



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

ESTUDIO DE VIRTUALIZACIÓN DE UN SISTEMA LEGADO EN UNA EMPRESA GUATEMALTECA

Luis Alberto Cano Moreno

Asesorado por el Ing. Marlon Pérez Türk

Guatemala, octubre de 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**ESTUDIO DE VIRTUALIZACIÓN DE UN SISTEMA LEGADO EN
UNA EMPRESA GUATEMALTECA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR:

LUIS ALBERTO CANO MORENO

ASESORADO POR EL INGENIERO MARLON ANTONIO PÉREZ TÜRK

AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, OCTUBRE DE 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE LA JUNTA DIRECTIVA

DECANO	Ing.	Murphy Olympo Paiz Recinos
VOCAL I	Inga.	Glenda Patricia García Soria
VOCAL II	Inga.	Alba Maritza Guerrero de López
VOCAL III	Ing.	Miguel Ángel Dávila Calderón
VOCAL IV	Br.	José Milton De León Bran
VOCAL V	Br.	Isaac Sultán Mejía
SECRETARIA	Inga.	Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing.	Murphy Olympo Paiz Recinos
EXAMINADOR	Ing.	Marlon Antonio Pérez Türk
EXAMINADORA	Inga.	Floriza Felipa Ávila Pesquera
EXAMINADORA	Inga.	Sonia Yolanda Castañeda Ramírez
SECRETARIA	Inga.	Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

ESTUDIO DE VIRTUALIZACIÓN DE UN SISTEMA LEGADO EN UNA EMPRESA GUATEMALTECA,

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha agosto de 2007.

Luis Alberto Cano Moreno

AGRADECIMIENTOS A:

Mis padres

Luis Jesús Cano Pineda y Sandra Elizabeth Moreno Barrundia, por todo el apoyo, paciencia y cariño.

Mis hermanos

María Alejandra Cano Moreno y Juan Luis Angel Cano Moreno, por el soporte.

Mis familiares

Por apoyarme en todo este proceso.

Mis amigos

Alain Orión Reyes y Joel Delcompare.

El Ing. Marlon Pérez Türk Por su apoyo y guía.

El Ing. Armín Mazariegos Por motivarme con su dedicación y experiencia en el área educativa.

ACTO QUE DEDICO A:

- Mis padres** Luis Jesús Cano Pineda y Sandra Elizabeth Moreno Barrundia.
- Mis hermanos** María Alejandra Cano Moreno y Juan Luis Ángel Cano Moreno.
- Mis familiares** Por la motivación y consejos.
- Mis motivadores y amigos** Alain Orión Reyes, Joel Delcompare, Ing. Marlon Pérez Türk, Ing. Armín Mazariegos,

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	III
LISTA DE SÍMBOLOS.....	V
GLOSARIO.....	IX
RESUMEN.....	XV
VARIABLES.....	XVII
OBJETIVOS.....	XIX
HIPÓTESIS.....	XXI
INTRODUCCIÓN.....	XXIII
1. VIRTUALIZACIÓN.....	1
1.1. Introducción al concepto de virtualización.....	1
1.2. Historia de la virtualización de sistemas informáticos.....	5
1.3. Virtualización de sistemas informáticos.....	9
1.3.1. Sistema original.....	12
1.3.2. Capa de virtualización.....	14
1.3.3. Sistema virtualizado.....	15
1.3.4. Sistemas externos.....	17
1.4. Tipos de virtualización de sistemas informáticos.....	19
1.4.1. Emulación y simulación.....	20
1.4.2. Virtualización nativa.....	24
1.4.3. Virtualización asistida por el equipo informático.....	26
1.4.4. Virtualización parcial.....	29
1.4.5. Paravirtualización.....	31
1.4.6. Virtualización a nivel de sistema operativo.....	33
1.4.7. Virtualización a nivel de aplicación.....	34
1.5. Máquinas virtuales.....	36
1.6. Virtualización de la arquitectura x86.....	38
1.7. Ventajas y desventajas de virtualizar un sistema informático.....	40
1.7.1. Ventajas.....	40
1.7.2. Desventajas.....	43
1.8. Herramientas de virtualización gratuitas.....	45

1.8.1. VMWare Server.....	47
1.8.2. Microsoft Virtual PC 2007.....	49
1.8.3. Innotek VirtualBox.....	51
1.8.4. XEN.....	53
1.8.5. QEMU.....	55
1.8.6. Bochs.....	57
1.8.7. DOSBox.....	59
1.8.8. DOSEmu.....	61
2. SISTEMAS LEGADOS.....	63
3. VIRTUALIZACIÓN DEL SISTEMA LEGADO.....	65
3.1. Sistema de pruebas de aptitud.....	65
3.2. Plataforma original del Sistema de Pruebas de Aptitud.....	70
3.3. Requerimientos del sistema de pruebas de aptitud.....	71
3.3.1. Requerimientos funcionales.....	72
3.3.2. Requerimientos no funcionales.....	109
3.4. Selección de la máquina virtual a utilizar.....	113
3.4.1. Descripción de la plataforma destino.....	114
3.4.2. Análisis de factibilidad.....	119
3.4.3. Comparativa de las alternativas.....	142
3.5. Implementación.....	145
3.5.1. Preparación de la instalación.....	145
3.5.2. Instalación de la máquina virtual.....	147
3.5.3. Configuración de la máquina virtual.....	154
3.6. Validación de la solución virtual.....	173
3.6.1. Validación técnica.....	173
3.6.2. Validación del usuario.....	175
3.6.3. Comparativa de rendimiento.....	177
CONCLUSIONES.....	183
RECOMENDACIONES.....	185
REFERENCIAS.....	187
BIBLIOGRAFÍA.....	189
APÉNDICE.....	193

ÍNDICE DE ILUSTRACIONES

TABLAS

I. Ficha técnica VMWare Server.....	48
II. Ficha técnica Microsoft VirtualPC.....	50
III. Ficha técnica Innotek Virtualbox.....	52
IV. Ficha técnica de XEN.....	54
V. Ficha técnica QEMU.....	56
VI. Ficha técnica Bochs.....	58
VII. Ficha técnica DOSBox.....	60
VIII. Ficha técnica DOSEmu.....	62
IX. Requerimientos Sistema de Pruebas de Aptitud (SPA).....	70
X. Dispositivos externos utilizados por el sistema de pruebas.....	110
XI. Plataforma destino.....	115
XII. Soporte de la plataforma destino por máquina virtual.....	117
XIII. Análisis de factibilidad económico para VMWare Server.....	120
XIV. Análisis de factibilidad legal para VMWare Server.....	122
XV. Configuración de dispositivos dentro y fuera de la máquina virtual para VMWare Server 2.0.....	125
XVI. Herramientas administrativas de VMWare.....	127
XVII. Análisis de factibilidad económico para QEMU.....	129
XVIII. Análisis de factibilidad legal para QEMU 0.9.0.....	130
XIX. Configuración de dispositivos dentro y fuera de la máquina virtual para QEMU 9.0.....	131
XX. Herramientas administrativas de QEMU 0.9.0.....	133
XXI. Análisis de factibilidad económico para Bochs 2.3.6.....	135

XXII. Análisis de factibilidad legal para Bochs 2.3.6.....	136
XXIII. Configuración de dispositivos dentro y fuera de la máquina virtual para Bochs 2.3.6.....	138
XXIV. Herramientas administrativas de Bochs 2.3.6.....	140
XXV. Comparativa de las máquinas virtuales candidatas.....	142
XXVI. Especificaciones técnicas pruebas de rendimiento.....	178
XXVII. Comparativa de rendimiento.....	181

LISTA DE SÍMBOLOS

Símbolo	Descripción
3D	Tres Dimensiones
ATA	<i>Advanced Technology Attachment</i> (Tecnología Avanzada de Acoplamiento)
CGA	<i>Color Graphics Adapter</i> (Adaptador Gráfico en Color)
CLR	<i>Common Language Runtime</i> (Código común de la plataforma .NET)
COM	COMunicaciones
CPU	<i>Central Processing Unit</i> (Unidad central de procesamiento)
DDR	<i>Double Data Rate</i> (Taza doble de datos)
DOS	<i>Disk Operating System</i> (Sistema Operativo de Disco)
EGA	<i>Enhanced Graphics Adapter</i> (Adaptador Gráfico Mejorado)
FAT	<i>File Allocation Table</i> (Tabla de asignación de archivos)

Símbolo	Descripción
GB	Giga <i>Byte</i>
GHZ	Giga Hertz
GPL	<i>General Public License</i> (Licencia General Pública)
HLT	<i>HaLT</i> (Parar)
HZ	Hertz (Ciclos por segundo)
IBM	<i>International Business Machines</i> (Empresa de servicios informáticos)
LGPL	<i>Lesser General Public License</i> (Licencia General Pública Menor)
LPT	<i>Line Print Terminal</i> (Terminal de impresión por línea)
MB	Mega <i>Byte</i>
MHZ	Mega Hertz
PC	<i>Personal Computer</i> (Computadora Personal)
Q (GTO)	Quetzales
SATA	<i>Serial Advanced Technology Attachment</i> (Tecnología Avanzada de Acompañamiento Serial)

Símbolo	Descripción
SCSI	<i>Small Computer System Interface</i> (Interfaz del Sistema de Computadoras Pequeñas)
SPA	Sistema de Pruebas de Aptitud
TCP	<i>Transmission Control Protocol</i> (Protocolo del control de la transmisión)
UDP	<i>User Datagram Protocol</i> (Protocolo de Datagramas de Usuario)
USB	<i>Universal Serial Bus</i> (Bus Serial Universal)
USPTO	<i>United States Patent and Trademark Office</i> (Oficina de patentes y marcas de comercio de Estados Unidos)
VFAT	<i>Virtual File Allocation Table</i> (Tabla de asignación de archivos virtual)
VM	<i>Virtual Machine</i> (Máquina Virtual)
VNC	<i>Virtual Networking Computing</i> (Computación de Red Virtual)

GLOSARIO

Término	Definición
Capa de virtualización	Es una abstracción del equipo informático, la máquina virtual.
Caso de uso	Es un caso en que el usuario usa el sistema informático.
Componente físico	Es toda parte tangible de un sistema informático. También llamado <i>hardware</i> .
Componente informático	Es toda parte intangible de un sistema informático. Son las aplicaciones y programas. También llamado <i>software</i> .
Componente lógico	Componente informático.
<i>Copyleft</i>	Es un conjunto de derechos otorgados a los usuarios de algún material informático. No es lo opuesto a <i>Copyright</i> .
<i>Copyright</i>	Derechos de Autor , es un conjunto de derechos exclusivos para el creador de una obra intelectual. No se debe confundir con la patente.
<i>Diskette</i>	Medio de almacenamiento, es un disco magnético flexible.

Término	Definición
Emulación	Es utilizar un programa de computadoras en una plataforma distinta para la cual fue diseñado.
Equipo informático	Componente físico. También llamado <i>Hardware</i> .
FAT	<i>File Allocation Table</i> , es el sistema de archivos utilizado en el sistema D.O.S.
<i>File Allocation Table</i>	Sistema de archivos utilizado en el sistema operativo D.O.S. y en dispositivos portátiles.
Funcionalidad	Son todas las tareas que un sistema puede realizar.
GNU/Linux	Sistema operativo compuesto por el <i>kernel</i> Linux y el sistema GNU.
<i>Hardware</i>	Componentes tangibles de una computadora.
HLT	<i>Halt</i> , es una instrucción del procesador que la utilizan los procesos cuando no están realizando tareas. Esto permite que el procesador consuma menos potencia o que otros procesos puedan utilizarlo.
<i>Hypervisor</i>	Es un tipo de virtualización que permite utilizar varios sistemas operativos en una misma máquina al mismo tiempo.

Término	Definición
i386	De la familia x86 de Intel, es la arquitectura que cumple con las instrucciones de los procesadores 80386.
Linux	Núcleo de un sistema operativo, utiliza la licencia GPL versión 2.0 y se distribuye gratuitamente.
Mac OSX	Sistema operativo de la empresa Apple.
Máquina virtual	Es un programa informático que simula ser una equipo físico, esto permite ejecutar dentro de la máquina virtual un sistema operativo.
Microsoft Windows	Sistema operativo de la empresa Microsoft.
Modelo	Es una representación lógica de la realidad.
OMR	<i>Optical Mark Recognition</i> , es un proceso que se utiliza para reconocer marcas en papel trasladando la posición de las marcas a un medio digital. Las marcas son comúnmente hechas con lápiz o lapicero.
Para-virtualización	Es una técnica de virtualización en la cual se modifica el sistema operativo para que funcione como una o más máquinas virtuales.

Término	Definición
Requerimientos	Son todas las características tanto funcionales como de calidad que debe cumplir un sistema informático para que cumpla con las expectativas del cliente.
Scantron	Es una marca de productos OMR, estos productos permiten calificar pruebas hechas con papel y contestadas a lápiz o lapicero.
<i>Script</i>	Un conjunto de instrucciones que el sistema debe realizar.
Simulación	Es la representación de los fenómenos de la realidad utilizando modelos, lo cual permite la experimentación de los mismos.
Sistema	Conjunto de elementos que inter-actúan para alcanzar un objetivo.
Sistema anfitrión	Es el sistema donde está instalada la máquina virtual.
Sistema huésped	Es el sistema que está instalado dentro de la máquina virtual.
Sistema informático	Es un sistema que gestiona principalmente la información.

Término	Definición
Sistema operativo	Es el sistema que está formado por un conjunto de programas y un núcleo, el cual gestiona los recursos de un dispositivo informático, permitiendo así la comunicación entre el usuario y los componentes del dispositivo.
<i>Software</i>	Son todos los programas de una computadora.
SPA	Sistema de Pruebas de Aptitud, es el sistema que se virtualizó en este estudio.
TMER	Tiempo medio entre recuperaciones, es el tiempo promedio que tarda un sistema en recuperarse de un fallo.
TMPF	Tiempo medio entre fallas, es el tiempo transcurrido promedio entre cada falla de un sistema.
VFAT	Virtual FAT, sistema de archivos que no reside en un disco de almacenamiento, sino es simulado por la máquina virtual para que el sistema huésped lo utilice.
Virtualización	Es la abstracción del equipo informático.
x86	Es una familia de procesadores de la compañía Intel.

RESUMEN

Muchos procesos informáticos en las empresas están siendo soportados por computadoras personales, esto es especialmente cierto en las pequeñas y medianas empresas de Guatemala. En estas empresas las computadoras personales contienen las aplicaciones informáticas con las reglas del negocio. La decisión de utilizar ese tipo de computadoras reside en su bajo costo comparado con alternativas empresariales como servidores y equipo de alto rendimiento. Esto presenta un problema, debido a que las computadoras personales tienen un tiempo de vida en el mercado, mucho menor al del equipo empresarial, los dispositivos se desactualizan y conseguir repuestos o reemplazos se vuelve, en corto tiempo, una necesidad muy costosa. Los costos se incrementan de tal forma que esa es una de las razones por las cuales las empresas migran sus procesos de negocio a equipos más modernos.

En ocasiones se deben desarrollar las aplicaciones informáticas nuevamente para el nuevo equipo de cómputo porque el traslado de las aplicaciones informáticas se ve frustrado al encontrar que las computadoras personales modernas no soportan tecnologías viejas. El costo de desarrollar las aplicaciones utilizando las nuevas tecnologías es significativo y poco justificado cuando los sistemas antiguos sirven adecuadamente su propósito, y el único motivo por el cual se desarrollan las nuevas aplicaciones es porque no se encuentra reemplazo para el equipo que ha dejado de funcionar.

Por lo anterior, se justifica el presente estudio el cual intenta utilizar técnicas de virtualización para extender la vida útil de una aplicación informática de una empresa guatemalteca. De esta manera, aún cuando se adquiriera una computadora personal moderna que no soporte las aplicaciones informáticas legado, estas aplicaciones puedan seguir funcionando en el equipo nuevo con la ayuda de la capa de virtualización.

Para lograr el objetivo principal, el cual es extender la vida útil del sistema legado, se plantearon tres objetivos específicos: el primero, que las aplicaciones informáticas legado funcionen correctamente, en otras palabras que pasen el 100% de los casos de prueba; el segundo, que el uso de una computadora personal moderna signifique un incremento en el rendimiento de la aplicación de al menos 10% aun cuando la capa de virtualización agregue un costo al rendimiento; y el tercero, no interrumpir las actividades de la empresa donde se realice el estudio, esto quiere decir que se trabajará en una computadora que se designe como “libre” para la virtualización y las pruebas del sistema legado.

Utilizar una capa de virtualización para extender la vida útil del sistema legado presenta varias ventajas, entre ellas, la eliminación de la necesidad de desarrollar nuevamente las aplicaciones para el equipo nuevo, o si hay planes de migración de las tecnologías legado a nuevas tecnologías entonces la virtualización sirve de paso intermedio para mezclar aplicaciones nuevas y legado en un mismo equipo de computo.

VARIABLES

- **Costo directo de implementación (en quetzales):** es el costo directo de los insumos, recursos, aplicaciones informáticas y demás gastos directos que se necesiten para implementar la propuesta. En esta variable no se incluyen los costos indirectos, como el pago de luz eléctrica.
- **Tiempo de implementación (en días):** es el tiempo utilizado para implementar la propuesta.
- **Número de funciones (en casos de uso):** es el número de funcionalidades esenciales del sistema.
- **Número de funciones realizadas correctamente (en casos de uso):** es el número de funciones que fueron realizadas correctamente por el sistema en un ambiente virtual
- **Número de funciones realizadas incorrectamente (en casos de uso):** es el número de funciones realizadas incorrectamente por el sistema en un ambiente virtual.
- **Tiempo fuera de línea del sistema (en horas):** es el tiempo que el sistema no está disponible para los usuarios de la empresa durante el tiempo de implementación de la propuesta.

- **Tiempo de ejecución prueba de rendimiento original (en segundos):** es el tiempo transcurrido durante la realización de la prueba de rendimiento para la plataforma actual.
- **Tiempo de ejecución prueba de rendimiento virtualizada (en segundos):** es el tiempo transcurrido durante la realización de la prueba de rendimiento en el sistema virtualizado.
- **Uso del procesador prueba de rendimiento original (en porcentaje):** es porcentaje de procesador que utiliza el sistema actual durante la prueba de rendimiento.
- **Uso del procesador prueba de rendimiento virtualizada (en porcentaje):** es el porcentaje que el caso de prueba utiliza en promedio durante las pruebas de rendimiento dentro del sistema virtual.
- **Número de incompatibilidades con dispositivos (en unidades):** es el número de incompatibilidades que tenga el sistema actualmente.
- **Número de incompatibilidades con dispositivos bajo virtualización (en unidades):** es el número de incompatibilidades luego de implementar la propuesta.

OBJETIVOS

El objetivo general de este trabajo es evaluar técnicas de virtualización para extender la vida útil de un sistema legado en una empresa guatemalteca.

La solución virtual se considerará satisfactoria si cumple los siguientes objetivos específicos:

1. Migración satisfactoria del sistema legado de una empresa guatemalteca a un ambiente virtual, esto supone que el sistema virtualizado pase el 100% de los casos de prueba establecidos por el contacto de la empresa.
2. Mejora al rendimiento del sistema legado del 10% o más, haciendo que este pueda aprovechar de mejor manera el equipo más moderno en el cual corre gracias a la capa de virtualización.
3. Este estudio de virtualización no interrumpirá la productividad de la empresa.

HIPÓTESIS

Un sistema legado realizado en Foxpro 2.6 para D.O.S. aumentará su tiempo de vida y mejorará su seguridad, compatibilidad y rendimiento si se utiliza una capa de virtualización gratuita, la cual permita la ejecución del sistema en equipo moderno, utilizando las plataformas Windows Vista y OpenSUSE 10.3.

INTRODUCCIÓN

Empresas guatemaltecas pequeñas y medianas utilizan computadoras personales para hospedar servicios informáticos. Dichas computadoras resultan ser servidores de aplicaciones de bajo costo comparadas con alternativas empresariales de alto rendimiento. El ahorro en la adquisición del equipo de cómputo viene acompañado de soporte inexistente o insuficiente, lo que limita grandemente el tiempo de vida del equipo. Los problemas surgen cuando los repuestos o reemplazos para dicho equipo están desactualizados, lo cual ocurre con frecuencia dado lo cambiante de la tecnología en el área de las computadoras personales; simplemente, es muy costoso arreglar el equipo, al punto que las aplicaciones en dicho servidor deben ser desarrolladas nuevamente.

Lo anterior no es un problema en una empresa que cuente con recursos disponibles para expansión y que esté dispuesta a invertir en aplicaciones nuevas, sin embargo, en las empresas pequeñas y medianas, en las cuales se pueden desarrollar nuevamente una aplicación que ya está funcionando y sirve su propósito adecuadamente puede ser considerado como un gasto.

Por lo tanto, en el presente trabajo se muestra un método para extender el tiempo de vida de aplicaciones hechas en FoxPRO el cual, bueno o malo, es un ambiente que ha sido utilizado grandemente en empresas guatemaltecas.

Por lo anterior, extender el tiempo de vida de ese tipo de aplicaciones merece atención para que las empresas guatemaltecas, que aún cuentan con ese tipo de sistemas, decidan el momento adecuado para desarrollar nuevas aplicaciones informáticas, y que no sean forzadas por un desperfecto en su equipo de cómputo.

Para lograr el objetivo de extender el tiempo de vida de las aplicaciones legado, se utilizó en este estudio técnicas de virtualización, las cuales permiten hacer independientes las aplicaciones del equipo, haciendo posible el traslado de dichas aplicaciones a otro tipo de equipo, más moderno, para el cual no fueron desarrolladas.

Este estudio está dividido en tres capítulos, el primero establece las bases teóricas de virtualización; desde la historia hasta los tipos de herramientas para virtualizar, son parte del contenido de dicho capítulo. El segundo, es una explicación de los sistemas legados en el contexto de este trabajo. El tercer capítulo es el estudio por sí mismo, iniciando con la descripción del sistema que se virtualizará, siguiendo con los aspectos de funcionalidad y calidad que debe cumplir la solución virtual, para luego seleccionar la herramienta más apta para la virtualización, entonces continúa con la virtualización propiamente dicha para terminar con la validación de la solución virtual.

1. VIRTUALIZACIÓN

1.1. Introducción al concepto de virtualización

La palabra Virtual significa “algo que tiene existencia aparente y no real” (RAE, 2006). Este concepto se ha utilizado ampliamente en muchas áreas tecnológicas, en especial en el área de la tecnología de información. En el área de la tecnología de la información se le llama virtualización al conjunto de técnicas que se utilizan para crear sustitutos de aquellos componentes que son esenciales para el funcionamiento de otros componentes informáticos. Estos componentes pueden ser de cualquier índole que se encuentre en el ámbito de la informática, tanto equipo o aplicaciones informáticas, e.g., computadoras, equipo de comunicaciones, sistemas operativos, procesadores de texto, entre otros.

Cuando se sustituyen componentes informáticos, ya sea equipo o aplicaciones, por otros componentes que podrían ser componentes virtuales, se debe incurrir en el menor número de cambios o modificaciones a los componentes anteriores que dependan del componente que se está sustituyendo. Esto quiere decir, que al cambiar un componente por otro, todo debe seguir funcionando igual que antes (o mejor).

Por ejemplo, cambiar una impresora por otra de distinto fabricante que inherentemente es distinta en diseño, debe implementar una capa de virtualización, esto es una capa que imita la impresora original, los componentes que dependen de ella deben funcionar sin ningún cambio, la computadora, los programas de computo no necesitaran ser alterados para interactuar con la nueva impresora.

Entonces, para que los componentes informáticos funcionen correctamente, los componentes virtuales deben tener la apariencia de los componentes originales. Esta apariencia, en el área de tecnología de información, se refiere a la forma en que interactúan, en otras palabras, los componentes virtuales deben interactuar con los componentes anteriores como los componentes que desean reemplazar.

Una de las características que se debe resaltar, de los componentes virtuales es que el componente virtual no tiene el mismo diseño interno que el componente real. Aunque el componente virtual debe aparentar fielmente el componente real. El componente virtual es una imitación y en esa calidad su forma de funcionar interna es distinta a la del componente real. El imitador además de tener la funcionalidad que tiene el original debe tener la funcionalidad inherente a la de un imitador. La virtualización permite que se pueda trasladar un sistema a diferentes ambientes siempre y cuando éstos ambientes imiten el ambiente original.

Una analogía de este concepto en otro campo es la siguiente: En un zoológico se intenta imitar el ambiente natural de cada especie. Así, en la jaula de los osos se construyen cuevas artificiales que se asemejen a las cuevas de su hábitat natural, esto es para que los osos no tengan problemas para vivir en su nuevo ambiente; en el área de cocodrilos existen zanjas, las cuales se llenan de agua para que parezcan ríos, los ríos son el ambiente natural de este animal, de esta manera pueden sobrevivir. Cada una de las jaulas y ambientes del zoológico no intenta más que simular el ambiente natural de algún animal, de esta manera cada especie puede sobrevivir en ese ambiente artificial. Entonces podríamos decir utilizando el término virtual, que el zoológico es un ambiente virtual que simula el hábitat de cada animal que allí vive. Virtualización bajo esta analogía es el conjunto de recursos y acciones necesarios para mantener en funcionamiento el zoológico y que los animales que allí viven, puedan desarrollarse plenamente.

