.1	Arquitectura del hipervisor de XenServer	49
Figura 3.2	Ambiente virtual de Xen	51
Figura 3.3	Esquema de virtualización UML	54
Figura 4.1	Comando para medir consumo de memoria RAM con la herramienta sar	62
Figura 4.2	Captura de hardinfo del consumo de memoria RAM ...	63
Figura 4.3	Captura del Administrador de Tareas de Windows	67

Figura 4.4	Comando para medir consumo de CPU con la herramienta sar	66
Figura 4.5	Captura del resultado obtenido del comando top	67
Figura 4.6	Captura del Monitor de Rendimiento de Windows	69
Figura 5.1	Imagen del Servidor Sun Microsystems Sunfire X2200	71
Figura 5.2	Imagen del Computador de Escritorio	72
Figura 5.3	Imagen del Servidor HP Proliant DL160 G6	73
Figura 5.4	Imagen de los parámetros de una máquina virtual desde Citrix XenCenter	74
Figura 5.5	Diagrama de Flujo para elaborar las pruebas	77
Figura 5.6	Comando para generar estrés al sistema	79
Figura 5.7	Comando para generar carga de tráfico	80
Figura 5.8	Respuesta de ejecución del comando ab en un cliente Linux	81
Figura C- 1	Esquema de directorios de la ubicación de scripts y resultados	

ÍNDICE DE TABLAS

Tabla I	Comparativa entre Virtualización y Paravirtualización	41
Tabla II	Resumen de soluciones a evaluar	54
Tabla III	Resumen de soluciones a evaluar (continuación)	55
Tabla IV	Resumen de hardware y las plataformas de Virtualización	72
Tabla V	Resumen de arquitecturas y sujetos de medición	74
Tabla VI	Resumen de pruebas sobre un sujeto de medición Linux	81
Tabla VII	Resumen de pruebas sobre un sujeto de medición Windows	82
Tabla VIII	Resultados promedio de consumo de RAM obtenidos en Centos sobre el servidor SUN	86
Tabla IX	Resultados promedio de consumo de RAM obtenidos en Fedora sobre el servidor SUN	86
Tabla X	Resultados promedio de consumo de RAM obtenidos en Windows sobre el servidor SUN	87
Tabla XI	Resultados promedio de consumo de RAM obtenidos en Centos sobre el computador de escritorio	87
Tabla XII	Resultados promedio de consumo de RAM obtenidos en Fedora sobre el computador de escritorio	87
Tabla XIII	Resultados promedio de consumo de RAM obtenidos en Windows sobre el computador de escritorio	88

Tabla XIV	Resultados promedio de consumo de RAM obtenidos en Centos sobre el servidor HP	88
Tabla XV	Resultados promedio de consumo de RAM obtenidos en Fedora sobre el servidor HP	88
Tabla XVI	Resultados promedio de consumo de RAM obtenidos en Windows sobre el servidor HP	89
Tabla XVII	Resultados promedio de consumo de CPU obtenidos en Centos sobre el servidor SUN	89
Tabla XVIII	Resultados promedio de consumo de CPU obtenidos en Fedora sobre el servidor SUN	89
Tabla XIX	Resultados promedio de consumo de CPU obtenidos en Windows sobre el servidor SUN	90
Tabla XX	Resultados promedio de consumo de CPU obtenidos en Centos sobre el computador de escritorio	90
Tabla XXI	Resultados promedio de consumo de CPU obtenidos en Fedora sobre el computador de escritorio	90
Tabla XXII	Resultados promedio de consumo de CPU obtenidos en Windows sobre el computador de escritorio	91
Tabla XXIII	Resultados promedio de consumo de CPU obtenidos en Centos sobre el servidor HP	91
Tabla XXIV	Resultados promedio de consumo de CPU obtenidos en Fedora sobre el servidor HP	91
Tabla XXV	Resultados promedio de consumo de CPU obtenidos en Windows sobre el servidor HP	91
Tabla A- 1	Estudio previo de consumo de RAM sobre sujeto de medición CentOS	

Tabla A- 2	Estudio previo de consumo de RAM sobre sujeto de medición CentOS (continuación)
Tabla A- 3	Estudio previo de consumo de RAM sobre sujeto de medición Fedora
Tabla A- 4	Estudio previo de consumo de RAM sobre sujeto de medición Windows
Tabla A- 5	Estudio previo de consumo de CPU sobre sujeto de medición CentOS
Tabla A- 6	Estudio previo de consumo de CPU sobre sujeto de medición CentOS (continuación)
Tabla A- 7	Estudio previo de consumo de CPU sobre sujeto de medición Fedora
Tabla A- 8	Estudio previo de consumo de CPU sobre sujeto de medición Windows
Tabla B- 1	Escenarios de prueba sobre la plataforma VMware vSphere
Tabla B- 2	Escenarios de prueba sobre la plataforma Citrix XenServer
Tabla B- 3	Escenarios de prueba sobre la plataforma Xen
Tabla D- 1	Resultados obtenidos en CentOS sobre la plataforma VMware vSphere
Tabla D- 2	Resultados obtenidos en Fedora sobre la plataforma VMware vSphere
Tabla D- 3	Resultados obtenidos en Windows sobre la plataforma VMware vSphere
Tabla D- 4	Resultados obtenidos en CentOS sobre la plataforma Citrix XenServer

- Tabla D- 5 Resultados obtenidos en Fedora sobre la plataforma Citrix XenServer
- Tabla D- 6 Resultados obtenidos en Windows sobre la plataforma Citrix XenServer
- Tabla D- 7 Resultados obtenidos en CentOS sobre la plataforma Xen
- Tabla D- 8 Resultados obtenidos en Fedora sobre la plataforma Xen
- Tabla D- 9 Resultados obtenidos en Windows sobre la plataforma Xen
- Tabla D- 10 Resultados obtenidos en CentOS sobre la plataforma VMware VSphere
- Tabla D- 11 Resultados obtenidos en Fedora sobre la plataforma VMware VSphere
- Tabla D- 12 Resultados obtenidos en Windows sobre la plataforma VMware VSphere
- Tabla D- 13 Resultados obtenidos en CentOS sobre la plataforma Citrix XenCenter
- Tabla D- 14 Resultados obtenidos en Fedora sobre la plataforma Citrix XenCenter
- Tabla D- 15 Resultados obtenidos en Windows sobre la plataforma Citrix XenCenter
- Tabla D- 16 Resultados obtenidos en CentOS sobre la plataforma Xen
- Tabla D- 17 Resultados obtenidos en Fedora sobre la plataforma Xen
- Tabla D- 18 Resultados obtenidos en Windows sobre la plataforma Xen
- Tabla D- 19 Resultados obtenidos en CentOS sobre la plataforma VMware VSphere

- Tabla D- 20 Resultados obtenidos en Fedora sobre la plataforma VMware VSphere
- Tabla D- 21 Resultados obtenidos en Windows sobre la plataforma VMware VSphere
- Tabla D- 22 Resultados obtenidos en CentOS sobre la plataforma Citrix XenCenter
- Tabla D- 23 Resultados obtenidos en Fedora sobre la plataforma Citrix XenCenter
- Tabla D- 24 Resultados obtenidos en Windows sobre la plataforma Citrix XenCenter
- Tabla D- 25 Resultados obtenidos en CentOS sobre la plataforma Xen
- Tabla D- 26 Resultados obtenidos en Fedora sobre la plataforma Xen
- Tabla D- 27 Resultados obtenidos en Windows sobre la plataforma Xen
- Tabla E- 1 Comparativa de técnicas en CentOS, Ambiente Individual
- Tabla E- 2 Comparativa de técnicas en CentOS, Ambiente Simultáneo
- Tabla E- 3 Comparativa de técnicas en Fedora, Ambiente Individual
- Tabla E- 4 Comparativa de técnicas en Fedora, Ambiente Simultáneo
- Tabla E- 5 Comparativa de técnicas en Windows, Ambiente Individual
- Tabla E- 6 Comparativa de técnicas en Windows, Ambiente Simultáneo
- Tabla E- 7 Comparativa de técnicas en CentOS, Ambiente Individual
- Tabla E- 8 Comparativa de técnicas en CentOS, Ambiente Simultáneo

- Tabla E- 9 Comparativa de técnicas en Fedora, Ambiente Individual
- Tabla E- 10 Comparativa de técnicas en Fedora, Ambiente Simultáneo
- Tabla E- 11 Comparativa de técnicas en Windows, Ambiente Individual
- Tabla E- 12 Comparativa de técnicas en Windows, Ambiente Simultáneo
- Tabla E- 13 Comparativa de técnicas en CentOS, Ambiente Individual
- Tabla E- 14 Comparativa de técnicas en CentOS, Ambiente Simultáneo
- Tabla E- 15 Comparativa de técnicas en Fedora, Ambiente Individual
- Tabla E- 16 Comparativa de técnicas en Fedora, Ambiente Simultáneo
- Tabla E- 17 Comparativa de técnicas en Windows, Ambiente Individual
- Tabla E- 18 Comparativa de técnicas en Windows, Ambiente Simultáneo
- Tabla E- 19 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual
- Tabla E- 20 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo
- Tabla E- 21 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual
- Tabla E- 22 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo
- Tabla E- 23 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual

- Tabla E- 24 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo
- Tabla E- 25 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual
- Tabla E- 26 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo
- Tabla E- 27 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual
- Tabla E- 28 Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo
- Tabla E- 29 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual
- Tabla E- 30 Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo

INTRODUCCIÓN

En la actualidad el campo de la virtualización tiene un gran impacto en el diseño y funcionalidad de los centros de datos tanto empresarial como de investigación. La virtualización completa y la paravirtualización son las técnicas mayormente utilizadas, cada una tiene aplicaciones específicas en función del hardware y de las necesidades de los centros de datos. En los últimos años, ambas técnicas han tenido actualizaciones con el fin de entregar una solución completa a los clientes, pero con eficiencias comparables entre ellas. Este trabajo se centra en una comparativa de las mismas, utilizando carga de uso de recursos y de tráfico con solicitudes web sobre hardware heterogéneo, para de esta forma poder correlacionar no sólo el software sino también la influencia del hardware sobre el rendimiento de la virtualización en cuanto a la memoria RAM, CPU y consumo de recursos.

CAPÍTULO 1

1. GENERALIDADES

En la actualidad el campo de la virtualización tiene un gran impacto en el diseño y funcionalidad de los centros de cómputo tanto empresarial como de investigación. La técnica de virtualización completa, que en este estudio la llamaremos virtualización, y la técnica de paravirtualización son dos de las técnicas de virtualización más utilizadas [1]. Todos los modelos de virtualización tienen como objetivo utilizar al máximo las capacidades y recursos de cómputo de los equipos y por lo tanto, permiten ejecutar varios sistemas en el mismo dispositivo de forma concurrente maximizando el uso de dichos recursos físicos, tales como son la memoria, el procesador y las unidades de almacenamiento.

La principal diferencia entre la paravirtualización y virtualización; es que la paravirtualización realiza modificaciones en los sistemas invitados, con el objetivo de alcanzar un mejor rendimiento. Esta diferencia de características influye en los sistemas que no son de código abierto, tal es el caso de Microsoft Windows o Mac OS, que no permiten dichas modificaciones. Debido a esta característica de la paravirtualización, múltiples empresas tales como VMware Inc. y Citrix Systems, Inc., han creado plataformas para la gestión de entornos virtualizados aprovechando la ventaja de la virtualización que trabaja con sistemas no modificados.

A la hora de seleccionar la técnica de virtualización adecuada, la infraestructura existente es fundamental, por lo que surge la idea de llevar a cabo un estudio comparativo en ambiente de laboratorio entre las dos técnicas para detectar cuál es la mejor para cada escenario y justificar el por qué.

1.1 Antecedentes

Muchos usuarios hoy en día, están interesados en ejecutar múltiples sistemas operativos en forma concurrente, recurriendo a la solución de virtualización para la infraestructura de sus centros de datos. Una de las grandes empresas en virtualización tal como se la mencionó anteriormente es Citrix

Systems, Inc. Según estudios de esta empresa [2], los principales factores que impulsan el crecimiento de la virtualización son:

- Acceso seguro a los dispositivos móviles corporativos y de propiedad del usuario.
- Seguridad mejorada para datos y aplicaciones.
- Agilidad empresarial para una creciente fuerza laboral móvil.
- Gestión simplificada del riesgo.

La virtualización ofrece administración centralizada de escritorios, aplicaciones y datos entregados a cualquier dispositivo de punto final. La virtualización proporciona un control de acceso detallado, basado en políticas y admitiendo requisitos de cumplimiento de estándares. La virtualización además permite actualizar un conjunto completo de computadoras y dispositivos de computación, entregando datos y aplicaciones en forma segura [2].

Según un estudio de Tendencias de virtualización de escritorios para pequeñas y medianas organizaciones (PYMES) [3] realizado por *Spiceworks Community*, el 53% de las instalaciones de virtualización de escritorio en el sector de las

PYMES se desplegaron usando el programa VMware View, así mismo el programa Citrix Virtual Desktop fue utilizado en 30%, representando los otros proveedores únicamente el diecisiete por ciento (17%); demostrando así una clara tendencia de uso de plataformas de las compañías VMware Inc. y Citrix Systems, Inc.

Debido a lo anteriormente expuesto, realizamos una comparativa entre las principales técnicas: la virtualización y la paravirtualización; estudio que estuvo basado en pruebas de rendimiento de CPU y memoria.

Para determinar el número de repeticiones en las pruebas, se realizó un estudio previo con los diferentes escenarios, de los cuales se obtuvieron resultados preliminares. A partir de los cuales se determinó que el número de repeticiones para todas las pruebas fuese 200, esta información puede encontrarse en el Anexo A.

1.2 Justificación

Existen muchas razones para la adopción y la utilización de técnicas y tecnologías de virtualización en el campo empresarial. Escoger la técnica adecuada depende del

proyecto que se va a realizar; siendo en muchos casos el ahorro de costos el principal factor a la hora de empezar un proyecto de virtualización. Una ventaja de la virtualización es reducir los equipos, no sólo disminuyendo el gasto en nuevo hardware sino también reduce los consumos de electricidad, refrigeración, espacio físico y mantenimiento. Otra de las ventajas es la simplificación de gestión puesto que al encontrarse los sistemas centralizados se optimizan los recursos de personal lo cual reduce la necesidad de tener varios administradores para cada sistema en ejecución; resultando esto en que uno solo, debidamente capacitado, gestione todos los sistemas.

Mediante el estudio comparativo de las técnicas de virtualización sobre los diferentes escenarios se pudo verificar las ventajas que tiene cada una de las diferentes tecnologías de virtualización. Este estudio se realizó a través de diferentes pruebas en las que se midió el rendimiento de procesador y consumo de memoria RAM, factores críticos en los centros de cómputos. Los sistemas fueron sometidos a cargas de estrés de uso de recursos y carga de estrés de tráfico utilizando solicitudes web. Las pruebas de laboratorio escogidas nos ayudaron a simular entornos reales de trabajo tomando en

cuenta las plataformas más recientes desarrolladas y las que mayor uso han tenido en los últimos años.

El hardware utilizado para realizar las pruebas fue seleccionado para abarcar diferentes entornos arquitectónicos como son servidores y estaciones de trabajo, similares a los que se utiliza en el medio de producción con el objetivo de obtener datos parecidos y poder emitir con criterio cuál de estas dos técnicas será la más recomendable para los diferentes entornos de trabajo que tenga una empresa.

1.3 Objetivos

Los objetivos de este trabajo se dividen en dos áreas: el objetivo general abarca el trabajo completo y los objetivos específicos que ayudarán a detallar el objetivo general.

1.3.1 Objetivo General

Comparar la eficiencia de las técnicas de virtualización y paravirtualización utilizando hipervisores de distribución libre y propietaria sobre hardware heterogéneos en condiciones de laboratorios en función del rendimiento y el consumo de recursos.

1.3.2 Objetivos Específicos

- Investigar el estado actual de las plataformas de virtualización y paravirtualización.
- Instalar y configurar diferentes hipervisores y plataformas de virtualización sobre hardware heterogéneo.
- Ejecutar pruebas de eficiencia, en ambiente de laboratorio, del uso de recursos por parte del software de virtualización sobre las diferentes plataformas previamente instaladas.
- Realizar análisis estadísticos de los resultados proporcionados por las diferentes pruebas realizadas anteriormente.
- Exponer las ventajas de estas dos técnicas de virtualización en el medio empresarial y dar recomendaciones en base a las fortalezas de los tipos de sistemas.

1.4 Alcances y Restricciones

El estudio de este proyecto se centró en dos de las técnicas de virtualización más utilizadas y las pruebas se basaron en los niveles de consumo de memoria y de procesador; debido a que estos recursos o factores son los más relevantes a la hora de seleccionar algunas de las tecnologías.

Una restricción en el estudio fue la disponibilidad de los códigos fuentes o documentación especializada sobre los hipervisores o virtualizadores, debido a que estos son en algunos casos restringidos por derechos intelectuales. Además al adquirir las licencias no nos brindaba derecho sobre el código fuente.

1.5 Descripción del Proyecto

Existen empresas y comunidades que están aprovechando la necesidad de tener múltiples sistemas operativos en forma concurrente. Por esta razón podemos ver en el mercado múltiples soluciones de virtualización, cada una cumpliendo demandas específicas. Para tener un conocimiento pleno de dichas soluciones primero se debe conocer las definiciones básicas y las técnicas en las que se han basado la mayoría de las soluciones del mercado.

El proyecto comenzó con una introducción de la virtualización, las diferentes técnicas que existen, sus ventajas, desventajas y clasificación; donde nos concentramos en las diferencias de las dos técnicas de estudio escogidas para realizar una comparativa que nos mostró una visión general del desempeño de ambas. Además de un estudio del estado actual en el que se encuentra esta área; presentando las plataformas más utilizadas tanto de código libre, como propietarias con y sin adquisición de licencia.

Se escogió las plataformas: VMware vSphere ESXI (virtualización), Citrix XenServer (virtualización) y Xen (paravirtualización). Especificamos los diferentes programas de medición de recursos computacionales, algunos fueron personalizados por los autores de este proyecto y otros fueron herramientas prediseñadas. Para medición de consumo de memoria RAM tuvimos: para Linux el script *ram.sh* y el programa *hardinfo*, para Windows el *Administrador de tareas*; así mismo para medir el consumo de procesador tenemos: para Linux el script *cpu.sh* y el programa *top*; y para Windows el *Monitor de rendimiento*.

Los equipos que se utilizaron fueron: un servidor SUN Microsystems, un computador de escritorio y un servidor HP

como se aprecia en la Figura 1.1. Sobre estos tres equipos se instalaron las tres plataformas escogidas, dando un total de nueve arquitecturas de virtualización (tres plataformas por tres tipos de equipos). Para las máquinas virtuales a instalar se escogió dos distribuciones Linux con diferentes núcleos: CentOS 5.6 con núcleo 2.6.18 y Fedora 15 con núcleo *vanille* 3.6.3; así como Windows Server 2008 Standard Edition.

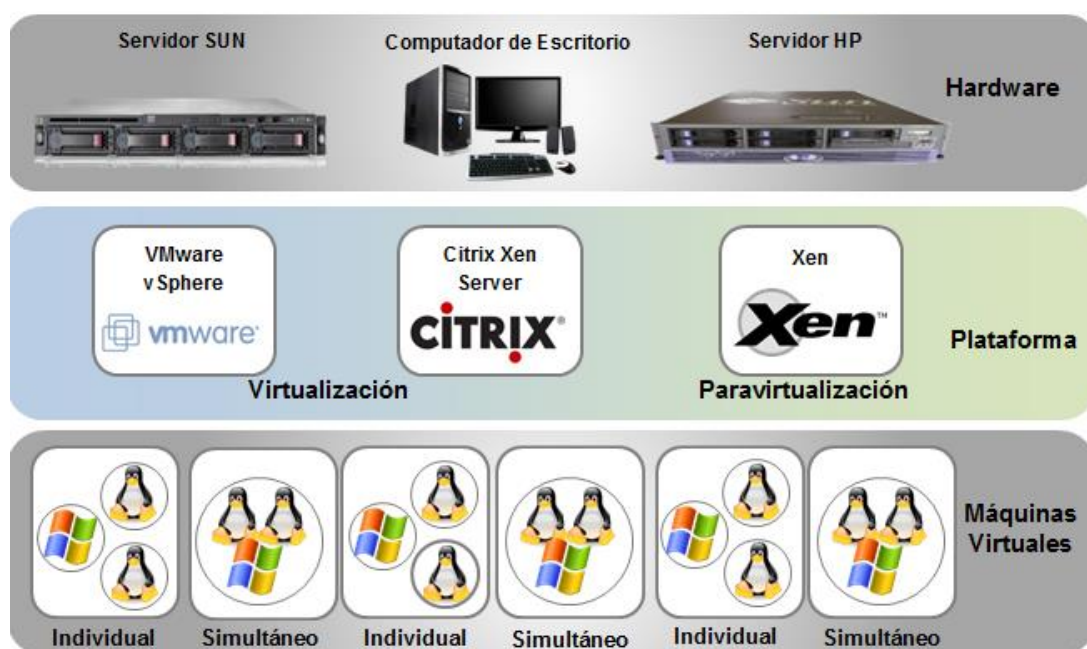


Figura 1.1 Entorno de Trabajo del Proyecto de Graduación

Se definió además las variables a medir (consumo de memoria RAM y de procesador) y los métodos para realizar las diferentes pruebas: carga de estrés de uso de recursos y carga de estrés de tráfico utilizando solicitudes web. La Figura 1.1 muestra los

dos ambientes que existieron: individual y simultáneo. La forma individual, ejecuta las pruebas usando cualquier método en solo la máquina virtual que se está evaluando, y la forma simultánea, ejecuta las pruebas con cualquier método en todas las máquinas virtuales.

Posteriormente se realizó un resumen de los escenarios para las pruebas, que consistió en la combinación de los sujetos de medición y las pruebas planteadas. Dando un total de 40 pruebas por cada plataforma, es decir que para cada equipo tendremos 120 escenarios, resultando un total de 360 por los tres tipos de hardware. En el Anexo B se detallan todos los escenarios para las pruebas. Una vez obtenidos todos los datos de las mismas se procedió a realizar la comparativa de las dos técnicas estudiadas sobre los diversos escenarios en base a los resultados para poder determinar si la hipótesis planteada en cada caso ha sido cumplida.

1.6 Metodología

Tal como vemos en la Figura 1.2 la primera etapa de la investigación estuvo orientada a recalcar las diferentes técnicas de virtualización existentes y cuáles fueron los principales programas de libre distribución como propietario. Con la

información recopilada clasificamos y documentamos las herramientas actuales, así como sus ventajas y desventajas. Del resumen obtenido, se seleccionaron las soluciones de virtualización de mayor reconocimiento y preferencia en el mercado para el proyecto de comparativa de virtualización y paravirtualización.

Continuamos con la selección de los programas de medición de rendimiento, los mismos que fueron personalizados por nosotros en unos casos y herramientas prediseñadas en otros. A partir de las soluciones de virtualización, seleccionamos los equipos como se indica en el siguiente recuadro para formar las diferentes arquitecturas de virtualización y continuamos estableciendo los sujetos de medición. Antes de realizar las pruebas, como se observa en la parte derecha de la Figura 1.2 procedimos a definir las variables o recursos a evaluar: consumo de memoria y procesador. Continuamos definiendo los métodos y ambientes usados para llevar a cabo las pruebas. Seguidamente presentamos un resumen de los escenarios y procedimos a realizar las pruebas para obtener los resultados y procesarlos; a los mismos que se les aplicaron criterios estadísticos para la eliminación de datos aberrantes para su posterior análisis y estudio. Finalmente argumentamos las

conclusiones obtenidas y la justificación de haber aceptado las hipótesis planteadas.

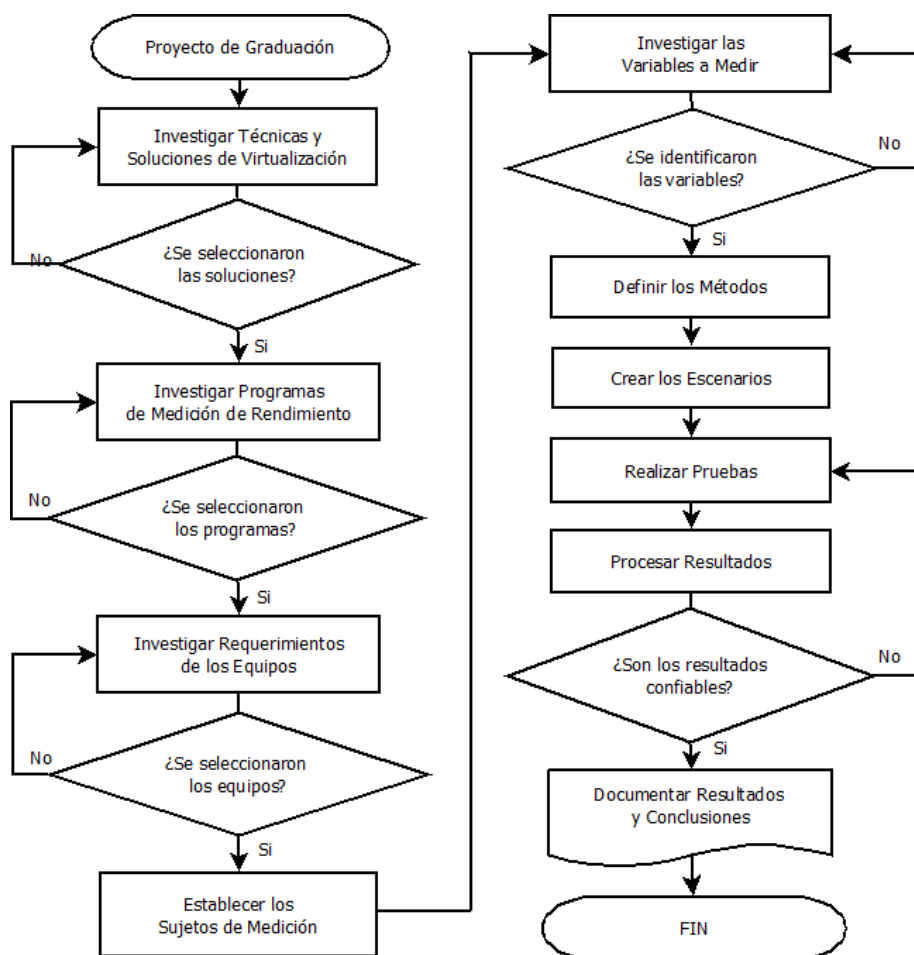


Figura 1.2 Metodología del Proyecto de Graduación

1.7 Planteamiento de la Hipótesis

Para llevar a cabo la comparativa de eficiencia entre virtualización completa y paravirtualización utilizamos diferentes hipervisores de distribución libre y propietaria sobre hardware heterogéneo en condiciones de laboratorio en función del

rendimiento y el consumo de recursos. El presente estudio tuvo dos niveles. La hipótesis de nivel uno, radicó en determinar cuál de las técnicas de virtualización presentó un mejor rendimiento para cada escenario, y la hipótesis de nivel dos, consistió en determinar la independencia del rendimiento de las técnicas sobre el hardware que se estuvo utilizando. A continuación planteamos las hipótesis que nos ayudaron a inferir en los resultados obtenidos de las pruebas realizadas.

- Hipótesis con respecto a las técnicas de virtualización:

H0: Las medias de las pruebas son estadísticamente iguales.

H1: La media de la técnica de virtualización es estadísticamente mayor o menor que la media de la técnica de paravirtualización.

- Hipótesis con respecto al hardware heterogéneo:

H0: El rendimiento de la técnica es estadísticamente independiente del hardware que se utilice para la ejecución de la misma.

H1: El rendimiento de la técnica es estadísticamente dependiente del hardware que se utilice para la ejecución de la misma.

CAPÍTULO 2

2. TECNOLOGÍAS DE VIRTUALIZACIÓN

La virtualización es una tecnología que está transformando la industria de la información y está jugando un papel muy importante en la construcción de los centros de datos. A mediados de la década de los 60, la empresa International Business Machines (IBM), dio la invención del concepto de virtualización, desarrollando las primeras soluciones [4]. Posteriormente en 1972 con la llegada del sistema VM/370 apareció el concepto de máquina virtual (VM) [5].

En los años 2005 y 2006 aparecen nuevas tecnologías desarrolladas tanto por Intel Corporation (Intel VT) como por Advanced Micro Devices Inc. (AMD-V), quienes trabajaron de forma independiente. Esto dio paso a una nueva era de soluciones de gran escala, cuyos

equipos implementan extensiones en los procesadores con arquitecturas x86 [6], esto nos permite tener diferentes soluciones de virtualización mejorando notablemente el rendimiento de los equipos de cómputo.

La consolidación de servidores es un ejemplo de virtualización y se ha convertido en una parte integral de la planificación de los centros de datos [7]. La consolidación de servicios, es decir la reducción del número de servidores aumenta el porcentaje de uso de los recursos computacionales, proporcionando ventajas en rendimiento, escalabilidad y portabilidad para un sistema. Durante este capítulo el lector encontrará una descripción detallada de los términos y las tecnologías utilizadas en nuestro proyecto, adquiriendo de una forma más sencilla los conocimientos necesarios para comprenderlo en su totalidad.

2.1 Virtualización

La virtualización se refiere a la abstracción de los recursos de una computadora [8], los mismos que serán utilizados en diferentes escenarios de operación; una capa lógica es la encargada de gestionar y entregar los recursos de hardware como CPU, memoria y dispositivos de entrada y salida, a la otra capa de clientes que son los que realizan las peticiones.