La virtualización utiliza grandemente las técnicas de modelación y simulación para poder imitar el ambiente original. Por una parte modelar significa extraer las características de un sistema de la realidad y simplificarlo de tal forma que se pueda analizar para resolver problemas. Por la otra parte, simular es llevar el modelo a experimentación utilizando variables con validez científica. Con la modelación y simulación se puede entonces simplificar algún fenómeno de la realidad con el objeto de predecir los resultados de las acciones realizadas sobre ese fenómeno.

Entonces, la virtualización se vale de los métodos y técnicas de modelado y simulado para crear ambientes que aparenten ser la realidad y sin embargo no lo son. La idea central de la virtualización es por lo tanto, simular o emular el ambiente de un sistema, esto es, imitar la realidad del sistema.

En resumen, la virtualización se le denomina al conjunto de técnicas utilizadas para que los componentes informáticos o parte de ellos interactúen con un ambiente simulado o virtual. En otras palabras, un sistema informático es engañado para funcionar en un ambiente distinto para el cual fue diseñado. Los componentes virtuales son imitaciones de componentes reales pero de distinta naturaleza, es importante notar esto porque el crear un componente virtual significa agregar la funcionalidad de los componentes originales y la que permite imitarlo.

1.2.Historia de la virtualización de sistemas informáticos

No se puede exponer la historia de la virtualización sin mencionar algunos aspectos de la simulación de un sistema y su historia. La virtualización esta muy relacionada con la simulación porque la virtualización utiliza técnicas de simulación para lograr su objetivo. La simulacion es un pilar en el área de virtualización, desde el punto de vista que un sistema virtual debe aparentar o mejor dicho simular otro ambiente, para lo cual se utilizan las técnicas de modelación y simulación de sistemas. En otras palabras, la simulación en este tema se utiliza para lograr compatibilidad entre el sistema virtual y el virtualizado.

Uno de los principales exponentes en el campo de la simulación fue John von Neuman, a finales de la década de 1940 él fue el primero en usar un modelo matemático repetidas veces para obtener datos estadísticos y así analizar el comportamientos de los sistemas (Smith, 1998). Al inicio uno de los problemas que presenta la simulación de hechos complejos, es la incertidumbre de ciertas variables, para incorporar esos elementos que no se conocen exactamente, von Neuman introdujo variables aleatorias, las cuales aunque aleatorias tienen una representación estadística válida, con esto él pudo simular de forma innovadora eventos de la vida real con mayor grado de exactitud comparado con los modelos ideales.

A este método se le nombró El Método Monte Carlo. John von Neuman utilizó este método para analizar experimentos de neutrones y la efectividad de bombardeos aéreos. Sin embargo, la simulación sigue siendo utilizada en gran medida por empresarios que quieren predecir la mayor productividad en sus fábricas.

Además de los empresarios, los investigadores en el área de informática encontraron útil la simulación, entre estos un grupo del Instituto de Tecnología de Massachussets (MIT), el cual estaba dirigido por el profesor Fernando Ellos publicaron en 1959 un artículo titulado *“Compatible Time-Sharing System” (CTSS)* el cual marco el inicio de la virtualización en los ambientes informáticos (Varian, 1997), aunque el término virtualización fue acuñado pocos años después (Singh, 2006). El artículo trataba sobre compartir los recursos de un sistema para que otros sistemas pudieran trabajar en él. A esto se le llamo un Sistema de Tiempo Compartido.

Pocos años después del artículo del MIT en el centro Watson de desarrollo de la empresa IBM se estaba desarrollando un proyecto llamado M44/44X, el cual tenía como objetivo evaluar nuevas técnicas para compartir el tiempo de procesamiento de un sistema informático. La idea principal era tener una máquina cuyo modelo era IBM 7044 y simular en esa misma máquina varias máquinas del mismo modelo. A la máquina física se le llamó M44 y las máquinas simuladas eran las 44X de ahí el nombre M44/44X.

Lo anterior significaba crear varios ambientes de cómputo del mismo tipo pero aislados uno del otro de tal forma que pudieran coexistir simultáneamente. Aunque este sistema implementaba varios conceptos de lo que hoy conocemos como virtualización, no implementaba una virtualización completa, porque requería que el sistema físico fuera del mismo tipo que el sistema simulado.

El artículo del grupo del MIT y el proyecto M44/44X dieron lugar a otro proyecto denominado CP-40 de IBM, el cual permitía el uso de varios sistemas operativos en una misma máquina. El líder de este proyecto era Robert Creasy quien era miembro del grupo del MIT quienes habían publicado el artículo antes mencionado.

El sistema CP-40 tenía como objetivo probar nuevos conceptos como la memoria virtual, la cual se explicara más adelante, y los conceptos de tiempo compartido. Aunque este, al igual que el proyecto M44/44X, era solo un proyecto de investigación en él se implementó el concepto de virtualización completa (Adair et al, 1966) (Comeau, 1982).

Las técnicas de virtualización desarrolladas en el sistema CP-40 fueron incorporadas en el sistema de control CP/CMS, el cual fue instalado en las máquinas IBM S/360 las cuales fueron un éxito debido a sus mejoras. Para el año de 1969 CP/CMS estaba siendo usado en 15 sitios, para ese entonces era un número significativo (Creasy, 1981).

Luego IBM reemplazo CP/CMS para llamarle VM (del inglés *Virtual Machina*). Con estos hechos se marco el inicio de las máquinas virtuales corporativas.

Excluyendo los emuladores de consolas de video juegos, las máquinas virtuales habían sido usadas exclusivamente en ambientes empresariales y de alto costo hasta que la empresa VMWare implemento una forma de virtualización de sistemas para el hogar y servidores. Con esto nació la época de la virtualización de escritorio, en plataforma x86, la cual se explicará con detalle en la sección 1.6.

1.3.Virtualización de sistemas informáticos

Al conjunto de componentes tanto físicos (equipo informático) como lógicos (aplicaciones informáticas) que interactúan entre sí con un objetivo definido y que proveen información sensible y útil para una persona se le llama Sistema Informático.

Por una parte los componentes físicos, también llamados equipos informáticos, son las partes físicas del sistema, tales como los teclados, ratones, impresoras, monitores y en general todo aquello que se puede tocar y ver directamente. Para denominar este tipo de componente se usa comúnmente la voz inglesa *hardware*, refiriéndose a la parte dura de la computadora.

Por la otra los componentes lógicos, también llamados aplicaciones informáticas, son programas de cómputo. Estas aplicaciones no se pueden observar directamente, ya que estas funcionan en partes del equipo informático que no se pueden observar a simple vista, y su función es dirigir las operaciones del equipo informático. Por ello, lo único que se observa de las aplicaciones informáticas son los resultados de las órdenes a estos equipos. Se puede decir que el equipo informático es análogo al cuerpo humano y las aplicaciones informáticas análogas a los pensamientos y motivaciones.

Los componentes informáticos forman parte y funcionan bajo un ambiente, este ambiente es por lo regular otro componente informático. Por ejemplo, una hoja electrónica de cálculo es un componente informático que necesita para funcionar un sistema operativo que resulta ser otro componente informático. La hoja electrónica de cálculo no funcionaria sin el sistema operativo.

Virtualizar un componente informático es reemplazar su ambiente por un componente virtual. Por ejemplo, virtualizar una hoja de cálculo significa reemplazar el sistema operativo, por otro que imite ser el original, con esto la hoja de cálculo debe seguir funcionando a pesar de estar en un ambiente distinto al original.

Hay casos en los cuales el ambiente del componente informático es equipo físico. La mayoría de sistemas operativos funcionan de esta forma, virtualizar este tipo de sistemas requiere por lo regular el uso de una capa de virtualización lógica. Esta capa lógica es la encargada de proveer la funcionalidad del equipo físico y de imitarlo, así permitiendo que el sistema a virtualizar funcione de forma adecuada. A esta capa se le llama comúnmente Máquina Virtual o Emulador. Lo que hay que notar de esto es que para estos casos el sustituto del equipo físico, la computadora, es una aplicación informática.

Lo anterior plantea que un equipo informático puede ser virtualizado utilizando una aplicación lógica. Este tipo de virtualización ha sido bastante utilizado en ocasiones cuando no se cuenta con el equipo informático en cuestión.

Para virtualizar un equipo informático entonces se recurre a imitaciones lógicas (como modelos) para simular su funcionamiento. Este punto se explica con más detalle en la sección 1.5.

Para llevar a cabo la virtualización de algún componente informático es necesario identificar los componentes que interactúan en todo el sistema. Para luego identificar aquellos que serán afectados por la virtualización. A continuación se enumeran los componentes más importantes en la virtualización de un sistema informático:

- Sistema original
- Sistema virtualizado
- Capa de virtualización
- Sistema externo

1.3.1.Sistema original

La virtualización de los sistemas informáticos tiene como objetivo reemplazar un componente informático por una alternativa virtual, el componente que se reemplazara se le llama sistema original refiriéndose a su complejidad y posibilidad de ser aislado del resto del sistema informático. En otras palabras el sistema original es el componente que se pretende imitar. El sistema puede ser un componente lógico (aplicaciones informáticas) o físico (equipo informático). Si fuera un sistema físico por lo regular se reemplaza con un componente lógico como se explica en la sección 1.5.

Es importante notar el nuevo nombre que adopta el componente informático luego de ser identificado. Antes se le denomina un componente y pasa a ser referido como un sistema. Esto es porque el componente debe ser aislado del sistema informático para manejarlo como una unidad independiente la cual puede ser identificada completamente. Esta identificación permite seleccionar la mejor herramienta de virtualización que se puede utilizar en cada caso. Existen muchas herramientas de virtualización cada una especializada en un tipo de sistema, en la sección 1.8 se detalla al respecto.

El aislamiento del sistema original implica un período de tiempo en el cual los sistemas que dependen de el no funcionarán. A este tiempo se le llama tiempo de recuperación (TDR).

Al virtualizar un sistema informático se debe disminuir el tiempo de recuperación en lo menos posible. El funcionamiento de algunas empresas depende de sus sistemas informáticos, por esta razón virtualizar un sistema crítico es una tarea que se debe justificar plenamente.

No siempre es conveniente virtualizar un sistema informático, sin embargo, algunas de las razones para contar con una alternativa virtual para esos sistemas son:

- Baja utilización del sistema externo.
- Costos altos de infraestructura de computación, equipos variados.
- Altos costos de mantenimiento del sistema actual, cuando la alternativa virtual en algunos casos es más barata.
- La capa de virtualización puede manejar de mejor manera la portabilidad.
- Capa para disminuir el riesgo de desastres.

1.3.2.Capa de virtualización

La capa de virtualización es un componente lógico que imitando el sistema que esta siendo reemplazado permite que los demás componentes en el sistema informático funcionen adecuadamente.

La capa de virtualización es el componente virtual con el cual se sustituye el sistema original. Para que la capa de virtualización funcione adecuadamente debe proveer una interfaz idéntica a la del sistema original. Esta interfase es la parte del componente virtual que interactua con los demás componentes.

Existen varias formas en que se presenta la capa de virtualización, si se trata de la virtualización de un equipo informático se le llama máquina virtual. La máquina virtual provee toda la funcionalidad de un equipo real, como una computadora, implementando mucho de los componentes de esta de forma virtual, como los dispositivos de comunicaciones y los dispositivos internos de una computadora. Las máquinas virtuales se detallan en la sección 1.5.

Antes de virtualizar hay que seleccionar o desarrollar la capa de virtualización que se utilizara. Las distintas herramientas gratuitas de virtualización son expuestas en la sección 1.8.

1.3.3.Sistema virtualizado

Todos los componentes que dependen del sistema original se pueden agrupar lógicamente en un sistema llamado sistema virtualizado, también se le llama sistema huésped. Este sistema debe de funcionar en el ambiente virtual sin modificaciones, esto quiere decir que la funcionalidad que este sistema necesita debe de ser completamente proporcionada por la capa de virtualización. Sin embargo, hay casos para los cuales se admiten modificaciones al sistema virtualizado siempre y cuando esto esté considerado para el caso específico, algunos tipos de virtualización necesitan hacer cambios en los sistemas a virtualizar como se detalla en la sección 1.4.5.

Los sistemas virtualizados deben funcionar como lo hacían inicialmente, en algunos casos no es así porque la capa de virtualización no provee la funcionalidad adecuada, por esta razón cuando se virtualiza un sistema debe de ser llevado a prueba para comprobar su adecuado funcionamiento.

Otra de las dificultades encontradas al virtualizar un sistema es cuando estos sistemas no siguen la documentación establecida para su construcción y se salen de los lineamientos establecidos en su diseño. Entonces la capa de virtualización debe de implementar también los funcionamientos erróneos, estos son los que no cumplen con el diseño original, de los sistemas originales.

Lo anterior significa que para hacer una capa virtual en la cual los sistemas virtualizados funcionen adecuadamente se debe de tener un amplio conocimiento del sistema original. Afortunadamente existen actualmente varias soluciones que implementan adecuadamente muchas de las plataformas más populares, estas soluciones se detallan en la sección 1.8.

1.3.4.Sistemas externos

El sistema externo es el ambiente donde funciona el sistema original, es importante identificar este ambiente porque componentes del sistema externo interactúan con el sistema virtualizado, la capa de virtualización debe permitir esta comunicación como lo hace el sistema original. La importancia radica en que en algunos tipos de virtualización se agrega una capa adicional al sistema informático, la capa de virtualización no solo reemplaza al sistema original, también es utilizada con otros sistemas de soporte que permiten su funcionamiento, por esa razón verificar que el sistema huésped pueda comunicarse adecuadamente con el sistema externo, esto es con los componentes que están fuera de la capa de virtualización.

Por ejemplo, el sistema externo de una computadora es el ambiente donde funciona dicha computadora, esto incluye las computadoras de la misma red, la fuente de poder eléctrico, impresoras, escáneres, ratones, monitores, teclados, en general todo dispositivo que interactúa con la computadora.

La capa de virtualización debe permitir que los sistemas huéspedes tengan la capacidad de utilizar las impresoras o los escáneres y en general aquellos componentes del sistema externo que se necesiten.

En algunos tipos de virtualización como se muestra en la siguiente sección, no simulan todas las partes del sistema original, así como tampoco dejan que el sistema huésped se comunique con el sistema exterior, esto depende de las necesidades de la organización que implemente la solución de virtualización.

1.4. Tipos de virtualización de sistemas informáticos

La virtualización se puede proveer a distintos niveles, esto depende de en que medida se necesite simular el sistema original para lograr que el sistema huésped funcione adecuadamente. Un nivel es por lo tanto, el grado de funcionalidad que provee la capa de virtualización.

A continuación se presenta una lista donde se enumeran los tipos de virtualización, según el nivel de emulación que provean. Luego se explicara cada uno de los incisos, en orden de funcionalidad, el primero provee la mayor funcionalidad y esta va decreciendo con cada uno.

- Emulación y simulación.
- Virtualización nativa.
- Virtualización soportada por el equipo informático.
- Virtualización parcial.
- Paravirtualización.
- Virtualización a nivel del sistema operativo.
- Virtualización a nivel de aplicación.

1.4.1.Emulación y simulación

Según el Diccionario de la Lengua Española, la palabra emular significa “Imitar las acciones de otro procurando igualarlas e incluso excederlas”, con ese sentido, la virtualización que utiliza emulación intenta imitar todo el sistema y no parte de él. La emulación permite crear ambientes virtuales, en los cuales puedan funcionar aplicaciones hechas para otro sistema, el sistema original.

Lo que se virtualiza en el caso de la emulación es el equipo informático (*hardware*), por eso la emulación es una forma de virtualización completa, ya que intenta imitar todo el sistema original. La emulación permite que las aplicaciones funcionen sin modificaciones en el ambiente virtual. El hecho que una aplicación pueda ser utilizada en un ambiente virtual sin modificación alguna, hace de este tipo de virtualización bastante atractiva, sin embargo, el costo de crear un sistema de virtualización de este tipo es mayor comparado con los otros tipos.

Los sistemas informáticos que emulan otro sistema se les conocen como emuladores. Estos emuladores pueden ser lógicos o físicos, los emuladores lógicos son aplicaciones informáticas, mientras los emuladores físicos son equipos informáticos. El tipo de emulador dependerá de lo que se pretende emular y las condiciones para ello.

Los emuladores lógicos funcionan como cualquier aplicación informáticas, con la salvedad que sirven como ambiente en el cual funcionará una aplicación diseñada para funcionar en otra plataforma.

Los emuladores pueden emular una gran variedad de sistemas informáticos, entre ellos están: Computadoras, asistentes personales digitales, calculadoras digitales, consolas de video, impresoras, entre otros.

Un uso muy frecuente de los emuladores es la emulación de sistemas de entretenimiento digital o videojuegos, con un emulador una persona podría jugar algún videojuego en su computadora, aún cuando el juego haya sido diseñado para una consola totalmente distinta.

Por otra parte, la simulación según el diccionario es “Representar algo, fingiendo o imitando lo que no es”, con ese sentido, la simulación aunque sea similar al concepto de emulación difiere en su alcance. Simular es simplemente imitar algo, mientras emular intenta simularlo y superarlo.

En emulación y en general en todo tipo de virtualización se utilizan técnicas de modelación y simulación, sin embargo, un emulador difiere de un simulador, un emulador crea un ambiente virtual, por otra parte, un simulador imita únicamente el sistema, sin tener una capa de virtualización (Marat, 2000).

Por ejemplo, un emulador de una computadora permitiría que todos los sistemas diseñados para esa computadora funcionaran sin modificaciones, mientras un simulador ofrecería la funcionalidad de la computadora, pero sin utilizar los sistemas diseñados para ella, sino otros simulando ser los originales.

Algo que hay que notar en este tipo de virtualización es que la emulación oculta el equipo informático de forma que el sistema huésped no puede utilizarlo directamente. Entonces los emuladores sirven como cajas de arena en las cuales los sistemas huéspedes están limitados por el emulador. Esto permite que se puedan crear políticas más rígidas de seguridad y de protección al sistema externo en caso de fallas del sistema huésped, sin embargo conlleva una penalización al rendimiento ya que el emulador debe transformar toda la funcionalidad del sistema exterior a la funcionalidad requerida por el sistema huésped, esa transformación es costosa en el tiempo de operación, ya que en un procesador cada transformación utiliza tiempo de procesamiento.

Ejemplos de emuladores y simuladores:

En la actualidad hay varios emuladores que se pueden obtener de forma gratuita, uno de ellos es el emulador de la consola de entretenimiento Super Nintendo llamado ZSNES, el cual permite que el usuario use los juegos realizados para dicha consola, sin necesidad de poseer la consola.

Para Zsnes algunos de los requisitos son poseer una computadora donde correr el emulador y el juego en forma de imagen lógica. Para más información la página Web <http://www.zsnes.com> tiene indicaciones para obtener el emulador y hacerlo funcionar.

En la página <http://www.hpmuseum.org/simulate/simulate.htm> se encuentran simuladores de calculadoras científicas, las cuales pueden ser usadas en una computadora. Este tipo de simulación imita toda la funcionalidad de esas calculadoras permitiendo que el usuario pueda observar una en funcionamiento sin tenerla físicamente.

1.4.2.Virtualización nativa

La virtualización nativa permite que un sistema operativo sea virtualizado sin modificaciones. A diferencia de la emulación la virtualización nativa no emula todas las funcionalidades del sistema original. Este tipo de virtualización aprovecha los componentes del sistema original para ofrecer la funcionalidad a los sistemas huésped. En otras palabras, la virtualización nativa no reemplaza del todo el sistema original, pero agrega una capa de virtualización que permite simular las parte que reemplaza y deja al sistema huésped utilizar libremente las que no. Así la virtualización nativa típicamente se utiliza para crear un ambiente similar al del sistema original pero separado por la capa de virtualización para que puedan coexistir varios sistemas huéspedes que de otra forma no podrían hacerlo.

Este tipo de virtualización permite que sistemas excluyentes puedan coexistir en un mismo ambiente siempre que estos estén diseñados para una misma arquitectura, por ejemplo, se podría tener un sistema operativo Linux i386 funcionando dentro de una capa virtualización que funciona dentro de en un sistema operativo OSX Intel. Ambos sistemas operativos tanto Linux i386 como OSX Intel utilizan arquitectura de procesador Intel i386 para funcionar, entonces la capa de virtualización no simula todo el procesador, únicamente las instrucciones que le permitan que los dos sistemas coexistan en un la misma máquina.

Además de permitir que dos sistemas operativos coexistan en una misma máquina, la virtualización nativa mejora el rendimiento comparado con las técnicas de emulación y simulación, la razón para esta mejora de rendimiento es que la capa de virtualización no utiliza emulación para todas las funciones de los sistemas huéspedes.

A cambio de emular todas las funciones este tipo de virtualización deja que parte de estos sistemas utilicen directamente el procesador y algunos recursos del equipo informático, de esta forma se aumenta el rendimiento considerablemente.

La virtualización nativa virtualiza las parte que aíslan al sistema huésped de las demás aplicaciones informáticas que funcionan al mismo nivel de la capa de virtualización, sin embargo, deja que el sistema huésped utilice directamente partes del equipo informático, en especial el procesador.

1.4.3.Virtualización asistida por el equipo informático

El término x86 es utilizado actualmente para nombrar a la arquitectura que utilizan procesadores compatibles con el conjunto de instrucciones creado por Intel en sus procesadores 8086. Estos procesadores han ido evolucionando pero conservan muchas de las instrucciones originales, aún hoy se encuentran muchos programas hechos para la arquitectura i386. La arquitectura i386 se hizo famosa por el procesador 80386 de Intel el cual salió a la venta en 1985. Los procesadores x86 son bastamente utilizados en computadoras de escritorio y otros dispositivos informáticos, habiendo más de un fabricante de procesadores compatibles x86. Uno de los fabricantes de procesadores compatibles es la compañía AMD.

Tanto Intel como AMD han visto valor en el futuro de la virtualización, esto se ve reflejado en el hecho que ambas empresas han desarrollado tecnología en sus procesadores para brindar funciones de virtualización.

Las funciones de virtualización que brindan AMD e Intel se llaman respectivamente Intel VT y AMD-V. La propuesta de Intel es llamada Intel VT (IVT), esto es una extensión que le permite a los procesadores de Intel proveer un sistema de virtualización para la arquitectura x86. AMD por su parte incorporó su tecnología de virtualización AMD-V a sus procesadores porque ésta provee funcionalidad para virtualizar sistemas de 64 bits.

En la práctica los procesadores más modernos de tanto Intel como AMD brindan estas tecnologías de virtualización. Esto permite que la capa de virtualización utilice la funcionalidad del procesador para aumentar el rendimiento de los sistemas huéspedes.

Como los procesadores proveen extensiones que facilitan la virtualización, la capa de virtualización es más ligera, esta ligereza ayuda al rendimiento general del sistema, porque bajo este ambiente el sistema huésped corre directamente en el procesador y solo existe la capa de virtualización que sirve de monitor de las actividades de ese sistema.

En este tipo de virtualización se hace uso de los recursos del procesador para crear el ambiente virtual como se menciona en el párrafo anterior, aunque este tipo de virtualización en este punto se parece a la virtualización nativa, hay que notar que son dos aproximaciones distintas. La virtualización nativa a diferencia de este tipo de virtualización, deja que el huésped utilice la parte genérica del procesador, esta parte provee las instrucciones aritméticas y lógicas elementales, así la capa de virtualización no tiene que agregar carga adicional al procesador tratando de interpretar dichas operaciones. En este tipo por el contrario, el procesador tiene una extensión del mismo diseñada para proveer toda la funcionalidad del procesador al sistema huésped, siempre siendo aislado de los demás sistema huésped si hubiera.

Para completar la diferencia, en el caso de la virtualización nativa el sistema huésped no puede utilizar directamente todo el procesador ya que este no fue diseñado para virtualizar equipos por lo que la capa de virtualización debe de implementar toda la funcionalidad que le permita virtualizarlo, así aislarlo.

1.4.4.Virtualización parcial

Este tipo de virtualización no trata de imitar completamente el sistema original. Su uso es para compartir recursos entre usuarios. Históricamente este fue el tipo de virtualización utilizada en el sistema IBM M44/44X, en donde no se emulaba todo el sistema, sino una parte de él.

La razón por la cual surgió este tipo de virtualización fue porque se necesitaba compartir recursos entre diversos usuarios o aplicaciones. Con esto nació el concepto de Compartimiento de tiempo (*time sharing*), este concepto se refiere al compartir un mismo recurso entre varios usuarios. De tal forma que la apreciación del usuario fuera que el era el único que utilizaba el recurso. Por esta razón, se intentó imitar parte del sistema para lograr esta ilusión.

La parte que se virtualizó en los sistemas M44/X44 fue inicialmente la memoria, logrando que varias instancias del sistema corrieran simultáneamente en la misma máquina. La segunda instancia del sistema se decía que estaba en una máquina virtual, la cual era otro espacio de memoria que se activaba y servía como memoria principal del sistema virtual.