En el estudio realizado por el Instituto Nacional de Estándares y Tecnología (NIST) del Departamento de Comercio de los Estados Unidos [9], se indica que la virtualización es la simulación del software y/o hardware sobre el que se ejecuta otro software. Además existen distintas formas de virtualización, que dependerá principalmente de la arquitectura de hardware. Podemos nombrar algunos ejemplos de virtualización: consolidación de servidores, creación de entornos virtualizados de alta redundancia y con balanceo de cargas, emulación de redes, para entornos de pruebas de laboratorio, validación de software, virtualización de aplicaciones y virtualización de recursos de almacenamiento [10].

Los principales conceptos en la virtualización son: el recurso disponible para virtualizar y el ente virtualizado que va a disponer de dichos recursos para usarlos. Ambos conceptos en conjunto crean las diferentes técnicas de virtualización más conocidas hoy en día, las cuales son: Virtualización de plataforma, Virtualización de recursos, Virtualización de aplicaciones y Virtualización de escritorio.

2.2 Virtualización de Plataforma

Es la técnica de virtualización donde el recurso que se abstrae es un sistema entero, consiste en aislar todo el hardware de un equipo de cómputo [8] [11], para que múltiples instancias de sistemas operativos trabajen de forma independiente y accedan a dichos recursos, teniendo la ilusión de que estos le pertenecen en exclusiva a la máquina invitada, pero realmente está compartiendo los recursos con otros sistemas virtualizados.

Tal y como se define en el artículo de la Compañía IBM, “*Virtualización en la Educación*” [12], “La virtualización de servidores permite que un servidor físico pueda dividirse para ejecutar varios servidores virtuales seguros, esto crea una oportunidad para consolidar servidores físicos”. La consolidación de servidores actualmente es un tema necesario en los centros de datos, puesto que la reestructuración de las infraestructuras existentes a sistemas virtualizados tienen un gran adelanto principalmente en la optimización de recursos de cómputo y reducción de costos.

En este grupo de virtualización de plataforma tenemos: emulación, virtualización, paravirtualización y virtualización a

nivel de sistemas operativos. En el presente proyecto nos enfocamos en virtualización y paravirtualización.

La **emulación** es una técnica de virtualización tal como la presentamos en la Figura 2.1, que consiste en recrear una arquitectura completa de hardware a través de software de emulación especializado [13], que incluye la funcionalidad del procesador y varios periféricos, tales como discos duros, USB, CD-ROM; logrando ejecutar diferentes sistemas operativos sin modificarlos. Este programa de emulación tiene el control total de los sistemas emulados, incluyendo la ejecución de las instrucciones a nivel de CPU [14].

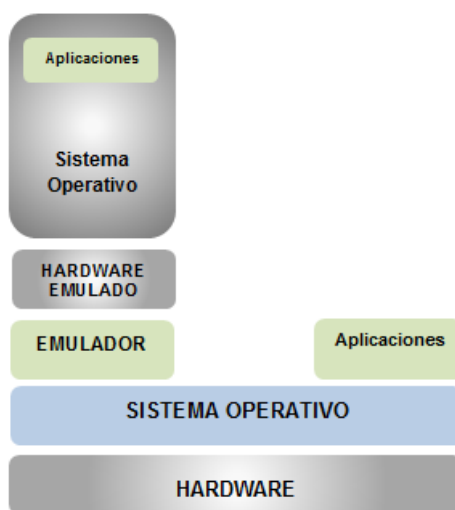


Figura 2.1 Arquitectura general de la técnica de Emulación

En esta técnica, el tiempo de respuesta para resolver una petición de algún recurso es alto, presentando penalizaciones en el rendimiento. Para cada solicitud que realice el sistema emulado puede ser necesario ejecutar más de 1000 instrucciones a la CPU real [14], ya que cada instrucción en las máquinas virtuales debe ser interpretada por el sistema anfitrión, causando problemas de lentitud en el sistema. Por lo tanto esta técnica no es la más eficiente, pero para los escenarios en donde el tiempo de respuesta no es un factor crítico, la emulación podría ser la mejor opción de virtualización, a pesar de ser la más costosa.

Una ventaja interesante de esta técnica es que permite ejecutar sistemas operativos y aplicaciones de distintas arquitecturas al del sistema anfitrión, dándonos la posibilidad de poder ejecutar un sistema operativo diseñado para una plataforma concreta sobre otra distinta, sin tener que modificarlo; emulando las condiciones necesarias para que pueda funcionar. Entre las soluciones más conocidas tenemos: Bochs, MAME, y Qemu.

La Virtualización Completa o “Virtualización” como se puede observar en la Figura 2.2, introduce el concepto de hipervisor que hace de intermediario entre el hardware del equipo y los

sistemas clientes virtualizados, en el cual se incluye código para emular el hardware subyacente de un sistema completo [1]. Este mediador conocido también como monitor de máquinas virtuales (VMM) se encarga de interceptar, traducir y ejecutar de forma segura instrucciones privilegiadas [13].

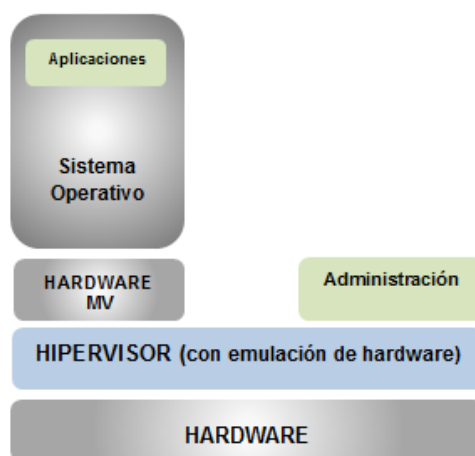


Figura 2.2 Arquitectura general de la técnica de Virtualización

Como se expuso, la virtualización incluye código para emular hardware dentro del hipervisor lo que ocasiona penalizaciones de rendimiento, pero con las nuevas tecnologías de microprocesadores: Intel-VT y AMD-V; se ha mejorado notablemente el rendimiento del sistema y adicional a ellos los sistemas virtualizados se pueden ejecutar sin necesidad de modificar su código, por lo tanto tiene la ventaja de trabajar con todo tipo de sistemas, incluido Windows.

En estos sistemas con apoyo de hardware, lo que se hace es ejecutar el hipervisor con el máximo nivel de acceso a la CPU (Anillo -1 en procesadores AMD o Intel) y los sistemas invitados se ejecutan a un nivel inferior (Anillo 0 en procesadores AMD e Intel, que era el máximo nivel cuando los procesadores no incorporaban apoyo para la virtualización) [14]. Con esta técnica apoyada por hardware, los sistemas invitados no saben que están siendo virtualizados y no tienen acceso directo a los dispositivos reales, el hipervisor será el que permite la gestión de los recursos a través de las instrucciones que el CPU informe al hipervisor.

El reto principal de la virtualización completa es la intersección y simulación de operaciones que son privilegiadas, como las instrucciones de entrada/salida. Por otro lado, las instrucciones generadas por máquinas virtuales que no afectan ni acceden a otras y a la máquina anfitriona suelen ser ejecutadas directamente por el hardware, sin simulación alguna [1].

Adicional a las mejoras que está teniendo esta técnica, empresas como VMware Inc. y Citrix Systems, Inc. han realizado adiciones a su software para optimizar el rendimiento del sistema, estas sí ayudan, pero se sigue hablando de virtualización ya que los sistemas invitados no saben que están

siendo virtualizados y será el hipervisor el encargado de gestionar todos los recursos computacionales. Ejemplos de esta técnica tenemos: VMware vSphere, Citrix XenServer y Microsoft Hyper-V.

En la **paravirtualización** también se hace uso del hipervisor como capa de virtualización como observamos en la Figura 2.3, pero además de no incluir emulación de hardware, introduce modificaciones en los sistemas invitados [13] [1], es decir dichos sistemas necesitan ser diseñados de tal manera que sepan que van a ser virtualizados y así puedan cooperar de mejor forma con la virtualización.

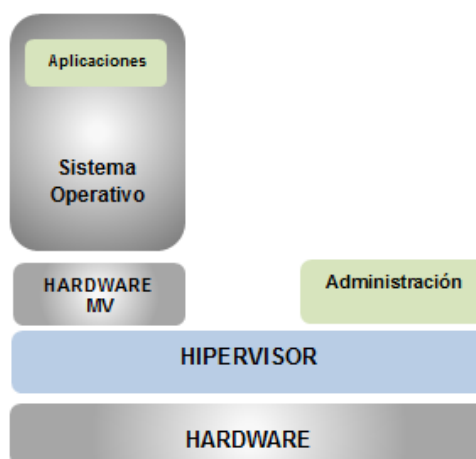


Figura 2.3 Arquitectura general de la técnica de Paravirtualización

La diferencia con la virtualización, es que al incluir estas modificaciones en lugar que el hipervisor capture las

instrucciones privilegiadas o problemáticas, es el sistema invitado quién llamará al hipervisor cuando sea necesario [14]. Esta interacción mutua impone una sobrecarga menor en el rendimiento con respecto a las demás técnicas de virtualización de plataforma. Por lo tanto la capa delgada de software de paravirtualización actúa más como un policía de tráfico, la que permite que un sistema operativo invitado acceda a los requerimientos físicos del hardware mientras se detiene a todos los demás sistemas operativos invitados el acceso a los mismos recursos a la vez [11].

Los sistemas invitados deberán estar basados en sistemas operativos que permitan estas modificaciones. A consecuencia de ello, esta técnica es usada en sistemas operativos de código libre tales como: Linux, BSD y UNIX. Así la paravirtualización se ve típicamente en entornos Linux con núcleos Xen incluido, aunque es cada vez más común encontrar vendedores de virtualización completa, que incluyen algunos controladores de paravirtualización [15], en sus últimos productos. Esta técnica aún no es utilizada en sistemas Windows debido a que estos no pueden ser modificados. En cuanto a las librerías y utilidades ejecutadas por los sistemas virtualizados deberán estar compiladas para el mismo hardware y juego de instrucciones

que el de la máquina física anfitriona. Las soluciones más extendidas e importantes dentro del paradigma de la paravirtualización son: Denali, Xen, User Mode Linux.

La Virtualización a nivel de sistema operativo Consiste en utilizar el mismo núcleo del sistema operativo de una máquina única para crear diferentes instancias o entornos virtuales [13], que son sistemas operativos invitados más ligeros, se refieren a menudo como servidores privados virtuales (VPS). Los VPS compartirán el mismo núcleo con el sistema anfitrión y será el núcleo el que determinará para quién trabaja en un momento determinado [16]. Desde la perspectiva del usuario, el sistema en su conjunto actúa como si realmente existiesen varios servidores ejecutándose en varias máquinas distintas.

Para conseguir aislar los diferentes entornos virtuales, la virtualización a nivel del sistema operativo extiende el concepto *chroot* permitiendo al sistema iniciar servidores virtuales, que trabajan de manera aislada e independiente dentro de los límites de su propio sistema de ficheros raíz [1], con este concepto se establecen diferentes directorios raíces para cada sistema creado. Esta técnica aporta bastante seguridad en lo que se refiere a accesos a sistemas de ficheros ajenos, ya que

si un servidor virtual se viese comprometido, esto afectaría únicamente a su propio sistema de ficheros raíz. En la Figura 2.4 se muestra de forma gráfica la arquitectura general de esta técnica, como observamos no existe una capa intermedia entre el hardware y los sistemas invitados. La ausencia de la capa de virtualización intermedia implica que sea necesario realizar modificaciones en el núcleo del sistema operativo anfitrión, y esto se puede realizar solamente en sistemas de distribución libre.

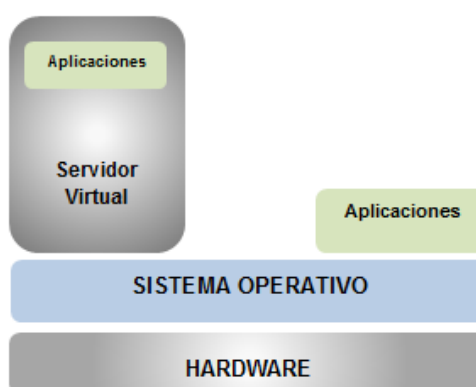


Figura 2.4 Arquitectura general de la técnica Virtualización a nivel de sistema operativo

La virtualización a nivel de sistema operativo no necesita apoyo de hardware y en muchos casos es más eficiente que otras soluciones de virtualización; pues al no usar el hipervisor el tiempo de respuesta del sistema es más eficiente y se consumen menos recursos debido a que solo existe un único

núcleo. Pero debido a esta característica de usar el mismo núcleo del sistema operativo, se corre el riesgo que ante algún daño del sistema automáticamente todos los sistemas quedan en desuso. En esta categoría tenemos a las Zonas de Solaris (Solaris Zones), OpenVZ, la técnica de Micro Partioning de AIX y FreeBSD Jails.

Todos los modelos de virtualización que hemos mencionado hasta ahora tienen elementos comunes en su arquitectura. La Figura 2.5 muestra los elementos de la arquitectura básica de la virtualización de plataforma.

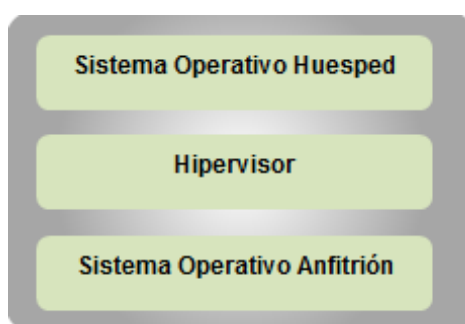


Figura 2.5 Arquitectura básica de la virtualización de plataforma

El **hipervisor** o monitor de máquinas virtuales VMM [17], es la capa lógica intermedia entre los recursos físicos del computador y los del cliente, que son los que realizan peticiones para usar los mismos; puede correr directamente sobre el equipo o sobre un sistema operativo anfitrión. Su función principal es gestionar

todas las peticiones que realizan los sistemas virtualizados y muchas veces traducirlas para poder obtener y entregar los recursos de cómputo.

Según la investigación de Eugenio Villar y Julio Gómez, el hipervisor proporciona dos funcionalidades básicas: “identifica, capta, maneja y responde a operaciones de CPU e instrucciones privilegiadas o protegidas emitidas por las máquinas virtuales; y maneja el encolado, envío y devolución de resultados de peticiones de acceso a los recursos hardware instalados en el host anfitrión por parte de las máquinas virtuales” [1].

El VMM asigna procesadores virtuales a los sistemas invitados a partir de los procesadores físicos que disponga el equipo anfitrión. De hecho, normalmente habrá más máquinas virtuales ejecutándose en el servidor, que CPU físicas tenga éste, lo que requiere de algún tipo de solución. Por lo tanto, un VMM usa un mecanismo de planificación para compartir los procesadores físicos con cada procesador virtual. Así mismo el VMM deberá manejar la asignación de memoria para los sistemas virtualizados y hacerse cargo del posible *swapping* [18] que es la técnica mediante la cual se intercambia un proceso que está en memoria por otro que no lo está, usando el espacio de

intercambio o swap, el swapping se ejecuta cuando hay escasos recursos [19].

Sobre el hipervisor se encuentran los diferentes sistemas virtualizados ejecutándose al mismo tiempo realizando peticiones simultáneas, el VMM resuelve dichas peticiones multiplexando recursos entre ellas, logrando independizar cada sistema y evitando que los recursos en las mismas sean interrumpidos u ocupados, es decir cada máquina virtual trabaja independiente de la otra ya que los recursos de memoria o espacio físico son aislados para cada sistema. El hipervisor puede ser categorizado en dos grupos [13]: VMM Tipo I y VMM Tipo II como se puede observar en la Figura 2.6.

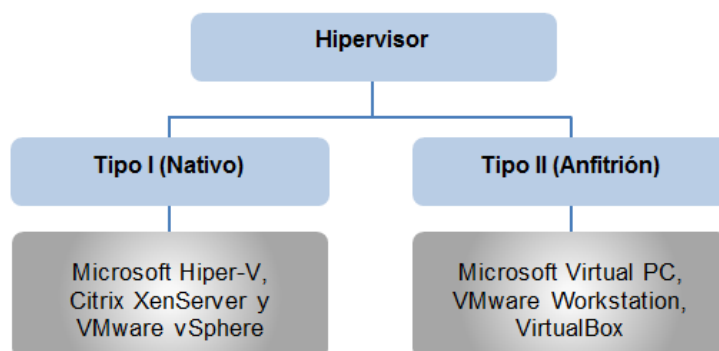


Figura 2.6 Tipos de Hipervisores

- Los **VMM Tipo I**, conocidos como nativos o *bare-metal*, son aquellos que se instalan directamente en el hardware [13], similar a cómo instalar un sistema operativo regular y tienen el

total control de la asignación de los recursos del sistema para los diferentes entornos virtualizados, la Figura 2.7 muestra la arquitectura de este tipo.

La ventaja de los hipervisores de esta categoría es que su rendimiento es mayor que en los de Tipo II ya que actúan directamente sobre el equipo, gestionar estos recursos ayuda a que las peticiones sean resueltas de forma más rápida y eficaz, logrando que el sistema sea más seguro. Ejemplos de este tipo son: VMware vSphere, Microsoft Hyper-V y Citrix XenServer.

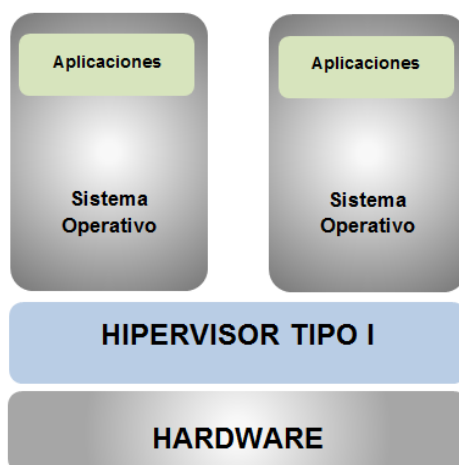


Figura 2.7 Arquitectura del Hipervisor Tipo I

- Los **VMM Tipo II**, se instalan sobre un sistema existente como se muestra en la Figura 2.8, es decir el hipervisor es una aplicación que corre sobre el sistema operativo anfitrión, ocupando recursos del sistema [13]. Debido a esta

característica, los recursos del equipo los controla el sistema anfitrión, lo que ocasiona que exista sobrecarga al pasar por una capa adicional, resultando un rendimiento menor ya que el tiempo de respuesta es más alto.

Los hipervisores de este tipo son menos seguros debido a que cualquier vulnerabilidad que comprometa al sistema operativo anfitrión afectará a los sistemas virtualizados. Ejemplos de este tipo son: VMware Workstation, Microsoft Virtual PC y VirtualBox.



Figura 2.8 Arquitectura del Hipervisor Tipo II

El **sistema operativo anfitrión** es el que ejecuta la solución de virtualización para gestionar los recursos computacionales [13], como memoria, procesador, almacenamiento y red; los mismos que son abstraídos del equipo físico para cuando los requieran

los diferentes sistemas invitados. Además es el primer sistema instalado en el equipo para soportar las diferentes máquinas virtuales.

Los **sistemas invitados** conocidos también como máquinas virtuales (VM) o sistemas virtualizados, corren simultáneamente sobre el sistema anfitrión y comparten los recursos de cómputo que son separados de éste. Una VM es un duplicado de una máquina real, eficiente y aislado [20]. Cada máquina virtual es independiente de la otra, ocupando de forma virtual los recursos de hardware sin interferencias y con diferentes cargas de trabajo.

2.3 Virtualización de Recursos

Esta técnica tiene como objetivo abstraer el recurso físico individual de un computador (espacio en disco, memoria, red, CPU, etc) [1]. Con este aislamiento de recursos ofrecemos un servicio más fiable y potente debido a las ventajas en redundancia, flexibilidad y abstracción de servicios.

Los entes que se aíslan son recursos individuales de un computador, que podríamos agruparlos para que sean vistos como un sólo recurso, o al revés, dividir un recurso en múltiples

instancias independientes [21]. Dentro de la virtualización de recursos podemos mencionar los principales modelos: virtualización de almacenamiento, de red, de entrada/salida y de memoria.

En el estudio realizado por Eugenio Villar y Julio Gómez [1], detallan a profundidad los diferentes usos de la virtualización de recursos.

- En la **virtualización de almacenamiento** el recurso aislado es el almacenamiento, creando espacios lógicos que son utilizados para distribuir la información del sistema a través de la red y poder compartirla con todos los sistemas invitados para que los usuarios accedan a ella sin conocer necesariamente donde se encuentra físicamente esa información. Esta solución es empleada comúnmente en redes de área de almacenamiento SAN y en otras como: RAID, LVM y NAS.
- La **virtualización de red** permite aislar la misma red para poder crear entornos virtuales más flexibles, permitiendo a los usuarios gestionar accesos y recursos desde cualquier ubicación a través de internet, como es el caso de la creación de redes privadas virtuales o VPNs, y el de OpenVPN o FreeS/Wan.

- La **virtualización de entrada/salida** permite aislar los protocolos de capas superiores de las conexiones físicas o del transporte físico. Los recursos que se sacan son las conexiones de entrada/salida y transporte. Ejemplos de este escenario son: Xsigo Systems, 3Leaf Systems.
- La **virtualización de memoria** es el aislamiento de los recursos de memoria, permitiendo por ejemplo que sistemas invitados consuman este recurso en forma dinámica de acuerdo con las cargas de trabajo actuales. Siendo su objetivo hacer creer a los sistemas que disponen de espacio en memoria de forma virtual, ejemplo es el espacio de intercambio en Unix.

2.4 Virtualización de Aplicaciones

La virtualización de aplicaciones describe las tecnologías de software que mejoran la portabilidad, facilidad de administración y compatibilidad de aplicaciones mediante la encapsulación de ellas desde el sistema operativo subyacente en el que se ejecutan [22]. Con esta técnica se puede ejecutar aplicaciones de cualquier arquitectura sobre sistemas para los cuales no han sido implementadas.

De la publicación “Exploring the SoftGrid Platform v4.1” realizado por Microsoft Corporation [23], adaptamos la Figura

2.9 para ilustrar cómo trabaja la solución de virtualización de aplicaciones SoftGrid de Microsoft. La aplicación virtualizada se ejecuta en el sistema operativo utilizando sus recursos (información, servicios del sistema y configuraciones); pero no se le permite modificar nada. En su lugar, la aplicación se ejecuta sobre el entorno virtual SystemGuard que actúa como una capa entre la aplicación y el sistema operativo. La capa virtual es pequeña (por lo general sólo un par de megabytes de memoria) y carga justo antes de la ejecución de la aplicación.

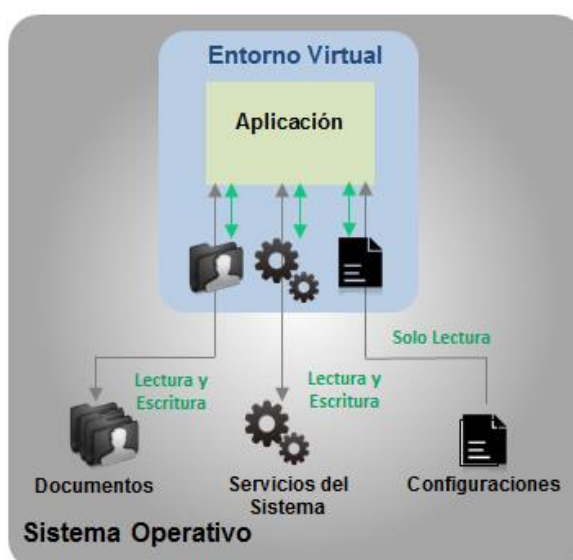


Figura 2.9 Entorno Virtual SystemGuard de Microsoft [23]

Wine es un ejemplo de programa que implementa esta técnica, el cual permite ejecutar aplicaciones Windows sobre arquitectura GNU/Linux. También dentro de este grupo está el

entorno de ejecución para lenguaje Java (JVM) de Sun Microsystems y el Common Language Runtime (CLR) para la plataforma .NET de Microsoft Corporation.

2.5 Virtualización de Escritorio

Es el concepto de aislar una instancia lógica de un sistema operativo, a la que un cliente accederá para usarla. Existen varios modelos de la virtualización de escritorio, que a grandes rasgos se pueden dividir en dos categorías en función de si la instancia del sistema operativo se ejecuta de forma local o remota [24].

El sistema operativo, aplicaciones y datos residen en un sistema centralizado, en el que podemos gestionar todos los entornos de escritorio de forma segura. Para que los usuarios interactúen con sus escritorios virtuales deben hacerlo a través de la red usando programas que ofrezcan soporte para protocolos de escritorio remoto tales como el RDP.

Esta técnica es usada en la infraestructura de escritorio virtual (VDI), donde cada usuario tiene una instancia del sistema operativo en el servidor y se ejecuta como una máquina virtual [25]. La principal ventaja que proporciona es que el sistema

responde a las diversas necesidades de los usuarios de forma flexible y administra de forma eficiente todos los entornos. Ejemplos de esta categoría son: VMware View, Sun VDI, Microsoft VDI y Citrix XenDesktop.

2.6 Comparativa entre la técnica de Virtualización y Paravirtualización

Las principales ventajas de implementar cualquiera de las técnicas que mencionamos son la optimización de recursos computacionales y la reducción de costos. Estudios realizados estiman que el porcentaje de utilización de un servidor de media está por debajo del 20% [26]. Además, los centros de datos están incluidos dentro de los grandes consumidores de energía, al ser responsables del uno punto tres por ciento (1,3%) del consumo energético mundial, según un estudio realizado por el profesor Jonathan G. Koomey [27] de la Universidad de Stanford. Con el uso de la virtualización de plataforma ha disminuido el desperdicio de los recursos de cómputo, pues dentro de un solo equipo puede haber más de un sistema virtualizado, esto ha ayudado a bajar los costos de adquisición de nuevos equipos, así como a la adecuación, mantenimiento, refrigeración, consumo de electricidad y espacio de los mismos.

Los entornos de gestión permitirán ahorrar costos de instalación, configuración, monitoreo, administración y soporte del servicio; ya que una vez establecidos todos estos procesos podrán ser gestionados de forma automática y por un sólo administrador en el mejor de los casos, en donde las características de hardware y software de las máquinas virtuales son establecidas de acuerdo a lo que realmente necesitamos. Esto logra crear una administración centralizada del sistema, es decir desde una sola consola se puede controlar todas las máquinas virtuales de la infraestructura virtualizada y realizar diferentes actividades de administración.

La migración en caliente y el balanceo dinámico de máquinas virtuales se usan en la consolidación de servidores para replicar máquinas virtuales y tener una alta disponibilidad en los sistemas, lo cual es otra ventaja que nos proporciona virtualizar los servidores. En la migración en caliente generalmente se presenta una parada de alrededor de 60 a 300 ms [16], que es necesaria para realizar la sincronización final antes de que la máquina virtual comience a ejecutarse en su destino final. Al presentarse alguna falla de un sistema invitado, se replica de forma rápida y segura los sistemas lógicos hacia otra máquina virtual del mismo equipo o hacia otro equipo y al agregar

balanceo dinámico a nuestra infraestructura proporcionamos altas mejoras de rendimiento, garantizando que cada máquina virtual ejecute en el servidor físico que se encuentre en condiciones óptimas.

Por otra parte, la independencia de entornos virtualizados proporciona mayor nivel de seguridad, ya que si un sistema se encuentra comprometido no afectaría al resto. Cada máquina virtual tendrá un acceso diferente al modo superusuario, si ocurre algún ataque de seguridad solamente se afectaría la máquina donde sucedió la intrusión.

A pesar de sus múltiples ventajas, la virtualización tiene también algunas desventajas. Con la virtualización al tener diferentes sistemas virtualizados ocupando los mismos recursos de un computador puede provocar problemas, ya que los equipos de cómputo se ven sometidos a altas cargas de procesamiento y consumo de memoria, introduciendo penalizaciones, por lo tanto un sistema operativo virtualizado no tendrá el mismo rendimiento que si estuviera directamente instalado sobre el hardware. Además las aplicaciones y procesos se ejecutarán más despacio.

Para evitar estos problemas de rendimiento es necesario disponer de equipos con suficientes recursos para que soporten toda la infraestructura de nuestro proyecto de virtualización, esto puede provocar aumento en gastos de equipos de cómputo de última generación; pero a largo plazo presentaría una mayor ventaja, por lo cual es indispensable estudiar el proyecto en detalle antes de implementarlo y escoger el equipo más acorde a la situación que se quiere solucionar.

El software de virtualización nos impondrá una serie de dispositivos virtuales como tarjetas de vídeo y red que no se podrán modificar [1]. Además a pesar de que son sistemas independientes, puede darse que el sistema anfitrión sea el comprometido, en este caso corremos el riesgo que toda nuestra infraestructura se vea afectada. En la Tabla I se puede observar una comparativa de las dos técnicas de nuestro estudio: virtualización y paravirtualización; de acuerdo al estudio previo realizado en cuanto a sus características, ventajas y desventajas.

Tabla I Comparativa entre Virtualización y Paravirtualización

Característica	Virtualización	Paravirtualización
Soporte de Hardware	Arquitecturas x86	Arquitecturas x86, x86-64, IA-64
Recurso Abstraído	Plataforma Hardware Completa	Plataforma Hardware Completa
Administración centralizada y simplificada	Los gestores de administración incluyen mejoras para el manejo de los recursos computacionales	Los programas de administración permiten configuraciones básicas
Seguridad	La seguridad la garantiza el software de virtualización	Los sistemas invitados saben que están siendo virtualizados y ayudan a mejorar la seguridad del sistema
Modificación del Sistema Invitado/ Compatibilidad	Sistema invitados no modificados y excelente compatibilidad	Sistemas invitados modificados, poca compatibilidad y no disponible para sistemas Windows
Rendimiento	Bajo rendimiento cuando no se usan equipos de última generación	Rendimiento estable

En este capítulo hemos presentado los conceptos básicos para comprender las diferentes técnicas de virtualización, en el siguiente nos enfocamos en conocer las diferentes soluciones de virtualización, para escoger las plataformas de virtualización y realizar la comparativa.

CAPÍTULO 3

3. SOFTWARE DE VIRTUALIZACIÓN Y PARAVIRTUALIZACIÓN

La tecnología de virtualización tiene muchos beneficios como hemos mencionado con anterioridad. Existen diversas compañías que ofrecen soluciones de virtualización, entre estas podemos encontrar las de código libre y las propietarias con y sin costo de adquisición de licencia, con las que se puede montar plataformas serias de bajo costo, pasando por las opciones gratuitas de los grandes fabricantes VMware Inc., Microsoft Corporation y Citrix Systems, Inc., hasta las soluciones comerciales de los mismos fabricantes, más potentes y con mayor número de funcionalidades. De entre todas ellas nombraremos

a continuación algunas de las soluciones de virtualización y paravirtualización más utilizadas y comercializadas en la actualidad.

3.1 Software de Virtualización

Entre las soluciones de virtualización disponibles en el mercado podemos encontrar gratuitas y propietarias. Comenzaremos nombrando algunas soluciones bases que han servido para que las grandes empresas ya mencionadas presenten plataformas robustas.

QEmu es un emulador de procesadores que soporta dos modos de ejecución: emulación del sistema completo y emulación del modo usuario [28]. Este programa ha sido usado como complemento de otras soluciones como KVM, proporcionando toda la funcionalidad de hardware virtual para los sistemas virtualizados.

KVM (*Kernel-Based Virtual Machines*) apareció como una solución de virtualización de código libre para GNU/Linux que se incluyó oficialmente a partir del núcleo 2.6.20 en el año 2007 [29], el cual utiliza el núcleo de Linux como hipervisor, de manera que el control de los dispositivos reales, la planificación

de tareas y la gestión de memoria del sistema anfitrión los hace el núcleo. Este programa se compone de varios módulos cargables para cada tipo de procesador, sea este Intel o AMD, entre los cuales tenemos el "*kvm.ko*", que proporciona la infraestructura de virtualización [30].

VirtualBox es otra solución de virtualización que forma parte de la familia de productos de Oracle; ofreciendo algunas funcionalidades como la ejecución de máquinas virtuales de forma remota, la implementación de controladores virtuales USB y arranque por red PXE. Además permite la implementación de software adicional para los sistemas operativos virtualizados logrando una integración y un mayor rendimiento; y en cuanto a la emulación de hardware, se presenta la opción de montar imágenes ISO como unidades virtuales ópticas, los discos duros de los sistemas invitados son almacenados como archivos individuales en un contenedor, y tiene un paquete de controladores que permiten aceleración en 3D, pantalla completa, placas PCI Ethernet, integración con teclado y ratón [31].

VMware Inc. es una empresa pionera en el sector de virtualización [32]. En la actualidad ofrece la solución **VMware vSphere**, siendo el primer sistema basado en la informática de nube [33], que usa un hipervisor de Tipo I. Existen dos arquitecturas de VMware vSphere en el mercado: vSphere ESX, y vSphere ESXi. En el núcleo de la arquitectura ESX, se aumentó una partición de gestión conocido como Sistema Operativo de Consola, COS; en cambio en la arquitectura ESXi, la COS se ha eliminado y todos los agentes se ejecutan directamente en el VMkernel [34]. VMware vSphere además para mejorar el rendimiento y la eficiencia en la comunicación entre el sistema invitado y el hipervisor, ha proporcionado una interfaz, el Virtual Machine Interface (VMI) que habilita la paravirtualización transparente permitiendo que el núcleo del sistema invitado pueda ejecutarse de forma nativa en el hardware virtualizado o en el hipervisor en modo paravirtualizado. El enfoque de VMware vSphere, frente a la virtualización puede ser descrito como una traducción binaria; que modifica instrucciones sensibles que se están ejecutando a instrucciones virtualizables. El resto de instrucciones, tales como las que se ejecutan en el modo de usuario, se realizan sin modificar. Debido a que la traducción binaria se efectúa en el

código binario que ejecuta el procesador no requiere cambios en el núcleo del sistema operativo invitado [32].

Virtual PC, es la solución de virtualización desarrollado originalmente por Connectix [35]. Este programa virtualiza sistemas Windows y Macintosh. Para el caso de Windows, no emula el procesador sino que deja que él mismo ejecute las instrucciones en forma nativa; por el contrario, en el caso de Mac OS se da una recompilación dinámica para traducir el código x86 a código de un Macintosh con procesador PowerPC [36]. La última versión de este software es Windows Virtual PC, que fue renombrado por motivo de la salida al mercado de Windows 7.

Microsoft posteriormente lanzó la solución de virtualización **Hyper-V**; conocida anteriormente como Windows Server Virtualization, que es un hipervisor nativo que permite la virtualización de plataformas en sistemas de 64 bits con procesadores que tengan tecnología de virtualización [37]. El programa implementa el aislamiento de las máquinas virtuales en términos de una partición. Una instancia del hipervisor debe tener al menos una partición primaria, corriendo una versión compatible de Windows Server, la misma que será el dominio

de control. En cambio, las particiones secundarias no tienen acceso directo a los recursos de hardware, sino que tienen una vista virtual de los recursos. Cualquier solicitud de los dispositivos virtuales y su correspondiente respuesta es redirigida, a través de los canales de comunicación, *VMBus* [38].

Citrix Systems, Inc. es otra de las principales empresas desarrolladora de soluciones de virtualización, que en la actualidad ofrece la solución de virtualización de servidores **Citrix XenServer**, basada en el potente hipervisor de 64-bit, Xen [39]. Está diseñada para una gestión eficiente de servidores virtuales Windows y Linux, proporcionando consolidación de servidores. El programa utiliza una combinación de paravirtualización y virtualización asistida por hardware, como lo podemos apreciar en la Figura 3.1. Las peticiones de las máquinas virtuales son realizadas de dos formas: la primera es que el sistema invitado llama al hipervisor para resolver dichas peticiones y la otra forma es aprovechar la virtualización asistida por hardware que ofrece los procesadores Intel VT y AMD-V en el caso de Windows, y de forma directa en el caso de Linux [40].

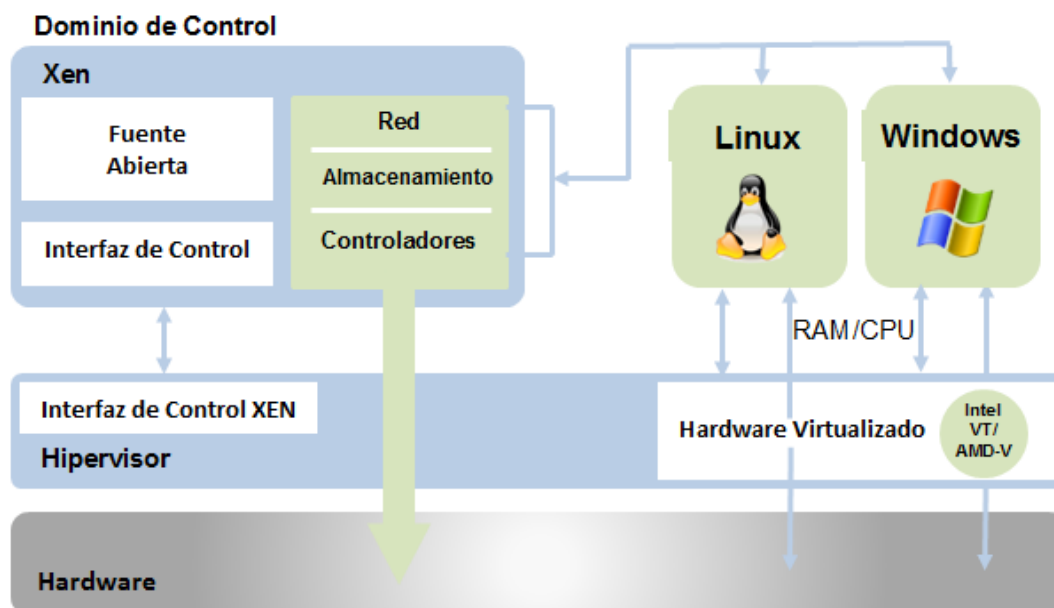


Figura 3.1: Arquitectura del hipervisor de XenServer [40]

Con Citrix XenServer, todas las interacciones de la máquina virtual con el hardware se gestionan a través del *domain 0*, el cual es una máquina virtual con privilegios especiales que se ejecuta en la parte superior del hipervisor. Este dominio realiza una instancia fortalecida y optimizado de Linux, lo que resulta en un soporte de hardware extremadamente amplio [40].

3.2 Software de Paravirtualización

En el área de paravirtualización también existen múltiples soluciones, tanto de libre distribución como propietarias; profundizaremos el estudio de la solución Xen.

Denali es un monitor de máquinas virtuales, que permite que los servicios que no son de confianza se ejecuten en dominios aislados [41]. Utiliza la paravirtualización para dar soporte de alto rendimiento a las máquinas virtuales, para ejecutar sistemas operativos ligeros con aplicaciones que brindan servicios de Internet [42].

Xen es la solución más conocida que utiliza conceptos de paravirtualización hasta la actualidad [11], es de código libre y proporcionado por XenSource para obtener paravirtualización a nivel de sistema operativo [43]. Su objetivo es poder ejecutar instancias de sistemas operativos con todas sus características, de forma completamente funcional en un equipo sencillo. Xen precisa colaboración (modificaciones en el sistema operativo alojado), por lo cual, solo pueden paravirtualizarse sistemas operativos modificados. Debido a esta característica, Xen en el caso de los sistemas operativos que no son de código libre, nos ofrece la opción de virtualización completa únicamente bajo procesadores Intel VT o AMD-V [43], ya que estos procesadores permiten al hipervisor de Xen ejecutarse en el anillo de máxima prioridad. La documentación oficial de XenSource acerca del producto Xen [44], nos indica que el

ambiente virtual de Xen tiene varios elementos que trabajan en conjunto para ofrecer los entornos de virtualización que un cliente desea implementar. En la Figura 3.2, podemos apreciar estos elementos:

- Hipervisor Xen
- Dominio 0 invitado o Dominio de Administración y Control (Xen DM&C)
- Dominio U invitado (Dom U) o PVM invitado o HVM invitado



Figura 3.2 Ambiente virtual de Xen [44]

El **hipervisor Xen** como vemos en la Figura 3.2, es la capa base y la primera que arranca, que se instala directamente en el hardware por debajo de todos los sistemas operativos [44]. Es responsable de la administración de la CPU y de la gestión de la memoria de las diferentes máquinas virtuales que se

ejecutan. El hipervisor no sólo separa el hardware de las máquinas virtuales, también controla la ejecución de las mismas.

El **Dominio 0**, es una máquina virtual con una versión modificada del núcleo Linux, que se ejecuta en el hipervisor Xen con privilegios de acceso a los recursos físicos de entrada y salida, e interactúa con las otras máquinas (Dominio U: PVM o HVM invitadas) que se ejecuten [44]. Todos los entornos de virtualización Xen requiere del dominio 0 para poder iniciar las otras máquinas virtuales, la Figura 3.2 ilustra de forma gráfica la ubicación de este elemento.

Los **Dominios U invitados** (domU = Unprivileged domain, o dominio sin privilegio), como se aprecia en la Figura 3.2, no tienen acceso directo al hardware físico en el equipo como el dominio 0. Del mismo informe difundido por XenSource podemos decir que Xen despliega dos clases de máquinas virtuales: [44].

- Las máquinas virtuales que se ejecutan paravirtualizadas en un hipervisor Xen se las conoce como dominio U o PVM invitado, donde los sistemas operativos son modificados, tales como: Linux, Solaris, FreeBSD y otros sistemas operativos UNIX. En estos dominios, la

máquina virtual es consciente que otros equipos virtuales se están ejecutando en la misma máquina.

- Las máquinas completamente virtualizadas que se ejecutan en un hipervisor Xen se les conoce como dominio U o HVM invitado, tales como Windows estándar o cualquier otro sistema operativo sin cambios. En estos dominios, la máquina virtual no es consciente de que comparte el tiempo de procesamiento del hardware y que otras máquinas virtuales estén presentes.

UML (User-mode Linux), es un núcleo Linux que se ejecuta como un proceso; éste nos permite que un sistema Linux ejecute otros sistemas Linux en el espacio del usuario. Cada sistema operativo alojado existe como un proceso en el anfitrión [45]; lo que permite a varios sistemas ejecutarse en el contexto de un solo núcleo. Puesto que los núcleos alojados se ejecutan en el espacio del usuario, estos deben estar compilados para este uso (aunque puede tratarse de diferentes versiones de núcleos).

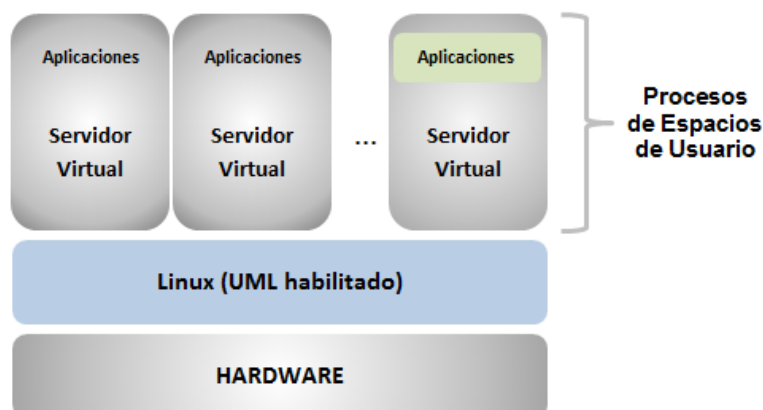


Figura 3.3 Esquema de virtualización UML

Como podemos observar en la Figura 3.3, existe un núcleo anfitrión (que se ejecutará sobre el hardware) y uno o varios núcleos alojados (que se ejecutarán en el espacio de usuario del núcleo anfitrión) [45]. Debido a que los sistemas alojados y el sistema anfitrión son ambos Linux con estructuras prácticamente idénticas, la comunicación pasa muy eficientemente del sistema alojado al anfitrión, requiriendo muy pocas traducciones.

3.3 Comparativa

Como podemos observar en la Tabla II, el emulador QEmu puede virtualizar una amplia gama de sistemas, en comparación con el resto de soluciones; pero esa técnica de emulación sufre penalizaciones en el rendimiento de las máquinas virtuales por

la sobrecarga que se ejerce al traducir las peticiones de las mismas, ocasionando respuestas más lentas en comparación con el resto de soluciones. En cambio las alternativas como KVM y VirtualBox están siendo fuertemente promovidas por la fácil gestión de administración y las mejoras de rendimiento con respecto a los programas de emulación.

Tabla II Resumen de soluciones a evaluar.

SOLUCIÓN	TÉCNICA DE VIRTUALIZACIÓN	LICENCIA	SISTEMA ANFITRIÓN	SISTEMA INVITADO
QEmu	Emulación	Código libre	Windows, Linux, Mac OS, Solaris, FreeBSD.	Varios [46]
KVM	Virtualización	Código Libre	FreeBSD, Linux	FreeBSD, Linux, Solaris, Windows
Virtual Box	Virtualización	Propietario con y sin costo de adquisición	Windows, Mac OS, Linux, Solaris.	Windows, Linux, Solaris, BSD, Mac OS
Virtual PC	Virtualización	Propietario sin costo de adquisición	Windows, Mac OS	Windows.
VMware vSphere	Virtualización	Propietario con y sin costo de adquisición	No existe	Windows, Linux, Solaris, FreeBSD, Mac OS
Hyper-V	Virtualización	Propietario con y sin costo de adquisición	Windows Server 2008 o superior	Windows, Linux

Tabla III Resumen de soluciones a evaluar (continuación).

SOLUCIÓN	TÉCNICA DE VIRTUALIZACIÓN	LICENCIA	SISTEMA ANFITRIÓN	SISTEMA INVITADO
Citrix Xen Server	Virtualización	Propietario con y sin costo de adquisición	No existe	Linux, Windows
Denali	Paravirtualización	Código libre	Denali	NetBSD
UML	Paravirtualización	Código Libre	Linux	Linux
Xen	Paravirtualización y Virtualización (con procesadores Intel-VT o AMD-V)	Código libre	NetBSD, Linux, Solaris	FreeBSD, NetBSD, Linux, Solaris, Windows

En cuanto a las soluciones de virtualización propietaria como se apreciar en la Tabla II, la empresa Microsoft Corporation ofrece como solución: Virtual PC y Hyper-V como sus soluciones más conocidas en su comunidad de usuarios, siendo estas muy óptimas a la hora de virtualizar sistemas Windows, pero solo Hyper-V permite virtualizar otros sistemas tales como Linux. VMware Inc. en cambio ofrece VMware vSphere, el cual utiliza a un hipervisor *bare-metal* y puede agregar otros componentes, como por ejemplo el VMI, que habilita la paravirtualización transparente para mejorar el rendimiento de las máquinas virtuales. Citrix Systems, Inc. por su parte destaca a Citrix

XenServer, el cual está basado en el reconocido hipervisor Xen, que incorpora los conceptos de la paravirtualización de forma nativa, permitiendo de esta forma que las máquinas virtuales presenten mejor rendimiento y no requiere de un sistema anfitrión.

El programa UML usa conceptos de la paravirtualización, para crear máquinas virtuales, esta característica es utilizada de manera que tanto el anfitrión como los invitados sean sistemas Linux. En cambio, el sistema Denali, usa los mismos conceptos para potenciar el rendimiento de las máquinas virtuales, pero estas son sistemas básicos usados para pruebas de códigos o programas poco confiables y no soportan sistemas complejos. Otra solución de paravirtualización muy conocida es Xen, por el rendimiento que presentan las máquinas virtuales y por el amplio número de sistemas virtualizados que son soportados en comparación con los dos anteriores.

CAPÍTULO 4

4. SOFTWARE DE MEDICIÓN DE RENDIMIENTO

La virtualización de servidores ofrece un enorme potencial de ahorro debido a la consolidación de las cargas de trabajo, esto reduce notablemente tanto el consumo de energía como la demanda de refrigeración. Para garantizar el alto desempeño y la alta disponibilidad de los servicios alojados en los servidores; la capacidad de procesamiento y la cantidad de memoria del equipo anfitrión son factores críticos al momento de adoptar esta tecnología ya que todos estos servicios residen en un único equipo. Por lo anteriormente expuesto, las pruebas para la presente comparativa nos basamos en el rendimiento de CPU y de memoria RAM.

Las técnicas de medición de rendimiento, *benchmarks* en inglés, han evolucionado hasta su situación actual. Un *benchmark* es la prueba que se realiza a través de un programa que evalúa las prestaciones de un sistema, con el objetivo de estimar el rendimiento de un elemento concreto para poder comparar los resultados con sistemas similares [47]. Los resultados de las comparaciones y la metodología abren grandes debates en las diferentes comunidades aportando de esta manera nuevos aspectos para que los diseñadores mejoren rápidamente dichos elementos.

En general los *benchmarks* se agrupan en paquetes o *suites*, conjuntos de programas que miden aspectos de un sistema. Existen paquetes especializados para cada sistema operativo, en el caso de Linux resalta *SYSSTAT*, el cual consta de varias herramientas (*sar*, *iostat*, *mpstat* y otros comandos Linux) que ofrecen funcionalidades avanzadas de monitoreo del sistema [48]. Cada sistema además brinda herramientas de medición propias como en el caso de Windows que trae al *Administrador de tareas*, el mismo que proporciona información sobre los programas y procesos que se ejecutan mostrando indicadores claves de rendimiento [49].

4.1 Evaluación de Memoria RAM

Según la investigación de Carlos Torres y Elena Ballano, en cuanto a la evaluación de memoria RAM: “Existen políticas descritas en lo que se refiere a la gestión de la memoria: unas se apoyan en las funciones que les ofrecen los procesadores para controlar las direcciones de memoria mientras que otras aplican sus propios algoritmos” [50]. El sistema de gestión de Linux hace uso de las ventajas de la segmentación y de los circuitos de los procesadores para traducir direcciones [51]. Además gestiona niveles de memoria principal y secundaria estos son: paginación, por demanda y por expulsión [50]. Una porción de memoria RAM está permanentemente asignada al núcleo, la cual es utilizada para almacenar el código y estructuras de datos estáticas del mismo. La parte restante de memoria RAM se denomina memoria dinámica, y ésta es asignada sólo cuando es estrictamente necesaria y liberada tan pronto como sea posible. Para aumentar la eficiencia se ha definido un mecanismo en el cual las páginas (espacio contiguo de direcciones de tamaño fijo que tiene una dirección base y atributos) son agrupadas en marcos (espacio de la memoria principal) de tamaño fijo. Para reemplazar páginas, se utiliza el algoritmo del reloj, en el cual tienen asignada una variable de

edad. Cuando es necesario sustituir una página, la que no ha sido referenciada en bastante tiempo es la mejor candidata a ser reemplazada [51].

Cuando un proceso se ejecuta en Windows, el sistema establece un espacio de direcciones virtuales propio de 32 bits, que permite habilitar un espacio de hasta 4 gigabytes de memoria. Éste está formado por la suma de la memoria RAM instalada, más la memoria virtual asignada. No todos los procesos caben en esta memoria a la vez, debido a que existen muchos de ellos o su tamaño es demasiado grande. Cuando esto sucede, Windows alterna la permanencia de éstos en memoria, sacando unos y poniendo otros para que todos puedan ejecutarse. Utiliza la memoria virtual asignada a cada proceso para guardar los datos cada vez que se saca de la memoria RAM, a éste intercambio entre memoria física y memoria virtual se le conoce como *swapping* [50]. Mientras que en Linux el *swapping* se da cuando “expulsa” un proceso (*swap out*) copiando su imagen a una área de intercambio, de acuerdo a diversos criterios como prioridad o preferencia y cuando existe espacio en la memoria principal, se intercambian los procesos a la misma copiando la información desde el área de intercambio (*swap in*) [51].

Debido a la gestión diferente de la memoria RAM por parte de cada sistema operativo, se seleccionaron programas de medición compatibles. Para el caso de Linux, tuvimos dos ambientes en los que trabajan los programas de medición, líneas de comandos e interfaz gráfica. En el ambiente de líneas de comandos se usó el programa *sar*, el cual es una de las herramientas que recoge, muestra y guarda la información de la actividad del sistema ya sea esta de CPU, memoria, discos, interrupciones, interfaces de red, tablas del núcleo, entre otras [52]. Se puede especificar parámetros para seleccionar que información acerca del sistema se desea mostrar. En el caso de requerir un informe de memoria usada que conste de cinco muestras; tomada cada dos segundos, sería requerido el comando que se muestra en la Figura 4.1.

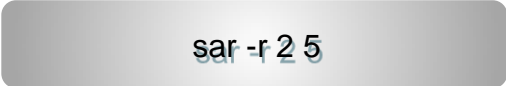
A screenshot of a terminal window showing the command `sar -r 2 5` entered. The text is centered within a light gray rounded rectangular background.

Figura 4.1 Comando para medir consumo de memoria RAM con la herramienta *sar*

Para la ejecución de esta herramienta se implementó un script personalizado, al mismo que lo denominamos *ram.sh*, para estandarizar parámetros y automatizar la obtención de resultados, el mismo puede verse en el Anexo C.

En el ambiente gráfico en los entornos Linux se utilizó el programa *hardinfo*, que permite obtener información sobre el sistema operativo, generar informes en formato HTML o texto plano de la utilización de hardware, y ejecutar ciertas pruebas *benchmark* [53].

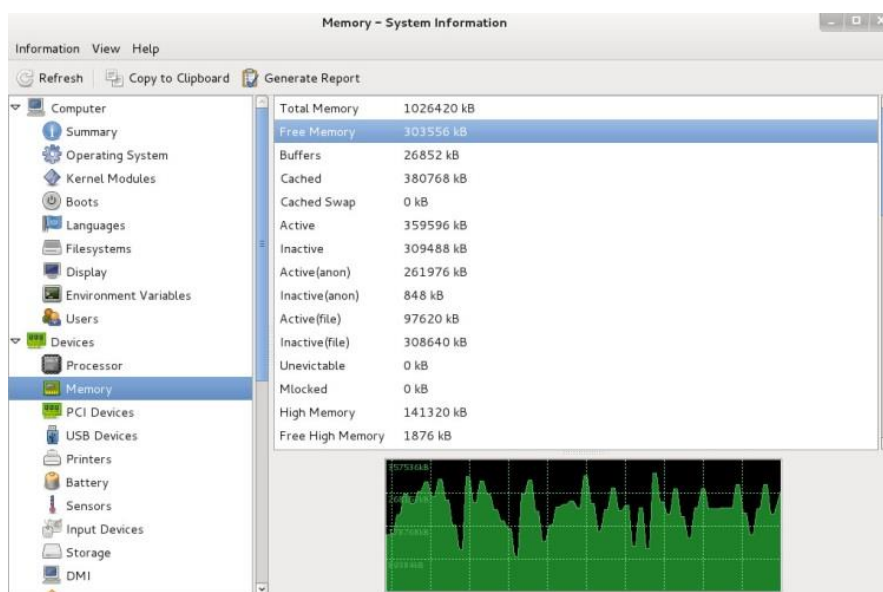


Figura 4.2 Captura de *hardinfo* del consumo de memoria RAM

Como podemos apreciar en la Figura 4.2, existen en la parte izquierda tres secciones principales: *computer*, *devices* y *benchmark*. En *computer* podemos encontrar detalladamente información básica del sistema como es: la versión del núcleo, sistema de archivo utilizado, usuarios del sistema entre otras características; en *devices* encontramos la descripción del hardware como es: tipo de procesador, memoria RAM,

dispositivos PCI, entre otros; finalmente en *benchmark* se puede ejecutar ciertas pruebas como *CPU blowfish*, *CPU cryptohash*, *CPU fibonacci*, entre otras. Nosotros nos centraremos en la sección de memoria RAM de *devices*, la misma que nos permite ver de forma dinámica la memoria consumida por el sistema.

Igualmente que en el caso del comando *sar*, se implementó un script personalizado denominado *ramGUI.sh*, para realizar la captura de imágenes de forma automática. En el cual se añadió el paquete *ImageMagick*, que permite crear, editar, componer, convertir o capturar imágenes [54]. El código del mismo se puede encontrar en el Anexo C.

Como mencionamos anteriormente existen herramientas de medición propias de cada sistema; en el caso de Windows se utilizó el *Administrador de tareas*, el cual nos permite ver los programas, procesos y servicios que está ejecutando el sistema. Además de supervisar el rendimiento del equipo a nivel de memoria y capacidad de procesador, ver el estado de la red y qué usuarios están conectados a la PC [49].

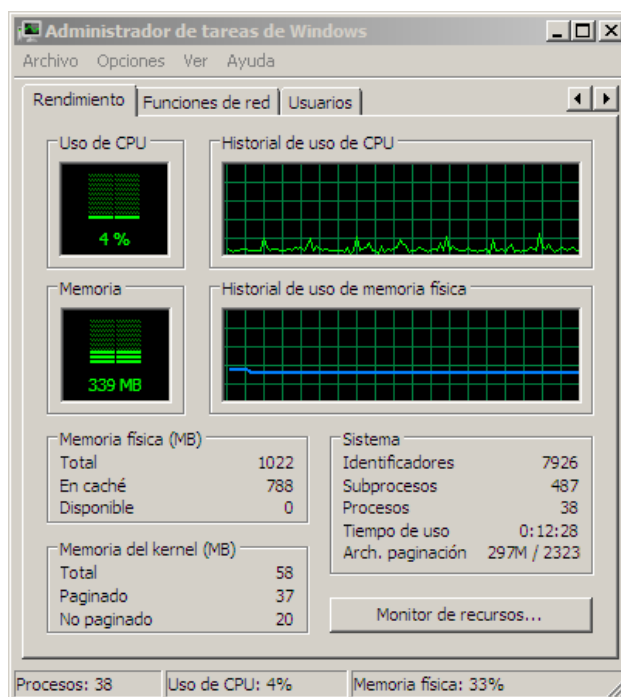


Figura 4.3 Captura del *Administrador de tareas* de Windows.

En la Figura 4.3, en la sección inferior observamos el número procesos en ejecución, el porcentaje usado del procesador, y el porcentaje de memoria utilizada; en la sección superior encontramos varias pestañas: aplicaciones, procesos, servicios, rendimiento, funciones de red y usuarios. Siendo la de mayor interés para nuestro estudio la de Rendimiento; la cual muestra dos gráficos en tiempo real (una representa el uso del procesador y la otra la memoria en uso) junto con varios detalles como el tiempo que el sistema ha estado funcionando y la cantidad de memoria virtual disponible.

4.2 Evaluación de CPU

El uso del procesador se debe a los diferentes procesos o servicios que se ejecutan en el sistema. Al igual que en el caso de la evaluación del consumo de RAM, para el CPU existen paquetes especializados o herramientas propias de cada sistema.

Para el caso de Linux, en el ambiente de líneas de comandos se volvió a usar la herramienta *sar* pero sin ningún parámetro, ya que de forma predeterminada devuelve consumo del procesador [52]. Así por ejemplo, en el caso de requerir un informe de consumo de procesador que conste de cinco muestras tomadas cada dos segundos, sería necesario el comando como se observa en la Figura 4.4.

A terminal window with a dark background and light text. The text displayed is the command 'sar 2 5'.

Figura 4.4 Comando para medir consumo de CPU con la herramienta *sar*

La implementación de esta herramienta se realizó a través de un script personalizado, al que se lo denominó *cpu.sh*, para estandarizar parámetros y automatizar la obtención de resultados; el código del mismo puede ser revisado en el Anexo C de este documento.