Actualmente, muchas de estas implementaciones no se consideran máquinas virtuales y han tomado más el nombre de memoria virtual, sin embargo fue el crear la memoria virtual lo que llevó a la creación de las máquinas virtuales modernas.

La virtualización parcial es significativamente más fácil de implementar debido a sus limitaciones, ya que esta no implementa todas las funciones del sistema original, y se centra en virtualizar los componentes mínimos para que dos o más usuarios o procesos puedan coexistir. Dada la complejidad de los sistemas modernos este tipo de virtualización se utiliza principalmente para virtualizar recursos como la memoria, discos duros, entre otros.

Entre las desventajas de este tipo de virtualización esta la poca portabilidad de los sistemas huéspedes a otras plataformas, debido a que esta virtualización es parcial, mucha de la funcionalidad que requiere el sistema huésped puede no estar simulada y trasladar ese sistema a otra plataforma podría resultar retador si la otra plataforma no implementa las funcionalidades que se necesitan.

1.4.5.Paravirtualización

Este tipo de virtualización requiere cambios en el sistema huésped para que pueda ser virtualizado. La capa de virtualización solo provee una interfaz para que el sistema huésped pueda utilizar las funciones del equipo informático. La capa de virtualización en este tipo de virtualización se llama *hypervisor* (XenSource, 2004). El *hypervisor* es el encargado de manipular las llamadas al equipo informático, cada llamada que hace el sistema huésped de este tipo se le llama *hypercall*. En los últimos años este tipo de virtualización ha sido muy exitosa debido a la compañía XenSource que la promueve. XenSource tiene una solución de virtualización llamada Xen la cual inicialmente se utilizaba para correr un Linux en una máquina virtual del mismo tipo.

El *hypervisor* puede ser una aplicación informática o en el caso de Xen un sistema operativo dedicado a la virtualización de sistemas huéspedes. Cuando el *hypervisor* es un sistema operativo no necesita de otro sistema más que del equipo para funcionar y este provee toda la funcionalidad para que varios sistemas operativos funcionen en la misma máquina.

En el caso anterior, el uso del hypervisor se resume a una consola de administración para varios sistemas huéspedes.

La paravirtualización no es como los demás tipos de virtualización en el sentido que para este tipo de virtualización es necesario modificar el sistema huésped. Sin embargo, su uso actual es de la misma índole que el de las máquinas virtuales tradicionales (XenSource, Ibíd.). Y ha encontrado más aceptación en la virtualización de sistemas con código abierto, por su facilidad de modificación, tales como Linux, OpenSolaris, entre otros.

La paravirtualización funciona de tal forma que las llamadas a los dispositivos son capturadas por el *hypervisor* antes de llegar a los dispositivos y son manejadas por este. Esto permite que las llamadas conflictivas y que afectarían a los demás sistemas huéspedes sean manejadas de forma no conflictiva.

La virtualización de este tipo se utiliza mayoritariamente en hacer que varios sistemas operativos funcionen al mismo tiempo. Así se puede tratar un mismo equipo informático como varias máquinas, esto permite que los recursos del equipo sean utilizados al máximo. El utilizar una misma máquina para varios sistemas operativos que sirven de servidores en una empresa se le llama Consolidación de servidores.

Otras compañías como Sun Microsystems, la compañía famosa por Java, utilizan paravirtualización en sus soluciones. Sun provee una solución para administrar sistemas tecnológicos llamada Sun N1 la cual utiliza paravirtualización para consolidar servidores y aplicaciones.

1.4.6.Virtualización a nivel de sistema operativo

Este tipo de virtualización se enfoca en la virtualización de sistemas operativos, permitiendo que varios sistemas operativos funcionen en una misma máquina. A diferencia de la paravirtualización la virtualización a este nivel no necesita modificaciones en los sistemas huéspedes para que funcionen.

Por lo regular, este tipo de emulación se lleva a cabo con máquinas virtuales, las cuales emulan todo el equipo informático para que los sistemas operativos huésped funcionen sin problemas. Entre las máquinas virtuales que brindan emulación del sistema operativo están: VMWare Server, VirtualPC, QEMU, VirtualBox. En la sección 1.8 se encuentra más información acerca de estas máquinas virtuales.

1.4.7.Virtualización a nivel de aplicación

La virtualización a este nivel permite que una aplicación pueda funcionar en diversas plataformas sin ser modificada. La idea principal es crear una aplicación que sea portable en los diversos sistemas operativos y entornos computacionales. Este tipo de virtualización no reemplaza al sistema operativo, es una capa que se coloca entre la aplicación y el sistema operativo para que simule un ambiente conocido, el cual puede ser implementado en una gran diversidad de sistemas operativos diferentes, permitiendo el traslado de la aplicación de un sistema operativo a otro sin modificaciones.

Aunque lenguajes de programación como C permitían y facilitaban la portabilidad de aplicaciones, la desventaja de C era que se debía contar con el código fuente para poder obtener una aplicación para las diversas plataformas, en muchas ocasiones el código fuente de la aplicación debía ser modificado para ajustarse a las peculiaridades de cada plataforma. En cambio la virtualización de aplicaciones permite el funcionamiento de las aplicaciones informáticas en diversas plataformas, sin modificaciones y sin necesidad de simular todo el sistema operativo para que funcionen, como sucede con la virtualización completa.

El mejor ejemplo de este tipo de virtualización es la máquina virtual de Java, la cual existe en diversas plataformas.

Si un programador hace una aplicación correctamente en Java puede, según lo dice el motto de la tecnología, hacerla una vez correrla donde sea. Java permite que una aplicación se haga para un sistema operativo y permite que esa aplicación funcione en varios sistemas operativos siempre y cuando cuenten con la capa de virtualización llamada Máquina Virtual de Java.

Como lo indica el índice de popularidad de lenguajes de programación de TIOBE, el lenguaje más popular es Java, el cual se usa en la máquina virtual del mismo nombre (TIOBE, 2007), lo que demuestra su éxito por la flexibilidad que provee.

Siguiendo este concepto, y dada la popularidad de Java, Microsoft también creó un tipo de máquina virtual en la forma de un motor llamado CLR, el cual provee la funcionalidad necesaria para que funcionen las aplicaciones dentro de él. El motor CLR es el que está detrás de la tecnología .NET, la cual es la competencia propuesta por Microsoft para Java.

1.5.Máquinas virtuales

Una máquina virtual es un sistema informático que simula ser una máquina real. La idea es tener una aplicación informática que simule todos los dispositivos de algún equipo informático. Con una máquina virtual se puede simular la existencia de equipo informático que no se tiene. En general, las máquinas virtuales son sistemas informáticos que proveen toda la funcionalidad necesaria para virtualizar otro sistema informático.

Una máquina virtual permite la ejecución de sistemas operativos en un ambiente de *software*, esto quiere decir que programas para computadora como los sistemas operativos, los cuales son diseñados para funcionar directamente con el equipo informático, funcionen dentro de una aplicación informática y que interactúen directamente con ella como si fuera el equipo informático para el cual fueron diseñados.

Como la máquina virtual funciona de intermediario, entre el equipo informático y el sistema operativo, puede adaptar el sistema operativo para que funcione en ambientes distintos sin modificaciones. Esto permite que hayan dos tipos de máquinas, una pública que es donde se esta ejecutando la máquina virtual, y otra privada donde se ejecuta el sistema operativo huésped.

Entonces si en un sistema operativo se instala una o más máquinas virtuales, estas permiten ejecutar uno o más sistemas operativos dentro del sistema operativo inicial. Esto permite consolidar los servicios de varias máquinas en una sola máquina utilizando máquinas virtuales para cada una. Cada sistema huésped estará aislado por la máquina virtual, siendo independiente en todo a excepción del rendimiento el cual es compartido entre todos los sistemas huéspedes.

La consolidación de servidores se lleva a cabo trasladando cada uno de los sistemas operativos y sus aplicaciones a una o más máquinas virtuales. Al final de la consolidación queda un solo servidor, i.e. un solo equipo de computo, con varias máquinas virtuales en las cuales se tienen todos los servicios disponibles. La ventaja de consolidar es aprovechar al máximo el equipo de computo, trasladando a un mismo servidor todos los servicios que antes se encontraban en varios servidores.

1.6.Virtualización de la arquitectura x86

Gerald Popek y Robert Goldberg publicaron un artículo en 1974 titulado “*Formal Requirements for Virtualizable Third Generation Architectures*” en el cual trataban los requerimientos que una arquitectura informática debía cumplir para ser virtualizable.

Según Popek y Goldberg para que un sistema fuera virtualizable debía cumplir tres aspectos: equivalencia, control de recursos y eficiencia. Hoy en día los requerimientos más importantes que se toman en cuenta son la equivalencia y el control de recursos. La eficiencia aunque muy deseable no es parte de los requerimientos actuales, debido a la complejidad de algunas arquitecturas como la x86.

El requisito de equivalencia se refiere a que un programa funcionando bajo un ambiente virtual debe funcionar de igual manera que fuera de este. En otras palabras, los programas informáticos deben funcionar igual en una máquina virtual así como directamente en el equipo informático real. Para comprobar este requisito se evalúan las instrucciones disponibles en las unidades de procesamiento también conocidas como C.P.U. y se catalogan aquellas instrucciones que sean sensibles al contexto, en otras palabras las que pueden funcionar de distinta manera según lo que esta ocurriendo. Luego se hacen casos de prueba para el sistema virtualizado para comprobar el requisito.

El control de recursos se refiere a que la máquina virtual o *hypervisor* deben poseer control total de los recursos que brindan a los sistemas huéspedes. Esto permite mejorar la funcionalidad del sistema huésped permitiendo funcionalidad adicional. Para que se cumpla este requerimiento las instrucciones del CPU deben de ser atrapables, el término atrapable en este contexto se refiere a que en cualquier momento la máquina virtual debe tener el control del CPU. La arquitectura x86 antes de la introducción de los nuevos modelos para virtualización tenía varias instrucciones no atrapables, lo que impedía el cumplimiento de este requisito.

En 1999, la compañía VMWare sacó al mercado su primer producto de virtualización llamado “VMWare Virtual Platform”, el cual era una máquina virtual para arquitecturas x86. Por la complejidad de la arquitectura x86 esta plataforma de virtualización era lenta a comparación de las máquinas reales, sin embargo, implementaba algoritmos complejos que le permitían virtualizar esta plataforma como antes no se había hecho. Los algoritmos utilizados por VMWare fueron patentados en la USPTO y se les otorgó el número de patente 6,397,242.

Como se mencionó antes la arquitectura x86 en un inicio no pasaba la prueba de Popek y Goldberg, sin embargo con la introducción de la máquina virtual de VMWare la industria vio el potencial de la tecnología de virtualización para esta arquitectura y las empresas Intel y AMD sacaron a la venta procesadores amigables para la virtualización.

La tecnología de los procesadores de Intel y AMD para virtualización llevan el nombre de Intel VT y AMD-V respectivamente. Estas dos tecnologías permiten el cumplimiento del requisito de control de recursos y eficiencia en la arquitectura x86.

1.7.Ventajas y desventajas de virtualizar un sistema informático

Como toda tecnología la virtualización tienes sus ventajas y desventajas, a continuación se enumeran las ventajas de virtualizar un sistema informático, seguido por las desventajas de lo mismo.

1.7.1.Ventajas

Las ventajas de utilizar máquinas virtuales son variadas, según un artículo publicado por IBM titulado “*Driving business value with a virtualized infrastructure*” las ventajas para implementar una solución virtual son las siguientes:

- Aislamiento de los sistemas operativos en ambientes privados, una máquina virtual proporciona aislamiento de los sistemas operativos que se ejecutan en cada una de ellas, aparte del rendimiento las máquinas virtuales son independientes y funcionan como máquinas separadas.

- Estandarización, las máquinas virtuales presentan un ambiente estándar para todas las aplicaciones dentro de ellas, esto quiere decir que la plataforma que se presenta a los huéspedes es el mismo para todos y no depende del equipo o sistema operativo donde este siendo ejecutada.
- Consolidación, la unión de distintos servidores en un mismo equipo de computo disminuye los costos y hace la administración de los servicios más fácil por su centralización. El mantenimiento de equipo informático es una tarea regular por la fragilidad de los equipos de cómputo comparado con las aplicaciones informáticas.
- Facilidad de pruebas, las máquinas virtuales permiten crear ambientes de pruebas rápidamente, esto gracias a las herramientas que proveen la mayoría de ellas, como restauración del estado de la máquina a un momento del tiempo, manejar redes virtuales, entre otras.
- Movilidad, trasladar un sistema virtual es tan fácil como copiar un archivo. Como el sistema de una máquina virtual esta estandarizado la copia del sistema virtual funcionara en distintos equipos informáticos con el mismo tipo de máquina virtual.

- Reducción de costos, la eliminación de equipo adicional y su mantenimiento reducen costos en este sentido, además empresas como VMWare y Microsoft ofrecen soluciones de virtualización gratuitas, sin embargo los costos pueden verse incrementados si los administradores de los sistemas informáticos no tienen los conocimientos necesarios para el manejo de la nueva infraestructura con máquinas virtuales.

1.7.2.Desventajas

Las desventajas son las siguientes:

- Impacto en el rendimiento, una máquina virtual agrega costo de procesamiento como cualquier aplicación informática. Además si el sistema que se esta emulando es muy complejo la capa de virtualización tiene simular su funcionamiento lo que implica agregar tiempo de ejecución al sistema huésped. Por esa razón fabricantes como Intel y AMD están preparando sus procesadores para que sean más amigables con las máquinas virtuales, esto se menciona en la sección 1.4.3.
- Riesgo en la concentración de equipo informático, la consolidación de servidores puede ser riesgoso si el equipo donde corre no se tiene con el suficiente respaldo. Si este servidor falla todos los servicios que se hospedan en el fallaran.
- Soporte del equipo informático, como la máquina virtual provee equipo estándar a los sistemas huéspedes, estos sistemas no pueden hacer uso de funcionalidad avanzada del equipo informático donde se hospedan, debido a que la capa de virtualización solo provee la funcionalidad necesaria para que funcionen.

- Licenciamiento de software, es otra desventaja si no se manejan adecuadamente las licencias del software que se virtualiza.
- Necesidad de más capacitación técnica, mantener un ambiente virtual requiere menos *hardware* pero se necesita que el administrador comprenda los conceptos de virtualización. Además requiere otras habilidades de administración por la naturaleza de las máquinas virtuales. Estas habilidades son relacionadas con el manejo de dispositivos virtuales y planificación de utilización de recursos.

1.8.Herramientas de virtualización gratuitas

En esta sección se describen algunas de las herramientas de virtualización gratuitas en el mercado, la lista no intenta ser exhaustiva, sin embargo se incluyen las herramientas con mayor funcionalidad, por esa misma razón, se dejaron fuera de la lista las máquinas virtuales a nivel de aplicación como la máquina virtual de java, y la lista se centra en máquinas de virtualización completa o paravirtualización.

Para considerar una herramienta gratuita se revisó el valor de licenciamiento y su disponibilidad en Internet. Toda herramienta listada no tiene costo de licenciamiento y esta públicamente disponible para descargar desde su sitio oficial.

No se consideró como restricción que la compañía proveedora de las herramientas distribuya el código fuente con la misma.

La siguiente lista contiene máquinas y herramientas gratuitas para virtualización:

- VMWare Server de VMWare Inc.
- Virtual PC 2007 de Microsoft.
- VirtualBox de Innotek.

- XEN.
- QEMU.
- Boschs.
- DOSBox.
- DOSEmu.

En las siguientes secciones se expondrá de forma breve cada una de estas herramientas de virtualización.

1.8.1.VMWare Server

VMWare Server es una máquina virtual que utiliza la tecnología de VMWare Inc. para virtualizar la plataforma x86. La primera versión de VMWare Server fue publicada el 12 de julio de 2006. Es una continuación a la gama VMWare Server GSX pero gratuita. Puede correr además de las máquinas virtuales hechas con los otros productos de VMWare las hechas por VirtualPC. Actualmente soporta los sistemas operativos de 32 y 64 bits, a excepción de Windows Vista edición de 64 bits.

VMWare Server utiliza una arquitectura cliente/servidor la cual permite conectarse remotamente a las máquinas virtuales que se encuentren hospedadas en el servidor. Esto aumenta la facilidad de administración.

Además de la consola de administración llamada VMWare Server Console, el servidor provee servicios VNC los cuales se utilizan para administrar remotamente por medio de un explorador la máquina virtual.

Tabla I. Ficha técnica VMWare Server

Desarrollador	VMWare Inc.
Sistema operativo	Microsoft Windows XP, 2003, Vista, 2008. Linux i386.
Plataforma	x86
Licencia	Propietaria
Tamaño del instalador	150 MB
Sitio WEB	http://www.vmware.com
Limitaciones	No soporta conexiones FireWire. Soporte experimental de aceleración gráfica 3D. No hay soporte para más de 3 botones del ratón.

1.8.2. Microsoft Virtual PC 2007

Esta herramienta fue desarrollada inicialmente por la empresa Connectix, luego fue comprada por Microsoft. Esta máquina provee virtualización para sistemas operativos Windows y para Mac OS X. Actualmente provee emulación para los periféricos comunes de una PC como teclado, ratón entre otros.

Los requerimientos mínimos para VirtualPC es una máquina con plataforma x86 a 400Mhz de procesador, el espacio de memoria requerido para correr el sistema operativo y los sistemas huéspedes.

VirtualPC tiene problemas para correr otros sistemas operativos que no sean Windows o Mac OS X. La razón de esto esta en la emulación de la tarjeta de video, la cual no soporta todos los colores de una tarjeta tradicional (Brunelle, 2006).

La herramienta de virtualización puede ser obtenida sin costo desde la página de Microsoft para Virtual PC, enlace a la página principal se muestra en la ficha técnica. El costo de la guía impresa es de US \$10.

Tabla II. Ficha técnica Microsoft VirtualPC

Desarrollador	Microsoft Corporation
Sistemas operativos	Microsoft Windows Mac OS X MacOS (Desactualizado)
Licencia	Propietaria de Microsoft
Tamaño del instalador	31 MB
Sitio WEB	http://www.microsoft.com/windows/virtualpc/default.aspx

1.8.3. Innotek VirtualBox

VirtualBox es una máquina virtual desarrollada por la empresa Innotek, soporta una gran gama de sistemas operativos huéspedes y corre sobre Linux, Mac OS X y Windows. En enero del 2007 salió la versión de código abierto y gratis, la cual tiene limitaciones con los dispositivos USB.

Con la salida de su versión de código abierto, Innotek por el momento provee dos tipos de licencia para sus versiones de la máquina virtual. La primera licencia es propietaria y su uso es gratuito para estudiantes y uso personal. La segunda es GPL y es gratis para cualquier uso pero incorpora ciertas limitaciones con dispositivos USB.

VirtualBox emula los periféricos más comunes como, la tarjeta de red AMD PCNet, tarjeta de video VESA de 8 MB, discos duros ATA y SCSI, entre otros.

Los requerimientos mínimos para VirtualBox es una plataforma aceptada como Linux, Windows y Mac OS X (experimental), la memoria necesaria es la utilizada por el sistema operativo actual sumada a la del sistema huésped.

Tabla III. Ficha técnica Innotek Virtualbox

Desarrollador	Innotek
Sistema operativo	Linux Windows Mac OS X (Experimental en la version 1.4)
Plataforma	x86
Licencia	GPL y propietaria
Tamaño del instalador	17 MB
Sitio WEB	http://www.virtualbox.org

1.8.4.XEN

Xen es un monitor de una máquina virtual, también llamado *hypervisor*, XEN corre en sistemas modificados Linux o FreeBSD. Como plataforma permite la virtualización de arquitecturas i386, IA 64, x86-64 y PowerPC. Desde la versión 3.0 es posible correr sistemas operativos Windows sin ser modificados.

Xen originalmente fue un proyecto de desarrollo por un grupo de estudiantes de la universidad de Cambridge, ahora XenSource la compañía detrás de Xen es parte de Citrix Systems.

El código fuente de Xen esta bajo la licencia GPL, esto significa que al adquirir el hypervisor Xen también se adquiere el código fuente. Esto permite que se pueda modificar y ampliar según las necesidades de la empresa. Sin embargo, la licencia GPL tiene una clausula en la distribución, en la cual como se obtuve el código fuente, cualquier trabajo derivado que se quiera distribuir debe ir acompañado de su código fuente. Lo anterior solo aplica para la redistribución de la máquina virtual y no para las modificaciones que se hagan internamente.

Tabla IV. Ficha técnica de XEN

Desarrollador	XenSource Inc.
Sistema operativo	Linux FreeBSD
Plataforma emulada	Windows Linux FreeBSD Mac OS X OpenSolaris
Tamaño del instalador	6.51 MB
Sitio WEB	http://www.xensource.com/xen

1.8.5.QEMU

Es una máquina virtual desarrollada por Fabrice Bellard, en un inicio QEMU era gratis pero con licencia propietaria pero desde febrero de 2007 Bellard cambio la licencia a GPL, permitiendo así la redistribución de su código fuente.

QEMU es una máquina virtual muy versátil porque permite emular una gran diversidad de plataformas, no solo i386. Además puede funcionar en una gran variedad de plataformas, como x86, x86-64, PowerPC, SPARC entre otras.

Tabla V. Ficha técnica QEMU

Desarrollador	Fabrice Bellard y otros.
Sistema operativo	Windows Linux Mac OS X FreeBSD OpenSolaris
Plataforma emulada	ARM MIPS x86 x86-64 PowerPC Otras
Licencia	GPL
Tamaño del instalador	4 MB
Sitio WEB	http://fabrice.bellard.free.fr/qemu/

1.8.6.Bochs

Bochs es un emulador hecho en C++ de la plataforma x86 y corre en los sistemas operativos Windows, Linux, Mac OS X y XBox. Además del procesador i386 también emula periféricos de entrada y salida como teclado, ratón, discos duros, memoria.

La desventaja de Bochs con respecto a otras máquinas virtuales es su menor rendimiento, esto se debe a que Bochs es un emulador y no implementa virtualización nativa. Entonces todas las funciones que hace el sistema huésped tienen que ser interpretadas por este emulador, lo que disminuye de gran manera su rendimiento.

La ventaja de Bochs es que por ser implementado como emulador permite un mayor control sobre los sistemas huéspedes, según la página principal de Bochs, los desarrolladores prefieren Bochs para trabajar en el área de sistemas operativos.

Tabla VI. Ficha técnica Bochs

Desarrollador	Comunidad, compañía patrocinadora Mandriva
Sistema operativo	Linux Windows Mac OS X XBox
Plataformas emuladas	i386
Licencia	LGPL
Tamaño del instalador	4 MB
Sitio WEB	http://bochs.sourceforge.net/

1.8.7.DOSBox

DOSBox es un emulador de D.O.S., por lo que no emula el *hardware* totalmente, solo la funcionalidad requerida por las aplicaciones de D.O.S. Este emulador funciona en los sistemas operativos Windows, BeOS, Linux, Mac OS X y otros. Y su función es únicamente emular la funcionalidad de una máquina i286/i386 con sistema operativo D.O.S.

DOSBox provee emulación para los procesadores i286 e i386, tanto en modo protegido como en modo real. El enfoque que los desarrolladores tienen para este emulador se centra en un ambiente para hacer funcionar los juegos del sistema operativo D.O.S. Sin embargo DOSBox también puede ser utilizado para correr aplicaciones empresariales realizadas para D.O.S.

Para emular una máquina con procesador 80286 se necesita una máquina Pentium o compatible de 400 Mhz. Para emular un procesador 80386 DOSBox pide como requerimiento mínimo una máquina con la frecuencia del procesador en 1 Ghz.

Tabla VII. Ficha técnica DOSBox

Desarrollador	Varios, basado en la comunidad.
Sistema operativo	Linux Windows Mac OS X
Plataforma emulada	80286/80386 Sistema operativo D.O.S.
Licencia	GPL
Tamaño del instalador	1.2 MB
Sitio WEB	http://dosbox.sourceforge.net/

1.8.8.DOSEmu

DOSEmu es una máquina virtual de la plataforma i386 pero diseñada especialmente para ejecutar de manera óptima aplicaciones de D.O.S. Implementa virtualización nativa y funciona en plataformas Linux i386.

DOSEmu y FREEDOS han sido combinados en un mismo paquete de instalación para brindar un ambiente D.O.S. gratuito a plataformas Linux. FREEDOS es una alternativa libre de versiones de D.O.S. como MS DOS o IBM DOS. En el sitio Web de esta máquina virtual (<http://www.dosemu.org>) se puede obtener sin ningún costo la versión de DOSEmu y FREEDOS. Por el hecho de que DOSEmu utiliza un sistema maduro de D.O.S. como FREEDOS, DOSEmu es compatible con más aplicaciones para D.O.S. que DOSBox.

Tabla VIII. Ficha técnica DOSEmu

Desarrollador	Basado en comunidad
Sistema operativo	Linux i386
Plataforma	i286/i386 especialmente para D.O.S.
Licencia	GPL
Tamaño del instalador	2.30 MB
Sitio WEB	http://www.dosemu.org

2.SISTEMAS LEGADOS

Un sistema legado también llamado *legacy* es aquel sistema que su soporte técnico es mínimo o inexistente. Las razones por las cuales un sistema pierde su soporte son diversas, las más comunes son cuando la compañía desarrolladora deja de existir o cuando es comprada por otra dejando a los usuarios sin soporte. Con el pasar del tiempo la cantidad de sistemas legados aumenta y las compañías se ven forzadas a abandonar los sistemas legados para migrar su información a sistemas más modernos que brinden mejor soporte.

Uno de los requerimientos de las nuevas tecnologías en un ambiente comercial es la compatibilidad de sistemas legado, esto es porque el costo de migración de un sistema legado a un sistema moderno es elevado. Microsoft Windows implementó sus sistemas operativos NT la máquina virtual para D.O.S. llamada *Virtual DOS Machine*. El problema con la máquina virtual de Windows es que su compatibilidad es limitada, además ha sido discontinuada en versiones de 64 bits de los sistemas operativos Windows XP y Windows 2003 Server.

Uno de los problemas con la virtualización de sistemas legados son las practicas de programación utilizadas en esos sistemas, por ejemplo en D.O.S. se acostumbraba utilizar un ciclo infinito el cual usaba todo el procesador cuando un programa estaba funcionando, esto era posible porque D.O.S. era un sistema mono – usuario, el cual

solo se ejecutaba una aplicación en un tiempo, por lo que las aplicaciones podían utilizar todo el procesador sin ningún problema secundario. Hoy en día los procesadores son multitareas y esta práctica hace que la emulación de programas de D.O.S. sea más difícil porque el emulador no sabe si la aplicación de D.O.S. esta haciendo algo o solo gastando procesamiento.

Como se nota en el artículo de Wikipedia titulado “*Virtual DOS Machine*”, la máquina virtual de DOS para Windows tiene algunos inconvenientes como el hecho que los programas de DOS corran nativamente no provee un método para disminuir el costo del procesamiento. Además su pobre simulación de periféricos hace que los programas de D.O.S. no funcionen del todo bien.

Para solventar estos problemas en el siguiente capítulo se estudian las herramientas de virtualización y su uso en virtualización de la plataforma D.O.S. para lograr ejecutar de manera adecuada aplicaciones hechas en FoxPro 2.6 para la misma plataforma.

3.VIRTUALIZACIÓN DEL SISTEMA LEGADO

3.1.Sistema de pruebas de aptitud

Para este trabajo de investigación la empresa ServiSoft Guatemala ofreció su Sistema de Pruebas de Aptitud para hacer el estudio de virtualización, esto con el objetivo de hacer funcionar el sistema en el sistema operativo OpenSUSE 10.3 y Windows Vista.

Las pruebas de aptitud en general son utilizadas para identificar a las personas aptas para ocupar un puesto en una organización, también se usan como pruebas de admisión en las universidades a nivel mundial, y en muchos otros ambientes donde se requiere medir de alguna forma las aptitudes de una persona.

El producto de Servisoft se utiliza para administrar la información de los solicitantes de admisión a una organización y calificar sus pruebas de aptitud. Este producto es llamado Sistema de Pruebas de Aptitud o con sus siglas SPA: La empresa Servisoft encontró en la propuesta de este trabajo de investigación una posible solución a algunos de los problemas que han surgido con el uso del sistema y su inclusión en sistemas operativos modernos.

Por el motivo anterior la empresa ofreció su producto SPA para ser estudiado en este trabajo haciendo posible el estudio de virtualización de dicho sistema.

La necesidad que llevo a la empresa a considerar la virtualización del sistema son los problemas cada vez más frecuentes en el uso del sistema en equipos nuevos. Los problemas han surgido porque el sistema esta hecho para funcionar en una plataforma considerada sistema legado, esto es DOS y FoxPRO. Sin embargo, los equipos nuevos en especial las computadoras personales, vienen con sistemas operativos de punta, para el año de realización de este trabajo sistemas operativos como OpenSUSE 10.3, Windows Vista, Windows XP, Mac OS X son algunos de los sistemas operativos encontrados en computadoras personales modernas.

Además el mantenimiento del sistema se ha vuelto más problemático. Debido a que muchos de los periféricos, como impresoras y escaneres, que se usaban con la plataforma original se han vuelto cada vez más escasos y/o caros por ser tecnologías que se dejan de usar, también sucede que los equipos no son del todo compatibles con el sistema operativo más moderno.

Otro punto que llevo al estudio de la virtualización de SPA es el auge de sistemas de código libre/abierto como GNU/Linux. Estos están siendo usados por su bajo costo de adquisición, por lo que muchas empresas incluyen planes de migración de los sistemas legado a sistemas de este tipo.

En el caso de ServiSoft, una parte de sus ofertas en productos esta siendo orientado al sistema operativo OpenSUSE 10.3, por lo que requieren una solución que minimice los costos de migración del sistema legado SPA al sistema operativo antes mencionado, esta es la parte donde interviene la solución de virtualización que es objeto de estudio en este trabajo.

La virtualización del sistema SPA es una solución de bajo costo que permitirá el uso de este sistema en el sistema operativo que deseen. Este estudio intenta ser una guía general para la virtualización de este tipo de sistemas, sin embargo, por la extensión que supondría incluir todos y cada uno de los posibles sistemas operativos que existen, solo se incluyen dos de los sistemas operativos para computadoras de escritorio, Windows Vista y OpenSUSE 10.3. Esto debido al alcance propuesto para el presente trabajo y la similitud en los pasos a seguir en los demás sistemas operativos.

Para la empresa ServiSoft, desarrollar el sistema para otra plataforma tiene un costo injustificable ya que el sistema actual cuenta con toda la funcionalidad que se necesita.

Una solución de virtualización parece ser viable según lo visto en los capítulos anteriores. Sin embargo, no habría razón para el presente trabajo si se supiera con certeza que las herramientas de virtualización (Máquinas virtuales) pueden lograr el objetivo.

Por lo anterior, en el presente capítulo se presentan los requerimientos de la empresa para considerar la virtualización del sistema SPA como una solución viable a su problema, el cual como se menciono antes, es migrar su sistema SPA a un sistema operativo más moderno sin perdida de funcionalidad.

Para presentar los requerimientos que se deben cumplir y su validación correspondiente en la virtualización del sistema, este capítulo esta separado en subsecciones que presentan dichos requerimientos y la validación desde distintas perspectivas.

En la subsección 3.2, Plataforma original del Sistema de Pruebas de Aptitud, se empieza enumerando la plataforma original para SPA, la cual posee las características tanto de equipo como de aplicaciones necesarias para el correcto funcionamiento de SPA.

Luego, en la subsección 3.3, Requerimientos del sistema de pruebas de aptitud, se enumeran los requerimientos más importantes desde el punto de vista de la empresa, estos requerimientos están divididos en las áreas de funcionalidad, usabilidad, confiabilidad y rendimiento.

En el APÉNDICE III: Código fuente aplicación para medir el rendimiento se expone el código fuente de la aplicación utilizada para validar aspectos técnicos de la plataforma virtual, esta aplicación cuenta con operaciones básicas y sirven para simular la puesta en marcha del sistema de algunos aspectos críticos del sistema SPA.

La aplicación de pruebas no intenta duplicar la funcionalidad del sistema de Servisoft, únicamente servir de prototipo para probar aquellas partes del ambiente virtual que se consideran críticas como el rendimiento o el uso de dispositivos como la impresora.

En la subsección 3.4, Selección de la máquina virtual a utilizar, se selecciona la herramienta de virtualización más apta para ser utilizada en los dos sistemas operativos, tanto Windows Vista como en OpenSUSE 10.3.

En la subsección 3.5, Implementación, se muestra paso a paso la puesta en marcha de la máquina virtual y la virtualización del Sistema huésped.

La última subsección 3.6, Validación de la solución virtual, están los resultados de las pruebas de validación realizadas tanto por la aplicación de pruebas de rendimiento, la aplicación de pruebas de validación y el usuario final.

3.2. Plataforma original del Sistema de Pruebas de Aptitud

El Sistema de ServiSoft para pruebas de aptitud fue hecho en el año 1984, utilizando la plataforma de desarrollo FoxPRO 2.6, y fue realizado para funcionar en el sistema operativo D.O.S. Aunque el sistema fue hecho hace mas de una década aun se sigue usando tanto en la empresa ServiSoft como en otras organizaciones que lo han adquirido durante los años.

Requerimientos en equipo y aplicaciones informáticas para SPA:

Tabla IX. Requerimientos Sistema de Pruebas de Aptitud (SPA)

Equipo/Componentes físicos	
Computadora	IBM PC o compatible (i386 SX+).
Memoria primaria	4 MB de RAM
Almacenamiento	20 MB o más de disco duro
Video	Hércules/CGA/EGA/VGA
Periféricos	Scantron/ScanMark modelo 2000 o superior
Unidades utilizadas para respaldo de información	Unidad de diskette de capacidad 1.44 MB - La empresa intenta que la máquina virtual utilice en lugar de diskette una ruta de almacenamiento parametrizable.
Aplicaciones	
Sistema operativo	MS-DOS 3.1 o equivalente.
Interprete	MS FoxPRO 2.6 para D.O.S.
Manejador del Scantron	ScanBook 4.0+

3.3.Requerimientos del sistema de pruebas de aptitud

En esta sección se especifica como funciona el sistema en su plataforma original, lo que sirve de guía a seguir para verificar que el sistema en el ambiente virtual funcione de forma adecuada. Para claridad la sección esta dividida en dos partes, la primera es la especificación de requerimientos funcionales, estos requerimientos indican la forma en que funciona el sistema específicamente, se utilizaron casos de uso para describir dicha funcionalidad. Y la segunda parte es la lista de requerimientos más generales, los cuales son fundamentales para el correcto funcionamiento del sistema, estos son aspectos como usabilidad, rendimiento y confiabilidad también llamados requerimientos no funcionales.

Los requerimientos funcionales definen la funcionalidad específica del sistema, estos definen paso a paso como debe de comportarse el mismo, en este caso los requerimientos funcionales describen como se esta comportando el sistema actualmente. Los requerimientos no funcionales por su parte definen otros aspectos más generales, los cuales son indispensables para el buen funcionamiento del sistema, estos requerimientos muchas veces influyen directamente en la percepción que el usuario tiene del sistema. El presente trabajo tiene como objetivo que el sistema virtualizado cumpla los requerimientos, tal y como se describen a continuación.

3.3.1.Requerimientos funcionales

Para especificar las funciones que tiene el sistema y que deben de trasladarse al ambiente virtual se utilizaron en este trabajo especificaciones de caso de uso, un caso de uso es una forma en que el usuario usa el sistema. También se puede decir que es una razón por la cual el usuario utiliza el sistema. Por ejemplo, el usuario quiere usar el sistema para calificar las pruebas (exámenes) de aptitud, ese es un caso de uso el cual se podría llamar “Calificar pruebas”.

Cada caso de uso tiene los siguientes elementos:

- **Objetivo**, es la finalidad o producto resultante del caso de uso, en otras palabras es la razón por la cual el usuario usa esa parte del sistema.
- **Limites**, son limitaciones que se describen explícitamente en caso de no resultar obvias.
- **Nivel**, es la jerarquía del caso de uso, es decir, aquí se especifica si es una función primaria o secundaria del sistema, además se especifica en donde esta posicionada lógicamente en el sistema.

- **Actor principal**, es el tipo de usuario que usa frecuentemente esta funcionalidad del sistema, por ejemplo, el administrador, el examinador, entre otros.
- **Prioridad**, es la importancia que tiene esta funcionalidad desde el punto de vista del usuario. Esta prioridad indica el grado de fidelidad que se requiere cuando esta parte del sistema este en la máquina virtual.
- **Frecuencia**, indica la frecuencia de uso de la funcionalidad. Esto es que tanto se usa el sistema con el propósito del caso de uso. Sirve de indicador para establecer la prioridad de desarrollo, en el caso de este estudio se utilizará para asignarle prioridad al caso de prueba para ser probado.
- **Evento que lo activa**, aquí se describe la circunstancia que da inicio al caso de uso, por lo regular es la acción que el usuario debe de hacer para iniciar la tareas que desea realizar.
- **Escenario principal**, es una descripción de las acciones que el sistema realiza para esta funcionalidad, además se describen las acciones esperadas por parte del usuario para que el objetivo del caso de uso se realice exitosamente. Un caso de uso puede tener más de un escenario, esto es un camino para lograr el objetivo, sin embargo en este inciso solo se describen los pasos mas comunes.

- **Escenarios secundarios**, aquí se describen las otras alternativas que existen para cumplir el objetivo, se usan en caso que a mitad del caso de uso ocurriera algún evento fuera de lo común, aquí se describen las tareas que se deben realizar en ese caso.

Los casos de uso que se encontraron en el sistema son:

- Gestionar grupo *
- Gestionar candidatos *
- Consultar candidato por nombre por grupo *
- Consultar candidato por nombre *
- Calificar pruebas *
- Eliminar registro del candidato *
- Activar grupo de trabajo *
- Generar puestos *
- Imprimir percentiles del grupo. *

- Imprimir gráficas de grupo *
- Imprimir reportes varios *
- Crear copia de respaldo **
- Reindexar archivos *
- Editar archivo de Scantron *

* Estos casos de uso son los encontrados en el sistema actual, este trabajo intenta trasladar estos casos de uso tal y como se encuentran en el sistema actual al mismo sistema pero en un ambiente virtual, esto es, seguir las especificaciones de casos de uso tal como funcionan en su plataforma original, solo que ahora con el sistema corriendo en la máquina virtual.

** En este caso de uso la copia de respaldo es creada a un diskette, por requerimiento de la empresa se buscara una alternativa para hacer las copias de seguridad a otro tipo de medio. Esto es aceptable debido al desuso actual de los diskettes.

3.3.1.1.Gestionar grupo

3.3.1.1.1.Objetivo

Este caso de uso permite al usuario agregar y modificar grupos que son utilizados para identificar a los candidatos que se examinan juntos. En este además se definen las habilidades mínimas que requiere cada integrante del grupo, si un integrante tiene en todos los aspectos del examen mas de lo mínimo se considera un alumno aprobado por el contrario si no cumple con algún percentil mínimo se trata como un alumno no-aprobado..

3.3.1.1.2.Limites

En este caso de uso no se ingresan los datos de los alumnos, solo los datos del grupo. Además solo permite ingresar datos desde teclado. Para ingresar los alumnos se utiliza el caso de uso Gestionar Alumno.

3.3.1.1.3.Nivel

Es una tarea principal del sistema, se encuentra localizada en el menú principal.

3.3.1.1.4.Actor principal

Administrador

3.3.1.1.5.Prioridad

Alta, el caso de uso es imprescindible para el correcto funcionamiento del sistema.

3.3.1.1.6.Frecuencia

Alta-Media, antes de ingresar los alumnos del grupo se debe gestionar el grupo, por lo que este caso de uso se utiliza cada vez que se necesite examinar a un grupo de alumnos.

3.3.1.1.7.Evento que lo activa

El caso de uso es principalmente activado por:

La selección de la opción Datos del examen del menú principal

No hay otro evento que lo active.

3.3.1.1.8.Escenario principal

1. El sistema pide el código del grupo.
2. El usuario ingresa el código del grupo nuevo. (2.1)
3. El sistema muestra una forma de ingreso de los datos: grado, catedrático, lugar, fecha, persona responsable, persona que corrige el examen y el tipo de examen que se realizara.
4. El usuario ingresa los datos.

5. El sistema muestra mas opciones en la misma forma, los cuales son: la ruta de un archivo donde se leerán las calificaciones de los alumnos, una opción para configurar el Scantron para escanear por un lado de la hoja o doble lado y la lista de percentiles mínimos que debe tener un alumno para ser considerado como aprobado, estos percentiles son: Habilidad verbal, numérica, abstracta, espacial, mecánica y velocidad y exactitud.

6. El sistema pregunta si desea continuar o Terminar el ingreso.

7. El usuario selecciona terminar el ingreso. (7.1)

8. El caso de uso termina.

3.3.1.1.9.Escenarios secundarios

2.1El código de grupo existe.

2.1.1.El sistema muestra los datos del grupo.

2.1.2.Ir al paso 3 del escenario principal.

7.1El usuario selecciona continuar con el ingreso.

7.1.1.El sistema regresa al paso 1.

3.3.1.2.Gestionar candidatos

3.3.1.2.1.Objetivo

Este caso de uso permite al usuario agregar y modificar los datos de cada candidato.

3.3.1.2.2.Limites

Este caso de uso solo permite ingreso de datos por teclado.

3.3.1.2.3.Nivel

Es una tarea principal del sistema, se ingresa a ella desde el menú principal.

3.3.1.2.4.Actor principal

Administrador

3.3.1.2.5.Prioridad

Alta, esta es una funcionalidad indispensable del sistema.

3.3.1.2.6.Frecuencia

Alta-media, se utiliza cada vez que se desee ingresar un candidato a la base de datos.

3.3.1.2.7.Evento que lo activa

El caso de uso es principalmente activado por:

- La opción del menú principal Datos del candidato.

También puede ser activado por:

1. El caso de uso “Consultar candidato por nombre por grupo”
2. El caso de uso “Consultar candidato por nombre”

3.3.1.2.8.Escenario principal

1. El sistema pide el código del alumno.
2. El usuario ingreso un código nuevo de alumno. (2.1)
3. El sistema muestra una forma donde se ingresan los datos del alumno, los datos son: Nombre, genero, código de grupo.
4. El sistema muestra la información asociada al grupo, la cual es: grado, catedrático, lugar, fecha, persona responsable, persona que corrige el examen y el tipo de examen.
5. El sistema muestra una forma para el ingreso de las notas del alumno, las notas pueden ser de: Habilidad verbal, numérica, abstracta, espacial, mecánica y velocidad y exactitud.
6. El usuario ingresa las notas del alumno.
7. El sistema muestra las opciones de continuar, terminar, imprimir notas y graficar. (7.1) (7.2) (7.3)
8. El usuario selecciona la opción de terminar.

9. El caso de uso termina.

3.3.1.2.9.Escenarios secundarios

2.1El código del candidato ya existe.

2.1.1.El sistema muestra los datos del alumno.

2.1.2.El sistema continua en el paso 3.

7.1El usuario selecciona la opción de continuar.

7.1.1.El sistema continua en el paso 1.

7.2El usuario selecciona la opción de imprimir notas.

7.2.1.El sistema imprime las notas del grupo.

7.2.2.El sistema continua en el paso 7.

7.3El usuario selecciona la opción graficar.

7.3.1.El sistema muestra una gráfica de barras de las notas del candidato.

7.3.2.El sistema continua en el paso 7.

3.3.1.3.Consultar candidato por nombre por grupo

3.3.1.3.1.Objetivo

Este caso de uso se utiliza para obtener los datos del candidato que se encuentra específicamente en un grupo y no se cuenta con el código de registro de dicho candidato.

3.3.1.3.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.3.3.Actor principal

Examinador

3.3.1.3.4.Prioridad

Media-Baja, no es una prioridad fundamental del sistema ya que existen otras formas de acceder a la misma información dentro del sistema, como el caso de uso “Generar reportes varios” o también “Consultar candidato por nombre”.

3.3.1.3.5.Frecuencia

Baja, se utiliza en el caso que un candidato tenga problemas con su nota y se quieran saber detalles acerca del mismo. Los problemas de ese tipo son poco frecuentes ya que los candidatos reciben una gráfica de sus percentiles usando el caso “Imprimir gráficas de grupo” en la cual se encuentran los datos que se registran en el sistema.

3.3.1.3.6.Evento que lo activa

Es activado seleccionando la opción Consulta general->Por grupo del menú principal.

3.3.1.3.7.Escenario principal

1. El sistema pregunta por el código del grupo del candidato.
2. El usuario ingresa el código del grupo.
3. El sistema pregunta por parte del nombre del candidato.
4. El usuario ingresa parte del nombre.
5. El sistema muestra la lista de candidatos que sean del grupo y que parte de su nombre sea el ingresado.
6. El usuario selecciona un candidato de la lista.
7. El sistema muestra la información del candidato utilizando el caso de uso "Gestionar candidatos" paso 1.
8. El sistema muestra las opciones de continuar consultando o terminar. (8.1)
9. El usuario selecciona la opción de terminar.
- 10.El caso de uso termina.

3.3.1.3.8.Escenarios secundarios

- 8.1El usuario selecciona la opción de continuar consultando.
 - 8.1.1.El caso de uso continua en el paso 1.

3.3.1.4.Consultar candidato por nombre

3.3.1.4.1.Objetivo

Este caso de uso se utiliza para obtener los datos del candidato utilizando el nombre del mismo, es necesario cuando no se cuenta con el código de registro del candidato.

3.3.1.4.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.4.3.Actor principal

Examinador

3.3.1.4.4.Prioridad

Media-Baja, no es una prioridad fundamental del sistema ya que existen otras formas de acceder a la misma información dentro del sistema, como el caso de uso “Generar reportes varios” o también “Consultar candidato por nombre por grupo”.

3.3.1.4.5.Frecuencia

Baja, se utiliza en el caso que un candidato tenga problemas con su nota y se quieran saber detalles acerca del mismo, además no se tenga el grupo en el cual se examinó o se haya examinado en varios grupos. Los problemas de ese tipo son poco frecuentes ya que los candidatos reciben una gráfica de sus percentiles usando el caso “Imprimir gráficas de grupo” en la cual se encuentran los datos que se registran en el sistema.

3.3.1.4.6.Evento que lo activa

Es activado seleccionando la opción Consulta individual del menú principal.

3.3.1.4.7.Escenario principal

1. El sistema pregunta por parte del nombre del candidato.
2. El usuario ingresa parte del nombre.
3. El sistema muestra la lista de candidatos que sean del grupo y que parte de su nombre sea el ingresado.
4. El usuario selecciona un candidato de la lista.
5. El sistema muestra la información del candidato utilizando el caso de uso “Gestionar candidatos” paso 1.
6. El sistema muestra las opciones de continuar consultando o terminar. (6.1)

7. El usuario selecciona la opción de terminar.

8. El caso de uso termina.

3.3.1.4.8. Escenarios secundarios

6.1 El usuario selecciona la opción de continuar consultando.

6.1.1. El caso de uso continúa en el paso 1.

3.3.1.5. Calificar pruebas

3.3.1.5.1. Objetivo

El usuario utiliza esta funcionalidad para calificar las pruebas de un grupo, calificar significa corregir la prueba según la clave y asignarle un percentil por aspecto a cada candidato.

3.3.1.5.2. Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.5.3. Actor principal

Examinador

3.3.1.5.4.Prioridad

Alta, es una prioridad fundamental del sistema, en este caso de uso se encuentra la calificación automatizada de pruebas utilizando el Scantron.

3.3.1.5.5.Frecuencia

Alta-Media, este caso de uso se utiliza para cada grupo o cada vez que se desea recalificar el grupo. La recalificación puede ocurrir si se califico parcialmente al grupo.

3.3.1.5.6.Evento que lo activa

Es activado seleccionando la opción Cargar datos desde Scantron del menú principal.

3.3.1.5.7.Escenario principal

1. El sistema pregunta si se desean alimentar hojas al Scantron, .cargar un documento ya existente o validar las pruebas. (1.1) (1.2)
2. El usuario selecciona la opción de Alimentar hojas al Scantron.
3. El sistema pregunta el grupo que se va a calificar.
4. El usuario ingresa el grupo correspondiente.

5. El sistema le indica al dispositivo Scantron que escanee las hojas puestas por el usuario.

6. El sistema le indica al usuario que el Scantron esta listo para escanear.

7. El usuario coloca las hojas en el Scantron y le indica al Scantron que las hojas están puestas presionando el botón de escaneado en el aparato.

8. El Scantron inicia la captura de las respuestas de los candidatos, pasando hoja por hoja por el reconocedor de marcas.

9. El usuario le indica al Scantron cuando ya no tiene más pruebas para calificar, presionando el botón correspondiente en el aparato. (9.1)

10.El Scantron le indica al sistema que terminó de capturar los exámenes.

11.El sistema califica las pruebas y le asigna a los candidatos los percentiles correspondientes.

12.El caso de uso termina.

3.3.1.5.8.Escenarios secundarios

1.1El usuario selecciona la opción de cargar un documento ya existente.

1.1.1.El sistema pregunta la ubicación del documento que contiene las respuestas de las pruebas.

1.1.2.El usuario ingresa la ruta.

1.1.3.El caso de uso continua en el paso 11.

1.2El usuario selecciona la opción de validar las pruebas.

1.2.1.El sistema recalifica las últimas pruebas.

1.2.2.El caso de uso termina.

3.3.1.6.Eliminar registro del candidato

3.3.1.6.1.Objetivo

El usuario utiliza esta funcionalidad para eliminar del sistema un candidato ingresado.

3.3.1.6.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.6.3.Actor principal

Administrador

3.3.1.6.4.Prioridad

Baja, no es indispensable eliminar candidatos del sistema.

3.3.1.6.5.Frecuencia

Baja, el sistema esta pensado en mantener un historial de las personas que se han hecho la prueba, por lo que la frecuencia del uso de esta funcionalidad es muy baja. Sin embargo, hay una incongruencia porque el sistema tiene la opción en el nivel principal de la aplicación.