Para el ambiente gráfico en entornos Linux se utilizó el comando `top` que permite obtener una información en tiempo real del procesador además de proporcionar una lista de los procesos más intensos en lo referente a consumo de CPU [55] como podemos observar en la Figura 4.5.

```

virtual@virtual:/home/virtual
File Edit View Search Terminal Tabs Help
virtual@virtual:/home/virtual/Desktop/virtual/scripts
top - 15:48:33 up 14 min, 3 users, load average: 9.13, 8.17, 4.74
tasks: 146 total, 10 running, 136 sleeping, 0 stopped, 0 zombie
Cpu(s): 18.0%us, 81.3%sy, 0.0%ni, 0.0%id, 0.0%wa, 0.7%hi, 0.0%si, 0.0%st
Mem:  1031224k total,  622656k used,  408568k free,  23876k buffers
Swap: 2031612k total,    0k used, 2031612k free,  171864k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 1621 root        20   0  130m  69m  180  R  11.0   6.9   1:09.67 stress
 1622 root        20   0  2056    60    0  R  11.0   0.0   1:09.37 stress
 1619 root        20   0  2056    60    0  R  10.6   0.0   1:09.66 stress
 1620 root        20   0  2056    60    0  R  10.6   0.0   1:09.36 stress
 1623 root        20   0  130m  93m  180  R  10.6   9.3   1:09.66 stress
 1624 root        20   0  2056    60    0  R  10.6   0.0   1:09.37 stress
 1625 root        20   0  130m  89m  180  R  10.6   8.9   1:09.66 stress
 1626 root        20   0  2056    60    0  R  10.6   0.0   1:09.36 stress
 1627 root        20   0  2056    60    0  R  10.6   0.0   1:09.36 stress
  596 root         0   0    0    0    0  S  0.7   0.0   0:04.69 flush-253:1
 1741 root        20   0    0    0    0  S  0.7   0.0   0:00.22 flush-8:0
  986 root        20   0 55272  27m 6988  S  0.3   2.7   0:01.95 Xorg
 1742 root        20   0    0    0    0  S  0.3   0.0   0:00.19 flush-0:35
    1 root        20   0 15652  13m 1928  S  0.0   1.4   0:02.53 systemd
    2 root         0   0    0    0    0  S  0.0   0.0   0:00.00 kthreadd
    3 root         0   0    0    0    0  S  0.0   0.0   0:00.09 ksoftirqd/0
    5 root         0 -20    0    0    0  S  0.0   0.0   0:00.00 kworker/0:0H
    6 root         0   0    0    0    0  S  0.0   0.0   0:00.01 kworker/u:0
    7 root         0 -20    0    0    0  S  0.0   0.0   0:00.00 kworker/u:0H
    8 root        RT   0    0    0    0  S  0.0   0.0   0:00.00 migration/0
    9 root        RT   0    0    0    0  S  0.0   0.0   0:00.00 watchdog/0
   10 root         0 -20    0    0    0  S  0.0   0.0   0:00.00 cpuset
   11 root         0 -20    0    0    0  S  0.0   0.0   0:00.00 khelper
   12 root        20   0    0    0    0  S  0.0   0.0   0:00.00 kdevtmpfs
   13 root         0 -20    0    0    0  S  0.0   0.0   0:00.00 netns
   14 root        20   0    0    0    0  S  0.0   0.0   0:00.00 bdi-default

```

Figura 4.5 Captura del resultado obtenido del comando `top`

El comando `top` se actualiza cada cinco segundos de forma predeterminada. En la parte superior de la Figura 4.5, se muestra el tiempo que el sistema lleva activo, y los tres promedios de procesos ejecutándose en el sistema durante los últimos minutos: uno, cinco y quince respectivamente. A

continuación se encuentra el número total de procesos que se ejecutan actualmente desde la última actualización, estos se dividen en el número de tareas ejecutas, dormidas, detenidas o muertas. En la tercera línea, se presenta el porcentaje de tiempo de CPU (us = modo usuario, sy = modo del sistema, ni = modo de usuario de bajo prioridad, id = tarea inactiva, wa = "I/O en espera, hi = mantenimiento de IRQs, si = servicio IRQs flexibles, st = tiempo dedicado a otras instancias domU); en las siguientes líneas se demuestran estadísticas sobre el uso de memoria y el espacio de intercambio además del listado de procesos activos con sus respectivos identificadores [55].

En este caso, para la toma de datos se implementó el script *cpuGUI.sh* para capturar las imágenes del comando *top* en forma automatizada. El código del mismo puede verse en el Anexo C.

Como sucedió en la medición del consumo de RAM en ambientes Windows, para medir el consumo de CPU se utilizó la herramienta propia que tiene este sistema, el *Monitor de rendimiento* como se muestra en la Figura 4.6, que nos permite obtener de forma visual contadores de rendimientos del sistema, ya sea en tiempo real o histórico [56]. Este presenta en

forma más exacta el consumo de CPU, en comparación con el *Administrador de tareas*.

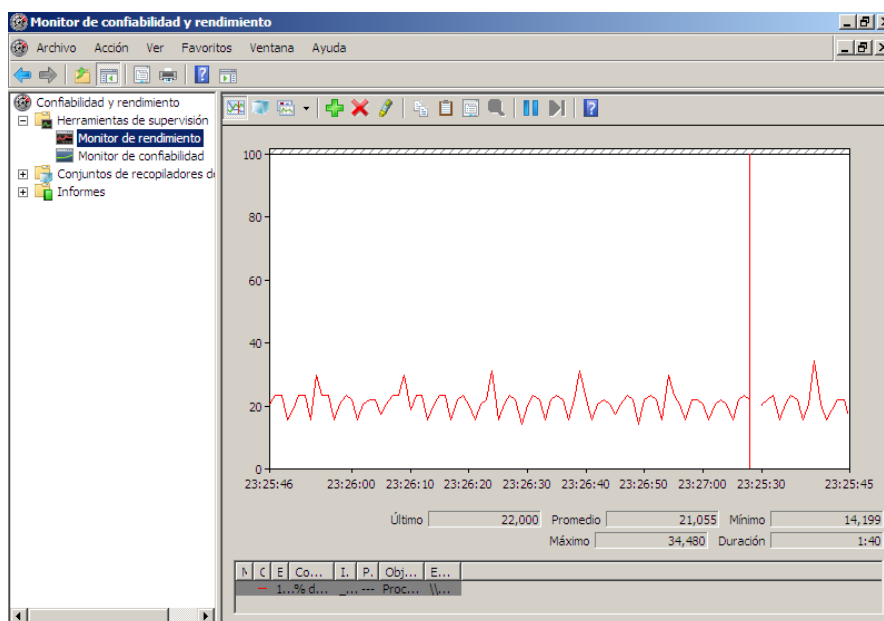


Figura 4.6 Captura del *Monitor de rendimiento* de Windows

La captura de pantalla del *Monitor de rendimiento*, Figura 4.6, muestra una gráfica histórica del consumo de CPU por defecto presentando el máximo, mínimo y el actual dato registrado durante todo el tiempo que se lleva monitoreando el recurso.

CAPÍTULO 5

5. ESCENARIOS

A continuación el lector encontrará las especificaciones de hardware de los sistemas anfitriones y las configuraciones de software de los sistemas invitados para llevar a cabo nuestra comparativa. Para cada máquina virtual se estandarizó la instalación y configuración de los sistemas operativos así como los programas de medición; con esto aseguramos que todos los sistemas virtualizados tengan las mismas características y las pruebas con las diferentes plataformas de virtualización se realicen bajo las mismas condiciones. Además se describe el proceso para realizar las pruebas sobre los diferentes escenarios.

5.1 Arquitecturas para las Pruebas

Para las pruebas se contó con: un Servidor Sun Microsystems Sunfire X2200, un Computador de Escritorio y un Servidor HP Proliant DL160 G6. El servidor “Sun Microsystems Sunfire X2200” mostrado en la Figura 5.1, es ideal para clústeres de cómputo de alto rendimiento que ejecutan aplicaciones que consumen mucha memoria, como los programas de análisis de fluidos en ingeniería: Fluent o PowerFLOW, así como los de alta escala para arquitecturas de aplicaciones Web [57]. Las características principales de este equipo son: procesador 4x Dual-Core AMD Opteron™ con frecuencia 1.8 GHz, memoria RAM de 6618MB y dos discos duros: ATA SEAGATE y ATA HITACHI de 250 GB cada uno. Además cuenta con una unidad óptica DVD RW AD-7530.



Figura 5.1 Imagen del Servidor Sun Microsystems Sunfire X2200 [57]

El computador de escritorio mostrado en la Figura 5.2, es un equipo de última generación a nivel de procesador, ideal para tareas digitales de alto rendimiento, el cual es escalable

permitiendo agregar dispositivos y características adicionales. El equipo cuenta con un procesador Intel® Core™ i7-2600 CPU @ 3.40GHz, memoria RAM de 8192 MB, un disco duro SAMSUNG SATA de 1TB y un DVDRAM HL-DT-ST GH22NS90.



Figura 5.2 Imagen del Computador de Escritorio

El servidor “HP Proliant DL160 G6” que se ilustra en la Figura 5.3, es un equipo de alto rendimiento, bajo costo, diseñado para entornos informáticos de alto rendimiento, e implantaciones de servidores web y de gran cantidad de memoria [58]. El servidor tiene un procesador 4x Intel® Xeon® CPU X3430 con frecuencia 2.40GHz, memoria RAM de 4258MB, un disco duro ATA de 250 GB y una unidad óptica MATSHITA DVD RAM UJ862A.



Figura 5.3 Imagen del Servidor HP Proliant DL160 G6 [58]

Como se planteó en la descripción del proyecto, sobre los tres equipos se instalaron las plataformas de virtualización. En la Tabla IV presentamos un resumen de las instalaciones (Inst.) de hardware y las plataformas de virtualización.

Tabla IV Resumen de hardware y las plataformas de virtualización

Inst.	Servidor SUN	Computador de Escritorio	Servidor HP
1	VMware vSphere VMW	Xen Paravirtualización XDP	Citrix XenServer XEN
2	Xen Paravirtualización XDP	Citrix XenServer XEN	VMware vSphere VMW
3	Citrix XenServer XEN	VMware vSphere VMW	Xen Paravirtualización XDP

En el caso de los sistemas invitados se establecieron parámetros de hardware que fueron configurados sobre cada plataforma en el momento de instalar las diferentes máquinas virtuales. Se asignó para cada máquina virtual 1024 MB de memoria RAM, 16 GB de disco duro, un procesador y una tarjeta de red en modo puente.

De lo mencionado, la Figura 5.4 muestra una captura de pantalla del administrador de máquinas virtuales de Citrix XenServer, donde se puede observar los recursos de procesador y memoria asignados a una máquina virtual.

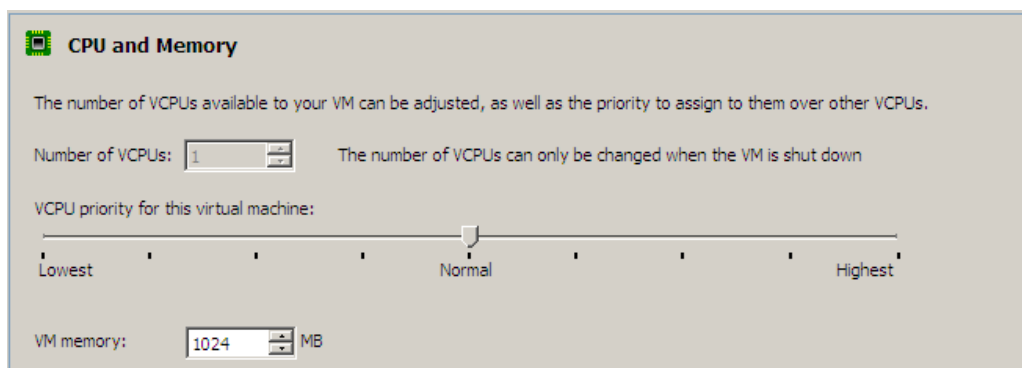


Figura 5.4 Imagen de los parámetros de una máquina virtual desde Citrix XenCenter

Una vez definidos los recursos de cómputo de los sistemas anfitriones y huéspedes, presentamos en la tabla IV un resumen de las nueve arquitecturas generadas por la combinación del hardware y las plataformas de virtualización, sobre cada combinación se instalaron las máquinas virtuales definidas anteriormente: Fedora 15, CentOS 5.6 y Windows Server 2008; resultando un total de 27 sujetos de medición (SDM), que son los entornos donde se realizaron las diferentes pruebas de rendimiento.

Tabla V Resumen de arquitecturas y sujetos de medición

SDM	Hardware	Plataforma	Máquinas virtuales
1	Servidor Sun Microsystems Sunfire X2200	VMware vSphere	Fedora 15 con <i>kernel vanille</i> 3.6.3
2			CentOS 5.6 con <i>kernel</i> 2.6.18
3			Windows Server 2008 Standard Edition
4		Citrix XenServer	Fedora 15 con <i>kernel vanille</i> 3.6.3
5			CentOS 5.6 con <i>kernel</i> 2.6.18
6			Windows Server 2008 Standard Edition
7		Xen	Fedora 15 con <i>kernel vanille</i> 3.6.3
8			CentOS 5.6 con <i>kernel</i> 2.6.18
9			Windows Server 2008 Standard Edition
10	Computador de Escritorio	VMware vSphere	Fedora 15 con <i>kernel vanille</i> 3.6.3
11			CentOS 5.6 con <i>kernel</i> 2.6.18
12			Windows Server 2008 Standard Edition
13		Citrix XenServer	Fedora 15 con <i>kernel vanille</i> 3.6.3
14			CentOS 5.6 con <i>kernel</i> 2.6.18
15			Windows Server 2008 Standard Edition
16		Xen	Fedora 15 con <i>kernel vanille</i> 3.6.3
17			CentOS 5.6 con <i>kernel</i> 2.6.18
18			Windows Server 2008 Standard Edition
19	Servidor Sun Microsystems Sunfire X2200	VMware vSphere	Fedora 15 con <i>kernel vanille</i> 3.6.3
20			CentOS 5.6 con <i>kernel</i> 2.6.18
21			Windows Server 2008 Standard Edition
22		Citrix XenServer	Fedora 15 con <i>kernel vanille</i> 3.6.3
23			CentOS 5.6 con <i>kernel</i> 2.6.18
24			Windows Server 2008 Standard Edition
25		Xen	Fedora 15 con <i>kernel vanille</i> 3.6.3
26			CentOS 5.6 con <i>kernel</i> 2.6.18
27			Windows Server 2008 Standard Edition

En cada sistema se instalaron programas necesarios para el funcionamiento del software de medición y ejecución de pruebas. Para todas las máquinas virtuales con distribución Linux se instalaron paquetes básicos de Linux como GNOME, *graphical internet*, *text-bases internet*, *base*, *dialup networking support*, *X Windows system*; paquetes adicionales como *gcc*, *stress*, *sysstat*, *ImageMagick*, el programa *hardinfo*, el servidor *Apache 2.2.22* y una página Web de prueba. Para el caso de Windows, también se utilizaron algunos programas como *google chrome*, *appmini*, *HeavyLoad*, *CPUKiller*, *winrar*, el servidor *Apache 2.2.22* y una página Web de prueba.

5.2 Descripción de las Pruebas

Una vez establecidos los sujetos y softwares de medición, inmediatamente detallaremos en profundidad los pasos previos para empezar a realizar las diferentes pruebas. Como se observa en la Figura 5.5 se empezó definiendo las variables a medir para realizar la comparativa estadística y los métodos escogidos. Con esta información resumimos los escenarios que se crearon como resultado de todas las combinaciones para finalmente proceder a realizar las pruebas.

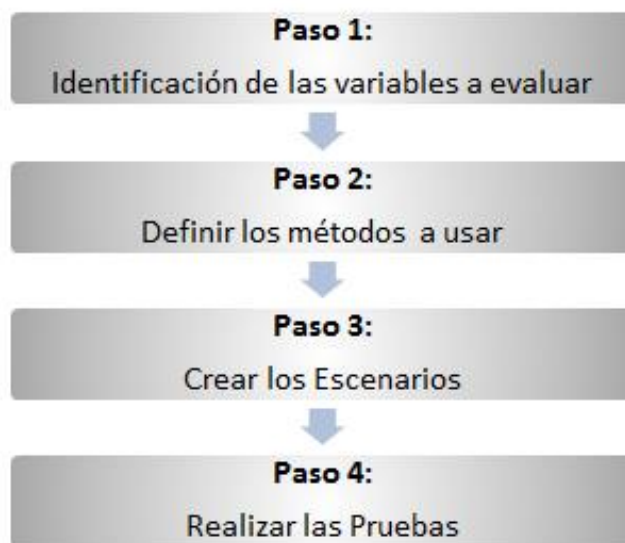


Figura 5.5 Diagrama de Flujo para elaborar las pruebas

A partir de la elección de las variables para la comparativa se procedió a realizar un conjunto de pruebas sobre los diferentes escenarios, usando métodos y estableciendo el número de interacciones para cada prueba; todos estos factores serán detallados en este apartado. Para nuestra comparativa entre virtualización y paravirtualización en el análisis del rendimiento del consumo de recursos se usaron las siguientes variables:

- **Consumo de Memoria:** Se refiere al porcentaje de consumo de memoria del sistema.
- **Consumo de CPU:** Se refiere al porcentaje de consumo de procesador del sistema.

Los métodos se refieren a las situaciones que fueron sometidos los sujetos de medición para realizar alguna prueba. Para el presente proyecto se estableció dos métodos: método de carga de estrés de uso de recursos y método de carga de estrés de tráfico utilizando solicitudes web, los mismos que fueron usados en dos ambientes diferentes: simultáneo e individual.

- **Individual:** Cuando se ejecuta la prueba utilizando cualquier método en solo la máquina virtual que se está evaluando
- **Simultáneo:** Cuando se ejecuta la prueba utilizando cualquier método en todas las máquinas virtuales

El método de carga de estrés de uso de recursos consistió en realizar carga al sistema, tanto al procesador como a la memoria; para conocer cómo responden los recursos de cómputo, con el fin de detectar si la plataforma de virtualización cumple con los requerimientos al sobrecargar el sistema. Tanto Linux como Windows utilizarán programas que serán los encargados de realizar esta carga.

Para los entornos Linux se utilizó el comando *stress*, que es un generador de carga de trabajo deliberadamente simple para sistemas *POSIX*, escrito en C, y es software libre licenciado con

licencia GPLv2. Este programa configura una cantidad de procesador, memoria, entrada/salida, y disco para generar carga al sistema [59]. Para usar este método, generamos estrés con el comando que se muestra en la Figura 5.6.

```
stress --cpu 1 --io 5 --vm 3 --vm-bytes 128M --timeout 3600s
```

Dónde:

--*cpu* significa el número de procesos vinculados a la carga

--*io* es el número de procesos de entrada y salida

--*vm* es el número de máquinas virtuales

--*vm-bytes* es el número de bytes usado por máquina virtual (por defecto es 256 MB)

--*timeout* es el tiempo de espera

Fuente: Sitio web del proyecto stress [59]

Figura 5.6 Comando para generar estrés al sistema

Para los entornos Windows en cambio, se utilizó dos programas: *HeavyLoad* para carga de memoria y *CPUKiller* para carga de procesador. *HeavyLoad* es una herramienta gratuita que permite crear una carga pesada con el objetivo de probar si el sistema seguirá funcionando con fiabilidad [60]. *CPUKiller* es un software capaz de ralentizar el computador hasta en un 99% [61], que fue usado para generar carga al procesador.

El método de carga de estrés de tráfico utilizando solicitudes web, consistió en sobrecargar con peticiones http al servidor

web instalado en los sistemas invitados para medir el rendimiento del mismo. Para todas las pruebas se utilizó el comando *ab* instalado en una máquina virtual Linux con distribución Ubuntu dentro de la misma LAN de nuestra infraestructura.

El programa *ab*, es una herramienta de evaluación comparativa de *Apache*, que está diseñado para dar una impresión de cómo una instalación web funciona; este en especial muestra cómo la instalación es capaz de servir muchas peticiones por segundo [62]. Una vez instalado podemos usarlo por ejemplo para ejecutar una prueba de 100 solicitudes con una concurrencia de diez conexiones, tal como se puede observar en la Figura 5.7.

```
ab -n 100 -c 10 http://[direcciónIP_VM]/
```

Dónde:

-n es el número de solicitudes, el valor predeterminado es una única solicitud.

-c es el número de solicitudes concurrentes, siendo el valor predeterminado uno

Fuente: Sitio web oficial de Apache [62].

Figura 5.7 Comando para generar carga de tráfico

La respuesta del comando *ab* para el ejemplo anterior después de culminar todas las peticiones http se muestra en la Figura 5.8. Entre los parámetros de respuesta podemos apreciar el

tiempo que llevo hacer la prueba (*time taken for test*), la cantidad de peticiones que el servidor pudo resolver por segundo (*requests per second*), la tasa de transferencia (*transfer rate*), y al final un detalle del porcentaje de conexiones según el tiempo que tomaron [63].

```

root@patu-VirtualBox:/home/patu# ab -n 720000 http://200.126.14.238/prueba/
This is ApacheBench, Version 2.3 <$Revision: 655654 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 200.126.14.238 (be patient)
Completed 72000 requests
Completed 144000 requests
Completed 216000 requests
Completed 288000 requests
Completed 360000 requests
Completed 432000 requests
Completed 504000 requests
Completed 576000 requests
Completed 648000 requests
Completed 720000 requests
Finished 720000 requests

Server Software:      Apache/2.2.3
Server Hostname:     200.126.14.238
Server Port:         80

Document Path:       /prueba/
Document Length:     11654 bytes

Concurrency Level:   1
Time taken for tests: 4440.196 seconds
Complete requests:   720000
Failed requests:     0
Write errors:        0
Total transferred:   8583120000 bytes
HTML transferred:    8390880000 bytes
Requests per second: 162.15 [#/sec] (mean)
Time per request:    6.167 [ms] (mean)
Time per request:    6.167 [ms] (mean, across all concurrent requests)
Transfer rate:       1887.74 [Kbytes/sec] received

Connection Times (ms)
              min    mean[+/-sd] median    max
Connect:     1     2   1.9      2     1000
Processing:  2     3   1.5      3     162
Waiting:     0     2   1.3      1     162
Total:       3     6   2.4      5     1007

Percentage of the requests served within a certain time (ms)
 50%    5
 66%    6
 75%    6
 80%    7
 90%    8
 95%    9
 98%   13
 99%   15
100%  1007 (longest request)

```

Figura 5.8 Respuesta de ejecución del comando *ab* en un cliente Linux

5.3 Escenarios para las Pruebas

Una vez detallados los requerimientos para realizar las pruebas, se procedió a establecer los diferentes escenarios, que consisten en la combinación de los sujetos de medición ya establecidos (Tabla V) y las pruebas (tablas VI y VII). Una prueba de nuestra comparativa es la combinación entre: los ambientes de trabajo, las variables a evaluar, los métodos y el software de medición. Toda la lista completa de los escenarios se la puede encontrar en el Anexo B.

Tabla VI Resumen de pruebas sobre un sujeto de medición Linux

Ambiente	Variable	Método	Software de Medición
Individual/ Simultaneo	RAM	carga de estrés de uso de recursos	<i>ram.sh</i>
		carga de estrés de tráfico utilizando solicitudes web	<i>ramGUI.sh</i>
	CPU	carga de estrés de uso de recursos	<i>cpu.sh</i>
		carga de estrés de tráfico utilizando solicitudes web	<i>cpuGUI.sh</i>

Tenemos un total de ocho pruebas para medir el rendimiento de consumo de memoria y procesador, multiplicando por los dos ambientes tenemos 16 pruebas para cada sujeto de medición Linux.

Tabla VII Resumen de pruebas sobre un sujeto de medición Windows

Ambiente	Variable	Método	Software de Medición
Individual/ Simultaneo	RAM	carga de estrés de uso de recursos	<i>Administrador de tareas</i>
		carga de estrés de tráfico utilizando solicitudes web	
	CPU	carga de estrés de uso de recursos	<i>Monitor de rendimiento</i>
		carga de estrés de tráfico utilizando solicitudes web	

Hay un total de cuatro pruebas para medir el rendimiento de consumo de memoria y procesador, al multiplicar por los dos ambientes tenemos ocho pruebas para cada sujeto de medición Windows. Resumiendo el número de pruebas para el caso de Linux fue de 288 escenarios que resultan de la combinación de los dieciocho sujetos de medición con las dieciséis pruebas correspondientes; en cambio para el caso de Windows totaliza 72 escenarios de la combinación entre los nuevos sujetos de medición y las ocho pruebas correspondientes, dando un resultado de 360 escenarios.

En el siguiente capítulo ponemos en práctica cada una de las pruebas sobre los diferentes sujetos de medición, para obtener los datos que serán procesados y presentados en diferentes tablas, a partir de los cuales procederemos a realizar la comparativa.

CAPÍTULO 6

6. PRUEBAS Y RESULTADOS

A partir de las hipótesis planteadas en el primer capítulo se procedió a realizar las comparativas sobre diferentes escenarios de pruebas. Comparativa de técnicas, la cual consiste en determinar cuál de las mismas presenta un mejor rendimiento en cada escenario. Y Comparativa de hardware heterogéneo, para determinar la independencia del rendimiento de las técnicas sobre el hardware que se esté utilizando.

6.1 Requerimientos mínimos

Para realizar las diferentes pruebas sobre los sujetos de medición, se emplearon los equipos anteriormente expuesto en

el capítulo 5, siendo estos: un servidor Sun Microsystems Sunfire X2200, una computador de Escritorio, y un servidor HP Proliant DL160 G6, teniendo como base las siguientes características: memoria RAM de 3072 MB o más, procesador de 4 núcleos o superior, una tarjeta de red, un disco duro de 100 GB y una unidad de reproducción DVD-RW.

6.2 Pruebas de Rendimiento

Como se explicó en el capítulo anterior, se generaron un conjunto de pruebas que tiene que ver con la combinación de los ambientes de trabajo, las variables a evaluar, los métodos y el software de medición. Estas pruebas se realizaron sobre los diferentes sujetos de medición y el número de datos que se obtuvieron por cada una de ellas fue de 200, valor obtenido por el estudio preliminar que se efectuó, la misma que puede ser revisado en el Anexo A.

6.2.1 Memoria RAM

Al medir el consumo de memoria RAM, en cuanto a Linux se realizaron scripts personalizados, *ram.sh* (usó el programa *sar*) y *ramGUI.sh* (usó el programa *hardinfo*), ambos nos ayudaron a manejar los datos de forma más

organizada, para su posterior análisis. En el caso de Windows se usó solamente el *Administrador de Tareas*. Todas las pruebas se ejecutaron con los métodos de carga de estrés de uso de recursos y carga de estrés de tráfico utilizando solicitudes web; así como bajo dos ambientes: individual y simultáneo.

6.2.2 CPU

Al evaluar el consumo de procesador, en Linux utilizamos scripts personalizados, *cpu.sh* (usó el programa *sar*) y el script *cpuGUI.sh* (usó el programa *top*), ambos nos ayudaron a manejar los datos de forma más automatizada, para su posterior análisis. En el caso de Windows se usó solamente el *Monitor de rendimiento*. Todas las pruebas se efectuaron con los métodos de carga de estrés de uso de recursos y carga de estrés de tráfico a través de solicitudes web; así como bajo dos ambientes: individual y simultáneo.

6.3 Resultados

Los valores que mostramos en la Tabla VIII a la Tabla XVI representan el valor promedio de consumo de memoria RAM

sobre los diferentes sujetos de medición. De la Tabla XVII a la Tabla XXV se observan los valores promedios de consumo de procesador sobre los diferentes sujetos de medición. Los valores promedios de las diferentes tablas han pasado el proceso de depuración de datos que consistió en eliminar los valores aberrantes. Toda la información detallada puede encontrarse en el Anexo D de este documento.

Resultados de las pruebas de RAM

Tabla VIII Resultados promedio de consumo de RAM obtenidos en Centos sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	21.26	20.16	19.02	21.58	20.52	19.72
	<i>ramGUI.sh</i>	21.48	21.18	20.69	21.96	21.20	20.80
ab	<i>ram.sh</i>	3.97	3.10	3.75	3.73	2.88	3.37
	<i>ramGUI.sh</i>	9.65	4.94	5.05	5.31	4.60	4.82

Tabla IX Resultados promedio de consumo de RAM obtenidos en Fedora sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	21.35	23.86	21.78	21.57	23.95	22.71
	<i>ramGUI.sh</i>	22.38	23.90	23.82	23.69	28.36	23.93
ab	<i>ram.sh</i>	3.52	3.46	3.43	3.82	3.50	3.46
	<i>ramGUI.sh</i>	5.70	5.64	5.12	12.95	5.57	5.25

Tabla X Resultados promedio de consumo de RAM obtenidos en Windows sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Administrador de tareas</i>	2.05	1.96	1.30	2.04	1.93	0.40
ab	<i>Administrador de tareas</i>	2.85	2.82	1.04	2.75	2.41	0.92

Tabla XI Resultados promedio de consumo de RAM obtenidos en Centos sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	20.03	19.84	19.62	20.70	20.38	19.89
	<i>ramGUI.sh</i>	21.71	22.86	21.10	21.93	23.08	21.02
ab	<i>ram.sh</i>	5.72	4.64	4.62	3.85	3.81	1.95
	<i>ramGUI.sh</i>	7.15	6.45	6.33	5.08	5.02	3.37

Tabla XII Resultados promedio de consumo de RAM obtenidos en Fedora sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	21.69	21.53	21.14	22.37	21.77	21.47
	<i>ramGUI.sh</i>	23.22	23.12	22.92	23.75	23.72	23.33
ab	<i>ram.sh</i>	6.21	5.51	5.33	5.03	4.95	4.64
	<i>ramGUI.sh</i>	11.62	7.83	7.67	6.78	6.14	5.65

Tabla XIII Resultados promedio de consumo de RAM obtenidos en Windows sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Administrador de tareas</i>	2.48	1.14	1.13	0.58	0.52	0.49
ab	<i>Administrador de tareas</i>	3.07	2.89	1.41	2.36	2.13	2.10

Tabla XIV Resultados promedio de consumo de RAM obtenidos en Centos sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	19.63	21.22	18.84	20.07	21.94	19.28
	<i>ramGUI.sh</i>	20.63	22.97	20.24	21.27	23.74	20.85
ab	<i>ram.sh</i>	3.72	3.22	3.91	3.49	3.04	3.58
	<i>ramGUI.sh</i>	5.28	4.72	5.50	5.02	4.61	5.04

Tabla XV Resultados promedio de consumo de RAM obtenidos en Fedora sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>ram.sh</i>	21.19	20.77	21.52	21.45	20.89	23.69
	<i>ramGUI.sh</i>	22.28	22.39	22.09	24.64	26.38	24.04
ab	<i>ram.sh</i>	3.72	3.32	3.29	3.53	3.52	3.45
	<i>ramGUI.sh</i>	8.67	5.73	5.46	4.94	5.44	5.34

Tabla XVI Resultados promedio de consumo de RAM obtenidos en Windows sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Administrador de tareas</i>	2.64	2.08	1.97	2.18	1.95	1.66
ab	<i>Administrador de tareas</i>	2.39	1.82	1.80	2.08	1.94	1.69

Resultados de las pruebas de CPU

Tabla XVII Resultados promedio de consumo de CPU obtenidos en Centos sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	90.05	84.94	89.14	90.18	87.22	89.29
	<i>cpuGUI.sh</i>	90.85	87.74	89.25	91.12	88.68	89.81
ab	<i>cpu.sh</i>	2.31	0.45	0.09	2.46	2.28	0.42
	<i>cpuGUI.sh</i>	2.40	0.78	0.32	2.95	2.67	0.77

Tabla XVIII Resultados promedio de consumo de CPU obtenidos en Fedora sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	83.64	82.66	82.63	83.08	82.46	82.35
	<i>cpuGUI.sh</i>	80.98	80.35	80.32	80.36	80.32	79.70
ab	<i>cpu.sh</i>	1.39	1.44	0.54	1.76	3.21	1.45
	<i>cpuGUI.sh</i>	1.99	2.02	0.78	2.89	4.41	1.97

Tabla XIX Resultados promedio de consumo de CPU obtenidos en Windows sobre el servidor SUN

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Monitor de rendimiento</i>	19.38	21.22	22.01	20.32	21.22	21.56
ab	<i>Monitor de rendimiento</i>	15.50	45.49	60.95	15.47	53.75	54.28

Tabla XX Resultados promedio de consumo de CPU obtenidos en Centos sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	88.73	88.83	87.40	88.90	91.25	87.40
	<i>cpuGUI.sh</i>	89.50	89.56	87.47	89.53	92.36	88.91
ab	<i>cpu.sh</i>	2.01	0.90	0.12	2.25	1.71	0.12
	<i>cpuGUI.sh</i>	2.15	1.09	0.40	2.60	2.41	0.41

Tabla XXI Resultados promedio de consumo de CPU obtenidos en Fedora sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	82.90	82.32	82.28	82.69	82.17	81.44
	<i>cpuGUI.sh</i>	81.87	80.68	80.62	81.72	80.58	80.02
ab	<i>cpu.sh</i>	1.72	2.36	1.70	1.18	2.37	0.56
	<i>cpuGUI.sh</i>	1.62	2.33	1.60	1.22	2.46	0.64

Tabla XXII Resultados promedio de consumo de CPU obtenidos en Windows sobre el computador de escritorio

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Monitor de rendimiento</i>	20.58	20.47	20.56	20.51	19.60	20.11
ab	<i>Monitor de rendimiento</i>	23.20	56.82	80.15	19.45	46.74	63.08

Tabla XXIII Resultados promedio de consumo de CPU obtenidos en Centos sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	91.03	90.51	88.59	91.85	90.33	88.74
	<i>cpuGUI.sh</i>	93.28	91.58	89.09	94.87	91.93	91.60
ab	<i>cpu.sh</i>	1.50	0.42	0.16	2.59	1.30	0.17
	<i>cpuGUI.sh</i>	1.52	1.38	0.42	2.70	1.40	0.46

Tabla XXIV Resultados promedio de consumo de CPU obtenidos en Fedora sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>cpu.sh</i>	83.27	82.99	82.96	83.07	82.59	82.47
	<i>cpuGUI.sh</i>	83.12	82.54	82.31	81.72	81.32	80.27
ab	<i>cpu.sh</i>	1.42	1.10	1.40	1.34	0.24	1.16
	<i>cpuGUI.sh</i>	1.49	1.30	1.45	1.36	0.51	1.41

Tabla XXV Resultados promedio de consumo de CPU obtenidos en Windows sobre el servidor HP

Método	Software de Medición	Individual			Simultáneo		
		VMW	XEN	XDP	VMW	XEN	XDP
estrés	<i>Monitor de rendimiento</i>	19.98	19.07	21.27	20.73	20.10	20.78
ab	<i>Monitor de rendimiento</i>	12.00	55.21	73.65	17.44	37.23	52.46

6.4 Análisis de Resultados

Una vez obtenidos los valores promedios de consumo de recursos sobre los diferentes sujetos de medición escogimos una plataforma de virtualización y otra de paravirtualización para llevar a cabo la comparativa de las técnicas de virtualización, en el Anexo E puede verse con mayor detalle dicha comparativa. A continuación se encontrará de forma resumida el análisis de los resultados obtenidos de la misma.

Estadísticamente la media de la técnica de virtualización es mayor que la media de paravirtualización, con un valor de P inferior a 0.05 en los siguientes casos:

- Las pruebas de consumo de memoria RAM en ambiente individual, usando el método de carga de estrés de recursos con cualquier software de medición sobre la máquina virtual CentOS en el servidor SUN.

- Las pruebas de consumo de memoria RAM en ambiente simultáneo, usando cualquier método, midiendo con cualquier software de medición sobre la máquina virtual CentOS en el computador de escritorio, a excepción de la prueba con el método de carga de estrés de recursos usando el programa *sar*.
- La prueba de consumo de memoria RAM en ambiente individual, usando el método de estrés de tráfico con el software de medición *hardinfo* sobre la máquina virtual Fedora en el servidor SUN.
- La prueba de consumo de memoria RAM en ambiente simultáneo, usando el método de estrés de tráfico con el software de medición *hardinfo* sobre la máquina virtual Fedora en el computador de escritorio.
- Las pruebas de consumo de memoria RAM en cualquier ambiente, usando cualquier método sobre los sujetos de medición Windows en cualquier hardware, a excepción de la prueba con el método de carga de estrés de recursos, en ambiente individual.
- Las pruebas de consumo de procesador en ambiente individual usando el método de carga de uso de recursos, midiendo con cualquier software de medición sobre la

máquina virtual CentOS, tanto en el computador de escritorio como en el servidor HP.

- Las pruebas de consumo de procesador en ambiente simultáneo usando el método de carga de uso de recursos, midiendo con el programa *sar* sobre los sujetos de medición Linux tanto en el computador de escritorio como en el servidor HP.
- Las pruebas de consumo de procesador en cualquier ambiente, usando el método de cargas de estrés de tráfico web, midiendo el consumo con cualquier software de medición sobre los sujetos de medición Linux en cualquier hardware; a excepción del servidor HP y la prueba midiendo con el programa *top* sobre el computador de escritorio.
- Las pruebas de consumo de procesador en ambiente simultáneo usando el método de carga de uso de recursos, con cualquier software de medición sobre la máquina virtual Fedora, en cualquier hardware.
- La prueba de consumo de procesador en ambiente individual usando el método de carga de uso de recursos, con el programa *hardinfo* sobre la máquina virtual Fedora en el servidor HP.

Estadísticamente la media de la técnica de virtualización es menor que la media de paravirtualización con un valor de P menor a 0.05 en los siguientes casos:

- Las pruebas de consumo de memoria RAM en cualquier ambiente, usando el método de carga de estrés de tráfico, midiendo con el programa *sar* sobre la máquina virtual CentOS tanto en el servidor SUN como en el servidor HP.
- Las prueba de consumo de memoria RAM en ambiente individual, usando el método de carga de estrés de tráfico y midiendo con el programa *hardinfo* sobre la máquina virtual CentOS en el servidor HP.
- La prueba de consumo de memoria RAM en ambiente simultáneo usando el método de carga de estrés de uso de recursos y el programa de medición *sar* sobre la máquina virtual Fedora tanto en el servidor SUN como el servidor HP.
- La prueba de consumo de memoria RAM en ambiente individual usando el método de carga de estrés de uso de recursos y el programa de medición *hardinfo* sobre la máquina virtual Fedora en el servidor SUN.

- Las pruebas de consumo de procesador en cualquier ambiente, usando el método de carga de estrés de uso de recursos y con cualquier software de medición sobre la máquina virtual CentOS en el servidor SUN.
- Las pruebas de consumo de procesador en cualquier ambiente, usando el método de carga de estrés de tráfico y con cualquier software de medición sobre la máquina virtual Fedora en el servidor HP.
- Las pruebas de consumo de procesador en cualquier ambiente, usando cualquier método y software de medición sobre la máquina virtual Windows en cualquier hardware, a excepción de la prueba con el método de carga de estrés de recursos en ambiente individual sobre el computador de escritorio.

Estadísticamente las medias de las técnicas de virtualización son iguales con un valor de P no inferior a 0.05 en los siguientes casos:

- Las pruebas de consumo de memoria RAM en el ambiente simultáneo usando el método de carga de estrés de uso de recursos, utilizando cualquier software

de medición sobre la máquina virtual CentOS, tanto en el servidor SUN como el servidor HP.

- Las pruebas de consumo de memoria RAM en el ambiente individual usando el método de carga de estrés de uso de recursos y con cualquier software de medición sobre cualquier entorno Linux tanto sobre el computador de escritorio como sobre el servidor HP.
- Las pruebas de consumo de memoria RAM en el ambiente individual usando el método de carga de estrés de tráfico y con cualquier software de medición sobre la máquina virtual Fedora tanto sobre el computador de escritorio como sobre el servidor HP.
- Las pruebas de consumo de memoria RAM en el ambiente simultáneo usando el método de carga de estrés de uso de recursos, midiendo con el programa *sar* sobre cualquier entorno Linux en el computador de escritorio.
- Las pruebas de consumo de memoria RAM en el ambiente simultáneo usando el método de carga de estrés de uso de recursos, midiendo con el programa *hardinfo* sobre la máquina virtual Fedora en cualquier hardware.

- Las pruebas de consumo de memoria RAM en cualquier ambiente, usando el método de carga de estrés de tráfico, midiendo con el programa *hardinfo* sobre la máquina virtual CentOS en el servidor SUN.
- Las pruebas de consumo de memoria RAM en ambiente individual, usando el método de carga de estrés de tráfico, con cualquier software de medición sobre los entornos Linux en el computador de escritorio.
- Las pruebas de consumo de memoria RAM en cualquier ambiente, usando el método de carga de estrés de tráfico, midiendo con el programa *sar*, sobre la máquina virtual Fedora en cualquier hardware.
- Las pruebas de consumo de memoria RAM en ambiente simultáneo usando el método de carga de estrés de tráfico y el software de medición *hardinfo* sobre cualquier entorno Linux en el servidor HP.
- La prueba de consumo de memoria RAM en el ambiente simultáneo usando el método de carga de estrés de tráfico, midiendo con el programa *hardinfo* sobre la máquina virtual Fedora en el servidor SUN.
- La prueba de consumo de memoria RAM en el ambiente individual usando el método de carga de estrés de uso de

recursos sobre la máquina virtual Windows en el computador de escritorio.

- La pruebas de consumo de procesador en el ambiente simultáneo usando el método de carga de estrés de uso de recursos, midiendo con el programa *top* sobre la máquina virtual CentOS tanto en el computador de escritorio como el servidor HP.
- Las pruebas de consumo de procesador en ambiente individual usando el método de carga de estrés de uso de recursos con cualquier software de medición sobre la máquina virtual Fedora, tanto en el servidor SUN como el computador de escritorio.
- La prueba de consumo de procesador en ambiente individual usando el método de carga de estrés de uso de recursos y el software de medición *sar* sobre la máquina virtual Fedora en el servidor HP.
- La prueba de consumo de procesador en ambiente individual usando el método de carga de estrés de tráfico y el software de medición *top* sobre la máquina virtual Fedora en el computador de escritorio.
- La prueba de consumo de procesador en ambiente individual usando el método de carga de estrés de uso de

recursos sobre la máquina virtual Windows en el
computador de escritorio.

CONCLUSIONES

1. La técnica de paravirtualización es la que presenta un mejor rendimiento a nivel de consumo de procesamiento, en el caso de CentOS, recordemos que esta distribución tiene un núcleo compatible con dicha técnica, esto hace que el sistema esté consciente de que está siendo virtualizado y tiene una mejor administración de CPU.
2. En el caso de tener un núcleo actualizado y sin modificaciones, como sucede en Fedora, la paravirtualización tiene un mejor rendimiento en el ambiente simultáneo, y en ciertas pruebas en el ambiente individual, esto nos hace pensar que al actualizar el núcleo del sistema operativo se pierde la comunicación completa con los recursos físicos del sistema anfitrión, por lo tanto una actualización de núcleo no asegura una perfecta compatibilidad con las diferentes técnicas de virtualización.

3. A nivel de rendimiento de memoria RAM, tenemos que es indiferente la técnica utilizada, esto se debe a que en la actualidad los virtualizadores presentan módulos o interfaces para agilizar la comunicación de las máquinas virtuales, presentando así un rendimiento similar a las plataformas de paravirtualización.

4. Con los sistemas operativos Windows, observamos que existe una mejor compatibilidad con la técnica de virtualización en lo referente a consumo de procesador, esta característica se da gracias a que los virtualizadores inicialmente fueron creados para explotar la condición propietaria de Windows, ya que esta técnica no necesita modificar el sistema invitado. Por otra parte observamos que a nivel del rendimiento de memoria RAM, la paravirtualización es la técnica de mejor desempeño, esto se debe a que no tenemos una paravirtualización pura, sino que se da gracias a la ayuda de la virtualización asistida por hardware.

5. Los resultados de las pruebas de consumo de memoria RAM en los sujetos de medición Linux ejecutando el método de carga de estrés de uso de recursos, obtenemos estadísticamente hablando independencia de hardware en casi todos los escenarios debido a que las medias de rendimiento son iguales, con valores del estadístico de

prueba entre 0.099 y 0.497. Concluyendo así, que sin importar el hardware que utilicemos obtendremos el mismo resultado al aplicar cualquiera de las técnicas analizadas. Por otra parte, en el método de carga de estrés de tráfico utilizando solicitudes web, se presenta una dependencia del hardware debido a que las medias son estadísticamente diferentes ya que el valor del estadístico de prueba se mantiene en casi todas ellas en cero.

6. Las medias de rendimientos de las pruebas de consumo de CPU nos permiten concluir estadísticamente, que existe una dependencia de hardware debido a que el valor del estadístico de prueba se mantiene en cero en todas ellas. Concluyendo así, que a nivel de consumo de procesador las especificaciones de éste influyen en el rendimiento de las técnicas de virtualización utilizadas.
7. Con el sujeto de medición Windows, se presenta una dependencia total del hardware, debido a que las medias de rendimiento presentan una diferencia estadística ya que el valor del estadístico de prueba se mantiene en todos los escenarios en cero.

RECOMENDACIONES

1. Para ampliar el alcance del proyecto sería recomendable aumentar el número de variables de medición, como por ejemplo latencia de respuesta de servicios que se implementaran en las máquinas virtuales. Ya que a nivel de producción se usa con frecuencia la virtualización para servidores web o de correos electrónicos, los cuales deben tener bajos niveles de latencia.
2. Para futuros nuevos escenarios a agregar a éste u otros proyectos sobre comparativas se deberá realizar un estudio previo para conocer de antemano cual es el número factible de repeticiones de pruebas y sí las herramientas seleccionadas generan los resultados esperados.
3. Para obtener un máximo rendimiento con cualquier técnica de virtualización, se deberá revisar todos los requerimientos tanto a nivel de hardware como sistemas operativos soportados por el hipervisor;

debido a que no todos los sistemas operativos son soportados por los hipervisores.

4. El rendimiento de la técnica depende del hardware sobre la cual estemos ejecutándola, es decir que en equipos especializados como servidores se obtendrá mejores resultados. Pero también se debe tener en cuenta las características que el equipo tenga ya que siempre se debe cumplir con los requerimientos mínimos de cada plataforma.
5. Al momento de implementar cualquier escenario expuesto en el presente proyecto se deberá contar con una conexión estable a internet, para que los paquetes necesarios en los diferentes procesos de instalaciones se descarguen debidamente y no exista ningún error o corrupción del proceso.

ANEXOS

ANEXO A: ESTUDIO PRELIMINAR

Para determinar el número de repeticiones de las pruebas utilizamos la fórmula A-1 (*tamaño de la muestra*), donde $n =$ es el tamaño de la muestra, $Z_{\alpha/2}$ = se refiere al porcentaje de confiabilidad que en nuestro caso es del 95%, $\sigma^2 =$ es la varianza de la muestra y $e =$ es el error estimado.

$$n = \frac{Z_{\alpha/2}^2 \times \sigma^2}{e^2} \quad (A-1)$$

Para el presente estudio tomamos como constantes los siguientes parámetros: $Z_{0,025} = 1.96$, tanto el error estimado para el método de carga de estrés de uso de recursos como para el método de carga de estrés de tráfico utilizando solicitudes web cuyo valor es del 0.1.

RESULTADOS DE LAS PRUEBAS DE CONSUMO DE MEMORIA RAM

Tabla A- 1: Estudio previo sobre sujeto de medición CentOS

		<i>stress</i>		<i>ab</i>		
		<i>ram.sh</i>	<i>ramGUI.sh</i>	<i>ram.sh</i>	<i>ramGUI.sh</i>	
Individual	XDP	Desv. St.	77,81	97,64	3,37	1,78
		e	20,25	19,55	1,13	16,33
		n	56,74	95,79	34,04	0,05
	VMW	Desv. St.	36,10	40,04	0,41	0,40
		e	17,41	20,98	0,68	0,45
		n	16,53	14,00	1,42	3,06
	XEN	Desv. St.	49,79	76,11	0,61	0,55
		e	19,93	16,92	0,84	0,31
		n	23,97	77,75	2,03	11,74
Simul-táneo	XDP	Desv. St.	64,63	89,22	0,68	1,03
		e	19,42	22,15	2,24	0,71
		n	42,54	62,32	0,35	8,21

Tabla A- 2: Estudio previo sobre sujeto de medición CentOS (continuación)

			<i>stress</i>		<i>ab</i>	
			<i>ram.sh</i>	<i>ramGUI.sh</i>	<i>ram.sh</i>	<i>ramGUI.sh</i>
Simultaneo	VMW	Desv. St.	71,33	78,81	1,13	0,32
		e	19,27	17,44	0,45	0,35
		n	52,65	11,81	0,00	0,01
	XEN	Desv. St.	78,52	55,50	0,40	0,72
		e	39,08	41,15	33,82	35,68
		n	15,51	6,99	0,00	0,00
Máximo n:			56,68			

Tabla A- 3: Estudio previo sobre sujeto de medición Fedora

			<i>stress</i>		<i>ab</i>		
			<i>ram.sh</i>	<i>ramGUI.sh</i>	<i>ram.sh</i>	<i>ramGUI.sh</i>	
Individual	XDP	Desv. St.	117,11	78,65	3,42	3,64	
		e	16,42	26,27	6,16	4,61	
		n	195,33	34,44	1,19	2,40	
	VMW	Desv. St.	67,35	125,25	1,83	0,91	
		e	26,36	26,02	0,26	4,17	
		n	25,09	89,01	188,57	0,18	
	XEN	Desv. St.	116,32	81,46	2,81	1,13	
		e	30,78	27,79	4,42	4,33	
		n	54,88	33,02	1,55	0,26	
Simultaneo	XDP	Desv. St.	116,85	71,38	2,22	0,88	
		e	22,75	27,91	5,81	6,17	
		n	101,35	25,12	0,56	0,08	
	VMW	Desv. St.	80,81	72,25	3,10	28,66	
		e	25,34	25,94	3,71	6,15	
		n	39,07	29,81	2,68	83,51	
	XEN	Desv. St.	111,61	73,74	0,62	0,72	
		e	29,93	24,31	4,01	4,18	
		n	53,41	35,35	0,09	0,11	
	Máximo n:			195,33			

Tabla A- 4: Estudio previo sobre sujeto de medición Windows

			<i>stress</i>	<i>ab</i>	
Individual	XPD	Desv. St.	12,98	3,20	
		e	4,97	2,05	
		n	26,24	9,35	
	VMW	Desv. St.	14,88	2,23	
		e	2,27	1,44	
		n	164,65	9,21	
	XEN	Desv. St.	8,44	2,47	
		e	2,78	3,06	
		n	35,38	2,51	
Simultaneo	TXPD	Desv. St.	6,71	6,73	
		e	1,82	1,33	
		n	52,22	98,79	
	TVMW	Desv. St.	6,36	2,20	
		e	2,84	0,59	
		n	19,24	54,00	
	TXEN	Desv. St.	7,86	4,54	
		e	2,06	1,72	
		n	55,97	26,75	
	Máximo n:			164,65	

RESULTADOS DE LAS PRUEBAS DE CONSUMO DE PROCESADOR

Tabla A- 5: Estudio previo sobre sujeto de medición CentOS

			<i>stress</i>		<i>ab</i>	
			<i>cpu.sh</i>	<i>cpuGUI.sh</i>	<i>cpu.sh</i>	<i>cpuGUI.sh</i>
Individual	XDP	Desv. St.	1,80	6,02	0,21	0,66
		e	0,88	0,86	0,11	0,16
		n	16,20	187,88	14,16	65,17
	VMW	Desv. St.	2,25	4,43	0,92	1,73
		e	0,90	0,89	0,34	0,34
		n	23,78	95,41	28,55	98,47
	XEN	Desv. St.	2,08	4,20	0,56	1,26
		e	0,88	0,88	0,31	0,32
		n	21,23	86,55	12,34	60,14
Simul-tánea	XDP	Desv. St.	1,54	3,35	0,47	0,62
		e	0,87	0,89	0,31	0,32
		n	12,17	54,31	8,73	14,18

Tabla A- 6: Estudio previo sobre sujeto de medición CentOS (continuación)

		<i>stress</i>		<i>ab</i>		
		<i>cpu.sh</i>	<i>cpuGUI.sh</i>	<i>cpu.sh</i>	<i>cpuGUI.sh</i>	
Simultaneo	VMW	Desv. St.	2,81	5,51	0,58	1,86
		e	0,90	0,92	0,09	0,18
		n	27,42	75,19	76,48	0,00
	XEN	Desv. St.	2,39	3,92	1,01	0,00
		e	0,89	0,89	0,23	0,37
		n	27,42	75,19	76,48	0,00
Máximo n:		187,88				

Tabla A- 7: Estudio previo sobre sujeto de medición Fedora

		<i>stress</i>		<i>ab</i>		
		<i>cpu.sh</i>	<i>cpuGUI.sh</i>	<i>cpu.sh</i>	<i>cpuGUI.sh</i>	
Individual	XDP	Desv. St.	0,17	0,33	0,17	0,73
		e	0,86	0,86	0,38	0,37
		n	0,16	0,57	0,78	15,33
	VMW	Desv. St.	0,18	0,42	1,02	2,08
		e	0,85	0,83	0,29	0,57
		n	0,17	1,00	47,60	51,67
	XEN	Desv. St.	0,20	0,62	0,37	1,07
		e	0,84	0,82	0,30	0,31
		n	0,21	2,17	5,63	45,28
Simultanea	XDP	Desv. St.	0,12	0,44	0,26	0,43
		e	0,85	0,86	0,47	0,47
		n	0,08	1,00	1,19	3,23
	VMW	Desv. St.	0,17	1,11	0,79	2,65
		e	0,85	0,82	0,53	0,77
		n	0,16	6,95	8,51	45,88
	XEN	Desv. St.	0,65	1,11	0,44	0,46
		e	0,84	0,82	0,08	0,07
		n	2,32	7,04	112,83	156,19
Máximo n:		156,19				

Tabla A- 8: Estudio previo sobre sujeto de medición Windows

			<i>stress</i>	<i>ab</i>
Individual	XDP	Desv. St.	2,84	12,10
		e	0,96	9,69
		n	33,56	6,00
	VMW	Desv. St.	6,58	7,04
		e	7,40	2,05
		n	3,04	45,41
	XEN	Desv. St.	11,62	20,43
		e	2,81	6,77
		n	65,86	34,97
Simultanea	XDP	Desv. St.	3,14	0,00
		e	0,74	10,00
		n	68,68	0,00
	VMW	Desv. St.	5,86	7,91
		e	7,18	2,03
		n	2,56	58,40
	XEN	Desv. St.	4,54	13,66
		e	2,02	8,04
		n	19,30	11,10
Máximo n:			68,68	

Como podemos observar, los máximos tamaños de muestras presentados en las tablas A-1 a la A-6, difieren entre 56.68 hasta 187.88, optamos por un número de repeticiones de 200 para cada prueba y de esta manera asegurar tener un error de estimación del 10%.

ANEXO B: LISTA DE ESCENARIOS PARA LAS PRUEBAS

Tabla B- 1: Escenarios de prueba sobre la plataforma VMware VSphere

N°	Hardware	Plataforma	MV	Ambiente	Variable	Método	Software de Medición
1	Servidor Sun Microsystems, Computador de Escritorio, Servidor HP	VMware VSphere	CentOS	Individual	RAM	Carga de estrés de uso de recursos	ram.sh
2						ramGUI.sh	
3						Carga de estrés de tráfico utilizando solicitudes web	ram.sh
4						ramGUI.sh	
5					CPU	Carga de estrés de uso de recursos	cpu.sh
6						cpuGUI.sh	
7						Carga de estrés de tráfico utilizando solicitudes web	cpu.sh
8						cpuGUI.sh	
9				Simultáneo	RAM	Carga de estrés de uso de recursos	ram.sh
10						ramGUI.sh	
11						Carga de estrés de tráfico utilizando solicitudes web	ram.sh
12						ramGUI.sh	
13					CPU	Carga de estrés de uso de recursos	cpu.sh
14						cpuGUI.sh	
15						Carga de estrés de tráfico utilizando solicitudes web	cpu.sh
16						cpuGUI.sh	
17			Fedora	Individual	RAM	Carga de estrés de uso de recursos	ram.sh
18						ramGUI.sh	
19						Carga de estrés de tráfico utilizando solicitudes web	ram.sh
20						ramGUI.sh	
21					CPU	Carga de estrés de uso de recursos	cpu.sh
22						cpuGUI.sh	
23						Carga de estrés de tráfico utilizando solicitudes web	cpu.sh
24						cpuGUI.sh	
25				Simultáneo	RAM	Carga de estrés de uso de recursos	ram.sh
26						ramGUI.sh	
27						Carga de estrés de tráfico utilizando solicitudes web	ram.sh
28						ramGUI.sh	
29					CPU	Carga de estrés de uso de recursos	cpu.sh
30						cpuGUI.sh	
31						Carga de estrés de tráfico utilizando solicitudes web	cpu.sh
32						cpuGUI.sh	
33			Windows Server	Individual	RAM	Carga de estrés de uso de recursos	Administrador de Tareas
34						Carga de estrés de tráfico utilizando solicitudes web	
35					CPU	Carga de estrés de uso de recursos	Monitor de Rendimiento
36						Carga de estrés de tráfico utilizando solicitudes web	
37				Simultáneo	RAM	Carga de estrés de uso de recursos	Administrador de Tareas
38						Carga de estrés de tráfico utilizando solicitudes web	
39					CPU	Carga de estrés de uso de recursos	Monitor de Rendimiento
40						Carga de estrés de tráfico utilizando solicitudes web	

Tabla B- 2: Escenarios de prueba sobre la plataforma Citrix XenServer

N°	Hardware	Plataforma	MV	Ambiente	Variable	Método	Software de Medición	
1	Servidor Sun Microsystems, Computador de Escritorio, Servidor HP	Citrix XenServer	CentOS	Individual	RAM	Carga de estrés de uso de recursos	ram.sh	
2						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
3					CPU	Carga de estrés de uso de recursos	cpu.sh	
4						Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh	
5					Simultáneo	RAM	Carga de estrés de uso de recursos	ram.sh
6						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
7					CPU	Carga de estrés de uso de recursos	cpu.sh	
8						Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh	
9				Fedora	Individual	RAM	Carga de estrés de uso de recursos	ram.sh
10							Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh
11						CPU	Carga de estrés de uso de recursos	cpu.sh
12							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
13					Simultáneo	RAM	Carga de estrés de uso de recursos	ram.sh
14						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
15						CPU	Carga de estrés de uso de recursos	cpu.sh
16							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
17			Windows Server	Individual	RAM	Carga de estrés de uso de recursos	Administrador de Tareas	
18						Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento	
19					CPU	Carga de estrés de uso de recursos	Administrador de Tareas	
20						Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento	
21				Simultáneo	RAM	Carga de estrés de uso de recursos	Administrador de Tareas	
22					Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento		
23					CPU	Carga de estrés de uso de recursos	Administrador de Tareas	
24						Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento	
25								
26								
27								
28								
29								
30								
31								
32								
33								
34								
35								
36								
37								
38								
39								
40								

Tabla B- 3: Escenarios de prueba sobre la plataforma Xen

N°	Hardware	Plataforma	MV	Ambiente	Variable	Método	Software de Medición	
1	Servidor Sun Microsystems, Computador de Escritorio, Servidor HP	Xen	CentOS	Individual	RAM	Carga de estrés de uso de recursos	ram.sh	
2						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
3					CPU	Carga de estrés de uso de recursos	ram.sh	
4						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
5					Simultáneo	RAM	Carga de estrés de uso de recursos	cpu.sh
6							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
7						CPU	Carga de estrés de uso de recursos	cpu.sh
8							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
9				Fedora	Individual	RAM	Carga de estrés de uso de recursos	ram.sh
10							Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh
11						CPU	Carga de estrés de uso de recursos	ram.sh
12							Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh
13					Simultáneo	RAM	Carga de estrés de uso de recursos	cpu.sh
14							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
15						CPU	Carga de estrés de uso de recursos	cpu.sh
16							Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh
17			Windows Server	Individual	RAM	Carga de estrés de uso de recursos	ram.sh	
18						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
19					CPU	Carga de estrés de uso de recursos	ram.sh	
20						Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh	
21				Simultáneo	RAM	Carga de estrés de uso de recursos	cpu.sh	
22						Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh	
23					CPU	Carga de estrés de uso de recursos	cpu.sh	
24						Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh	
25			Individual	RAM	Carga de estrés de uso de recursos	ram.sh		
26					Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh		
27				CPU	Carga de estrés de uso de recursos	ram.sh		
28					Carga de estrés de tráfico utilizando solicitudes web	ramGUI.sh		
29			Simultáneo	RAM	Carga de estrés de uso de recursos	cpu.sh		
30					Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh		
31				CPU	Carga de estrés de uso de recursos	cpu.sh		
32					Carga de estrés de tráfico utilizando solicitudes web	cpuGUI.sh		
33			Individual	RAM	Carga de estrés de uso de recursos	Administrador de Tareas		
34					Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento		
35				CPU	Carga de estrés de uso de recursos	Administrador de Tareas		
36					Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento		
37			Simultáneo	RAM	Carga de estrés de uso de recursos	Administrador de Tareas		
38					Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento		
39				CPU	Carga de estrés de uso de recursos	Administrador de Tareas		
40					Carga de estrés de tráfico utilizando solicitudes web	Monitor de Rendimiento		

ANEXO C: SCRIPTS PERSONALIZADOS

Requerimientos e Indicaciones

Para la ejecución de los scripts personalizados sobre los entornos Linux necesitamos tener instalado los paquetes: stress, SYSSTAT, httpd, ImageMagick y hardinfo. También es necesario crear el directorio *virtual* en la ruta */home/virtual/Desktop/* como se muestra en la Figura C-1, en esta carpeta se encuentran los scripts y resultados obtenidos de las diferentes pruebas realizadas.