3.3.1.6.6.Evento que lo activa

Es activado seleccionando la opción Eliminar registro del menú principal.

3.3.1.6.7.Escenario principal

1. El sistema pregunta el código del candidato que se desea eliminar.
2. El usuario ingresa el código.
3. El sistema pide confirmación de la eliminación del candidato.
4. El usuario confirma la eliminación. (4.1)
5. El sistema elimina el registro.
6. El caso de uso termina.

3.3.1.6.8.Escenarios secundarios

4.1 El usuario cancela la eliminación.

4.1.1.El sistema no elimina el candidato.

4.1.2.El caso de uso termina.

3.3.1.7.Activar grupo de trabajo

3.3.1.7.1.Objetivo

El usuario utiliza esta funcionalidad para activar el grupo de trabajo que aparecerá automáticamente cuando el sistema pregunte por este, también es usado para que el sistema coloque el prefijo del código del grupo de trabajo de forma que el usuario no tenga que estarlo escribiendo cada vez que el sistema pregunte por el código del grupo.

3.3.1.7.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.7.3.Actor principal

Administrador

3.3.1.7.4.Prioridad

Baja, no es indispensable para el correcto funcionamiento del sistema.

3.3.1.7.5.Frecuencia

Media-baja, se utiliza cuando se quiere establecer un grupo muy utilizado en las demás opciones del sistema.

3.3.1.7.6.Evento que lo activa

Es activado seleccionando la opción Utilidades->Activar grupo de trabajo del menú principal.

3.3.1.7.7.Escenario principal

1. El sistema pregunta el código del grupo que se desea activar como principal.
2. El usuario ingresa el código.
3. El sistema activa el grupo como principal. (3.1)
4. El caso de uso termina.

3.3.1.7.8.Escenarios secundarios

3.1 Si el grupo no existe.

3.1.1.El sistema guarda la cadena para se utilizada como prefijo en los dialogos que pidan el codigo de grupo.

3.1.2.El caso de uso termina.

3.3.1.8. Generar puestos

3.3.1.8.1. Objetivo

El usuario utiliza esta opción para comparar el rendimiento de los candidatos con otros del mismo grupo, para esto el sistema determina el puesto de un candidato dentro del grupo. El candidato en el puesto 1 de un grupo es aquel que tiene el mayor promedio de los percentiles en ese grupo.

3.3.1.8.2. Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.8.3. Actor principal

Examinador

3.3.1.8.4. Prioridad

Alta-media, parte fundamental del sistema es determinar los candidatos más aptos para ingresar, esta funcionalidad es indispensable para eso.

3.3.1.8.5.Frecuencia

Alta, es usada frecuentemente y se utiliza cada vez que ocurre algún cambio en los percentiles de algún candidato de un grupo.

3.3.1.8.6.Evento que lo activa

Es activado seleccionando la opción Generar puesto del menú principal.

3.3.1.8.7.Escenario principal

1. El sistema pregunta el código del grupo.
2. El usuario ingresa el código.
3. El sistema le asigna a cada candidato del grupo su número de posición (puesto). (3.1)
4. El caso de uso termina.

3.3.1.8.8.Escenarios secundarios

3.1 Si el grupo no existe.

3.1.1. El sistema informa que el grupo no existe.

3.1.2. El caso de uso termina.

3.3.1.9.Imprimir percentiles del grupo

3.3.1.9.1.Objetivo

El usuario utiliza esta opción para generar un listado impreso de los candidatos de un grupo y sus percentiles, los percentiles son las notas de 0 a 100 de cada habilidad examinada en la prueba.

3.3.1.9.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.9.3.Actor principal

Examinador

3.3.1.9.4.Prioridad

Alta, es parte fundamental del sistema, se utiliza para generar un listado que sirve como reporte del rendimiento de los candidatos.

3.3.1.9.5.Frecuencia

Alta-media, es usada frecuentemente, se utiliza cuando se necesita la información del grupo en forma impresa.

3.3.1.9.6.Evento que lo activa

Es activado seleccionando la opción Reportes->Reporte percentil grupo del menú principal.

3.3.1.9.7.Escenario principal

1. El sistema presenta las indicaciones para la impresión, le especifica al usuario que debe colocar las hojas y preparar la impresora, además el sistema le presenta las opciones de confirmar la puesta de hojas o de cancelar la generación del reporte.

2. El usuario confirma la preparación de la impresora. (2.1)

3. El sistema presenta una forma para ingresar el código de grupo y el orden de los candidatos, esto es, orden alfabético o por percentil.

4. El usuario ingresa el código del grupo y el orden de los candidatos.

5. El sistema genera el reporte y envía la información a la impresora. (5.1)

6. El caso de uso termina.

3.3.1.9.8.Escenarios secundarios

2.1El usuario cancela la impresión.

2.1.1.El caso de uso termina.

5.1 El sistema no puede enviar la información a la impresora por algún error en la comunicación.

5.1.1. El sistema muestra al usuario un mensaje de error informándole que no pudo completar la impresión.

5.1.2. El caso de uso termina.

3.3.1.10. Imprimir gráficas de grupo

3.3.1.10.1. Objetivo

El usuario utiliza esta opción para obtener una hoja impresa con información del candidato, en la cual se indica con una gráfica de barras los percentiles obtenidos por el candidato en cada área examinada, además cuenta con la información general del candidato (código, nombre, sexo) e información del grupo donde se examinó (código del grupo, examinador, fecha).

3.3.1.10.2. Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.10.3. Actor principal

Examinador

3.3.1.10.4.Prioridad

Alta, es parte fundamental del sistema, se utiliza para generar la gráfica que comúnmente se le entrega al candidato para que visualice sus percentiles y los compare con la media o con lo requerido, según se ingreso como requerimientos del grupo.

3.3.1.10.5.Frecuencia

Alta-media, es usada frecuentemente, se utiliza cuando se necesitan las gráficas de los percentiles de los candidatos de un grupo en forma impresa.

3.3.1.10.6.Evento que lo activa

Es activado seleccionando la opción Reportes->Imprime gráficas grupo del menú principal.

3.3.1.10.7.Escenario principal

1. El sistema presenta las indicaciones para la impresión, le especifica al usuario que debe colocar las hojas y preparar la impresora, además el sistema le presenta las opciones de confirmar la puesta de hojas o de cancelar la generación del reporte.
2. El usuario confirma la preparación de la impresora. (2.1)
3. El sistema presenta una forma para ingresar el código de grupo y el ultimo candidato del grupo que se le imprimió su

gráfica, esto con el fin de empezar la impresión con el candidato siguiente.

4. El usuario ingresa el código del grupo y deja en blanco el campo del ultimo candidato del grupo. (4.1)

5. El sistema genera el reporte y envía la información a la impresora. (5.1)

6. El caso de uso termina.

3.3.1.10.8.Escenarios secundarios

2.1El usuario cancela la impresión.

2.1.1.El caso de uso termina.

4.1El usuario ingresa el código del grupo y el código del candidato cuya gráfica fue la última que se imprimió la última vez que se uso este caso de uso.

4.1.1.El sistema genera el reporte y envía la información a la impresora empezando por el siguiente candidato en la lista del grupo que le sigue al código que ingreso el usuario.

4.1.2.El caso de uso termina.

5.1El sistema no puede enviar la información a la impresora por algún error en la comunicación.

5.1.1.El sistema muestra al usuario un mensaje de error informandole que no pudo completar la impresión.

5.1.2.El caso de uso termina.

3.3.1.11. Generar reportes varios

3.3.1.11.1. Objetivo

El usuario utiliza esta funcionalidad del sistema para imprimir varios tipos de reportes, principalmente el reporte de candidatos que pasaron la prueba.

3.3.1.11.2. Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.11.3. Actor principal

Examinador

3.3.1.11.4. Prioridad

Alta, es parte fundamental del sistema, se utiliza para imprimir los listados de candidatos aprobados y reprobados.

3.3.1.11.5. Frecuencia

Alta-media, es usada frecuentemente, se utiliza cuando se necesitan en forma impresa los datos de los candidatos que fueron aprobados..

3.3.1.11.6.Evento que lo activa

Es activado seleccionando la opción Reportes->Generar notas aprobadas y Reportes->Reporte notas reprobadas del menú principal.

3.3.1.11.7.Escenario principal

1. El sistema presenta las indicaciones para la impresión, le especifica al usuario que debe colocar las hojas y preparar la impresora, además el sistema le presenta las opciones de confirmar la puesta de hojas o de cancelar la generación del reporte.

2. El usuario confirma la preparación de la impresora. (2.1)

3. El sistema presenta una forma para ingresar el código de grupo y el orden de la impresión de los candidatos, esto es, alfabético o por nota.

4. El usuario ingresa el código del grupo y el orden de impresión.

5. El sistema genera el reporte y envía la información a la impresora. (5.1)

6. El caso de uso termina.

3.3.1.11.8.Escenarios secundarios

2.1El usuario cancela la impresión.

2.1.1.El caso de uso termina.

5.1 El sistema no puede enviar la información a la impresora por algún error en la comunicación.

5.1.1. El sistema muestra al usuario un mensaje de error informándole que no pudo completar la impresión.

5.1.2. El caso de uso termina.

3.3.1.12. Crear copia de respaldo

3.3.1.12.1. Objetivo

El usuario utiliza esta funcionalidad del sistema para almacenar una copia de los datos en una ubicación distinta a la del sistema para mitigar el riesgo de perderla.

3.3.1.12.2. Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.12.3. Actor principal

Administrador

3.3.1.12.4. Prioridad

Alta, es una funcionalidad fundamental del sistema en el sentido que se debe poder sacar copias que respalden la información periódicamente.

3.3.1.12.5.Frecuencia

Variable, no se puede definir una frecuencia para este caso de uso ya que depende del uso que se le de al sistema y de la política que tenga la empresa con respecto a las copias de información. En algunas empresas puede ser una frecuencia alta, mientras en otras baja.

3.3.1.12.6.Evento que lo activa

Es activado seleccionando la opción Utilidades->Copia de seguridad a Diskette del menú principal .

3.3.1.12.7.Escenario principal

1. El sistema muestra una forma donde le indica al usuario que se hara una copia de seguridad a un Diskette y pide una confirmación.
2. El usuario confirma la acción. (2.1)
3. El sistema hace una copia de las bases de datos al Diskette.
4. El caso de uso termina.

3.3.1.12.8.Escenarios secundarios

- 2.1El usuario cancela la operación.
 - 2.1.1.El caso de uso termina.

3.3.1.13.Reindexar archivos

3.3.1.13.1.Objetivo

Esta funcionalidad tiene el objetivo de arreglar algunos problemas con la base de datos después de mucho uso, así define la empresa este caso de uso. Se encontró que luego de largos periodos de uso algunas funcionalidades del sistema dejaban de funcionar debido a que los índices de los datos dejaban de ser exactos, por lo que los datos debían ser reindexados, lo que significa volver a generar dichos índices. Esto según la empresa es debido a la forma en que la plataforma FoxPRO 2.6 maneja los índices.

3.3.1.13.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.13.3.Actor principal

Administrador

3.3.1.13.4.Prioridad

Alta, es una funcionalidad fundamental del sistema porque permite continuar el uso del mismo si los índices no permiten el uso de la información correctamente.

3.3.1.13.5.Frecuencia

Muy baja, no es una operación que se use frecuentemente, es bastante baja en la vida de un sistema de esta naturaleza, pero se encuentra en el sistema por la eventualidad que se necesite.

3.3.1.13.6.Evento que lo activa

Es activado seleccionando la opción Utilidades->Reindexar archivos del menú principal .

3.3.1.13.7.Escenario principal

1. El sistema muestra una forma donde le indica al usuario que espere la finalización de la operación mientras reindexa los datos.
2. El caso de uso termina.

3.3.1.14.Editar archivo de Scantron

3.3.1.14.1.Objetivo

Este caso de uso permite arreglar los datos capturados por el Scantron antes de ser calificados, de esta manera se pueden arreglar errores en la captura. Por lo regular solo se utiliza en el caso que las hojas utilizadas en la captura tengas daños que impidan la correcta captura de las mismas.

3.3.1.14.2.Nivel

Es una funcionalidad principal del sistema, se encuentra en el menú principal.

3.3.1.14.3.Actor principal

Examinador

3.3.1.14.4.Prioridad

Alta, es una funcionalidad fundamental del sistema porque permite la revisión de los datos antes de ser procesados, sin esta funcionalidad la corrección de una nota supondría la calificación manual de la prueba.

3.3.1.14.5.Frecuencia

Media, se puede utilizar cada vez que se capture un conjunto de hojas de respuestas, sin embargo no es necesario si las hojas se encuentran en buen estado y los datos en dichas hojas están llenados correctamente.

3.3.1.14.6.Evento que lo activa

Es activado seleccionando la opción Utilidades->Revisar archivo del menú principal .

3.3.1.14.7.Escenario principal

1. El sistema muestra una forma para ingresar el código de grupo al cual pertenecen las notas capturadas en el archivo de la ultima captura.
2. El usuario ingresa el código del grupo.
3. El sistema muestra el archivo para ser modificado en una pantalla de edición.
4. El usuario hace los cambios pertinentes.
5. El caso de uso termina.

3.3.1.15.Seguridad

El sistema provee mecanismos de seguridad, este mecanismo es basado en usuarios y contraseñas. Al iniciar la ejecución del sistema, este pide una clave de identificación la cual es utilizada para autenticar el usuario que esta usando el sistema en ese momento. Eso se utiliza para permitir el acceso al sistema a usuarios autorizados únicamente y denegar el acceso cuando se ingrese una contraseña invalidad.

3.3.2.Requerimientos no funcionales

En esta sección se definen los requerimientos generales del sistema, estos son poco específicos comparándolos con los requerimientos funcionales. Sin embargo, son igual de importantes porque su cumplimiento afecta directamente la calidad del sistema.

3.3.2.1.Dispositivos externos requeridos

El sistema se comunica principalmente con dos tipos de dispositivos externos, uno se utiliza para ingresar de forma automática las respuestas de los solicitantes al sistema, este es el dispositivo OMR explicado a continuación. El segundo tipo es el utilizado para imprimir los reportes, este dispositivo es la impresora.

3.3.2.1.1.Dispositivo reconocedor óptico de marcas (OMR)

El sistema se comunica con un dispositivo reconocedor óptico de marcas, también conocido como dispositivo OMR (*Optical Mark Recognition*), este dispositivo se utiliza para leer de forma automática las pruebas de aptitud que luego son calificadas por el sistema. Cuando se pasa la prueba a los solicitantes se les provee una hoja de preguntas y una de respuestas, la hoja de respuestas es una hoja especial diseñada para ser leída por los dispositivos OMR.

La comunicación con este tipo de dispositivos es una parte importante del sistema, es utilizado con dispositivos OMR de la empresa Scantron, y esta diseñado especialmente para la línea Scanmark 2000, 2500 y 2550. El sistema y los dispositivos OMR se comunican utilizando un puerto serial RS-232 el cual es cada vez menos usado dado que esta siendo reemplazado por el mas moderno puerto serial USB. La virtualización de este sistema contempla ese hecho y se indica la forma de solucionar el problema de incompatibilidad entre ambos puertos de comunicación, en este caso USB y RS-232, ver la sección de implementación de la máquina virtual para más información.

A continuación se muestra un tabla resumida de los tipos de dispositivos externos que el sistema utiliza:

Tabla X. Dispositivos externos utilizados por el sistema de pruebas

Tipo de dispositivo	Especificaciones	
Dispositivo OMR	Fabricante	Scantron
	Linea	Scanmark 2000, 2500 y 2550.
	Interfaz	Puerto serial, RS-232 (DB9 y DB25)
Impresora	Interfaz	Puerto paralelo (LPT).

Cuando el sistema se encuentre en la máquina virtual se deben poder utilizar estos dispositivos, en el caso de las impresoras el sistema virtual debe ser capaz de utilizar impresoras más modernas, esto implica el uso de comunicación por el puerto USB entre el sistema y la impresora.

3.3.2.2. Usabilidad

El sistema virtualizado debe proveer el mismo nivel de usabilidad que el sistema en su forma actual. Esto significa que no se deben incrementar los pasos o el tiempo para poder usar el mismo.

Los puntos que se deben conservar en la usabilidad del sistema son:

- Seguir con el uso del teclado para la operación del sistema.
- Mantener o mejorar los tiempos de inicio y de salida del sistema.
- Mantener la posición y diseño general de las formas y diálogos.
- Mantener el número de pasos para realizar una operación.

3.3.2.3.Confiabilidad

Actualmente, el sistema utiliza discos flexibles (también llamados Diskettes) para realizar copias de respaldo. Sin embargo, este tipo de medio ha dejado de usarse por lo que las copias de seguridad deberán ser realizadas a otro medio de almacenamiento.

3.3.2.4.Rendimiento

El rendimiento del sistema virtualizado debe ser comparable o mayor al del sistema en su forma actual.

3.4. Selección de la máquina virtual a utilizar

Uno de los puntos más importantes de este estudio es identificar las herramientas de virtualización a utilizar. Para eso, el primer paso a tomar es identificar aquellas máquinas virtuales que cumplan con los siguientes tres aspectos básicos:

1. Funcionamiento en las plataformas deseadas.
2. Licencia sin costo.
3. Virtualización de los requerimientos del sistema operativo huésped, esto es, D.O.S.

Luego de seleccionar las máquinas virtuales que cumplan con esos aspectos, el segundo paso es seleccionar aquella que ofrezca más beneficios en los aspectos económico, técnico y operativo.

Por lo anterior, esta sección se divide en dos partes, en la primera se especifica la plataforma destino donde se seleccionaran las tres máquinas virtuales con mejor documentación y soporte a dicha plataforma. La segunda parte muestra un estudio y comparativa entre las tres máquinas seleccionadas, de este estudio se seleccionará la más indicada para la virtualización del sistema de pruebas de aptitud.

3.4.1.Descripción de la plataforma destino

El punto principal de este estudio es permitir al SPA funcionar en una plataforma más moderna. El uso de tecnologías de punta permitirán, según la hipótesis planteada, aprovechar el rendimiento, el soporte y garantía de equipo más moderno. Entonces, la plataforma destino es el conjunto de componentes donde funcionará el Sistema de Pruebas de Aptitud (SPA) como se muestra en la Tabla XI. Plataforma destino.

Sin embargo, estudiar todas y cada una de las posibles alternativas de virtualización estaría fuera del contexto de este trabajo, dado que el objetivo del mismo es presentar una solución de bajo costo, esto limita la selección de las máquinas virtuales.

También la solución debe permitir la virtualización de una plataforma específica, por eso antes de seleccionar una máquina virtual hay que reducir la lista de posibles máquinas virtuales candidatas a un número manejable. Por ello se seleccionaran las tres máquinas virtuales que en su documentación muestren más soporte para la plataforma destino.

En la siguiente tabla se muestra la plataforma original y destino del Sistema de Pruebas de Aptitud:

Tabla XI. Plataforma destino

Equipo (Componentes físicos)	Plataforma original (Sistema huésped)	Plataforma destino (Sistema anfitrión)
Procesador	IBM PC o compatible (i386 SX+).	Procesador Intel o AMD de 1 núcleo o más.
Memoria primaria	4 MB de RAM	512 MB de RAM o más
Almacenamiento	20 MB o más de disco duro	40 GB o más
Video	Hercules/CGA/EGA/VGA	Cualquiera
Periféricos	Scantron/ScanMark línea 2000 puerto serial (RS-232) Impresora con interfaz puerto paralelo (LPT)	Scantron/ScanMark línea 2000 conectado a un puerto USB. Impresora con interfaz USB.
Unidades utilizadas para respaldo de información	Unidad de diskette de capacidad 1.44 MB - La empresa intenta que la máquina virtual utilice en lugar de diskette una ruta de almacenamiento parametrizable.	Ruta parametrizable.
Sistema operativo	MS-DOS 3.1 o equivalente.	Windows XP Windows Vista Linux 2.6 o superior
Interprete	MS FoxPRO 2.6 para D.O.S.	MS FoxPRO 2.6 para D.O.S. (Virtualizado)
Manejador del Scantron	ScanBook 4.0+	ScanBook 4.0+ (Virtualizado)

En la tabla anterior, la plataforma original se refiere al sistema operativo y los requerimientos de equipo donde funciona actualmente el Sistema de Pruebas de Aptitud, a la par se muestra la plataforma destino, la cual es el nuevo ambiente donde funcionará el sistema luego de la virtualización.

Cabe observar que el Interpretador y el Manejador del Scantrol permanecen siendo los mismos, esto es porque es parte del sistema que se virtualizará y ellos permitirán el correcto funcionamiento dentro de la máquina virtual.

En la subsección siguiente se enumeran las tres máquinas virtuales que en su documentación muestran soporte para la plataforma destino.

3.4.1.1.Máquinas virtuales candidatas

De la siguiente lista se tomarán las tres primeras máquinas virtuales que cumplan con los requerimientos de la plataforma:

Tabla XII. Soporte de la plataforma destino por máquina virtual

Máquina Virtual	WinXP / Vista (Anfitrión)	Linux (Anfitrión)	Gratis	Virtualiza D.O.S. (Huésped)
VMWare Server 2.0	SI	SI	SI	SI
Microsoft Virtual PC 2007	SI	NO	SI	SI
Innotek VirtualBox 1.5.4 *	SI	SI	NO	SI
XenServer (Hypervisor)	NO	NO	SI	NO
QEMU 0.9.0	SI	SI	SI	SI
Bochs 2.3.6	SI	SI	SI	SI
DOSBox 0.72 **	SI	SI	SI	SI
DOSEmu 1.4.0	NO	SI	SI	SI

* Innotek tiene una versión gratis de su máquina virtual, sin embargo está limitada en funcionalidad por lo que no es adecuada para este estudio.

** DOSBox no es una alternativa viable debido a que es un emulador orientado a juegos de D.O.S. por esa razón provee poco soporte para funcionalidad importante para este estudio como la impresión de documentos.

Las siguientes tres máquinas virtuales se seleccionaron como candidatas para la virtualización del sistema:

- VMWare Server 2.0
- QEMU
- Bochs

3.4.2. Análisis de factibilidad

Cada una de las tres máquinas virtuales seleccionadas en la sección anterior será analizada desde cuatro perspectivas diferentes, esto para determinar cual de las tres es la indicada para su uso. Las cuatro perspectivas son las siguientes:

- **Análisis económico**, es un análisis de costos de la implementación, en esta fase del estudio el análisis económico se reduce al estudio de los costos de adquisición e instalación.
- **Análisis legal**, es un estudio de licencias y restricciones legales.
- **Análisis técnico**, en este análisis se revisa que la máquina virtual soporte las operaciones necesarias para el correcto funcionamiento del sistema, i.e. soporte de los dispositivos necesarios como la impresora y el Scantron.
- **Análisis operativo**, este análisis se centra en la facilidad de uso de la máquina virtual. La intención del mismo es medir el trabajo adicional que supondrá para el usuario del sistema.

A continuación se muestran los análisis de cada una de las tres máquinas virtuales candidatas.

3.4.2.1.VMWare Server

3.4.2.1.1.Análisis económico de VMWare Server 2.0

A continuación se muestran los costos más importantes de adquisición e instalación de la máquina virtual, sin embargo, ya que las tres máquinas virtuales son gratuitas tienen costos de adquisición e instalación similares.

Tabla XIII. Análisis de factibilidad económico para VMWare Server

Máquina virtual	VMWare Server 2.0
Adquisición	Se puede obtener gratuitamente desde la página web http://www.vmware.com/products/server/
Instalación	Utilizados por ancho de banda y tiempo. 350 MB de ancho de banda + 30 minutos de instalación En los escenarios comunes el ancho de banda es depreciable, sin embargo, se toma en cuenta cuando en la empresa no se cuenta con una conexión a Internet y se debe pagar por el ancho de banda.
Total	30-45 minutos de instalación (No incluye la virtualización del Sistema de Pruebas de Aptitud), 350 MB en ancho de banda y la media utilizada para almacenar el instalador.

Conclusión del análisis: El costo en total son 30-45 minutos de instalación y en los lugares donde no se cuente con una conexión permanente a Internet deben pagar 350 MB en ancho de banda y la media para almacenar el instalador. Lo anterior no representan un impedimento para la utilización de esta máquina virtual.

3.4.2.1.2. Análisis legal de VMWare Server 2.0

El documento llamado licencia en un producto informático es un contrato legal que permite al usuario final utilizar la máquina virtual siempre que se sigan los acuerdos contenidos en dicha licencia. VMWare y muchos otros fabricantes de aplicaciones informáticas no venden sus productos, en cambio, los licencian otorgando un permiso “temporal” a los usuarios, este permiso es la licencia. La siguiente tabla contiene información acerca de la licencia de VMWare Server:

Tabla XIV. Análisis de factibilidad legal para VMWare Server

Máquina Virtual	VMWare Server 2.0
Tipo de licencia	EULA, la licencia utilizada es “VMWARE SERVER MASTER END USER LICENSE AGREEMENT”
Limitaciones	La licencia prohíbe la copia directa de la máquina virtual cuando no sea para copias de seguridad, sin embargo, se pueden obtener tantas copias como sean necesarias desde la página de VMWare (http://www.vmware.com/products/server/). Lo que prohíbe la licencia explícitamente es redistribuir la máquina virtual. Solo se debe instalar una instancia de la máquina virtual por equipo de cómputo.

Máquina Virtual	VMWare Server 2.0
Restricciones	La venta, alquiler, licenciamiento o sub-licenciamiento de la máquina virtual a terceros. Modificar o crear trabajos derivadas de la máquina virtual. Esto se refiere a modificar la máquina virtual en formas no permitidas por la ley, en especial el desensamblaje y modificación del código de ejecución. Esto no aplica a este trabajo el cual no esta derivando un trabajo directo de la máquina virtual.
Soporte	Sin soporte más que la documentación incluida.
Garantía	Sin garantías.

La licencia no impone ninguna restricción que prohíba la utilización de VMWare Server en este trabajo, sin embargo, por cada instancia de VMWare Server que se utilice se debe obtener una licencia de VMWare, hay que hacer esto aunque la máquina virtual sea gratuita. En otras palabras, la licencia prohíbe la redistribución de la máquina virtual, por lo que no se podrá incluir directamente la máquina virtual en la media que acompaña este trabajo, en cambio se proveen los enlaces donde conseguirla (<http://www.vmware.com>).

La inexistencia de soporte y garantía no es una sorpresa ya que es una máquina virtual gratuita. Sin embargo, el soporte en forma de documentación de usuario y técnica es abundante.

Conclusión: La licencia no impone ninguna restricción que impida el uso de la máquina virtual en la virtualización del Sistema de Pruebas de Aptitud.

3.4.2.1.3. Análisis técnico de VMWare Server 2.0

Este análisis está orientado a descubrir si la máquina virtual soporta los dispositivos necesarios para el Sistema de Pruebas de Aptitud.

Como una máquina virtual es una capa de abstracción entre el sistema anfitrión y el sistema huésped, los dispositivos que se presentan al sistema huésped son virtuales, es decir, son dispositivos manejados por la máquina virtual, los cuales son presentados de una forma al sistema huésped cuando en realidad son de otra.

Para ilustrar lo anterior, se puede ver el caso de la impresora, en el sistema operativo D.O.S. las impresoras utilizan un puerto llamado LPT o paralelo. En el caso que el sistema operativo huésped fuera D.O.S., el puerto LPT será virtual y presentado de esa forma por la máquina virtual, sin embargo, fuera de la máquina virtual la impresora utilizará comúnmente un puerto USB. Es necesario entonces que la máquina virtual pueda presentar el dispositivo de impresión como LPT al sistema D.O.S. dentro de la máquina virtual y utilizar el puerto USB para sacar las impresiones por la impresora física.

Por lo anterior, en la tabla siguiente se presentan los dispositivos requeridos por el Sistema de Pruebas de Aptitud y la configuración soportada por VMWare para dichos dispositivos dentro y fuera de la máquina virtual:

Tabla XV. Configuración de dispositivos dentro y fuera de la máquina virtual para VMWare Server 2.0

Dispositivos requeridos por el sistema		Dispositivos soportados por VMWare Server 2.0	
Físicos (Dispositivos disponibles)	Virtuales	Virtuales	Físicos
Impresora conectada a USB	Impresora conectada a LPT	Puerto LPT (Máximo 3)	Puerto LPT (Bidireccional) Archivos
Scantron conectado a USB	Scantron conectado a RS-232	Puerto RS-232 (Serial) (Máximo 4)	Puerto RS-232 Tuberías Archivos
Unidad de respaldo de ruta parametrizable (Archivo)	Disco flexible (1.44")	Disco flexible (1.44")	Disco flexible (1.44") Imagen de disco (Ruta parametrizable)
Unidad de almacenamiento (Disco SATA)	Disco ATA	Disco ATA	Imagen de disco Disco SATA/ATA
Directorio compartido	Directorio o unidad de almacenamiento del sistema	Unidad compartida utilizando herramientas de red	Servidor de archivos
Adaptador gráfico (Cualquier adaptador)	Hercules/CGA/EGA/VGA	Hercules/CGA/EGA/VGA	Cualquier adaptador de video

Las primeras dos columnas muestran los dispositivos requeridos por el Sistema de Pruebas de Aptitud, mientras las dos últimas columnas muestran los dispositivos soportados por VMWare.

Como se puede observar en la tabla, la impresora y el Scantron no cuentan con un soporte físico por parte de la máquina virtual, ya que la impresora requiere un puerto USB y VMWare provee salida a un puerto LPT o archivos, sin embargo, esto no es problema porque se puede redireccionar la impresión desde un archivo a la impresora conectada al puerto USB desde los sistemas operativos Windows y Linux. El redireccionamiento de puertos se detalla en la sección de 3.5.3.5, Configuración de periféricos, de la configuración de la máquina virtual.

Conclusión: La máquina virtual no tiene impedimentos técnicos que impidan su uso para la virtualización del sistema. Sin embargo, la impresora y el Scantron no son soportados directamente por lo que hay que utilizar redireccionamiento de puertos tanto en Windows como en Linux. Además, para el directorio compartido se debe utilizar un servidor de archivos y herramientas de red para D.O.S. Lo cual incrementa el tiempo requerido de configuración.

3.4.2.1.4. Análisis operacional de VMWare Server 2.0

En la siguiente tabla se muestran aspectos de administración y uso de la máquina virtual, es importante que la máquina virtual cuente con herramientas que faciliten su uso para no dificultar la adopción de la misma por parte del usuario.

Tabla XVI. Herramientas administrativas de VMWare

Instalador	
OpenSUSE Linux 10.3	RPM y Paquete .tar.gz Configuración por script vmware-config.pl, necesita las fuentes del kernel para compilar módulos esenciales para el funcionamiento de la máquina virtual.
Windows	Ejecutable/Configuración utilizando asistente.

Herramientas de administración	
Linux y Windows	VI Web Access, es una consola de administración web en la cual se puede administrar el estado de las diferentes máquinas virtuales, además se puede controlar cada máquina virtual utilizando una consola gráfica dentro del explorador web.

Opciones de respaldo	
Instantáneas	Soportada, VMWare permite tomar una instantánea de la máquina virtual la cual puede ser utilizada como punto de restauración o como una máquina virtual independiente con el estado que tenía cuando se tomo la instantánea.
Clonación	Soportada, VMWare permite la clonación de máquinas virtuales, el clon puede ser transportado a otra máquina sin problemas de dependencias.

Interfaz de usuario	
Linux y Windows	Interfaz tipo web. Permite operaciones de administración y control de las máquinas virtuales desde clientes de la red. Esto permite centralizar las diferentes instancias del Sistema de Pruebas de Aptitud, sin embargo, se debe capacitar al usuario porque, aunque el sistema de pruebas de aptitud se sigue manejando igual, la forma de ingresar al sistema varía ya que ahora se debe utilizar el explorador de páginas web.
Interfaces externas para automatización	
Automatización	VMWare provee una interfaz de automatización para programadores y usuarios llamada VIX, esta interfaz permite que un usuario automatice sus operaciones con el sistema huésped.

VMWare provee un gran rango de herramientas administrativas y soporte, el hecho que las herramientas de administración sean web permite que el manejo de la máquina virtual pueda realizarse desde cualquier cliente conectado a la red. Esto facilitaría centralizar las múltiples instancias del sistema a virtualizar.

Conclusión: VMWare Server 2.0 posee herramientas de gestión y control apropiadas para la centralización y el manejo remoto de las máquinas virtuales. Para el Sistema de Pruebas de Aptitud, aunque no es necesario la centralización del mismo, ya que se cuenta con una sola instancia del mismo, el manejo de la máquina virtual desde dichas herramientas facilita de gran manera las tareas de arranque y apagado de la máquina virtual, pero añade tiempo de capacitación.

3.4.2.2.QEMU

3.4.2.2.1.Análisis económico de QEMU

0.9.0

Tabla XVII. Análisis de factibilidad económico para QEMU

Máquina virtual	QEMU 0.9.0
Adquisición	Se puede obtener gratuitamente desde la página web http://fabrice.bellard.free.fr/qemu/download.html
Instalación	Utilizados por ancho de banda, tiempo de instalación (Tiempo de servicio de un técnico o del equipo de informática) y la media de instalación. 3.7 MB de ancho de banda + 30 minutos de instalación. En los escenarios comunes el ancho de banda y la media es depreciable.
Total	15 minutos de instalación (No incluye la virtualización del Sistema de Pruebas de Aptitud).

Conclusión del análisis: El costo en total son 15 minutos de instalación, los cuales no representan un impedimento para la utilización de esta máquina virtual.

3.4.2.2.2. Análisis legal de QEMU 0.9.0

En la tabla siguiente se muestran los requerimientos, soporte y garantía establecidos en la licencia de QEMU, la GPL:

Tabla XVIII. Análisis de factibilidad legal para QEMU 0.9.0

Máquina Virtual	QEMU 0.9.0
Tipo de licencia	GPL, “GNU General Public License”, es una licencia de tipo <i>CopyLeft</i> en la cual se debe dar el código fuente cuando se distribuye QEMU. Las restricciones y limitaciones de este tipo de licencia no aplican al sistema huésped.
Limitaciones	La licencia no tiene limitaciones con respecto al uso que se le puede dar a la máquina virtual.
Restricciones	Si se distribuye la máquina virtual, debe ir acompañada del código fuente de la misma. Esto no es una limitación para el presente estudio porque no existe impedimento para distribuir el código fuente de QEMU. Se debe aclarar que la licencia no obliga a distribuir el código fuente del Sistema de Pruebas de Aptitud. Solo aplica a la distribución de la máquina virtual QEMU y a los cambios a ese código fuente.
Soporte	Sin soporte más que la documentación incluida.
Garantía	Sin garantías.

La licencia no impone ninguna restricción que prohíba la utilización de QEMU en este trabajo.

Conclusión: La licencia no impone ninguna restricción que impida el uso de esta máquina virtual.

3.4.2.2.3. Análisis técnico de QEMU 0.9.0

En la siguiente tabla se muestran los dispositivos requeridos por el sistema y los soportados por la máquina virtual:

Tabla XIX. Configuración de dispositivos dentro y fuera de la máquina virtual para QEMU 9.0

Dispositivos requeridos por el sistema		Dispositivos soportados por QEMU 0.9.0	
Físicos (Dispositivos disponibles)	Virtuales	Virtuales	Físicos
Impresora conectada a USB	Impresora conectada a LPT	Puerto LPT (Máximo 3)	Puerto LPT Archivos Tuberías Servidores UDP Servidores TCP Entrada estándar Telnet
Scantron conectado a USB	Scantron conectado a RS-232	Puerto RS-232 (Serial) (Máximo 4)	Puerto RS-232 Tuberías Archivos Servidores UDP Servidores TCP Entrada estándar Telnet
Unidad de respaldo de ruta parametrizable (Archivo)	Disco flexible (1.44")	Disco flexible (1.44")	Disco flexible (1.44") Imagen de disco (Ruta parametrizable)
Unidad de almacenamiento (Disco SATA)	Disco ATA	Disco ATA	Imagen de disco Disco SATA/ATA

Dispositivos requeridos por el sistema		Dispositivos soportados por QEMU 0.9.0	
Físicos (Dispositivos disponibles)	Virtuales	Virtuales	Físicos
Directorio compartido	Directorio o unidad de almacenamiento o del sistema	VFAT en una unidad del sistema	Ruta parametrizable
Adaptador gráfico (Cualquier adaptador)	Hercules/CGA/EGA/VGA	Hercules/CGA/EGA/VGA	Cualquier adaptador de video

A diferencia de VMWare, QEMU muestra una mayor variedad de dispositivos físicos soportados, además, una de las mayores ventajas de esta máquina virtual sobre las demás es el uso de un sistema VFAT para compartir archivos entre el sistema huésped y el sistema anfitrión.

Conclusión: La máquina virtual no tiene impedimentos técnicos que impidan su uso para la virtualización del sistema. Sin embargo, la impresora y el Scantron no son soportados directamente por lo que hay que utilizar redireccionamiento de puertos tanto en Windows como en Linux.

3.4.2.2.4. Análisis operacional de QEMU

0.9.0

En la siguiente tabla se muestran aspectos de administración y uso de la máquina virtual, es importante que la máquina virtual cuente con herramientas que faciliten su uso para no dificultar la adopción de la misma por parte del usuario.

Tabla XX. Herramientas administrativas de QEMU 0.9.0

Instalador	
OpenSUSE Linux 10.3	En el repositorio oficial de OpenSUSE Configuración por la línea de comandos.
Windows	Ejecutable (http://www1.interq.or.jp/t-takeda/top.html) Configuración por la línea de comandos.

Herramientas de administración	
Linux y Windows	Comandos de consola: qemu (Inicio de la máquina virtual) qemu-img (Creación de imágenes de disco) Además, al levantar la máquina virtual, QEMU crea una consola llamada Monitor la cual utiliza un puerto para la comunicación de los comandos, el puerto puede ser TCP, UDP, COM, LPT entre otros. No cuenta con interfaz gráfica para la administración de la máquina virtual.

Opciones de respaldo	
Instantaneas	Soportada, se utiliza el comando del monitor savevm.
Clonación	Soportada, copiando la imagen de disco y utilizando la misma línea de comandos.

Interfaz de usuario	
Linux y Windows	Para el inicio de la máquina virtual se utiliza un comando en línea. Para el uso del sistema operativo huésped se puede utilizar el mismo adaptador gráfico del servidor donde esta corriendo o puede levantar un servidor VNC, el cual permite el manejo remoto de la máquina virtual.

Interfaces externas para automatización	
Automatización	El código fuente de la máquina virtual esta disponible en la pagina principal de QEMU, sin embargo, no tiene una interfaz donde se implemente de forma directa tareas de automatización.

Conclusión: La máquina virtual no provee herramientas de administración con interfaz gráfica. La automatización de tareas requiere modificar el complejo código fuente de la máquina virtual, no provee como VMWare una interfaz exclusiva para automatización de tareas. Sin embargo, para el uso de la máquina virtual cuenta con un servidor VNC, así el entrenamiento requerido por el usuario final es mínimo, además se puede administrar remotamente utilizando el Monitor de QEMU.

3.4.2.3.Bochs

3.4.2.3.1.Análisis económico de Bochs

2.3.6

Tabla XXI. Análisis de factibilidad económico para Bochs 2.3.6

Máquina virtual	Bochs 2.3.6
Adquisición	Se puede obtener gratuitamente desde la página web http://bochs.sourceforge.net/getcurrent.html
Instalación	Utilizados por ancho de banda, tiempo de instalación (Tiempo de servicio de un técnico o del equipo de informática) y la media de instalación. 3.0 MB – 4.0 MB de ancho de banda + 30 minutos de instalación. En los escenarios comunes el ancho de banda y la media es depreciable.
Total	30 minutos de instalación (No incluye la virtualización del Sistema de Pruebas de Aptitud).

Conclusión del análisis: El costo en total son 30 minutos de instalación, los cuales no representan un impedimento para la utilización de esta máquina virtual.

3.4.2.3.2. Análisis legal de Bochs 2.3.6

En la tabla siguiente se muestran los requerimientos, soporte y garantía establecidos en la licencia de Bochs, la LGPL:

Tabla XXII. Análisis de factibilidad legal para Bochs 2.3.6

Máquina Virtual	Bochs 2.3.6
Tipo de licencia	LGPL, “ <i>GNU Lesser General Public License</i> ”, es una licencia de tipo <i>CopyLeft</i> . A diferencia de la licencia GPL, utilizada en la máquina virtual QEMU, la licencia LGPL no impone la restricción de liberar el código de los módulos que interactúan con la máquina virtual, para este estudio ambas licencias GPL y LGPL pueden ser utilizadas.
Limitaciones	La licencia no tiene limitaciones con respecto al uso que se le puede dar a la máquina virtual.
Restricciones (Prohibiciones)	Si se distribuye la máquina virtual, debe ir acompañada con código fuente. Esta restricción no aplica al sistema huésped o al Sistema de Pruebas de Aptitud, por lo que es seguro distribuir la máquina virtual y el Sistema de Pruebas de Aptitud.
Soporte	Sin soporte más que la documentación incluida.
Garantía	Sin garantías.

La licencia no impone ninguna restricción que prohíba la utilización de Bochs en este trabajo.

Conclusión: La licencia no impone ninguna restricción que impida el uso de la máquina virtual en la virtualización del Sistema de Pruebas de Aptitud.

3.4.2.3.3. Análisis técnico de Bochs 2.3.6

En la siguiente tabla se muestran los dispositivos requeridos por el sistema y los soportados por la máquina virtual:

Tabla XXIII. Configuración de dispositivos dentro y fuera de la máquina virtual para Bochs 2.3.6

Dispositivos requeridos por el sistema		Dispositivos soportados por Bochs 2.3.6	
Físicos (Dispositivos disponibles)	Virtuales	Virtuales	Físicos
Impresora conectada a USB	Impresora conectada a LPT	Puerto LPT (Máximo 2)	Puerto LPT Archivos
Scantron conectado a USB	Scantron conectado a RS-232	Puerto RS-232 (Serial) (Máximo 4)	Puerto RS-232 Servidores TCP Entrada/Salida estándar
Unidad de respaldo de ruta parametrizable (Archivo)	Disco flexible (1.44")	Disco flexible (1.44")	Disco flexible (1.44") Imagen de disco (Ruta parametrizable)
Unidad de almacenamiento (Disco SATA)	Disco ATA	Disco ATA	Imagen de disco Disco SATA/ATA
Directorio compartido	Directorio o unidad de almacenamiento del sistema	* Utilizando una pila TCP/IP para D.O.S.	Servidor de archivos
Adaptador gráfico (Cualquier adaptador)	Hercules/CGA/EGA/VGA	Hercules/CGA/EGA/VGA	Cualquier adaptador de video

* Nativamente la máquina virtual no cuenta con una forma de compartir transparentemente archivos entre los sistemas huésped y anfitrión, sin embargo, se puede instalar un conjunto de herramientas para comunicación TCP/IP en D.O.S.

Conclusión: La máquina virtual no tiene impedimentos técnicos que impidan su uso para la virtualización del sistema. Al igual que las demás máquinas virtuales, se debe redireccionar los puertos de impresión y para el Scantron.

3.4.2.3.4. Análisis operacional de Bochs

2.3.6

En la siguiente tabla se muestran aspectos de administración y uso de la máquina virtual, es importante que la máquina virtual cuente con herramientas que faciliten su uso para no dificultar la adopción de la misma por parte del usuario.

Tabla XXIV. Herramientas administrativas de Bochs 2.3.6

Instalador	
OpenSUSE Linux 10.3	En el repositorio oficial de OpenSUSE Configuración modificando archivo bochsrc.
Windows	Ejecutable (http://bochs.sourceforge.net/getcurrent.html) Configuración modificando archivo bochsrc.
Herramientas de administración	
Linux y Windows	Comandos de consola: textconfig, es una herramienta de administración basada en menús de texto. bximage, comando para crear imagenes de disco de las máquinas virtuales. bxcommit, se utiliza para recuperar información en caso de bajas de corriente y otros problemas que interrumpen la ejecución de la máquina virtual.
Opciones de respaldo	
Instantáneas	No soportada.
Clonacion	Soportada, copiando la imagen de disco.
Interfaz de usuario	
Linux y Windows	Para el inicio de la máquina virtual se utiliza el comando bochs.

Interfaces externas para automatización	
Automatización	El código fuente de la máquina virtual esta disponible en la pagina principal, sin embargo, no tiene una interfaz donde se implemente de forma directa tareas de automatización.

Conclusión: De las tres máquinas virtuales, esta es la mas limitada porque no cuenta herramientas que permitan su uso o configuración remota.

3.4.3.Comparativa de las alternativas

En la siguiente tabla se muestran las características mas relevantes de cada una de las máquinas virtuales, se seleccionará aquella máquina virtual con la mayoría de ellas:

Tabla XXV. Comparativa de las máquinas virtuales candidatas

Características	VMWare Server	QEMU	Bochs
Creador	VMWare	Fabrice Bellard ayudado por una comunidad de programadores	Kevin Lawton
Versión	2.0	0.9.0	2.3.6
Licencia	VMWare Server EULA	GPL	LGPL
Tamaño	350 MB	3.7 MB	3.5 – 4.0 MB
Administración remota	WEB VI Access	Monitor de QEMU (Puerto TCP/UDP)	N/A
Uso remoto	WEB VI Access/VNC	VNC	N/A
Directorio compartido	N/A	VFAT	N/A

La máquina virtual QEMU cuenta con todas las características deseadas, por lo que es la herramienta apta para la virtualización del Sistema de Pruebas de Aptitud. Por esta y otras razones QEMU es la máquina virtual seleccionada para la virtualización del sistema. En la siguiente subsección se detallan las razones por las cuales fue seleccionada.

3.4.3.1. Justificación de la máquina virtual seleccionada

La máquina virtual seleccionada es QEMU, las razones para utilizarla son las siguientes:

- Permite compartir transparentemente archivos entre el sistema anfitrión y el sistema huésped utilizando un sistema de archivos tipo VFAT (*Virtual File Allocation Table*). Esto no puede ser realizado con las otras dos alternativas sin el uso de herramientas de red para D.O.S., lo cual incrementa el tiempo de configuración.
- El instalador es el mas pequeño, eso facilita la obtención y el transporte de la máquina virtual.
- Es una tecnología que ha sido probada en ambientes empresariales.
- La licencia utilizada es la GPL, la cual es utilizada para asegurar la distribución del código fuente la máquina virtual. Es una licencia ideal para este estudio porque no impone restricciones sobre el uso de la herramienta.

- Los puertos para imprimir y capturar las pruebas en el sistema pueden ser utilizados desde la red, lo cual permite la implementación de un sistema centralizado de impresión y capturas de las prueba. Esto no es parte del presente estudio, pero es un punto que se debe considerar para reducir costos en caso de fallas de la impresora o del Scantron.

3.5.Implementación

En esta sección se indica paso a paso la puesta en funcionamiento del sistema de Pruebas de Aptitud en una máquina virtual, para ello la sección se divide en tres subsecciones; la preparación de la instalación, la instalación y la configuración de la máquina virtual.

Algunas de las subsecciones aplican únicamente a uno de los sistemas anfitriones, estos son Windows XP/Vista o OpenSUSE Linux, por ello se indica al inicio de la subsección si aplica únicamente a alguno de los dos, en caso de no indicarlo el paso se aplica para ambos sistemas.

3.5.1.Preparación de la instalación

Algunas de las herramientas utilizadas en este estudio están contenidas en el material digital adjunto. Lo incluido no viola los derechos de los autores, porque en las licencias de lo ello esta permitida la redistribución de dicho material. Cualquier material que prohíba su redistribución no se incluye pero se indica como obtenerlo en caso de ser necesario.

Antes de instalar la máquina virtual en el sistema anfitrión, ya sea OpenSUSE o Windows, se debe obtener el sistema huésped, el cual es D.O.S. Para el estudio se selecciono FreeDOS por ser gratuito y compatible con sistemas D.O.S. Comerciales.

3.5.1.1. Obtención del sistema huésped FreeDOS

FreeDOS es un sistema operativo totalmente compatible con MS-DOS, el cual es el utilizado en el Sistema de Pruebas de Aptitud. En el material digital adjunto a este trabajo se encuentra la versión 1.0 de este sistema operativo, bajo el nombre de **fdfullcd.iso** en la carpeta de **extras**. En caso de no poseer el material adjunto se puede conseguir de la pagina web de FreeDOS (<http://www.freedos.org/>).

3.5.1.2. Obtención de la máquina virtual para Windows XP y Vista

Esto aplica únicamente a la virtualización en sistemas Windows.

En el material adjunto bajo el nombre de **qemu-0.9.0-windows.zip** en la carpeta **VM** se encuentra el instalador de QEMU versión 0.9.0, también se puede obtener de la ruta <http://www1.interq.or.jp/t-takeda/qemu/qemu-0.9.0-windows.zip>

3.5.1.3. Obtención de la máquina virtual para OpenSUSE 10.3

Esto aplica únicamente a la virtualización en sistemas OpenSUSE.

Si se cuenta con una conexión activa a Internet, QEMU puede ser obtenido e instalado desde el manejador de paquetes como se indica en la subsección 3.5.2.2, Instalación en OpenSUSE. De lo contrario, el paquete se encuentra bajo el nombre de **qemu-0.9.0.cvs-35.i586.rpm** en la carpeta **VM** del material adjunto. Alternativamente se puede obtener de la dirección <ftp://mirrors.kernel.org/opensuse/distribution/10.3/repo/oss/suse/i586> bajo el mismo nombre.

3.5.2.Instalación de la máquina virtual

3.5.2.1.Instalación en Windows

Esto aplica únicamente a la virtualización en sistemas **Windows XP y Vista**.

Si esta disponible el material digital de este trabajo seguir la instalación automática, de lo contrario la instalación manual.

La instalación automática configura la máquina virtual, en cambio para la instalación manual se deben seguir los pasos en la sección 3.5.3 Configuración de la máquina virtual.