Antes de iniciar cualquier prueba se debe levantar el servicio a utilizar, por ejemplo el servicio web, y ejecutar la herramienta gráfica correspondiente de ser necesaria (*hardinfo* o *top*) para ejecutar el script principal (*main.sh* o *mainGUI.sh*) y posteriormente ejecutar la prueba correspondiente (*ram.sh*, *ramGUI.sh*, *cpu.sh* o *cpuGUI.sh*).

Proceso de ejecución de los scripts

1. Localizarse en la carpeta *scripts* ubicada dentro del directorio virtual como podemos observar en la Figura C-1.

2. Darle los permisos de ejecución al script

```
chmod +x ./[nombre_script]
```

3. Ejecutar el script de acuerdo al formato

```
./[nombre_script].sh [tipo_prueba].
```

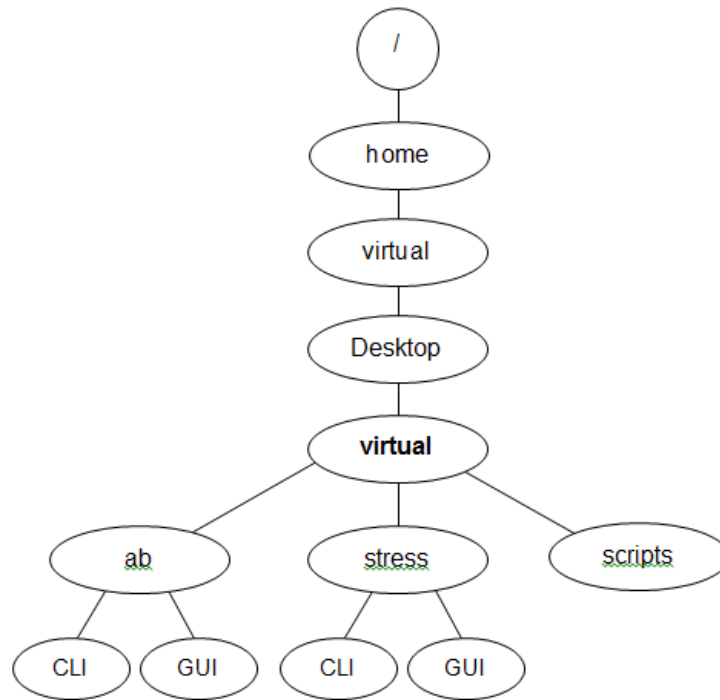


Figura C- 1: Esquema de directorios de la ubicación de scripts y resultados

Script main.sh

Prepara el ambiente de pruebas, creando directorios para el proceso de guardado de datos y genera la carga correspondiente de la prueba a ejecutar.

```

#!/bin/bash

#Nos ubicamos en la carpeta principal de las pruebas.
cd /home/virtual/Desktop/virtual/

#Creamos el directorio CLI correspondiente al ambiente a medir.
FLAG=$(ls | grep $1 | wc -l)

if [ $FLAG == 0 ]; then
    mkdir $1
    mkdir $1/CLI
fi

#Nos ubicamos en la carpeta correspondiente al ambiente.
cd /home/virtual/Desktop/virtual/$1/CLI

DATE=`date +%Y-%m-%d_%H%M:%S`

#Toma de datos de las condiciones iniciales.
sar -r 5 6 >> Rinicial${DATE}.txt
  
```

```

sar 5 6 >> Cincial${DATE}.txt

#Generación de carga dependiendo del tipo de prueba a ejecutar
if [ $1 = "stress" ]; then
    echo "Start test"

    echo ${DATE}': ejecucion de stress'>> ${1}.log
    stress --cpu 1 --io 5 --vm 3 --vm-bytes 128M --timeout 4200s
    sleep 4260

    echo ${DATE}': fin de stress'>> ${1}.log

elif [ $1 = "ab" ]; then
    echo "Start test"

    echo ${DATE}': ejecucion de ab'>> ${1}.log
    sleep 4260

    echo ${DATE}': fin de ab'>> ${1}.log
    echo "Finished test"

fi

#Toma de datos de las condiciones finales.
sar -r 5 6 >> Rfinal${DATE}.txt
sar 5 6 >> Cfinal${DATE}.txt

```

Script cpu.sh

Obtiene resultados por línea de comandos del consumo de CPU

```

#!/bin/bash

#Nos ubicamos en la carpeta correspondiente al ambiente a ejecutar.
cd /home/virtual/Desktop/virtual/$1/CLI
FLAG=$(ls | grep cpu | wc -l)

#Creamos el directorio de la variable a medir.
if [ $FLAG == 0 ]; then
    mkdir cpu
fi

DATE=`date +%Y-%m-%d_%H%M`

#Toma de datos
sar -u 15 200 >> cpu/testCPU${DATE}.txt

```

Script ram.sh

Obtiene resultados por línea de comandos del consumo de RAM

```
#!/bin/bash

#Nos ubicamos en la carpeta correspondiente al ambiente a ejecutar.
cd /home/virtual/Desktop/virtual/$1/CLI

FLAG=$(ls | grep ram | wc -l)

#Creamos el directorio de la variable a medir.
if [ $FLAG == 0 ]; then
    mkdir ram
fi

DATE=`date +%Y-%m-%d_%H%M`

#Toma de datos
sar -r 15 200 >> ram/testRAM${DATE}.txt
```

Script mainGUI.sh

Prepara el ambiente de pruebas, creando directorios para el proceso de guardado de datos y genera la carga correspondiente de la prueba a ejecutar.

```
#!/bin/bash

#Nos ubicamos en la carpeta principal de las pruebas.
cd /home/virtual/Desktop/virtual/

#Creamos el directorio GUI correspondiente al ambiente a medir.
FLAG=$(ls | grep $1 | wc -l)

if [ $FLAG == 0 ]; then
    mkdir $1
    mkdir $1/GUI
fi

#Nos ubicamos en la carpeta correspondiente al ambiente.
cd /home/virtual/Desktop/virtual/$1/GUI

DATE=`date +%Y-%m-%d_%H%M:%S`

#Toma de datos de las condiciones iniciales.
sar -r 5 6 >> Rinicial${DATE}.txt

sar 5 6 >> Cinicial${DATE}.txt
```

```

#Generación de carga dependiendo del tipo de prueba a ejecutar
if [ $1 = "stress" ]; then
    echo "Start test"

    echo ${DATE}': ejecucion de stress'>> ${1}.log

    stress --cpu 1 --io 5 --vm 3 --vm-bytes 128M --timeout 4200s

    sleep 4260

    echo ${DATE}': fin de stress'>> ${1}.log

elif [ $1 = "ab" ]; then
    echo "Start test"

    echo ${DATE}': ejecucion de ab'>> ${1}.log

    sleep 4260

    echo ${DATE}': fin de ab'>> ${1}.log

    echo "Finished test"

fi

#Toma de datos de las condiciones finales.
sar -r 5 6 >> Rfinal${DATE}.txt

sar 5 6 >> Cfinal${DATE}.txt

```

Script cpuGUI.sh

Obtiene resultados por interfaz gráfica de consumo de CPU

```

#!/bin/bash

#Nos ubicamos en la carpeta correspondiente al ambiente a ejecutar.
cd /home/virtual/Desktop/virtual/$1/GUI

FLAG=$(ls | grep cpu | wc -l)

#Creamos el directorio de la variable a medir.
if [ $FLAG == 0 ]; then
    mkdir cpu
fi

cont=1

#Toma y guardado de datos
while [ $cont -le 200 ]
do

```

```
        cont=`expr $cont + 1`  
        sleep 15  
        import -window root cpu/screenCPU$cont.png  
done
```

Script ramGUI.sh

Obtiene resultados por interfaz gráfica de consumo de RAM

```
#!/bin/bash  
  
#Nos ubicamos en la carpeta correspondiente al ambiente a ejecutar.  
cd /home/virtual/Desktop/virtual/$1/GUI  
  
#Creamos el directorio de la variable a medir.  
FLAG=$(ls | grep ram | wc -l)  
  
if [ $FLAG == 0 ]; then  
    mkdir ram  
  
fi  
  
cont=1  
  
#Toma y guardado de datos  
while [ $cont -le 200 ]  
do  
    cont=`expr $cont + 1`  
    sleep 15  
    import -window root ram/capturaRAM$cont.png  
  
done
```

ANEXO D: RESULTADOS DE LAS PRUEBAS DE ESTUDIO

Nomenclatura

Prom:	Valor Promedio
Desv. St:	Desviación Estándar
Error:	Error de Prueba
% Error:	Porcentaje de Error con respecto a la media
Decisión:	Decisión si la prueba es aceptada
Q1:	Primer Cuartil
Q3:	Tercer Cuartil
IRQ:	Rango Intercuartil que en formula es $IQR = Q_3 - Q_1$
Lim_{inf}:	Límite inferior que en formula es $Lim_{inf} = Q_1 - 3 \times IQR$
Lim_{sup}:	Límite superior que en formula es $Lim_{sup} = Q_3 + 3 \times IQR$

RESULTADOS DE LAS PRUEBAS SOBRE EL SERVIDOR SUN

Tabla D- 1: Resultados obtenidos en CentOS sobre la plataforma VMware vSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.26	21.48	3.97	9.65	90.05	90.85	2.31	2.40
Desv. St	5.38	5.24	1.71	4.60	2.28	3.95	0.26	0.27
Error	0.75	0.73	0.24	0.64	0.32	0.55	0.04	0.04
%Error	3.51	3.38	5.97	6.61	0.35	0.60	1.55	1.58
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.24	17.31	2.56	5.80	88.34	87.86	2.17	2.20
Q3	24.67	24.94	5.28	13.23	91.60	93.66	2.50	2.50
IRQ	6.43	7.64	2.72	7.43	3.26	5.80	0.33	0.30
Lim inf	8.60	5.85	-1.52	-5.35	83.45	79.16	1.68	1.75
Lim sup	34.32	36.40	9.36	24.38	96.49	102.36	3.00	2.95
CV	25.30%	24.40%	43.10%	47.70%	2.50%	4.30%	11.20%	11.40%
Ambiente: Simultáneo								
Prom.	21.58	21.96	3.73	5.31	90.18	91.12	2.46	2.95
Desv. St	4.54	6.17	1.45	2.50	2.28	4.02	0.14	0.36
Error	0.63	0.86	0.20	0.35	0.32	0.56	0.02	0.05
%Error	2.91	3.89	5.40	6.52	0.35	0.61	0.78	1.70
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.71	17.47	2.60	3.18	88.52	88.90	2.40	2.83
Q3	23.93	26.27	4.86	7.44	91.81	93.50	2.55	3.23
IRQ	5.22	8.80	2.26	4.25	3.29	4.60	0.16	0.40
Lim inf	10.87	4.26	-0.79	-3.20	83.58	82.00	2.16	2.23
Lim sup	31.76	39.47	8.25	13.82	96.75	100.40	2.78	3.83
CV	21.00%	28.10%	38.90%	47.10%	2.50%	4.40%	5.70%	12.30%

Tabla D- 2: Resultados obtenidos en Fedora sobre la plataforma VMware VSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.35	22.38	3.52	5.70	83.64	80.98	1.39	1.99
Desv. St	5.55	5.85	1.34	2.27	0.20	1.27	0.22	0.46
Error	0.77	0.81	0.19	0.31	0.03	0.18	0.03	0.06
%Error	3.60	3.62	5.28	5.52	0.03	0.22	2.23	3.23
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.36	18.30	2.37	3.81	83.53	80.00	1.20	1.50
Q3	25.64	25.56	4.63	7.65	83.79	82.30	1.47	2.50
IRQ	8.28	7.26	2.26	3.84	0.26	2.30	0.27	1.00
Lim inf	4.95	7.41	-1.01	-1.96	83.14	76.55	0.80	0.00
Lim sup	38.05	36.45	8.01	13.41	84.18	85.75	1.88	4.00
CV	26.00%	26.20%	38.10%	39.80%	0.20%	1.60%	16.10%	23.30%
Ambiente: Simultáneo								
Prom.	21.57	23.69	3.82	12.95	83.08	80.36	1.76	2.89
Desv. St	4.66	6.26	1.38	2.31	0.24	0.68	0.50	0.69
Error	0.65	0.87	0.19	0.32	0.03	0.09	0.07	0.10
%Error	3.00	3.66	5.03	2.47	0.04	0.12	3.90	3.30
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.05	18.82	2.63	11.13	82.92	79.84	1.40	2.37
Q3	24.25	28.97	4.95	14.90	83.26	80.94	2.09	3.37
IRQ	6.20	10.15	2.32	3.77	0.34	1.10	0.69	1.00
Lim inf	8.75	3.58	-0.86	5.48	82.41	78.19	0.37	0.87
Lim sup	33.55	44.20	8.43	20.55	83.77	82.59	3.11	4.87
CV	21.60%	26.40%	36.30%	17.80%	0.30%	0.80%	28.10%	23.80%

Tabla D- 3: Resultados obtenidos en Windows sobre la plataforma VMware VSphere

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	2.05	2.85	19.38	15.50	2.04	2.75	20.32	15.47
Desv. St	0.07	0.00	2.39	2.88	0.01	0.08	2.45	3.15
Error	0.01	0.00	0.33	0.40	0.00	0.01	0.34	0.44
%Error	0.49	0.01	1.71	2.58	0.04	0.40	1.67	2.82
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	1.98	2.85	16.98	13.91	2.04	2.72	18.88	13.67
Q3	2.10	2.85	21.66	17.03	2.04	2.82	22.00	16.80
IRQ	0.12	0.00	4.68	3.13	0.00	0.10	3.12	3.13
Lim inf	1.80	2.85	9.96	9.22	2.04	2.57	14.19	8.98
Lim sup	2.28	2.85	28.68	21.72	2.04	2.96	26.68	21.48
CV	3.50%	0.10%	12.30%	18.60%	0.30%	2.90%	12.10%	20.30%

Tabla D- 4: Resultados obtenidos en CentOS sobre la plataforma Citrix XenServer

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	20.16	21.18	3.10	4.94	84.94	87.74	0.45	0.78
Desv. St	4.75	5.74	1.19	2.06	4.65	7.72	0.06	0.15
Error	0.66	0.80	0.17	0.28	0.65	1.07	0.01	0.02
%Error	3.27	3.76	5.34	5.76	0.76	1.22	1.87	2.62
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.26	17.13	2.04	3.19	81.68	80.63	0.43	0.60
Q3	23.80	25.73	4.13	6.73	88.41	97.23	0.50	0.90
IRQ	6.55	8.60	2.09	3.54	6.72	16.60	0.07	0.30
Lim inf	7.43	4.22	-1.10	-2.11	71.59	55.73	0.33	0.15
Lim sup	33.62	38.64	7.26	12.03	98.49	122.13	0.61	1.35
CV	23.60%	27.10%	38.50%	41.60%	5.50%	8.80%	13.50%	18.90%
Ambiente: Simultáneo								
Prom.	20.52	21.20	2.88	4.60	87.22	88.68	2.28	2.67
Desv. St	5.82	5.37	1.06	1.78	5.26	1.83	0.20	0.29
Error	0.81	0.74	0.15	0.25	0.73	0.25	0.03	0.04
%Error	3.93	3.51	5.11	5.37	0.84	0.29	1.21	1.52
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.70	17.65	1.95	3.09	83.50	87.33	2.13	2.50
Q3	24.97	24.70	3.79	6.17	91.43	89.63	2.40	2.90
IRQ	8.27	7.06	1.84	3.08	7.93	2.30	0.27	0.40
Lim inf	4.30	7.06	-0.81	-1.53	71.59	83.88	1.73	1.90
Lim sup	37.38	35.29	6.55	10.79	103.33	93.08	2.81	3.50
CV	28.40%	25.30%	36.90%	38.80%	6.00%	2.10%	8.70%	11.00%

Tabla D- 5: Resultados obtenidos en Fedora sobre la plataforma Citrix XenServer

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	23.86	23.90	3.46	5.64	82.66	80.35	1.44	2.02
Desv. St	6.48	6.32	1.38	2.26	0.90	0.74	0.13	0.21
Error	0.90	0.88	0.19	0.31	0.12	0.10	0.02	0.03
%Error	3.76	3.67	5.54	5.57	0.15	0.13	1.25	1.44
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.71	19.36	2.28	3.65	82.05	79.83	1.35	1.93
Q3	28.47	28.30	4.66	7.52	83.35	80.96	1.55	2.15
IRQ	9.76	8.94	2.38	3.87	1.30	1.13	0.20	0.22
Lim inf	4.07	5.96	-1.28	-2.16	80.10	78.14	1.05	1.60
Lim sup	43.10	41.71	8.23	13.33	85.30	82.64	1.85	2.48
CV	27.20%	26.50%	40.00%	40.20%	1.10%	0.90%	9.00%	10.40%
Ambiente: Simultáneo								
Prom.	23.95	28.36	3.50	5.57	82.46	80.32	3.21	4.41
Desv. St	4.84	5.94	1.37	2.31	0.54	0.64	0.51	0.60
Error	0.67	0.82	0.19	0.32	0.07	0.09	0.07	0.08
%Error	2.80	2.90	5.41	5.76	0.09	0.11	2.22	1.89
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	20.76	24.75	2.35	3.59	82.31	79.80	2.90	3.97
Q3	26.32	33.27	4.64	7.55	82.83	80.73	3.64	4.77
IRQ	5.56	8.52	2.29	3.96	0.52	0.93	0.74	0.80
Lim inf	12.43	11.97	-1.09	-2.35	81.53	78.41	1.79	2.77
Lim sup	34.66	46.06	8.08	13.49	83.61	82.13	4.75	5.97
CV	20.20%	21.00%	39.10%	41.50%	0.70%	0.80%	16.00%	13.70%

Tabla D- 6: Resultados obtenidos en Windows sobre la plataforma Citrix XenServer

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	1.96	2.82	21.22	45.49	1.93	2.41	21.22	53.75
Desv. St	0.14	0.06	2.80	5.72	0.14	0.08	2.80	6.18
Error	0.02	0.01	0.39	0.79	0.02	0.01	0.39	0.86
%Error	1.01	0.31	1.83	1.74	0.98	0.44	1.83	1.59
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	1.87	2.75	18.88	40.85	1.88	2.35	18.88	50.00
Q3	2.07	2.85	23.56	48.44	1.98	2.45	23.56	57.81
IRQ	0.20	0.10	4.68	7.58	0.10	0.10	4.68	7.81
Lim inf	1.58	2.60	11.86	29.48	1.74	2.21	11.86	38.28
Lim sup	2.36	3.00	30.58	59.82	2.13	2.60	30.58	69.53
CV	7.30%	2.20%	13.20%	12.60%	7.10%	3.20%	13.20%	11.50%

Tabla D- 7: Resultados obtenidos en CentOS sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	19.02	20.69	3.75	5.05	89.14	89.25	0.09	0.32
Desv. St	4.68	5.18	1.61	2.08	1.95	5.73	0.04	0.16
Error	0.65	0.72	0.22	0.29	0.27	0.79	0.01	0.02
%Error	3.41	3.47	5.94	5.70	0.30	0.89	5.47	6.66
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	14.87	17.26	2.35	3.14	87.84	87.63	0.07	0.27
Q3	23.15	23.90	5.15	6.84	90.69	90.63	0.13	0.27
IRQ	8.28	6.64	2.80	3.70	2.85	3.00	0.06	0.00
Lim inf	2.45	7.31	-1.86	-2.42	83.56	83.13	-0.02	0.27
Lim sup	35.57	33.86	9.35	12.40	94.97	95.13	0.22	0.27
CV	24.60%	25.00%	42.90%	41.10%	2.20%	6.40%	39.40%	48.00%
Ambiente: Simultáneo								
Prom.	19.72	20.80	3.37	4.82	89.29	89.81	0.42	0.77
Desv. St	5.74	6.47	1.41	2.17	1.87	4.48	0.10	0.28
Error	0.80	0.90	0.19	0.30	0.26	0.62	0.01	0.04
%Error	4.03	4.31	5.77	6.25	0.29	0.69	3.45	5.00
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	15.43	16.54	2.16	2.95	87.98	87.20	0.33	0.70
Q3	23.60	25.17	4.59	6.69	90.63	92.90	0.47	1.00
IRQ	8.17	8.63	2.43	3.74	2.65	5.70	0.14	0.30
Lim inf	3.17	3.58	-1.49	-2.66	84.00	78.65	0.12	0.25
Lim sup	35.86	38.12	8.24	12.30	94.61	101.45	0.68	1.45
CV	29.10%	31.10%	41.70%	45.10%	2.10%	5.00%	24.90%	36.10%

Tabla D- 8: Resultados obtenidos en Fedora sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.78	23.82	3.43	5.12	82.63	80.32	0.54	0.78
Desv. St	6.30	6.38	1.30	2.06	0.23	0.84	0.13	0.21
Error	0.87	0.88	0.18	0.29	0.03	0.12	0.02	0.03
%Error	4.01	3.71	5.26	5.59	0.04	0.15	3.22	3.81
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.66	18.21	2.34	3.32	82.45	79.63	0.45	0.70
Q3	27.56	27.37	4.56	6.88	82.77	80.93	0.62	0.90
IRQ	8.90	9.16	2.23	3.56	0.32	1.30	0.17	0.20
Lim inf	5.31	4.47	-1.00	-2.02	81.97	77.68	0.20	0.40
Lim sup	40.92	41.12	7.90	12.21	83.25	82.88	0.88	1.20
CV	28.90%	26.80%	38.00%	40.30%	0.30%	1.10%	23.20%	27.50%
Ambiente: Simultáneo								
Prom.	22.71	23.93	3.46	5.25	82.35	79.70	1.45	1.97
Desv. St	5.93	7.28	1.31	2.05	0.41	0.62	0.12	0.30
Error	0.82	1.01	0.18	0.28	0.06	0.09	0.02	0.04
%Error	3.62	4.22	5.24	5.41	0.07	0.11	1.10	2.11
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.20	17.93	2.36	3.53	81.90	79.30	1.37	1.77
Q3	27.26	29.74	4.58	6.98	82.74	80.30	1.53	2.17
IRQ	9.06	11.82	2.23	3.45	0.84	1.00	0.16	0.40
Lim inf	4.61	0.20	-0.98	-1.65	80.64	77.80	1.13	1.17
Lim sup	40.86	47.47	7.92	12.16	84.00	81.80	1.77	2.77
CV	26.10%	30.40%	37.80%	39.00%	0.50%	0.80%	8.00%	15.20%

Tabla D- 9: Resultados obtenidos en Windows sobre la plataforma Xen

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmngr		perfmon		taskmngr		perfmon	
Prom.	1.30	1.04	22.01	60.95	0.40	0.92	21.56	54.28
Desv. St	0.12	0.07	2.09	6.31	0.08	0.09	2.02	7.97
Error	0.02	0.01	0.29	0.87	0.01	0.01	0.28	1.10
%Error	1.30	0.91	1.32	1.43	2.72	1.33	1.30	2.04
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	1.17	0.98	20.22	56.90	0.39	0.89	20.17	49.78
Q3	1.37	1.08	23.34	65.63	0.39	0.99	23.29	59.15
IRQ	0.20	0.10	3.12	8.73	0.00	0.10	3.12	9.38
Lim inf	0.88	0.83	15.53	43.81	0.39	0.74	15.49	35.71
Lim sup	1.66	1.22	28.02	78.71	0.39	1.14	27.97	73.21
CV	9.40%	6.60%	9.50%	10.30%	19.60%	9.60%	9.40%	14.70%

RESULTADOS DE LAS PRUEBAS SOBRE EL COMPUTADOR DE ESCRITORIO

Tabla D- 10: Resultados obtenidos en CentOS sobre la plataforma VMware
VSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	20.03	21.71	5.72	7.15	88.73	89.50	2.01	2.15
Desv. St	4.17	4.63	2.63	3.32	2.53	3.85	0.20	0.29
Error	0.58	0.64	0.37	0.46	0.35	0.53	0.03	0.04
%Error	2.88	2.96	6.38	6.43	0.40	0.60	1.37	1.90
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.33	18.98	3.18	2.10	87.14	86.40	1.90	1.93
Q3	22.88	25.09	7.28	3.39	90.82	92.87	2.17	2.23
IRQ	5.55	6.10	4.10	1.28	3.68	6.48	0.27	0.30
Lim inf	9.00	9.83	-2.97	0.18	81.63	76.68	1.50	1.48
Lim sup	31.21	34.24	13.43	5.31	96.33	102.58	2.58	2.68
CV	20.80%	21.30%	46.00%	46.40%	2.90%	4.30%	9.90%	13.70%
Ambiente: Simultáneo								
Prom.	20.70	21.93	3.85	5.08	88.90	89.53	2.25	2.60
Desv. St	6.03	5.72	1.53	2.31	2.67	5.25	0.18	0.37
Error	0.84	0.79	0.21	0.32	0.37	0.73	0.02	0.05
%Error	4.04	3.62	5.50	6.30	0.42	0.81	1.11	1.98
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	15.83	17.93	2.52	2.64	86.85	84.93	2.13	2.27
Q3	25.25	25.58	5.17	5.74	90.89	93.93	2.40	2.97
IRQ	9.42	7.65	2.65	3.10	4.04	9.00	0.27	0.70
Lim inf	1.69	6.45	-1.46	-2.02	80.80	71.43	1.73	1.22
Lim sup	39.39	37.06	9.14	10.40	96.94	107.43	2.81	4.02
CV	29.10%	26.10%	39.70%	45.40%	3.00%	5.90%	8.00%	14.30%

Tabla D- 11: Resultados obtenidos en Fedora sobre la plataforma VMware VSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.69	23.22	6.21	11.62	82.90	81.87	1.72	1.62
Desv. St	5.91	6.78	2.72	3.96	0.18	0.41	0.13	0.24
Error	0.82	0.94	0.38	0.55	0.02	0.06	0.02	0.03
%Error	3.78	4.05	6.06	4.72	0.03	0.07	1.04	2.01
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.11	17.53	3.95	8.25	82.81	81.63	1.65	1.33
Q3	24.86	27.84	8.49	15.00	83.09	82.37	1.85	1.93
IRQ	7.75	10.30	4.54	6.75	0.28	0.74	0.20	0.60
Lim inf	5.48	2.08	-2.86	-1.87	82.39	80.53	1.36	0.43
Lim sup	36.48	43.29	15.30	25.12	83.51	83.47	2.14	2.83
CV	27.20%	29.20%	43.70%	34.10%	0.20%	0.50%	7.50%	14.50%
Ambiente: Simultáneo								
Prom.	22.37	23.75	5.03	6.78	82.69	81.72	1.18	1.22
Desv. St	5.60	6.12	1.89	2.67	0.14	0.34	0.14	0.25
Error	0.78	0.85	0.26	0.37	0.02	0.05	0.02	0.03
%Error	3.47	3.57	5.20	5.46	0.02	0.06	1.60	2.77
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.06	19.58	3.32	4.52	82.60	81.57	1.06	1.20
Q3	25.55	27.76	8.63	11.67	82.79	81.97	1.25	1.50
IRQ	7.49	8.19	5.31	7.15	0.19	0.40	0.19	0.30
Lim inf	6.83	7.30	-4.64	-6.21	82.32	80.97	0.78	0.75
Lim sup	36.78	40.04	16.60	22.39	83.08	82.57	1.54	1.95
CV	25.00%	25.80%	37.60%	39.40%	0.20%	0.40%	11.50%	20.00%