3.5.2.1.1.Instalación automática

1. Iniciar sesión con una cuenta con privilegios administrativos.
2. Insertar el medio donde se encuentra el material digital.
3. Explorar el medio utilizando el explorador de Windows (Combinación: WINDOWS+e).
4. Entrar a la carpeta **windows\bin**
5. Ejecutar el archivo **win_instalar_qemu.cmd**, el cual es un archivo por lotes.

Al terminar los pasos la máquina virtual estará lista para ser usada.

3.5.2.1.2.Instalación manual

1. Iniciar sesión con una cuenta con privilegios administrativos.
2. Crear una carpeta llamada **QEMU** dentro de la carpeta principal (raíz) de la unidad **C:** (C:\QEMU).

3. Con el explorador de Windows, abrir el contenido del material digital adjunto.

4. Abrir la carpeta **windows\vm** en la cual se encuentran las herramientas de virtualización.

5. Localizar el archivo **qemu-0.9.0-windows.zip** dentro de la carpeta. Alternativamente el archivo se puede obtener de la página <http://www1.interq.or.jp/t-takeda/qemu/qemu-0.9.0-windows.zip>.

6. Copiar el archivo a la carpeta C:\QEMU creada anteriormente.

7. Una vez copiado el archivo se descomprime utilizando la opción **Extraer** del menú contextual (sobre el archivo, botón derecho del ratón). Opcionalmente, al finalizar la extracción se puede eliminar el archivo.

En el caso de la instalación manual se debe configurar la máquina virtual antes de poder ser usada.

3.5.2.2.Instalación en OpenSUSE 10.3

Esto aplica únicamente a la virtualización en sistemas **OpenSUSE**.

Esta sección contiene los pasos para instalar la máquina virtual en el sistema operativo OpenSUSE 10.3, para ello se exponen dos métodos, la instalación en línea y fuera de línea.

La instalación en línea se utiliza cuando se tiene la media original de OpenSUSE o una conexión a Internet activa, de lo contrario se utiliza la instalación fuera de línea.

Luego de ser instalada, la máquina virtual debe ser configurada siguiendo los pasos contenidos en la sección 3.5.3 Configuración de la máquina virtual.

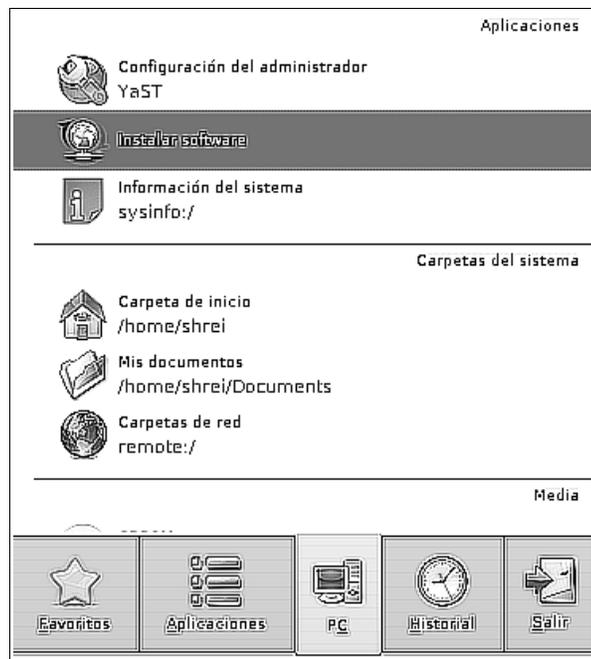
3.5.2.2.1.Instalación en línea

Para la instalación en línea se debe contar con la media de instalación completa de OpenSUSE o con una conexión activa a Internet y el repositorio principal de OpenSUSE correctamente configurado, de lo contrario se debe utilizar la instalación fuera de línea.

Para instalar QEMU se debe utilizar la opción de Instalar Software del menú principal de OpenSUSE. Para ello primero se debe ingresar el menú principal el cual comúnmente tiene el siguiente ícono:

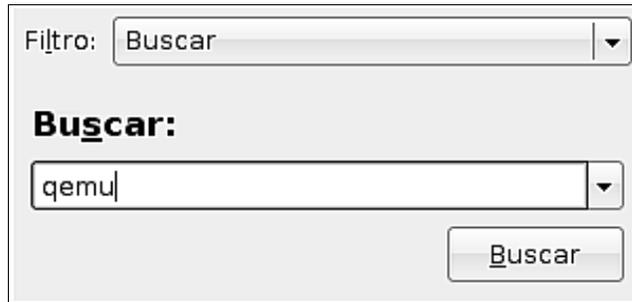


Luego se debe ingresar en la opción de Instalar Software en la pestaña de PC.



El sistema pedirá la contraseña de administrador, una vez ingresada mostrará la ventana principal del manejador de paquetes.

En la casilla de buscar se debe colocar la palabra `qemu` y presionar buscar.



Filtro: Buscar

Buscar:

qemu

Buscar

De los resultados de la búsqueda se deben seleccionar dos paquetes, el primero es **qemu** y el segundo el llamado **kqemu-kmp-default**, el cual es utilizado para mejorar el rendimiento de la máquina virtual.

Paquete	Resumen
<input type="checkbox"/> kqemu-kmp-bigsmp	The QEMU Accelerator Module
<input type="checkbox"/> kqemu-kmp-debug	The QEMU Accelerator Module
<input checked="" type="checkbox"/> kqemu-kmp-default	The QEMU Accelerator Module
<input type="checkbox"/> kqemu-kmp-xen	The QEMU Accelerator Module
<input type="checkbox"/> kqemu-kmp-xenpae	The QEMU Accelerator Module
<input checked="" type="checkbox"/> qemu	Universal CPU emulator

Al terminar de seleccionar los paquetes correspondientes, se presiona el botón aceptar el cual se encuentra en la esquina inferior derecha, esto instalará los paquetes correspondientes los cuales estarán listos para ser configurados.

3.5.2.2.2. Instalación fuera de línea

Si no se cuenta con la media original de OpenSUSE 10.3 o una conexión a Internet, entonces se deben seguir los siguientes pasos para instalar QEMU 0.9.0:

1. Insertar el medio con el material digital adjunto.
2. Abrir en el navegador de archivos de OpenSUSE (Konqueror) el material digital, por lo regular al insertar el medio digital OpenSUSE automáticamente muestra una ventana con el contenido.
3. Una vez abierta la media en el navegador de archivos, ingresar a la carpeta llamada **VM** y darle un clic de ratón al archivo **qemu-0.9.0.cvs-35.i586.rpm**.
4. Inmediatamente el sistema pedirá la contraseña de administrador la cual es necesaria para instalar la máquina virtual.
5. Luego del ingreso de la contraseña, el sistema informará el avance de la instalación, al terminar la máquina virtual QEMU debe ser configurada como se explica en la sección siguiente. Utilizando el mismo procedimiento se puede instalar el paquete **kqemu-kmp-default-1.3.0pre11_2.6.22.5_31-6.i586.rpm** el cual se utiliza para aumentar el rendimiento de la máquina virtual.

3.5.3. Configuración de la máquina virtual

La presente sección indica los pasos para configurar la máquina virtual, también se incluye en el material digital adjunto archivos de comandos para realizar esta configuración automáticamente.

Para la configuración automática se deben realizar los siguientes pasos:

- 1 (Solo Windows) Iniciar sesión con una cuenta de administrador.

- 2 Insertar la media que contiene el material digital de este trabajo.

- 3 Navegar por la media extraíble, entrando a la carpeta correspondiente al sistema operativo:

- 3.1 Carpeta **windows** para el sistema operativo Windows XP/Vista.

- 3.2 Carpeta **opensuse** para el sistema operativo OpenSUSE 10.3.

- 4 Entrar a la carpeta **bin**.

- 5 Ejecutar el comando correspondiente a la configuración. En Windows **win_configurar.cmd**, en OpenSUSE **lin_configurar.cmd**.

- 6 La máquina virtual esta configurada y lista para ser utilizada.

En caso de no configurar la máquina virtual con el procedimiento anterior, a continuación se indican paso a paso la configuración tanto en Windows como en OpenSUSE.

3.5.3.1.Creación de la jerarquía de carpetas

Tanto en Windows como en OpenSUSE se creará la siguiente jerarquía de carpetas (directorios), para la organización de la máquina virtual.



En cada carpeta se almacenarán las siguientes partes del sistema:

En la carpeta **spa/bin** se almacenaran los archivos de comandos para iniciar y parar la máquina virtual.

En la carpeta **spa/compartido** se almacenará la aplicación que se desea virtualizar, para este estudio es el Sistema de Pruebas de Aptitud, este sistema no pudo ser incluido en el material digital por la licencia del mismo, sin embargo, el lector puede copiar a esta carpeta el sistema que quiera virtualizar.

En la carpeta **spa/discos** se almacenarán las imágenes de los discos virtuales utilizados por la máquina virtual.

3.5.3.1.1.En Windows

La siguiente jerarquía de carpetas debe ser creada en la unidad **C:**

Nombre	Tipo de archivo
c:\	Carpeta
spa	Carpeta
bin	Carpeta
compartido	Carpeta
discos	Carpeta

Para crearla se puede utilizar el archivo de comandos incluido en el material digital adjunto el cual se encuentra dentro de la media en la carpeta **windows\bin** y es llamado **win_crear_jerarquia.cmd**.

Manualmente se puede utilizar la consola de comandos de Windows, siguiendo los siguientes pasos:

1. Abrir el diálogo ejecutar del menú de inicio (Si no existe la opción ejecutar en el menú de inicio presionar la combinación de teclas **WINDOWS+r**).
2. Escribir el comando **cmd** dentro del diálogo y presionar aceptar.
3. En la consola ejecutar el siguiente grupo de comandos:

```
mkdir c:\spa  
  
mkdir c:\spa\bin  
  
mkdir c:\spa\compartido  
  
mkdir c:\spa\discos
```

4. Cerrar la consola.

3.5.3.1.2. En Linux

La siguiente estructura de directorios debe ser creado en el directorio del usuario (~/).

Nombre	Tipo de archivo
~/	Carpeta
spa	Carpeta
bin	Carpeta
compartido	Carpeta
discos	Carpeta

Para crearla se puede utilizar el archivo de comandos incluido en el material digital adjunto el cual se encuentra dentro de la media en la carpeta **linux\bin** y es llamado **lin_crear_jerarquia.cmd**.

Manualmente se puede utilizar la terminal, siguiendo los siguientes pasos:

4. Abrir el dialogo de ejecución presionando la combinación de teclas **ALT+F2**

5. Escribir el comando **konsole** dentro del dialogo y presionar el boton de ejecutar.

6. En la terminal ejecutar el siguiente grupo de comandos:

```
mkdir $HOME/spa
```

```
mkdir $HOME/spa/bin
```

```
mkdir $HOME/spa/compartido
```

```
mkdir $HOME/spa/discos
```

4. Cerrar la terminal.

3.5.3.2.Creación de la imagen de disco

Una imagen de disco es un archivo que la máquina virtual utiliza como una unidad de almacenamiento. En dicho archivo se almacenará la información del sistema huésped, en este caso FreeDOS, y es importante que al momento de ser creado se le de un tamaño adecuado para no tener problemas de espacio más adelante.

El espacio que FreeDOS ocupa es 200 MB por lo cual una imagen de disco de 300MB es razonable dado que en esta imagen no se almacenará la base de datos.

El material digital adjunto contiene una imagen de disco preparada y lista para ser usada. La imagen se encuentra en la carpeta **discos** el archivo se llama **disco1.img**. Sin embargo, a continuación se muestran los pasos para crear la imagen de disco.

Los comandos utilizados para crear la imagen de disco son los siguientes:

3.5.3.2.1.En Windows

1. Abrir la consola (**Inicio->Todos los programas->Accesorios->Símbolo del sistema**).
2. En la consola ingresar los siguientes comandos:

```
cd c:\qemu\bin
```

```
qemu-img create -f qcow2 c:\spa\discos\disco1.img  
300M
```

3.5.3.2.2.En Linux

```
qemu-img create -f qcow2 $HOME/spa/discos/disco1.img  
300M
```

El comando creará un archivo de 300,000,000 bytes el cual sera utilizado como disco de almacenamiento.

3.5.3.3.Instalación del sistema operativo FreeDOS

El material digital adjunto contiene una imagen de disco preparada y lista para ser usada, la cual contiene instalado el sistema operativo FreeDOS. La imagen se encuentra en la carpeta **discos** el archivo es llamado **disco1.img**. Si no se cuenta con el material digital adjunto a este trabajo, se deben seguir los siguientes pasos para preparar dicha imagen:

1. Crear la imagen de disco como se muestra en la sección anterior (Creación de la imagen de disco)
2. Obtener la imagen del sistema operativo FreeDOS como se indica en la sección 3.5.1.1 Obtención del sistema huésped FreeDOS.
3. Copiar la imagen del instalador (**fdfullcd.iso**) a la carpeta **discos** que se encuentra dentro de la carpeta **spa** (en windows es **c:\spa\discos**, en OpenSUSE **\$HOME/spa/discos**).

4. Correr la máquina virtual e instalar el sistema operativo FreeDOS, como se muestra a continuación:

3.5.3.3.1.En windows

Abrir la consola (**Inicio->Todos los programas->Accesorios->Símbolo del sistema**)

En la consola ejecutar los comandos siguientes:

```
cd c:\spa
```

```
qemu -L c:\qemu -boot d -hda discos\disco1.img -cdrom  
discos\fdfullcd.iso
```

3.5.3.3.2.En OpenSUSE

Abrir la terminal (**CTRL+F2 -> konsole**)

En la terminal ejecutar los comandos siguientes:

```
cd $HOME/spa
```

```
qemu -boot d -hda discos/disco1.img -cdrom  
discos/fdfullcd.iso
```

3.5.3.3.3.Windows y OpenSUSE

Al iniciar la máquina virtual se deberá completar la instalación siguiendo las instrucciones que presenta el instalador de FreeDOS.

Al terminar la instalación de FreeDOS, se deben configurar el almacenamiento compartido y los periféricos, esto se detalla en las siguientes secciones.

3.5.3.4.Configuración del almacenamiento compartido

Para que el sistema a virtualizar pueda ser usado por la máquina virtual debe ser copiado a una carpeta que será compartida tanto por el sistema anfitrión como por el sistema huésped. Esto es posible gracias al sistema de archivos vfat que utiliza la máquina virtual QEMU.

La carpeta **spa/compartido**, la cual fue creada en la sección de creación de jerarquía de carpetas, debe contener el sistema que se desea virtualizar, para este estudio es el SPA.

La opción de QEMU que permite compartir esta carpeta es:

```
-hdb fat:rw:compartido
```

Esa opción indica el disco virtual que se creará (**hdb**) con el contenido de una carpeta del sistema anfitrión (**compartido**).

Las partes de dicha opción son las siguientes: **hdb** se refiere al disco D: dentro de FreeDOS, **rw** se refiere a los permisos de la carpeta compartida en este caso de lectura y escritura, **compartido** es la carpeta que se utilizará para ser compartida entre el sistema anfitrión y el huésped.

Los siguientes comandos inician la máquina virtual utilizando la carpeta compartida **compartido**:

3.5.3.4.1.En Windows

```
cd c:\spa  
  
c:\qemu\qemu -L c:\qemu -hda discos\disco1.img -hdb  
fat:rw:compartido
```

3.5.3.4.2.En OpenSUSE

```
cd $HOME/spa  
  
qemu -hda discos/disco1.img -hdb fat:rw:compartido
```

3.5.3.5.Configuración de periféricos

La configuración de periféricos (impresora, Scantron y ratón) consiste en ajustar los puertos de comunicación entre el sistema huésped (FreeDOS) y el sistema anfitrión (OpenSUSE o Windows).

Los puertos de comunicación dentro de la máquina virtual difieren según el tipo de dispositivo, por ejemplo, la impresora utiliza un puerto paralelo, el Scantron un puerto serial y el ratón un puerto PS/2. En cambio, fuera de la máquina virtual, en el sistema anfitrión, los dispositivos por lo regular utilizan el mismo tipo de puerto USB. El problema de los dispositivos virtuales y los reales entonces se resume a un problema de comunicación entre un puerto específico virtual y un puerto USB real. Para resolver ese problema se tomó una aproximación distinta para cada dispositivo, la impresora, el Scantron y el ratón. A continuación se explica como configurar esos tres dispositivos.

3.5.3.5.1. Configuración del ratón

La máquina virtual configura automáticamente el ratón si encuentra uno en el sistema anfitrión. En el sistema huésped aparecerá el dispositivo listo para ser utilizado. Si el ratón no funciona en la máquina virtual puede ser por las siguientes razones:

1. La aplicación virtualizada no soporta operaciones con el ratón, esto es porque en D.O.S. abundan las aplicaciones con interfaz tipo consola y muchas de ellas no soportan el ratón en su programación.
2. El controlador del ratón no inició al inicio del sistema FreeDOS. Esto puede ser arreglado corriendo el comando **mouse** dentro de la máquina virtual.

3.5.3.5.2. Configuración de la impresora

La impresión depende de la comunicación entre el sistema FreeDOS y el sistema anfitrión (Windows u OpenSUSE). La configuración que se realizará debe permitir a las aplicaciones dentro de la máquina virtual poder utilizar la impresora física. Así dentro del sistema de pruebas de aptitud se podrá utilizar la impresora USB aún cuando ese sistema no haya sido programado para utilizar ese tipo de impresoras.

Existe un problema para la impresión desde la máquina virtual, éste radica en la forma en que se comunican los programas de FreeDOS con la impresora y lo diferente en que esto mismo sucede en Windows y OpenSUSE.

Entonces la configuración de la impresora consiste en homogeneizar la comunicación de la máquina virtual con el sistema anfitrión en los puertos de impresión. En los sistemas operativos Windows esto se logrará compartiendo la impresora a la red para luego mapear la impresora compartida a un puerto local de tipo paralelo. Por otra parte, en OpenSUSE se logrará utilizando el sistema CUPS, en específico el comando **lp**, el cual se puede utilizar para imprimir desde archivos y en el caso de este estudio desde una tubería. La configuración para cada tipo de sistema anfitrión se explica a continuación con mayor detalle:

3.5.3.5.2.1. En OpenSUSE

Para que la impresión funcione desde la máquina virtual la impresora debe estar correctamente configurada en el sistema operativo OpenSUSE.

Los pasos básicos para configurar la impresora son los siguientes:

1. Entrar al Centro de Control de **YaST**.
2. Entrar al apartado **Hardware** y seleccionar la opción **Impresora**.
3. Seleccionar si la impresora es local o de red. Se seleccionó una impresora de tipo local, aunque esto variará dependiendo de donde se encuentre conectada la impresora.
4. Seleccionar el tipo de impresora que se usara, en este estudio se utilizó una impresora con puerto USB.
5. Seleccionar el controlador de la impresora.
6. Finalizar el asistente presionando el botón finalizar.

Luego de configurar la impresora en el sistema operativo anfitrión, se debe crear una tubería la cual servirá como comunicación entre el sistema huésped y OpenSUSE.

Para crear la tubería se deben seguir lo siguientes pasos:

1. Abrir la terminal (**CTRL+F2, konsole**)
2. Ejecutar en la terminal los siguientes comandos:

```
cd $HOME/spa
```

```
mkfifo impresion
```

3. Cerrar la terminal.

La tubería se utilizará para redireccionar las impresiones de la máquina virtual a la impresora conectada físicamente. Sin embargo, antes que funcione completamente la impresión se debe crear un método que detecte los fines de página en la impresión, esto debido a la diferencia en el manejo de las páginas entre OpenSUSE y FreeDOS. Para detectar correctamente estos fines de página se creó en este estudio un programa en lenguaje C el cual detecta el fin de página y es parte de la tubería que se utiliza en la impresión, el programa es llamado filtro.

El programa filtro es realizado en C, el código fuente se encuentra en el material digital adjunto o en este documento en el APÉNDICE I: Código fuente del filtro de impresión.

Antes de iniciar la máquina virtual, y que esta pueda mandar a imprimir, se debe ejecutar un proceso que monitoree la tubería y al encontrar información que la envíe a la impresora, para eso se creó el siguiente archivo de comandos, el cual debe ser ejecutado antes de iniciar la máquina virtual QEMU.

El archivo está en el material digital dentro de la carpeta **opensuse/bin**, con el nombre **lin_monitor.sh**, debe ser colocado en **\$HOME/spa/bin/lin_monitor.sh**, en caso de no contar con el material digital el contenido del archivo es el siguiente:

```
#!/bin/bash

cd $HOME/spa

while true
do

    cat impresion |

    $HOME/bin/filtro 24 |

    lp -o cpi=12 -o lpi=8 -o page-left=72

done
```

Con esto la configuración de la impresora en OpenSUSE esta completa. Al iniciar **qemu** se debe agregar la siguiente opción:

```
-parallel pipe:$HOME/spa/impresion
```

3.5.3.5.2.2.En Windows

Si la impresora utiliza un puerto USB existe la necesidad de mapear dicho puerto a uno paralelo. Esto es debido a que FreeDOS utiliza puertos paralelos para la impresión. Para hacer esto en Windows los pasos para realizar eso son los siguientes:

1. Compartir la impresora, en este estudio el recurso compartido se llamará **Impresora**.
2. Mapear la impresora compartida a un puerto paralelo, esto se realiza con el comando siguiente:

```
NET USE LPT1: \\localhost\Impresora /persistent:yes
```

En el comando utilizado, **localhost** es la máquina donde se encuentra conectada la impresora. Si la impresora esta en la misma máquina donde funcionará la máquina virtual esto no debe cambiarse, sin embargo, si la máquina donde esta conectada la impresora es otra, **localhost** debe cambiarse por el nombre de la máquina.

Ambos pasos solo son necesarios realizarlos una vez cuando se instala la máquina virtual.

Para que la impresora USB pueda utilizar el puerto LPT1 correctamente, se deben cumplir las siguientes condiciones:

1. El puerto LPT1 debe estar libre, de lo contrario se debe utilizar otro, como LPT2.
2. La impresora debe estar compartida.
3. La impresora debe estar conectada a la misma máquina.

Con esto la configuración de la impresora en Windows esta completa. Al iniciar **qemu** se debe agregar la siguiente opción:

```
-parallel LPT1
```

3.5.3.5.3. Configuración del Scantron

El Scantron que se utiliza en el SPA tiene un puerto tipo RS-232 (Serial), el cual difiere de los puertos USB utilizados con más frecuencia en la actualidad, por lo cual, para conectar ese dispositivo a máquinas más modernas es necesario adquirir un adaptador RS-232 a USB. En la fecha en que se escribió este trabajo (2007) dichos adaptadores se adquieren a un precio promedio de Q. 200.00 en tiendas de accesorios electrónicos como Office Depot e Intelaf.

Al instalar el adaptador RS-232 a USB, también llamado USB a Serial, se pone a la disposición del sistema anfitrión un puerto tipo Serial (**COM**) el cual puede ser utilizado por la máquina virtual. Para utilizarlo se deben usar la siguiente opción:

3.5.3.5.3.1.En OpenSUSE

-serial /dev/ttySn

En donde n es el número de puerto asignado al adaptador USB a Serial, empezando por el 0.

3.5.3.5.3.2.En Windows

-serial COMn

En donde n es el número de puerto asignado al adaptador USB a Serial, empezando por el 1.

3.5.3.6.Optimización de rendimiento

El rendimiento de la máquina virtual se mejoró utilizando el Acelerador de QEMU versión 1.3.0pre11. Además, como los sistemas operativos compatibles con D.O.S., entre estos FreeDOS, no utilizan la instrucción del procesador **HLT** (*Halt*), por lo que en D.O.S. el procesador está usándose continuamente, lo cual disminuye el rendimiento en general del sistema anfitrión. Para resolver lo anterior se instalará una aplicación que utilizará dentro de la máquina virtual la instrucción **HLT** del procesador.

3.5.3.6.1.En OpenSUSE

Para mejorar el rendimiento en este sistema se debe instalar el paquete **kqemu**, lo cual puede ser realizado desde el manejador de paquetes (**YaST**) o desde la línea de comandos con la versión del paquete que se incluye en el material digital.

Si se siguieron las instrucciones para instalar **QEMU** desde el repositorio como se mostró en la sección 3.5.2.2, Instalación de la máquina virtual, entonces este paquete estará instalado correctamente. De lo contrario, se puede instalar con el paquete contenido en el material digital el cual se llama **kqemu-kmp-default-1.3.0pre11_2.6.22.5_31-6.i586.rpm**.

Desde la línea de comandos se puede instalar utilizando el comando dentro del directorio donde se encuentre el archivo:

```
rpm --install kqemu*
```

3.5.3.6.2.En Windows y OpenSUSE

Para aumentar el rendimiento en ambos sistemas anfitriones se debe instalar una utilidad llamada DOSIdle, la cual reduce el consumo de FreeDOS. Para instalarlo se debe copiar el archivo DOSIdle210.zip a la carpeta **compartido** dentro de la carpeta **spa**. Una vez copiado con la terminal o consola se inicia la máquina virtual con los siguientes parámetros:

Dentro de la carpeta **spa**:

En OpenSuse

```
qemu -hda discos/disco1.img -hdb fat:compartido
```

En Windows

```
c:\qemu\qemu -hda c:\spa\discos\disco1.img -hdb  
fat:c:\spa\compartido
```

Dentro del sistema FreeDOS se ejecutan los siguientes comandos:

```
cd fdos
```

```
cd bin
```

```
unzip d:\dosidle2.zip
```

```
cd c:\
```

```
echo dosidle >> autoexec.bat
```

Con esto al iniciar FreeDOS también iniciará DOSidle mejorando el rendimiento de los programas dentro y fuera de la máquina virtual.

3.6. Validación de la solución virtual

La validación de la solución virtual está dividida en dos partes, la primera es una validación técnica. La segunda es una validación por parte del usuario final.

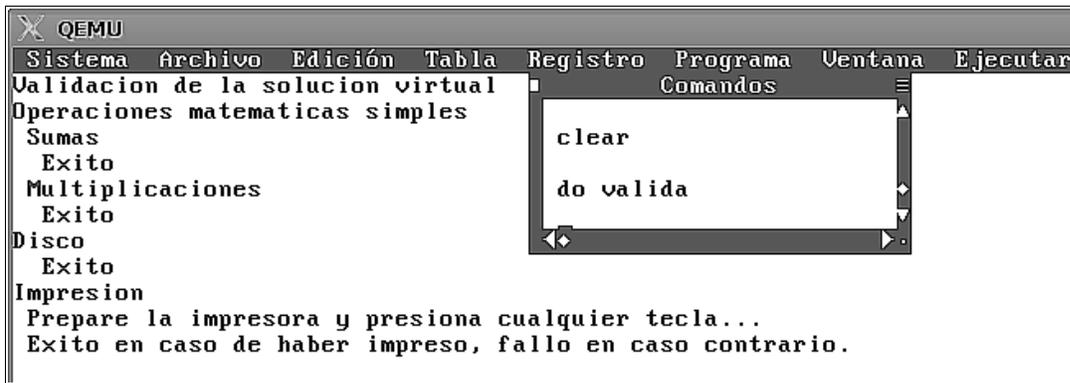
La diferencia entre la validación técnica y la de usuario radica en quién la realiza, en el caso de la validación técnica es una aplicación automatizada la que indica si se valida o no la solución virtual, por otro lado en la validación de usuario es el usuario final el que da su visto bueno siguiendo los casos de uso que se obtuvieron del sistema en la plataforma original y que se muestran en la sección 3.3.

3.6.1. Validación técnica

La validación técnica se realizó desarrollando una aplicación de pruebas, la cual tiene como propósito descubrir errores en el comportamiento del sistema huésped, esta aplicación esta desarrollada en FoxPRO 2.6 debido a que ese es el interprete del Sistema de Pruebas de Aptitud.

El código fuente de la aplicación de validación se encuentra en el APÉNDICE II: Código fuente de la aplicación para validar la solución virtual.

La siguiente imagen muestra la corrida de la aplicación **valida.prg** la cual revisa aspectos técnicos de la solución virtual, dicho programa esta funcionando bajo FoxPRO 2.6 en FreeDOS dentro de la máquina virtual QEMU:



Como se puede observar en la imagen, el sistema virtual paso las pruebas técnicas, además durante dichas pruebas el sistema estuvo estable.

3.6.2. Validación del usuario

El usuario utilizó el sistema virtualizado siguiendo los casos de uso planteados en la sección 3.3. En la sesión de pruebas se encontró lo siguiente:

Variable	Valor
Costo directo de implementación	Asesoría técnica (Honorarios por la asesoría) en el caso de este estudio la asesoría fue brindada gratuitamente por Luis Cano el autor de este trabajo.
Tiempo de implementación	2 días (4 horas cada día)
Número de caso de uso	14
Número de casos de uso realizados correctamente	14
Número de funciones realizadas incorrectamente	0 casos de uso
Tiempo fuera de línea del sistema	0 horas (Se implemento en una máquina adicional, la cual era poco utilizada)
Número de incompatibilidades con dispositivos	0 incompatibilidades
Número de incompatibilidades con dispositivos bajo virtualización	0 incompatibilidades

El sistema en el ambiente virtual esta en acordanza con los requerimientos del usuario, por lo que se validó el sistema virtualizado como alternativa viable al sistema en la plataforma original.

3.6.3.Comparativa de rendimiento

En esta sección se hace una comparativa del rendimiento del Sistema de Pruebas de Aptitud. Se comparan dos aspectos del rendimiento del sistema, estos son el tiempo en completar una tarea y el uso del procesador en porcentaje, en tres ambientes diferentes, el primero es la plataforma original donde corre actualmente el sistema, el segundo es en la máquina virtual corriendo sobre un sistema Windows Vista y el tercero en la máquina virtual corriendo sobre OpenSUSE 10.3.

Para la comparativa se utilizó un proceso desarrollado con el fin de estresar los recursos del sistema, en especial el procesador y el disco de almacenamiento. El código fuente del proceso se encuentra en el APÉNDICE III: Código fuente aplicación para medir el rendimiento.

Las especificaciones técnicas más relevantes de los equipos donde se realizaron las pruebas son las siguientes:

Tabla XXVI. Especificaciones técnicas pruebas de rendimiento

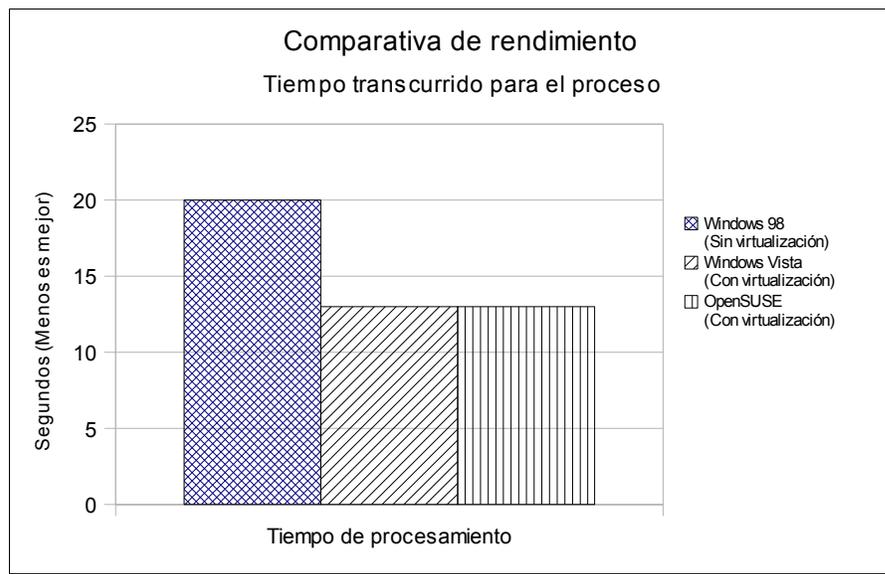
	Plataforma original (Windows 98)	Plataforma virtualizada (Windows Vista y OpenSUSE)
Marco y modelo	GreatQuality Green 320	Compaq Presario C500
Procesador	VIA C3 1.3GHZ	Intel Pentium Dual Core processor T2060
Memoria principal	128MB DDR	2GB DDR2
Memoria secundaria	40GB	120GB

3.6.3.1.Comparativa de tiempos

La métrica utilizada en esta comparativa es el tiempo en segundos que tarda la plataforma en terminar las tareas establecidas. Las tareas están definidas en el proceso desarrollado, el código fuente del mismo se encuentra en el APÉNDICE III: Código fuente aplicación para medir el rendimiento.

Menos tiempo significa mayor rendimiento, en este estudio se esperaba inicialmente que la plataforma virtual fuera cuando menos igual de rápida que el sistema en la plataforma original, esto era de esperarse ya que la virtualización permite utilizar sistemas legados en equipo nuevo. Los tiempos encontrados en las pruebas de rendimiento no son significativamente diferentes, lo que válida la solución virtual como una alternativa a la solución original.

La siguiente gráfica muestra el tiempo utilizado por cada una de las plataformas, Windows 98 (Sin virtualización, esta es la plataforma original), Windows Vista (Con virtualización) y OpenSUSE 10.3 (Con virtualización).

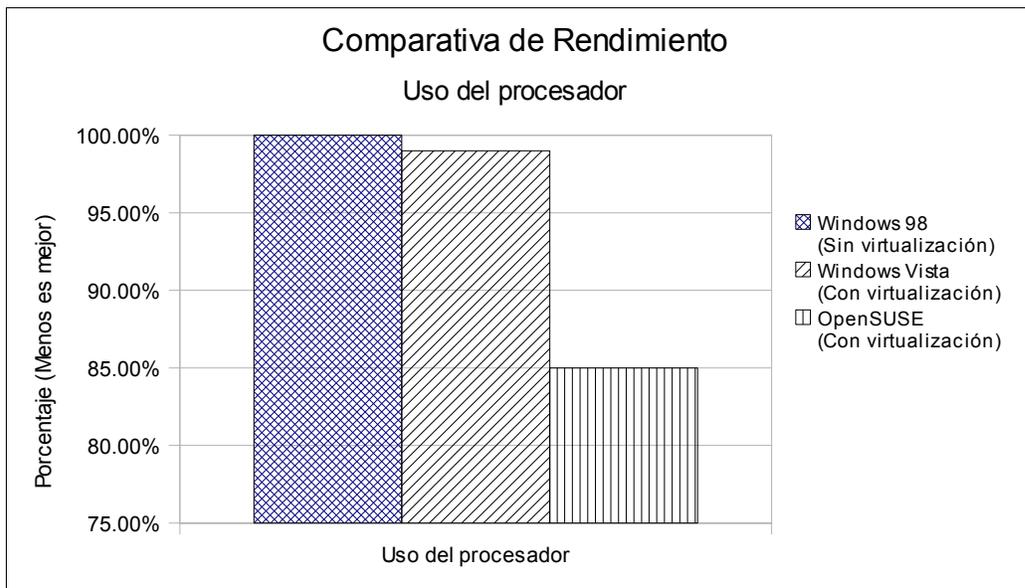


El tiempo transcurrido en los ambientes virtualizados es menor debido a la utilización de un equipo más moderno, sin embargo, la virtualización **no** aumenta el rendimiento de un sistema, lo que aumenta el rendimiento es que la virtualización permite utilizar equipos más modernos con sistemas legados.

3.6.3.2.Comparativa de uso del procesador

Otro beneficio de la virtualización es que se puede utilizar el equipo informático donde esta siendo usado el sistema legado, esto porque se puede controlar la cantidad de recursos que se le asigna a la máquina virtual.

En la siguiente prueba se midió el uso del procesador al realizar la prueba de rendimiento y así comparar que tanto estresa el sistema al equipo legado, se descubrió que la plataforma original toma el control total del equipo y lo vuelve inutilizable, en cambio las soluciones virtuales dejan un margen al procesador el cual puede aumentar o disminuir según los demás procesos en ejecución, en la siguiente gráfica se muestran los resultados obtenidos del uso del procesador en las pruebas de rendimiento:



En resumen la comparativa se muestra en la siguiente tabla:

Tabla XXVII. Comparativa de rendimiento

	Sin virtualización	Con virtualización	
	Plataforma original (Equipo antiguo)	en Windows	en OpenSUSE
Tiempo total (Menos es mejor)	20 segundos	13 segundos	13 segundos
Uso del procesador (Menos es mejor)	100%	99%	85%
Asignación de recursos	Fijos	Parametrizables	Parametrizables

La fila de Asignación de recursos muestra que en los ambientes virtuales se pueden asignar los recursos de forma parametrizable a la máquina virtual. El asignar únicamente los recursos necesarios permite controlar que tanto poder de procesamiento utilizará la máquina virtual en procesos intensivos. Lo anterior es muy importante en equipos modernos donde comúnmente se procesan varios servicios al mismo tiempo en una misma máquina. Por otro lado el hecho que se puedan asignar los recursos agrega un nivel de seguridad al sistema legado, lo que permite controlar de mejor manera el acceso de este a los dispositivos de disco así evitando que el sistema legado pueda corromper la información de otro usuario o de otro sistema.

Esto ultimo no era posible en la plataforma original ya que el sistema legado tenia acceso y permisos para utilizar todo el disco de almacenamiento.

Conclusión de la comparativa de tiempos: La solución virtual mejoro su tiempo en más del 10% lo que valida dicha solución como una alternativa a la solución original en cuestión de rendimiento.

CONCLUSIONES

1. De las herramientas virtuales gratuitas la más apta para virtualizar el sistema operativo D.O.S. es QEMU, como se muestra en la sección 3.4.
2. Es posible extender la vida útil de un sistema hecho en FoxPRO para D.O.S. utilizando la herramienta de virtualización QEMU, como se muestra en la sección 3.5.
3. Utilizando un equipo más moderno, la solución virtual aumentó su rendimiento en un 65%. Esto significa que la capa de virtualización no agrega un costo al rendimiento que impida su uso en aplicaciones empresariales.
4. La capa de virtualización permite gestionar de mejor manera los recursos, permitiendo asignar a cada aplicación legado la cantidad de memoria principal, secundaria y poder de procesamiento que utilizará, esto permite consolidar varias aplicaciones en una misma máquina, reduciendo así costos en equipo. Eliminando la necesidad de adquirir equipo de cómputo exclusivo para cada aplicación informática.

5. QEMU demostró ser una herramienta capaz en la virtualización del sistema D.O.S., siendo gratuito el costo de trasladar la aplicación legado a un nuevo equipo se elevó al pago en honorarios técnicos y de asesoría, el traslado se llevo a cabo en 8 horas.

RECOMENDACIONES

1. Aunque QEMU fue la máquina virtual utilizada en este estudio, no quiere decir que las otras sean menos funcionales que ésta, las otras son aptas para virtualizar otro tipo de sistemas. Por lo tanto, si se desea virtualizar un sistema distinto a D.O.S., se deben analizar las alternativas para encontrar la herramienta que cumpla con los requerimientos establecidos.
2. La virtualización de aplicaciones hechas con FoxPRO 2.6 es únicamente un caso para el uso de QEMU como herramienta para extender el tiempo de vida de una aplicación, existen incompatibilidades entre equipos modernos y otros sistemas legado, por lo cual se puede utilizar una capa de virtualización como QEMU para muchos de ellos.
3. Procesadores modernos como los vistos en la sección 1.4.3 son aptos para la virtualización, es importante considerar dichas alternativas para el aumento al rendimiento de las soluciones virtuales.
4. La virtualización de un sistema legado se puede utilizar como un paso intermedio en la migración a tecnologías más modernas. Ya que con el uso de una máquina virtual

puede coexistir en una misma máquina aplicaciones modernas y aplicaciones legado, aisladas por la capa de virtualización.

5. La virtualización de un sistema legado permite que las empresas seleccionen por cuenta propia el momento adecuado para modernizar sus procesos informáticos, de esta forma, el equipo de cómputo ya no es una limitante ni un factor decisivo, porque las herramientas de virtualización independizan las aplicaciones informáticas de dicho equipo.

REFERENCIAS

1. Nelson, R. A. *Mapping Devices and the M44 Data Processing System*. Centro de investigación IBM Thomas J. Watson, 1964, Reporte de desarrollo RC 1303.
2. Varian, Melinda. *VM and the VM community, past present, and future*. Universidad de Princeton [En línea], 1977 [Consulta: 5 oct. 2007]. SHARE 89 Sesión 9059-9061. Disponible en la dirección www.princeton.edu/~melinda
3. Adair, R.J.; Bayles, R.U.; Comeau, L.W.; Creasy, R.J. *A virtual machine system for the 360/40*. Centro científico IBM Cambridge, Cambridge, MA. Mayo 1966. Reporte 320-2007.
4. Comeau, L.W. *CP-40, the Origin of VM/370*. Procedimientos de SEAS AM82. Septiembre 1982.
5. Creasy, R. J. *The origin of the VM/370 time-sharing system*. IBM Journal of Research & Development: Revista de investigación y desarrollo de IBM. Vol. 25, No. 5 sept. 1981, pp. 483-90.
6. TIOBE, *IOBE Programming Community Index for October 2007*. TIOBE [en línea], 2007 [Consulta: 5 oct. 2007]. Disponible en la dirección: <http://www.tiobe.com/tpci.htm>

7. Wellie Chao, *The Pros and Cons of Virtual Machines in the Datacenter*. DevX [en línea]. 30 ene. 2006 [Consulta: 5 octubre 2007]. Disponible en la dirección: www.devx.com

8. Brunelle, David. *Installing Fedora Core on Microsoft Virtual PC 2004*. davidbrunelle.com [en línea] 23 sept. 2006 [Consulta: 5 oct. 2007]. Disponible en la dirección: <http://davidbrunelle.com/2006/09/23/installing-fedora-core-on-microsoft-virtual-pc-2004/>.

9. IBM, "Driving business value with a virtualized infrastructure", IBM, Marzo 2007.

BIBLIOGRAFÍA

1. Wikipedia. *Virtualization*. Wikipedia.org [en línea]. ca. 2007 [Consulta 8 oct. 2007]. Disponible en la dirección: <http://en.wikipedia.org/wiki/Virtualization>
2. Wikipedia. *x86 virtualization*. Wikipedia.org [en línea]. ca. 2007 [Consulta 8 oct. 2007]. Disponible en la dirección: http://en.wikipedia.org/wiki/X86_virtualization
3. Wikipedia. *Popok and Goldberg virtualization requirements*. Wikipedia.org [en línea]. ca. 2007 [Consulta 8 oct. 2007]. Disponible en la dirección: http://en.wikipedia.org/wiki/Popok_and_Goldberg_virtualization_requirements
4. Nelson, R. A. *Mapping Devices and the M44 Data Processing System*. Centro de investigación IBM Thomas J. Watson, 1964, Reporte de desarrollo RC 1303.
5. Varian, Melinda. *VM and the VM community, past present, and future*. Universidad de Princeton [En línea], 1977 [Consulta: 5 oct. 2007]. SHARE 89 Sesion 9059-9061. Disponible en la dirección www.princeton.edu/~melinda

6. Adair, R.J.; Bayles, R.U.; Comeau, L.W.; Creasy, R.J. *A virtual machine system for the 360/40*. Centro científico IBM Cambridge, Cambridge, MA. Mayo 1966. Reporte 320-2007.
7. Comeau, L.W. *CP-40, the Origin of VM/370*. Procedimientos de SEAS AM82. Septiembre 1982.
8. Creasy, R. J. *The origin of the VM/370 time-sharing system*. IBM Journal of Research & Development: Revista de investigación y desarrollo de IBM. Vol. 25, No. 5 sept. 1981, pp. 483–90.
9. TIOBE, *IOBE Programming Community Index for October 2007*. TIOBE [en línea], 2007 [Consulta: 5 oct. 2007]. Disponible en la dirección: <http://www.tiobe.com/tpci.htm>
10. Wellie Chao, *The Pros and Cons of Virtual Machines in the Datacenter*. DevX [en línea]. 30 ene. 2006 [Consulta: 5 octubre 2007]. Disponible en la dirección: www.devx.com
11. Brunelle, David. *Installing Fedora Core on Microsoft Virtual PC 2004*. davidbrunelle.com [en línea] 23 sept. 2006 [Consulta: 5 oct. 2007]. Disponible en la dirección: <http://davidbrunelle.com/2006/09/23/installing-fedora-core-on-microsoft-virtual-pc-2004/>.

12. Singh, Amit. *An Introduction to Virtualization*. Publicado en Kernelthread.com [en línea]. ca. 2006 [Consulta: 5 oct. 2007]. Disponible en <http://www.kernelthread.com/publications/virtualization/>
13. Smith, Roger D. *Simulation Article*. Model Benders [en línea]. 1998 [Consulta: 5 oct. 2007] Disponible en la dirección: <http://www.modelbenders.com/encyclopedia/encyclopedia.html>
14. Fayzullin, Marat. *How To Write a Computer Emulator*. Página personal [en línea]. 2000 [Consulta: 5 oct. 2007]. Disponible en la dirección <http://fms.komkon.org/EMUL8/HOWTO.html>

APÉNDICE

APÉNDICE I: Código fuente del filtro de impresión

En este apéndice se presenta el código fuente del programa utilizado para detectar los fines de páginas en la tubería realizada en OpenSUSE. Es un programa necesario en el proceso de impresión desde la máquina virtual.

Para generar el programa primero se debe crear el archivo con el nombre de **filtro.c** en la carpeta **\$HOME/spa/bin** y contener lo siguiente:

```
/* filtro.c */  
  
/* Nombre: Filtro de impresión */  
  
/* Trabajo: Estudio de virtualización */  
  
/* Autor: Luis Cano */  
  
/* ( c ) 2007 */  
  
#include <stdio.h>  
  
#include <stdlib.h>  
  
  
  
#define LINEAS_POR_PAGINA 80
```

```

int
main(int argc, char **argv)
{
    int contador = 0;

    char ch = 0;

    int lineas = LINEAS_POR_PAGINA;

    /* Viene algún parámetro? */

    if (argc > 1)
    {
        /* Es numérico? */

        lineas = atoi(argv[1]);

        if (lineas <= 0)
        {
            /* No es numérico, entonces se vuelve al número de
            líneas en caso de omisión */

                lineas = LINEAS_POR_PAGINA;

            }

        }

        while (!feof(stdin))
        {

```

```

        ch = getc(stdin);

/* Pregunta si es un salto de línea */
        if (ch == 0x0a)
        {

/* Si es un salto de línea lo cuenta */
                contador++;

/* Número de líneas por hoja, si es mayor o igual
termina el filtro */
                if (contador >= lineas)
                {
                        return EXIT_SUCCESS;
                }
        }

        fputc(ch, stdout);
}

return EXIT_SUCCESS;
}

```

Para compilarlo se deben instalar el paquete **gcc** y utilizar la línea de comandos:

```
cd $HOME/spa/bin
```

```
gcc filtro.c -o filtro
```

APÉNDICE II: Código fuente de la aplicación para validar la solución virtual

El siguiente código fuente es del programa valida.prg, se utilizó para validar las operaciones básicas de la máquina virtual.

```
* Inicialización del programa

clear

set talk off

set echo off

set status off

on error do fallido with error(),message()

* Inicio de las pruebas

?"Validación de la solución virtual"

?"Operaciones matemáticas simples"

do prueba_sum

do prueba_mul

?"Disco"

do prueba_dsk

?"Impresión"

do prueba_imp

return
```

```
procedure prueba_sum
```

```
    ?" Sumas"
```

```
    store 0 to I
```

```
    store 0 to D
```

```
    for I = 1 to 100
```

```
        D = D + I + (100-I)
```

```
    endfor
```

```
    if (D <> 10000) then
```

```
        ?" Fallido"
```

```
    else
```

```
        ?" Éxito"
```

```
    endif
```

```
return
```

```
procedure prueba_mul
```

```
    ?" Multiplicaciones"
```

```
    store 0 to I
```

```
    store 0 to D
```

```
    for I = 1 to 100
```

```

        D = D + I * (100-I)

    endfor

    if (D <> 166650) then

        ?" Fallido"

    else

        ?" Éxito"

    endif

return

procedure prueba_imp

    ?" Prepare la impresora y presiona cualquier
    tecla..."

    store col() to x

    store row() to y

    read

    @y,x say ""

    ???"Éxito de impresion"

    ?" Éxito en caso de haber impreso, fallo en
    caso contrario."

return

```

```

procedure prueba_dsk

    store 0 to I

    store 0 to archivo

    if (file('prueba.txt')) then

        archivo = fopen('prueba.txt',12)

    else

        archivo = fcreate('prueba.txt')

    endif

    if (archivo >= 0) then

        for I = 0 to 1000

            =fputs(archivo,str(I))

        endfor

        =fclose(archivo)

        if (.not. file('prueba.txt')) then

            ?" Fallo"

            return

        endif

        archivo = fopen('prueba.txt',12)

        if (archivo < 0) then

```

```

        ?" Fallo"

        return

    endif

    store space(80) to dato
    dato = fgets(archivo,80)
    =fclose(archivo)

    ?" Éxito"

else

    ?" Fallo"

endif

return

procedure fallido
parameters codigo,mensaje

    ?" Fallo: " + mensaje

return

```


APÉNDICE III: Código fuente aplicación para medir el rendimiento

El siguiente código fuente se realizó con el fin de medir el rendimiento en el sistema virtualizado y sin virtualización, está hecho en FoxPRO 2.6 y la intención del mismo es estresar el procesador y las operaciones a disco:

```
clear

set talk off

set echo off

set status off

store seconds() to inicio

store 0 to archivo

store 0 to I

store 0 to J

store 0 to H

store 0 to K

store 0 to L

if (file('rend.txt')) then
```

```

        archivo = fopen('rend.txt',12)
else
        archivo = fopen('rend.txt')
endif
for I = 1 to 100
        for J = 1 to 1000
* Estas son las tareas que estresan el procesador y
el disco:
                H = I * J * sqrt(I + J)
                K = H / 2
                L = H * sin(K) + H / (abs(K) + 1)
                =fwrite(archivo, str(H))
                =fwrite(archivo, str(K))
                =fwrite(archivo, str(L))
        endfor
        ?str(I)+"%"
endfor
=fclose(archivo)
store seconds() to fin
?"Tiempo de inicio: " + str(inicio)

```

```
? "Tiempo final: " + str(fin)
```

```
? "Tiempo total: " + str(fin - inicio)
```