Tabla D- 12: Resultados obtenidos en Windows sobre la plataforma VMware VSphere

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	2.48	3.07	20.58	23.20	0.58	2.36	20.51	19.45
Desv. St	0.13	0.07	2.96	5.17	0.14	0.05	2.86	4.10
Error	0.02	0.01	0.41	0.72	0.02	0.01	0.40	0.57
%Error	0.70	0.32	1.99	3.09	3.21	0.29	1.93	2.92
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	2.45	3.03	17.32	20.31	0.49	2.35	18.88	15.63
Q3	2.54	3.13	23.56	26.56	0.68	2.35	23.48	21.88
IRQ	0.10	0.10	6.24	6.25	0.20	0.00	4.60	6.25
Lim inf	2.30	2.89	7.96	10.94	0.20	2.35	11.97	6.25
Lim sup	2.69	3.28	32.92	35.94	0.98	2.35	30.39	31.25
CV	5.00%	2.30%	14.40%	22.30%	23.20%	2.10%	13.90%	21.10%

Tabla D- 13: Resultados obtenidos en CentOS sobre la plataforma Citrix XenCenter

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	19.84	22.86	4.64	6.45	88.83	89.56	0.90	1.09
Desv. St	6.97	6.74	2.09	3.00	2.68	5.06	0.07	0.21
Error	0.97	0.93	0.29	0.42	0.37	0.70	0.01	0.03
%Error	4.87	4.08	6.25	6.45	0.42	0.78	1.01	2.64
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	14.32	18.14	2.85	3.88	87.14	84.93	0.87	0.87
Q3	24.85	27.23	6.45	9.00	90.64	93.33	0.93	1.17
IRQ	10.52	9.09	3.60	5.12	3.50	8.40	0.06	0.30
Lim inf	-1.46	4.51	-2.55	-3.81	81.88	72.33	0.78	0.42
Lim sup	40.63	40.86	11.85	16.69	95.89	105.93	1.02	1.62
CV	35.10%	29.50%	45.10%	46.60%	3.00%	5.70%	7.30%	19.10%
Ambiente: Simultáneo								
Prom.	20.38	23.08	3.81	5.02	91.25	92.36	1.71	2.41
Desv. St	6.76	6.10	1.80	2.41	2.64	4.54	0.15	0.34
Error	0.94	0.85	0.25	0.33	0.37	0.63	0.02	0.05
%Error	4.59	3.67	6.56	6.66	0.40	0.68	1.22	1.95
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	14.97	18.82	1.35	2.72	89.38	89.40	1.60	2.27
Q3	25.20	27.00	1.74	7.29	93.18	96.40	1.82	2.67
IRQ	10.23	8.19	0.39	4.57	3.80	7.00	0.22	0.40
Lim inf	-0.37	6.54	0.77	-4.13	83.68	78.90	1.28	1.67
Lim sup	40.53	39.28	2.32	14.13	98.88	106.90	2.14	3.27
CV	33.10%	26.50%	47.40%	48.00%	2.90%	4.90%	8.80%	14.00%

Tabla D- 14: Resultados obtenidos en Fedora sobre la plataforma Citrix XenCenter

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.53	23.12	5.51	7.83	82.32	80.68	2.36	2.33
Desv. St	5.06	5.18	2.43	3.26	0.11	0.48	0.17	0.21
Error	0.70	0.72	0.34	0.45	0.01	0.07	0.02	0.03
%Error	3.25	3.10	6.11	5.77	0.02	0.08	1.01	1.24
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.42	18.94	3.01	5.12	82.24	80.21	2.21	2.17
Q3	25.35	25.40	5.51	10.63	82.35	81.11	2.51	2.37
IRQ	7.93	6.46	2.50	5.51	0.11	0.90	0.30	0.20
Lim inf	5.54	9.25	-0.73	-3.14	82.08	78.86	1.76	1.87
Lim sup	37.24	35.09	9.25	18.89	82.52	82.46	2.96	2.67
CV	23.50%	22.40%	44.10%	41.60%	0.10%	0.60%	7.30%	8.90%
Ambiente: Simultáneo								
Prom.	21.77	23.72	4.95	6.14	82.17	80.58	2.37	2.46
Desv. St	6.22	6.90	2.29	2.48	0.17	0.45	0.18	0.31
Error	0.86	0.96	0.32	0.34	0.02	0.06	0.02	0.04
%Error	3.96	4.03	6.40	5.60	0.03	0.08	1.05	1.73
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.63	18.32	2.97	4.35	82.06	80.26	2.23	2.23
Q3	25.85	29.42	4.95	7.84	82.30	80.86	2.47	2.73
IRQ	9.23	11.10	1.99	3.49	0.24	0.60	0.24	0.50
Lim inf	2.79	1.67	-0.01	-0.88	81.70	79.36	1.87	1.48
Lim sup	39.69	46.07	7.93	13.08	82.66	81.76	2.83	3.48
CV	28.60%	29.10%	46.20%	40.40%	0.20%	0.60%	7.50%	12.50%

Tabla D- 15: Resultados obtenidos en Windows sobre la plataforma Citrix XenCenter

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	1.14	2.89	20.47	56.82	0.52	2.13	19.60	46.74
Desv. St	0.05	0.06	3.04	6.25	0.10	0.03	3.04	6.30
Error	0.01	0.01	0.42	0.87	0.01	0.00	0.42	0.87
%Error	0.64	0.31	2.06	1.52	2.57	0.19	2.15	1.87
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	1.07	2.84	18.89	52.23	0.49	2.10	16.38	42.45
Q3	1.17	2.94	22.46	61.61	0.59	2.15	22.62	50.94
IRQ	0.10	0.10	3.57	9.38	0.10	0.05	6.24	8.49
Lim inf	0.93	2.69	13.54	38.17	0.34	2.03	7.02	29.72
Lim sup	1.32	3.08	27.82	75.67	0.73	2.23	31.98	63.67
CV	4.60%	2.20%	14.80%	11.00%	18.60%	1.40%	15.50%	13.50%

Tabla D- 16: Resultados obtenidos en CentOS sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	19.62	21.10	4.62	6.33	87.40	87.47	0.12	0.40
Desv. St	4.63	5.52	1.96	2.98	1.71	4.27	0.04	0.17
Error	0.64	0.77	0.27	0.41	0.24	0.59	0.01	0.02
%Error	3.27	3.63	5.89	6.51	0.27	0.68	4.32	5.85
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	15.85	17.33	2.99	3.74	86.34	84.70	0.07	0.30
Q3	22.81	24.63	6.28	8.95	88.68	89.00	0.13	0.40
IRQ	6.96	7.30	3.29	5.21	2.34	4.30	0.06	0.10
Lim inf	5.41	6.38	-1.95	-4.08	82.83	78.25	-0.02	0.15
Lim sup	33.24	35.58	11.22	16.77	92.19	95.45	0.22	0.55
CV	23.60%	26.20%	42.50%	47.00%	2.00%	4.90%	31.20%	42.20%
Ambiente: Simultáneo								
Prom.	19.89	21.02	1.95	3.37	87.40	88.91	0.12	0.41
Desv. St	5.29	5.33	0.57	1.19	1.54	5.16	0.04	0.18
Error	0.73	0.74	0.08	0.16	0.21	0.71	0.01	0.02
%Error	3.68	3.51	4.03	4.89	0.24	0.80	4.48	6.04
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.04	16.88	1.24	2.13	86.31	86.55	0.07	0.30
Q3	23.57	24.61	1.79	3.88	88.27	93.00	0.13	0.70
IRQ	7.53	7.73	0.56	1.75	1.97	6.45	0.06	0.40
Lim inf	4.74	5.28	0.40	-0.50	83.36	76.88	-0.02	-0.30
Lim sup	34.86	36.21	2.62	6.50	91.22	102.68	0.22	1.30
CV	26.60%	25.30%	29.10%	35.30%	1.80%	5.80%	32.40%	43.60%

Tabla D- 17: Resultados obtenidos en Fedora sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.14	22.92	5.33	7.67	82.28	80.62	1.70	1.60
Desv. St	5.58	6.66	2.12	3.11	0.50	0.70	0.12	0.20
Error	0.77	0.92	0.29	0.43	0.07	0.10	0.02	0.03
%Error	3.66	4.03	5.50	5.61	0.08	0.12	0.98	1.74
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.48	20.32	3.53	5.02	81.85	79.90	1.60	1.41
Q3	25.07	27.23	7.07	10.39	82.76	81.20	1.78	1.81
IRQ	6.59	6.91	3.54	5.37	0.91	1.30	0.18	0.40
Lim inf	8.60	9.96	-1.78	-3.04	80.49	77.95	1.33	0.81
Lim sup	34.95	37.59	12.38	18.44	84.13	83.15	2.05	2.41
CV	26.40%	29.10%	39.70%	40.50%	0.60%	0.90%	7.10%	12.50%
Ambiente: Simultáneo								
Prom.	21.47	23.33	4.64	5.65	81.44	80.02	0.56	0.64
Desv. St	6.25	6.81	1.98	2.77	0.18	0.43	0.08	0.12
Error	0.87	0.94	0.27	0.38	0.02	0.06	0.01	0.02
%Error	4.04	4.04	5.91	6.81	0.03	0.07	2.00	2.60
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.57	19.08	2.86	3.36	81.33	79.66	0.52	0.50
Q3	25.89	28.71	5.39	8.91	81.55	80.26	0.59	0.70
IRQ	9.33	9.63	2.53	5.55	0.22	0.60	0.07	0.20
Lim inf	2.58	4.64	-0.93	-4.97	81.00	78.76	0.42	0.20
Lim sup	39.88	43.15	9.17	17.24	81.88	81.16	0.70	1.00
CV	29.10%	29.20%	42.60%	49.10%	0.20%	0.50%	14.40%	18.80%

Tabla D- 18: Resultados obtenidos en Windows sobre la plataforma Xen

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	1.13	1.41	20.56	80.15	0.49	2.10	20.10	63.05
Desv. St	0.12	0.05	1.95	4.98	0.11	0.09	2.22	6.30
Error	0.02	0.01	0.27	0.69	0.02	0.01	0.31	0.87
%Error	1.46	0.54	1.32	0.86	3.12	0.58	1.53	1.38
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	0.98	1.37	19.81	76.74	0.39	2.05	18.88	59.38
Q3	1.18	1.47	21.37	82.95	0.59	2.20	22.00	67.19
IRQ	0.20	0.10	1.56	6.20	0.20	0.15	3.12	7.81
Lim inf	0.68	1.23	17.47	67.44	0.10	1.83	14.20	47.66
Lim sup	1.48	1.62	23.71	92.25	0.88	2.43	26.68	78.91
CV	10.50%	3.90%	9.50%	6.20%	22.50%	4.20%	11.00%	10.00%

RESULTADOS DE LAS PRUEBAS SOBRE EL SERVIDOR HP

Tabla D- 19: Resultados obtenidos en CentOS sobre la plataforma VMware
VSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	19.63	20.63	3.72	5.28	91.03	93.28	1.50	1.52
Desv. St	6.61	6.22	1.52	2.21	2.52	4.63	0.10	0.02
Error	0.92	0.86	0.21	0.31	0.35	0.64	0.01	0.00
%Error	4.66	4.18	5.69	5.80	0.38	0.69	0.93	0.14
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	15.04	16.11	2.43	3.38	89.33	89.77	1.44	1.50
Q3	24.45	25.53	5.06	7.20	92.77	97.07	1.57	1.53
IRQ	9.41	9.41	2.63	3.82	3.44	7.30	0.13	0.03
Lim inf	0.93	1.99	-1.52	-2.36	84.17	78.82	1.25	1.46
Lim sup	38.55	39.65	9.01	12.94	97.94	108.02	1.77	1.58
CV	33.70%	30.10%	41.00%	41.80%	2.80%	5.00%	6.70%	1.00%
Ambiente: Simultáneo								
Prom.	20.07	21.27	3.49	5.02	91.85	94.87	2.59	2.70
Desv. St	7.47	6.07	1.42	2.04	2.47	2.07	0.16	0.25
Error	1.04	0.84	0.20	0.28	0.34	0.29	0.02	0.03
%Error	5.16	3.96	5.63	5.64	0.37	0.30	0.85	1.28
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	14.40	16.75	2.27	3.31	89.93	94.08	2.49	2.63
Q3	25.77	25.06	4.74	6.76	93.93	96.46	2.71	2.93
IRQ	11.38	8.32	2.47	3.45	4.01	2.38	0.22	0.30
Lim inf	-2.67	4.27	-1.44	-1.87	83.92	90.52	2.16	2.18
Lim sup	42.83	37.54	8.45	11.94	99.94	100.02	3.04	3.38
CV	37.20%	28.50%	40.60%	40.70%	2.70%	2.20%	6.10%	9.30%

Tabla D- 20: Resultados obtenidos en Fedora sobre la plataforma VMware VSphere

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.19	22.28	3.72	8.67	83.27	83.12	1.42	1.49
Desv. St	6.28	5.46	1.28	2.34	0.25	0.82	0.15	0.24
Error	0.87	0.76	0.18	0.32	0.03	0.11	0.02	0.03
%Error	4.11	3.40	4.75	3.75	0.04	0.14	1.47	2.24
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.67	18.16	2.68	6.75	83.13	82.63	1.24	1.43
Q3	24.89	26.24	5.71	10.67	83.45	83.93	1.57	1.73
IRQ	8.22	8.08	3.03	3.91	0.32	1.30	0.33	0.30
Lim inf	4.34	6.05	-1.87	0.88	82.65	80.68	0.75	0.98
Lim sup	37.22	38.35	10.26	16.54	83.93	85.88	2.07	2.18
CV	29.60%	24.50%	34.20%	27.00%	0.30%	1.00%	10.60%	16.20%
Ambiente: Simultáneo								
Prom.	21.45	24.64	3.53	4.94	83.07	81.72	1.34	1.36
Desv. St	5.65	5.99	1.37	1.95	0.63	0.45	0.33	0.29
Error	0.78	0.83	0.19	0.27	0.09	0.06	0.05	0.04
%Error	3.65	3.37	5.35	5.47	0.11	0.08	3.37	2.97
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.70	19.36	2.41	3.22	82.26	81.43	1.11	1.13
Q3	25.53	29.33	3.53	6.77	83.67	82.03	1.59	1.53
IRQ	7.84	9.97	1.13	3.55	1.41	0.60	0.48	0.40
Lim inf	5.94	4.40	0.71	-2.11	80.15	80.53	0.39	0.53
Lim sup	37.28	44.28	5.23	12.11	85.79	82.93	2.31	2.13
CV	26.40%	24.30%	38.60%	39.50%	0.80%	0.50%	24.30%	21.50%

Tabla D- 21: Resultados obtenidos en Windows sobre la plataforma VMware VSphere

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	2.64	2.39	19.99	12.00	2.18	2.08	20.73	17.44
Desv. St	0.09	0.08	2.62	1.65	0.33	0.09	2.93	2.89
Error	0.01	0.01	0.36	0.23	0.05	0.01	0.41	0.40
%Error	0.47	0.49	1.82	1.91	2.07	0.62	1.96	2.30
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	2.54	2.35	17.97	10.68	1.96	2.05	17.32	15.63
Q3	2.70	2.45	22.65	12.24	2.35	2.15	23.56	18.75
IRQ	0.16	0.10	4.68	1.56	0.39	0.10	6.24	3.13
Lim inf	2.31	2.20	10.95	8.33	1.37	1.91	7.96	10.94
Lim sup	2.93	2.59	29.67	14.58	2.94	2.30	32.92	23.44
CV	3.40%	3.50%	13.10%	13.80%	14.90%	4.50%	14.20%	16.60%

Tabla D- 22: Resultados obtenidos en CentOS sobre la plataforma Citrix XenCenter

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.22	22.97	3.22	4.72	90.51	91.58	0.42	1.38
Desv. St	5.29	5.14	1.33	1.98	2.74	5.08	0.19	0.21
Error	0.73	0.71	0.18	0.27	0.38	0.70	0.03	0.03
%Error	3.45	3.10	5.71	5.81	0.42	0.77	6.16	2.06
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.54	19.68	1.16	2.23	88.77	88.46	0.30	1.17
Q3	25.21	26.58	1.41	4.32	92.52	96.86	0.70	1.57
IRQ	7.67	6.90	0.25	2.09	3.75	8.40	0.40	0.40
Lim inf	6.03	9.34	0.78	-0.90	83.14	75.86	-0.30	0.57
Lim sup	36.72	36.93	1.78	7.45	98.14	109.46	1.30	2.17
CV	24.90%	22.40%	41.20%	41.90%	3.00%	5.50%	44.50%	14.90%
Ambiente: Simultáneo								
Prom.	21.94	23.74	3.04	4.61	90.33	91.93	1.30	1.40
Desv. St	5.03	6.37	1.17	1.94	2.36	4.27	0.08	0.23
Error	0.70	0.88	0.16	0.27	0.33	0.59	0.01	0.03
%Error	3.17	3.72	5.33	5.83	0.36	0.64	0.87	2.28
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.50	18.93	2.07	2.94	88.78	88.97	1.27	1.17
Q3	24.87	28.78	3.90	6.27	92.18	96.05	1.33	1.57
IRQ	6.37	9.86	1.83	3.33	3.40	7.08	0.06	0.40
Lim inf	8.94	4.14	-0.68	-2.06	83.69	78.36	1.18	0.57
Lim sup	34.43	43.57	6.65	11.27	97.27	106.66	1.42	2.17
CV	22.90%	26.80%	38.50%	42.00%	2.60%	4.60%	6.30%	16.40%

Tabla D- 23: Resultados obtenidos en Fedora sobre la plataforma Citrix XenCenter

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	20.77	22.39	3.32	5.73	82.99	82.54	1.10	1.30
Desv. St	5.78	5.42	1.36	2.14	0.19	1.00	0.10	0.23
Error	0.80	0.75	0.19	0.30	0.03	0.14	0.01	0.03
%Error	3.86	3.36	5.66	5.19	0.03	0.17	1.25	2.48
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	17.44	18.85	2.15	3.86	82.87	81.70	1.04	1.03
Q3	24.46	25.89	4.31	7.53	83.19	83.60	1.17	1.43
IRQ	7.03	7.04	2.17	3.67	0.32	1.90	0.13	0.40
Lim inf	6.89	8.29	-1.10	-1.65	82.39	78.85	0.85	0.43
Lim sup	35.00	36.46	7.56	13.04	83.67	86.45	1.37	2.03
CV	27.80%	24.20%	40.90%	37.40%	0.20%	1.20%	9.00%	17.90%
Ambiente: Simultáneo								
Prom.	20.89	26.38	3.52	5.44	82.59	81.32	0.24	0.51
Desv. St	5.73	6.70	1.43	2.10	0.20	0.55	0.05	0.20
Error	0.79	0.93	0.20	0.29	0.03	0.08	0.01	0.03
%Error	3.80	3.52	5.63	5.35	0.03	0.09	3.05	5.41
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.53	19.70	2.32	3.66	82.46	80.99	0.21	0.47
Q3	24.80	32.10	4.74	7.24	82.74	81.76	0.27	0.67
IRQ	8.27	12.40	2.42	3.58	0.28	0.77	0.06	0.20
Lim inf	4.13	1.10	-1.30	-1.72	82.04	79.82	0.12	0.17
Lim sup	37.20	50.70	8.36	12.61	83.15	82.92	0.36	0.97
CV	27.40%	25.40%	40.60%	38.60%	0.20%	0.70%	22.00%	39.10%

Tabla D- 24: Resultados obtenidos en Windows sobre la plataforma Citrix XenCenter

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	2.08	1.82	19.07	55.21	1.95	1.94	20.10	37.23
Desv. St	0.07	0.07	2.62	4.95	0.13	0.13	2.70	5.55
Error	0.01	0.01	0.36	0.69	0.02	0.02	0.37	0.77
%Error	0.45	0.54	1.90	1.24	0.89	0.92	1.86	2.06
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	2.01	1.78	16.16	52.19	1.86	1.85	17.63	33.59
Q3	2.11	1.87	20.84	58.44	2.05	2.10	22.31	39.84
IRQ	0.10	0.10	4.68	6.25	0.19	0.25	4.68	6.25
Lim inf	1.86	1.63	9.14	42.81	1.58	1.48	10.61	24.22
Lim sup	2.26	2.02	27.86	67.81	2.34	2.48	29.33	49.22
CV	3.20%	3.90%	13.70%	9.00%	6.50%	6.60%	13.40%	14.90%

Tabla D- 25: Resultados obtenidos en CentOS sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	18.84	20.24	3.91	5.50	88.59	89.09	0.16	0.42
Desv. St	4.73	5.21	1.65	2.43	2.01	3.69	0.06	0.19
Error	0.66	0.72	0.23	0.34	0.28	0.51	0.01	0.03
%Error	3.48	3.57	5.85	6.12	0.32	0.57	5.09	6.42
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	15.21	16.54	2.46	3.29	86.98	87.90	0.13	0.30
Q3	21.63	23.83	5.28	7.25	90.03	89.90	0.20	0.70
IRQ	6.42	7.28	2.82	3.96	3.06	2.00	0.07	0.40
Lim inf	5.58	5.62	-1.76	-2.65	82.39	84.90	0.03	-0.30
Lim sup	31.26	34.75	9.50	13.20	94.61	92.90	0.31	1.30
CV	25.10%	25.70%	42.20%	44.20%	2.30%	4.10%	36.70%	46.30%
Ambiente: Simultáneo								
Prom.	19.28	20.85	3.58	5.04	88.74	91.60	0.17	0.46
Desv. St	5.06	5.10	1.48	2.13	1.82	4.30	0.06	0.21
Error	0.70	0.71	0.20	0.30	0.25	0.60	0.01	0.03
%Error	3.64	3.39	5.72	5.86	0.28	0.65	4.56	6.20
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	16.14	16.58	2.31	3.21	87.46	87.93	0.13	0.30
Q3	23.30	24.66	4.87	6.94	90.13	95.23	0.20	0.70
IRQ	7.16	8.08	2.56	3.72	2.68	7.30	0.07	0.40
Lim inf	5.39	4.47	-1.53	-2.37	83.44	76.98	0.03	-0.30
Lim sup	34.04	36.78	8.71	12.52	94.14	106.18	0.31	1.30
CV	26.30%	24.50%	41.20%	42.30%	2.10%	4.70%	32.90%	44.70%

Tabla D- 26: Resultados obtenidos en Fedora sobre la plataforma Xen

	RAM				CPU			
	stress		ab		stress		ab	
	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh	ram.sh	ramGUI.sh
Ambiente: Individual								
Prom.	21.52	22.09	3.29	5.46	82.96	82.31	1.40	1.45
Desv. St	5.51	5.04	1.31	2.02	0.63	0.46	0.14	0.24
Error	0.76	0.70	0.18	0.28	0.09	0.06	0.02	0.03
%Error	3.55	3.16	5.53	5.14	0.11	0.08	1.38	2.31
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	18.98	19.05	2.21	3.73	82.50	81.90	1.26	1.27
Q3	22.73	24.85	4.40	5.46	83.51	82.50	1.53	1.67
IRQ	3.75	5.80	2.19	1.74	1.01	0.60	0.27	0.40
Lim inf	13.36	10.36	-1.08	1.12	80.99	81.00	0.86	0.67
Lim sup	28.34	33.54	7.69	8.07	85.02	83.40	1.94	2.27
CV	25.60%	22.80%	39.90%	37.10%	0.80%	0.60%	10.00%	16.60%
Ambiente: Simultáneo								
Prom.	23.69	24.04	3.45	5.34	82.47	80.27	1.16	1.41
Desv. St	7.08	5.99	1.43	1.28	0.22	0.57	0.16	0.23
Error	0.98	0.83	0.20	0.18	0.03	0.08	0.02	0.03
%Error	4.14	3.45	5.74	3.31	0.04	0.10	1.90	2.27
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	19.95	20.84	2.29	4.26	82.32	79.84	1.06	1.23
Q3	28.45	27.46	4.66	6.29	82.64	80.64	1.27	1.53
IRQ	8.50	6.62	2.37	2.03	0.33	0.80	0.21	0.30
Lim inf	7.20	10.91	-1.26	1.22	81.83	78.64	0.75	0.78
Lim sup	41.20	37.39	8.21	9.34	83.13	81.84	1.59	1.98
CV	29.90%	24.90%	41.40%	23.90%	0.30%	0.70%	13.70%	16.40%

Tabla D- 27: Resultados obtenidos en Windows sobre la plataforma Xen

	Ambiente: Individual				Ambiente: Simultáneo			
	RAM		CPU		RAM		CPU	
	stress	ab	stress	ab	stress	ab	stress	ab
	taskmgr		perfmon		taskmgr		perfmon	
Prom.	1.97	1.80	21.27	73.65	1.66	1.69	20.78	52.46
Desv. St	0.26	0.06	2.83	5.34	0.49	0.07	2.95	6.39
Error	0.04	0.01	0.39	0.74	0.07	0.01	0.41	0.89
%Error	1.83	0.45	1.84	1.01	4.09	0.55	1.97	1.69
Decisión:	OK	OK	OK	OK	OK	OK	OK	OK
Q1	1.76	1.76	18.88	70.31	1.27	1.66	18.88	48.44
Q3	2.05	1.86	23.56	78.09	1.86	1.76	23.56	56.25
IRQ	0.29	0.10	4.68	7.78	0.59	0.10	4.68	7.81
Lim inf	1.32	1.61	11.86	58.64	0.39	1.51	11.86	36.72
Lim sup	2.50	2.01	30.58	89.77	2.74	1.91	30.58	67.97
CV	13.20%	3.20%	13.30%	7.30%	29.50%	3.90%	14.20%	12.20%

ANEXO E: COMPARATIVA DE ESTUDIO

COMPARATIVA DE LAS TÉCNICAS DE VIRTUALIZACIÓN

RESULTADOS DE LA COMPARATIVA SOBRE EL SERVIDOR SUN

Tabla E- 1: Comparativa de técnicas en CentOS, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.26	20.15	eq	XEN	0.01	Estadísticamente la media de VMW es mayor que la media de XEN
		GUI	21.47	21.17	eq	XEN	0.29	Estadísticamente ambas medias son iguales, se escoge la menor.
	ab	CLI	3.97	3.10	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
		GUI	9.64	4.94	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
CPU	stress	CLI	90.05	84.94	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
		GUI	90.84	87.74	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
	ab	CLI	2.30	0.45	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
		GUI	2.40	0.78	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.15	19.02	eq	XDP	0.01	Estadísticamente la media de XEN es mayor que la media de XDP
		GUI	21.17	20.69	eq	XDP	0.19	Estadísticamente ambas medias son iguales, se escoge la menor.
	ab	CLI	3.10	3.75	NO eq	XEN	0.00	Estadísticamente la media de XEN es menor que la media de XDP
		GUI	4.94	5.04	eq	Indepen- diente	0.31	Estadísticamente ambas medias son iguales.
CPU	stress	CLI	84.94	89.14	NO eq	XEN	0.00	Estadísticamente XEN tiene menor media que XDP
		GUI	87.74	89.25	NO eq	XEN	0.01	Estadísticamente XEN tiene menor media que XDP
	ab	CLI	0.45	0.09	NO eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP
		GUI	0.78	0.01	eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP

Tabla E- 2: Comparativa de técnicas en CentOS, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.58	20.52	NO eq	XEN	0.02	Estadísticamente VMW tiene una media mayor que XEN
		GUI	21.96	21.19	NO eq	XEN	0.09	Estadísticamente las medias de VMW y XEN son iguales
	ab	CLI	3.72	2.87	NO eq	XEN	0.00	Estadísticamente VMW tiene una media mayor que XEN
		GUI	5.31	4.60	NO eq	XEN	0.00	Estadísticamente VMW tiene una media mayor que XEN
CPU	stress	CLI	90.18	87.22	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
		GUI	91.11	88.68	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
	ab	CLI	2.46	2.28	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
		GUI	2.94	2.66	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.52	19.72	eq	Indepen- diente	0.08	Estadísticamente las media de XEN y VMW son iguales
		GUI	21.19	20.80	NO eq	Indepen- diente	0.25	Estadísticamente las media de XEN y VMW son iguales
	ab	CLI	2.87	3.37	NO eq	XEN	0.00	Estadísticamente la media de XEN es menor que la media de XDP
		GUI	4.60	4.82	NO eq	Indepen- diente	0.15	Estadísticamente las media de XEN y VMW son iguales
CPU	stress	CLI	87.22	89.28	NO eq	XEN	0.00	Estadísticamente la media de XEN es menor que la media de XDP
		GUI	88.68	89.81	NO eq	XEN	0.00	Estadísticamente la media de XEN es menor que la media de XDP
	ab	CLI	2.28	0.42	NO eq	XDP	0.00	Estadísticamente la media de XEN es mayor que la media de XDP
		GUI	2.66	0.76	eq	XDP	0.00	Estadísticamente la media de XEN es mayor que la media de XDP

Tabla E- 3: Comparativa de técnicas en Fedora, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.34	23.86	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
		GUI	22.37	23.89	eq	VMW	0.01	Estadísticamente VMW presenta una media menor que XEN
	ab	CLI	3.52	3.46	eq	XEN	0.32	Estadísticamente ambas tienen medias iguales, se escoge la de menor valor numérico
		GUI	5.70	5.63	eq	XEN	0.38	Estadísticamente ambas tienen medias iguales, se escoge la de menor valor numérico
CPU	stress	CLI	83.63	82.66	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	80.97	80.34	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	1.39	1.43	NO eq	VMW	0.01	Estadísticamente VMW presenta una media menor que XEN
		GUI	1.99	2.01	NO eq	VMW	0.24	Estadísticamente ambas tienen medias iguales, se escoge la de menor valor numérico
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.34	21.77	eq	Indepen- diente	0.23	Estadísticamente ambas tienen medias iguales.
		GUI	22.37	23.82	eq	VMW	0.01	Estadísticamente VMW presenta una media menor que XDP
	ab	CLI	3.46	3.43	eq	Indepen- diente	0.42	Estadísticamente ambas tienen medias iguales.
		GUI	5.63	5.11	eq	XDP	0.01	Estadísticamente VMW presenta una media mayor que XDP
CPU	stress	CLI	82.66	82.63	NO eq	Indepen- diente	0.32	Estadísticamente ambas tienen medias iguales.
		GUI	80.34	80.31	eq	Indepen- diente	0.33	Estadísticamente ambas tienen medias iguales.
	ab	CLI	1.39	0.54	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP
		GUI	1.99	0.77	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP

Tabla E- 4: Comparativa de técnicas en Fedora, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.56	23.94	eq	VMW	0.00	Estadísticamente VMW presenta menor media que XEN
		GUI	23.68	28.35	NO eq	VMW	0.00	Estadísticamente VMW presenta menor media que XEN
	ab	CLI	3.81	3.49	eq	XEN	0.01	Estadísticamente VMW presenta mayor media que XEN
		GUI	12.94	5.56	eq	XEN	0.00	Estadísticamente VMW presenta mayor media que XEN
CPU	stress	CLI	83.07	82.45	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	80.35	80.32	eq	XEN	0.31	Estadísticamente ambas medias son iguales, se escoge el de menor valor numérico
	ab	CLI	1.76	3.21	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
		GUI	2.89	4.41	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.56	22.71	NO eq	VMW	0.01	Estadísticamente VMW presenta una media menor que XDP
		GUI	23.68	23.93	NO eq	Indepen- diente	0.35	Estadísticamente ambas medias son iguales
	ab	CLI	3.49	3.46	eq	Indepen- diente	0.39	Estadísticamente ambas medias son iguales
		GUI	5.56	5.24	eq	Indepen- diente	0.07	Estadísticamente ambas medias son iguales
CPU	stress	CLI	82.45	82.34	NO eq	XDP	0.02	Estadísticamente XEN presenta una media mayor que XDP
		GUI	80.32	79.69	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab	CLI	1.76	1.45	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP
		GUI	2.89	1.96	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP

Tabla E- 5: Comparativa de técnicas en Windows, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.05	1.96	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		2.85	2.81	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	perfmon	19.36	21.22	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
	ab		15.50	45.49	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	1.96	1.29	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab		2.81	1.04	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	19.36	22.00	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP
	ab		15.50	60.95	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

Tabla E- 6: Comparativa de técnicas en Windows, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.03	1.92	NO eq	XEN	0.00	Estadísticamente VMW presente una media mayor que XEN
	ab		2.74	2.40	eq	XEN	0.00	Estadísticamente VMW presente una media mayor que XEN
CPU	stress	perfmon	20.31	21.22	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
	ab		15.46	53.75	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	1.92	0.40	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab		2.40	0.91	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	20.31	21.56	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP
	ab		15.46	54.28	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

RESULTADOS DE LA COMPARATIVA SOBRE EL COMPUTADOR DE ESCRITORIO

Tabla E- 7: Comparativa de técnicas en CentOS, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.03	19.83	NO eq	XEN	0.36	Estadísticamente medias iguales, se escoge XEN por ser el menor
		GUI	21.70	22.86	NO eq	VMW	0.02	Estadísticamente VMW presenta menor media que XEN
	ab	CLI	5.72	4.64	NO eq	XEN	0.00	Estadísticamente VMW presenta mayor media que XEN
		GUI	7.14	6.45	eq	XEN	0.01	Estadísticamente VMW presenta mayor media que XEN
CPU	stress	CLI	88.7	88.83	eq	VMW	0.44	Estadísticamente medias iguales, se escoge VMW por ser el menor
		GUI	89.5	89.56	eq	VMW	0.44	Estadísticamente medias iguales, se escoge VMW por ser el menor
	ab	CLI	2.01	0.89	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
		GUI	2.14	1.09	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	19.84	19.62	NO eq	Indepen- diente	0.35	Estadísticamente medias iguales
		GUI	21.70	21.10	NO eq	Indepen- diente	0.11	Estadísticamente medias iguales
	ab	CLI	4.64	4.61	eq	Indepen- diente	0.45	Estadísticamente medias iguales
		GUI	6.45	6.33	eq	Indepen- diente	0.35	Estadísticamente medias iguales
CPU	stress	CLI	88.72	87.40	NO eq	XDP	0.00	Estadísticamente VMW tiene mayor media que XDP
		GUI	89.50	87.46	NO eq	XDP	0.00	Estadísticamente VMW tiene mayor media que XDP
	ab	CLI	0.89	0.12	NO eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP
		GUI	1.09	0.39	NO eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP

Tabla E- 8: Comparativa de técnicas en CentOS, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.69	20.38	eq	XEN	0.31	Estadísticamente medias iguales, se escoge XEN por ser el menor
		GUI	21.92	23.07	eq	VMW	0.02	Estadísticamente VMW tiene menor media que XEN
	ab	CLI	3.85	3.81	NO eq	XEN	0.39	Estadísticamente medias iguales, se escoge XEN por ser el menor
		GUI	5.08	4.96	eq	XEN	0.31	Estadísticamente medias iguales, se escoge XEN por ser el menor
CPU	stress	CLI	88.89	91.25	eq	VMW	0.00	Estadísticamente la media de VMW es menor que la media de XEN
		GUI	89.53	92.36	NO eq	VMW	0.00	Estadísticamente la media de VMW es menor que la media de XEN
	ab	CLI	2.25	1.70	NO eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
		GUI	2.59	2.41	eq	XEN	0.00	Estadísticamente la media de VMW es mayor que la media de XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.38	19.89	NO eq	Indepen- diente	0.20	Estadísticamente medias iguales, se escoge XDP por ser el menor
		GUI	21.92	21.02	eq	XDP	0.05	No existe suficiente información estadística para decir que las medias son iguales.
	ab	CLI	3.81	1.94	No eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP
		GUI	4.96	3.37	NO eq	XDP	0.00	Estadísticamente XEN tiene mayor media que XDP
CPU	stress	CLI	88.89	87.40	NO eq	XDP	0.00	Estadísticamente VMW presenta media más alta que XDP
		GUI	89.53	88.91	eq	Indepen- diente	0.11	Estadísticamente medias iguales, se escoge XDP por ser el menor
	ab	CLI	1.70	0.12	NO eq	XDP	0.00	Estadísticamente XEN presenta media más alta que XDP
		GUI	2.41	0.40	NO eq	XDP	0.00	Estadísticamente XEN presenta media más alta que XDP

Tabla E- 9: Comparativa de técnicas en Fedora, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.69	21.53	NO eq	XEN	0.38	Estadísticamente ambas medias son iguales, se escoge el menor numéricamente
		GUI	23.22	23.12	NO eq	XEN	0.43	Estadísticamente ambas medias son iguales, se escoge el menor numéricamente
	ab	CLI	6.21	5.50	eq	XEN	0.01	Estadísticamente VMW presenta una media mayor que XEN
		GUI	11.61	7.82	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	CLI	82.90	82.32	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	81.86	80.68	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	1.71	2.36	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
		GUI	1.62	2.33	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtualizador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.53	21.13	eq	Independiente	0.23	Estadísticamente ambas medias son iguales
		GUI	23.12	22.91	NO eq	Independiente	0.36	Estadísticamente ambas medias son iguales
	ab	CLI	5.50	5.33	eq	Independiente	0.22	Estadísticamente ambas medias son iguales
		GUI	7.82	7.67	eq	Independiente	0.32	Estadísticamente ambas medias son iguales
CPU	stress	CLI	82.32	82.28	NO eq	Independiente	0.14	Estadísticamente ambas medias son iguales
		GUI	80.68	80.61	NO eq	Independiente	0.13	Estadísticamente ambas medias son iguales
	ab	CLI	1.71	1.69	eq	XDP	0.04	Estadísticamente no hay informacion suficiente para decir que ambas medias son iguales
		GUI	1.62	1.59	NO eq	Independiente	0.12	Estadísticamente ambas medias son iguales

Tabla E- 10: Comparativa de técnicas en Fedora, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	22.37	21.77	eq	XEN	0.15	Estadísticamente tienen medias iguales, se escoge la de menor valor numérico.
		GUI	23.75	23.72	eq	XEN	0.48	Estadísticamente tienen medias iguales, se escoge la de menor valor numérico.
	ab	CLI	5.02	4.95	NO eq	XEN	0.37	Estadísticamente tienen medias iguales, se escoge la de menor valor numérico.
		GUI	6.78	6.13	eq	XEN	0.01	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	CLI	82.68	82.17	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	81.71	80.58	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	1.18	2.37	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
		GUI	1.22	2.46	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtualizador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.77	21.47	eq	Independiente	0.31	Estadísticamente tienen medias iguales.
		GUI	23.72	23.33	eq	Independiente	0.28	Estadísticamente tienen medias iguales.
	ab	CLI	4.95	4.63	NO eq	Independiente	0.07	Estadísticamente tienen medias iguales.
		GUI	6.13	5.64	eq	XDP	0.03	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	CLI	82.17	81.43	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
		GUI	80.58	80.02	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab	CLI	1.18	0.55	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP
		GUI	1.22	0.63	NO eq	XDP	0.00	Estadísticamente VMW presenta una media mayor que XDP

Tabla E- 11: Comparativa de técnicas en Windows, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.48	1.13	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		3.07	2.89	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	perfmon	20.58	20.47	eq	XEN	0.35	Estadísticamente medias iguales, se escoge el de menor valor numérico
	ab		23.20	56.82	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	1.13	1.13	NO eq	Indepen- diente	0.29	Estadísticamente ambas medias son iguales
	ab		2.89	1.40	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	20.47	20.55	NO eq	Indepen- diente	0.36	Estadísticamente medias iguales.
	ab		23.20	80.15	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

Tabla E- 12: Comparativa de técnicas en Windows, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	0.58	0.52	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		2.36	2.12	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	perfmon	20.50	19.59	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		19.45	46.73	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	0.52	0.49	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab		2.12	2.10	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	19.59	20.10	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XDP
	ab		19.45	63.05	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

RESULTADOS DE LA COMPARATIVA SOBRE EL SERVIDOR HP

Tabla E- 13: Comparativa de técnicas en CentOS, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	19.63	21.21	NO eq	VMW	0.00	Estadísticamente VMW tiene menor media que XEN
		GUI	20.63	22.97	No eq	VMW	0.00	Estadísticamente VMW tiene menor media que XEN
	ab	CLI	3.71	3.22	NO eq	XEN	0.00	Estadísticamente VMW tiene mayor media que XEN
		GUI	5.28	4.72	eq	XEN	0.01	Estadísticamente VMW tiene mayor media que XEN
CPU	stress	CLI	91.03	90.50	eq	XEN	0.02	Estadísticamente XEN presenta una media menor que VMW
		GUI	93.27	91.57	eq	XEN	0.00	Estadísticamente XEN presenta una media menor que VMW
	ab	CLI	1.50	0.42	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que VMW
		GUI	1.51	1.38	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que VMW
			Virtualizador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	19.63	18.84	NO eq	Independiente	0.08	Estadísticamente ambas medias son iguales
		GUI	20.63	20.24	NO eq	Independiente	0.24	Estadísticamente ambas medias son iguales
	ab	CLI	3.22	3.90	NO eq	XEN	0.00	Estadísticamente XEN tiene una media menor que XDP
		GUI	4.72	5.49	eq	XEN	0.00	Estadísticamente XEN tiene una media menor que XDP
CPU	stress	CLI	90.50	88.58	NO eq	XDP	0.00	Estadísticamente XDP presenta menor media que XEN
		GUI	91.57	89.09	NO eq	XDP	0.00	Estadísticamente XDP presenta menor media que XEN
	ab	CLI	0.42	0.16	NO eq	XDP	0.00	Estadísticamente XDP presenta menor media que XEN
		GUI	1.38	0.42	eq	XDP	0.00	Estadísticamente XDP presenta menor media que XEN

Tabla E- 14: Comparativa de técnicas en CentOS, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.07	21.94	NO eq	VMW	0.00	Estadísticamente VMW tiene una media menor que XEN
		GUI	21.26	23.74	eq	VMW	0.00	Estadísticamente VMW tiene una media menor que XEN
	ab	CLI	3.48	3.03	NO eq	XEN	0.02	Estadísticamente VMW tiene una media mayor que XEN
		GUI	5.01	4.61	eq	XEN	0.02	Estadísticamente VMW tiene una media mayor que XEN
CPU	stress	CLI	91.84	90.33	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	94.86	91.92	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	2.58	1.30	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	2.70	1.40	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.07	19.27	NO eq	Indepen- diente	0.10	Estadísticamente ambas medias son iguales.
		GUI	21.26	20.85	eq	Indepen- diente	0.72	Estadísticamente ambas medias son iguales.
	ab	CLI	3.03	3.58	eq	XEN	0.02	Estadísticamente XEN tiene una media menor que XDP
		GUI	4.61	0.15	eq	Indepen- diente	0.97	Estadísticamente ambas medias son iguales.
CPU	stress	CLI	90.33	88.74	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
		GUI	91.92	91.59	eq	Indepen- diente	0.21	Estadísticamente ambas medias son iguales.
	ab	CLI	1.30	0.16	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
		GUI	1.40	0.46	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP

Tabla E- 15: Comparativa de técnicas en Fedora, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.18	20.77	eq	XEN	0.24	Estadísticamente iguales, se escoge el de menor valor numérico
		GUI	22.28	22.39	eq	VMW	0.41	Estadísticamente iguales, se escoge el de menor valor numérico
	ab	CLI	3.72	3.31	eq	XEM	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	8.67	5.72	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	CLI	83.27	82.98	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	83.12	82.54	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	1.41	1.10	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	1.49	1.29	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.77	21.51	eq	Indepen- diente	0.09	Estadísticamente ambas medias son iguales
		GUI	22.28	22.08	eq	Indepen- diente	0.35	Estadísticamente ambas medias son iguales
	ab	CLI	3.31	3.29	eq	Indepen- diente	0.43	Estadísticamente ambas medias son iguales
		GUI	5.72	5.46	eq	Indepen- diente	0.10	Estadísticamente ambas medias son iguales
CPU	stress	CLI	82.98	82.95	NO eq	Indepen- diente	0.26	Estadísticamente ambas medias son iguales.
		GUI	82.54	82.31	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab	CLI	1.10	1.40	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XDP
		GUI	1.29	1.44	eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XDP

Tabla E- 16: Comparativa de técnicas en Fedora, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	21.45	20.89	eq	XEN	0.16	Estadísticamente ambas medias son iguales, se escoge la menor
		GUI	24.64	26.38	eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
	ab	CLI	3.53	3.52	eq	XEN	0.47	Estadísticamente ambas medias son iguales, se escoge la menor
		GUI	4.93	5.43	eq	VMW	0.01	Estadísticamente VMW presenta una media menor que XEN
CPU	stress	CLI	83.07	82.59	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	81.71	81.31	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab	CLI	1.34	0.23	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
		GUI	1.36	0.51	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	CLI	20.89	23.68	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XDP
		GUI	24.64	24.03	eq	Indepen- diente	0.15	Estadísticamente medias iguales.
	ab	CLI	3.52	3.45	eq	Indepen- diente	0.30	Estadísticamente medias iguales.
		GUI	4.93	5.16	NO eq	Indepen- diente	0.08	Estadísticamente medias iguales.
CPU	stress	CLI	82.59	82.47	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XPD
		GUI	81.31	80.27	eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XPD
	ab	CLI	0.23	1.16	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XPD
		GUI	0.51	1.40	NO eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XPD

Tabla E- 17: Comparativa de técnicas en Windows, Ambiente Individual

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.64	2.08	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		2.39	1.82	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	perfmon	19.98	19.06	eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		12.00	55.21	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.08	1.96	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab		1.82	1.79	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	19.06	21.27	eq	XEN	0.00	Estadísticamente XEN presenta una media menor que XDP
	ab		12.00	73.64	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

Tabla E- 18: Comparativa de técnicas en Windows, Ambiente Simultáneo

			VMW	XEN	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	2.18	1.95	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
	ab		2.07	1.93	NO eq	XEN	0.00	Estadísticamente VMW presenta una media mayor que XEN
CPU	stress	perfmon	20.72	20.09	eq	XEN	0.01	Estadísticamente VMW presenta una media mayor que XEN
	ab		17.43	37.23	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XEN
			Virtuali- zador	XDP	Comp. Varianzas	Ganador	Valor P	Observación
RAM	stress	taskmgr	1.95	1.65	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
	ab		1.93	1.68	NO eq	XDP	0.00	Estadísticamente XEN presenta una media mayor que XDP
CPU	stress	perfmon	20.09	20.78	eq	XEN	0.01	Estadísticamente XEN presenta una media menor que XDP
	ab		17.43	52.45	NO eq	VMW	0.00	Estadísticamente VMW presenta una media menor que XDP

COMPARATIVA DE INDEPENDENCIA DE HARDWARE

RESULTADOS DE LA COMPARATIVA EN CENTOS

Tabla E- 19: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	19.63	20.15	No eq	Independiente	0.36	Estadísticamente las medias son iguales
		GUI	20.63	21.70	No eq	Dependiente	0.05	Estadísticamente las medias son diferentes
	ab	CLI	3.10	4.64	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	4.72	6.45	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	84.94	90.50	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	87.74	91.57	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.42	0.89	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.78	1.38	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 20: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	20.07	20.52	NO eq	Independiente	0.49	Estadísticamente las medias son iguales
		GUI	21.19	21.92	eq	Independiente	0.18	Estadísticamente las medias son iguales
	ab	CLI	2.87	3.81	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	4.60	4.96	NO eq	Independiente	0.11	Estadísticamente las medias son iguales
CPU	stress	CLI	87.22	90.33	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	88.68	91.92	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	1.30	2.28	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	1.40	2.66	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 21: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	18.84	19.62	eq	Independiente	0.09	Estadísticamente las medias son iguales
		GUI	20.24	21.10	eq	Independiente	0.11	Estadísticamente las medias son iguales
	ab	CLI	3.75	4.61	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	5.04	6.33	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	87.40	89.14	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	87.46	89.25	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.09	0.16	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.32	0.42	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 22: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	19.27	19.89	eq	Independiente	0.23	Estadísticamente las medias son iguales
		GUI	20.80	21.02	NO eq	Independiente	0.71	Estadísticamente las medias son iguales
	ab	CLI	1.94	3.58	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	3.37	5.03	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	87.40	89.28	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	88.91	91.59	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.12	0.42	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.40	0.76	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

RESULTADOS DE LA COMPARATIVA EN FEDORA

Tabla E- 23: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	20.77	21.53	eq	Independiente	0.16	Estadísticamente las medias son iguales
		GUI	22.28	23.12	eq	Independiente	0.11	Estadísticamente las medias son iguales
	ab	CLI	3.31	5.50	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	5.63	7.82	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	82.32	82.98	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	80.34	82.54	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	1.10	1.71	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	1.29	1.98	No eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 24: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	20.89	21.77	eq	Independiente	0.14	Estadísticamente las medias son iguales
		GUI	23.68	24.64	eq	Independiente	0.12	Estadísticamente las medias son iguales
	ab	CLI	3.497	4.95	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	4.93	6.13	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	82.17	82.59	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	80.32	81.31	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.23	1.76	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.51	2.89	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 25: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	21.13	21.77	eq	Independiente	0.28	Estadísticamente las medias son iguales
		GUI	22.08	23.82	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	3.291	5.33	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	5.11	7.67	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	CLI	82.28	82.95	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	80.31	82.31	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.54	1.69	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.77	1.59	eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 26: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	CLI	21.47	23.68	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	23.33	24.03	eq	Independiente	0.27	Estadísticamente las medias son iguales
	ab	CLI	3.45	4.63	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	5.16	5.64	NO eq	Dependiente	0.03	Estadísticamente las medias son diferentes
CPU	stress	CLI	81.43	82.47	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	79.69	80.27	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab	CLI	0.55	1.44	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
		GUI	0.63	1.96	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

RESULTADOS DE LA COMPARATIVA EN WINDOWS

Tabla E- 27: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	taskmgr	1.13	2.08	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		1.82	2.89	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	perfmon	19.06	20.47	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		12.00	23.20	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 28: Comparativa de independencia de hardware para la técnica de virtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	taskmgr	0.52	1.95	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		1.93	2.40	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	perfmon	19.59	20.31	NO eq	Dependiente	0.01	Estadísticamente las medias son diferentes
	ab		15.46	19.45	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 29: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Individual

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	taskmgr	1.13	1.96	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		1.04	1.79	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	perfmon	20.55	22.00	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		60.95	80.15	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes

Tabla E- 30: Comparativa de independencia de hardware para la técnica de paravirtualización, Ambiente Simultáneo

			Mínimo	Máximo	Comp. Varianzas	Resultado	Valor P	Observación
RAM	stress	taskmgr	0.40	1.65	NO eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		0.91	2.10	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
CPU	stress	perfmon	20.10	21.56	eq	Dependiente	0.00	Estadísticamente las medias son diferentes
	ab		52.45	63.05	eq	Dependiente	0.00	Estadísticamente las medias son diferentes

BIBLIOGRAFÍA

- [1] Almería Eugenio Villar y Julio Gómez, "VIRTUALIZACIÓN DE SERVIDORES DE TELEFONÍA IP EN GNU/LINUX," 2010.
- [2] Citrix Systems, "New Survey Shows Secure Mobile Workstyles Driving Desktop Virtualization Demand," 2012.
- [3] VMware Inc, *Trends around Desktop, Virtualization for Small and Mid-sized Organizations.*, 2012.
- [4] Jon Brodtkin, "WITH LONG HISTORY OF VIRTUALIZATION BEHIND IT," in *IBM LOOKS TO THE FUTURE.*, 2009.
- [5] R. J. Creasy, "THE ORIGIN OF THE VM/370 TIME-SHARING SYSTEM," in *IBM Journal of Research & Development.*
- [6] Wikipedia, The Free Encyclopedia. (Abril, 2010) X86 VIRTUALIZATION - HARDWARE ASSIST.
- [7] Xiaoyun Zhu, Zhikui Wang, Sharad Singhal, Kang G. Shin, Enterprise Systems and Software Laboratory Pradeep Padala, *PERFORMANCE EVALUATION OF VIRTUALIZATION TECHNOLOGIES FOR SERVER CONSOLIDATION.*, 2007.
- [8] Rose Robert, *SURVEY OF SYSTEM VIRTUALIZATION TECHNIQUES.*, 2004.
- [9] K Scarfone, M Souppaya, and Hoffman, *GUIDE TO SECURITY*

FOR FULL VIRTUALIZATION TECHNOLOGIES.: National Institute of Standards and Technology, 2011.

- [10] W. Fuertes y J. E. López de Vergara, "EVALUACIÓN DE PLATAFORMAS DE VIRTUALIZACIÓN PARA EXPERIMENTACIÓN DE SERVICIOS MULTIMEDIA EN REDES IP," 2008.
- [11] Bernard Golden y Clark Scheffy, *VIRTUALIZATION FOR DUMMIES, SUN AND AMD SPECIAL EDITION.*, 2008.
- [12] IBM, "VIRTUALIZATION IN EDUCATION," 2007.
- [13] Manas Ranjan Patra, Suresh Chandra Satapathy Rabi Prasad Padhy, "VIRTUALIZATION TECHNIQUES & TECHNOLOGIES: STATE OF THE ART," , 2011.
- [14] Sergio Talens-Oliag, "HERRAMIENTAS DE VIRTUALIZACIÓN LIBRES PARA SISTEMAS GNU/LINUX," , 2008.
- [15] Bill Hill, *INTRO TO VIRTUALIZATION: HARDWARE, SOFTWARE, MEMORY, STORAGE, DATA AND NETWORK VIRTUALIZATION DEFINED.*, 2012.
- [16] Ing. Érica.B. González, "GESTOR DE MÁQUINAS VIRTUALES," 2010.
- [17] Wikipedia, The Free Encyclopedia. (2012, Nov.) HYPERVISOR.
- [18] Miquel Angel Senar Rosell Carlos Miguel Martín Hernández,

"VIRTUALIZACIÓN SOBRE PLATAFORMAS OPEN SOURCE, ALTERNATIVAS A LAS SOLUCIONES COMERCIALES," , 2011.

- [19] Francesc Guim Bernat Francisco Javier Serrano Castaño, "MEMORIA: GESTIÓN DE PROCESOS EN LOS SISTEMAS OPERATIVOS," 2011.
- [20] Robert P. Goldberg Gerald J. Popek, "FORMAL REQUIREMENTS FOR VIRTUALIZABLE THIRD GENERATION ARCHITECTURES," 1974.
- [21] Elvira Mifsud, "INTRODUCCIÓN A LA VIRTUALIZACIÓN CON XEN," 2012.
- [22] Microsoft., "MICROSOFT APPLICATION VIRTUALIZATION TECHNICAL OVERVIEW,".
- [23] (2007, June) SITIO WEB DE MICROSOFT. [Online].
<http://technet.microsoft.com/en-us/library/bb608285.aspx>
- [24] (2011, Nov.) SITIO WEB SEARCH VIRTUAL DESKTOP DE TECHTARGET. [Online].
<http://searchvirtualdesktop.techtarget.com/definition/desktop-virtualization>
- [25] Tony Lock, Martin Atherton and Jon Collins Dale Vile, *DESKTOP VIRTUALIZATION FOR DUMMIES.*, 2010.
- [26] Brian T. Gold, Thomas F. Wensch, Publicación David Meisner,

- "POWERNAP: ELIMINATING SERVER IDLE POWER," 2009.
- [27] Jonathan G. Koomey, "BE ENERGY," *LA REVISTA DE LA EFICIENCIA Y EL AHORRO ENERGÉTICO*, 2012.
- [28] MAIN PAGE. [Online]. wiki.qemu.org
- [29] Kernel Linux. [Online]. <http://www.kernel.org/pub/linux/kernel/v2.6/>
- [30] KERNEL BASED VIRTUAL MACHINE. [Online]. www.linux-kvm.org
- [31] Oracle Corporation, "ORACLE VM VIRTUALBOX USER MANUAL," 2012.
- [32] VMware Inc, "PERFORMANCE OF VMWARE," 2008.
- [33] "VMWARE VSPHERE ENTERPRISE EDITION DATASHEET," 2012.
- [34] VMware Inc, "VMWARE ESX AND VMWARE ESXI THE MARKET LEADING PRODUCTION-PROVEN HYPERVISORS," 2009.
- [35] Wikipedia, The Free Encyclopedia. (2013, Mar.) WINDOWS VIRTUAL PC. [Online]. <http://es.wikipedia.org>
- [36] Jerry Honeycutt, "MICROSOFT® VIRTUAL PC 2004 TECHNICAL OVERVIEW," 2003.
- [37] Wikipedia, The Free Encyclopedia. (2013, Feb.) HYPER-V. [Online]. en.wikipedia.org

- [38] HYPER-V ARCHITECTURE. [Online]. msdn.microsoft.com
- [39] Citrix Systems Inc, "CITRIX XENSERVER, VIRTUALIZATION, AUTOMATION AND ADVANCED MANAGEMENT TOOLS FOR THE DATACENTER," 2011.
- [40] Citrix Systems Inc, "TECHNICAL AND COMMERCIAL COMPARISON OF CITRIX XENSERVER AND VMWARE VSPHERE," 2011.
- [41] AN INTRODUCTION TO VIRTUALIZATION. [Online]. kernelthread.com
- [42] DENALI: A BRIEF OVERVIEW. [Online]. denali.cs.washington.edu
- [43] Citrix System Inc., "WHAT IS XEN HYPERVISOR?," 2009.
- [44] Citrix System Inc., "HOW DOES XEN WORK?," 2009.
- [45] Fabrizio Ciachi, "LINUX DENTRO DE LINUX," 2013.
- [46] QEMU Official OS Support List. [Online]. www.claunia.com
- [47] CARACTERIZACIÓN DE LA CARGA: BENCHMARKS. [Online]. <http://geneura.ugr.es>
- [48] (2009, Sep.) SYSSTAT HOWTO: A DEPLOYMENT AND CONFIGURATION GUIDE FOR LINUX SERVERS. [Online]. <http://www.linux.com>
- [49] Microsoft Inc. TASK MANAGER OVERVIEW. [Online].

<http://www.microsoft.com>

- [50] Elena Ballano Hernández, Enrique López Mañas, Ismael Salgado Flores, José Abel Tamayo Carlos Miguel Torres Jiménez. [Online].
<http://www.neo-tech.es/>>,
- [51] WEB DE LA ASIGNATURA DISEÑO DE SISTEMAS OPERATIVOS: GESTIÓN DE MEMORIA. [Online].
<http://www.ual.es>
- [52] SAR MANUAL PAGE. [Online].
<http://sebastien.godard.pagesperso-orange.fr>
- [53] HARDINFO. [Online]. <http://www.linuxlinks.com>
- [54] ABOUT IMAGEMAGICK. [Online]. <http://www.imagemagick.org>
- [55] LINUX / UNIX COMMAND: TOP. [Online]. <http://linux.about.com>
- [56] (2013, Feb.) OVERVIEW OF WINDOWS PERFORMANCE MONITOR. [Online]. <http://technet.microsoft.com>
- [57] Sitio Web del grupo empresarial Óptima. [Online].
http://www.optimanet.com.mx/detalles_Sun-Microsystems-Sunfire-X2200-Server,340,36,2.htm
- [58] Sitio Web de la tienda oficial de HP España. [Online].
<http://h10010.www1.hp.com/wwpc/es/es/sm/WF05a/15351-15351-3328412-3328421-3328421-3884343.html?dnr=1>
- [59] Sitio Web de la Universidad of Oklahoma. [Online].

<http://weather.ou.edu/~apw/projects/stress>

- [60] Sitio Web de Jam Software. [Online]. <http://www.jam-software.com/heavyload>
- [61] Sitio Oficial de cpukiller. [Online]. <http://www.cpukiller.com>
- [62] Sitio Oficial de Apache. [Online].
<http://httpd.apache.org/docs/2.2/programs/ab.html>
- [63] Blog TechnoBlog. [Online].
<http://www.technoblog.com.ar/index.php/2010/01/como-medir-la-performance-de-apache>