



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

IMPLEMENTACIÓN DE UN SISTEMA PILOTO EN LA NUBE PARA EL APROVISIONAMIENTO DE MÁQUINAS VIRTUALES PARA EL LABORATORIO DE LA CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE INFORMACIÓN DE LA ESCUELA POLITÉCNICA NACIONAL – EPN

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y REDES DE INFORMACIÓN**

ESTEBAN JAVIER CALVACHE CUNALATA

stebancc_27@hotmail.es

SANTIAGO MARCELO FLORES NAVARRETE

santiago.floresnav@gmail.com

DIRECTOR: ING. PABLO WILLIAM HIDALGO LASCANO

pablo.hidalgo@epn.edu.ec

CODIRECTOR: ING. CATALINA IVANOVA AVILÉS BURBANO

caviles@ec.ibm.com

Quito, Abril 2016

DECLARACIÓN

Nosotros, Esteban Javier Calvache y Santiago Marcelo Flores, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Esteban Javier Calvache

Santiago Marcelo Flores

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por Esteban Javier Calvache y Santiago Marcelo Flores, bajo nuestra supervisión.

Ing. Pablo Hidalgo MSc.
DIRECTOR DEL PROYECTO

Ing. Catalina Avilés MSc.
CODIRECTOR DEL PROYECTO

AGRADECIMIENTOS

A nuestro Director de Proyecto, Ing. Pablo Hidalgo, quien de forma desinteresada y optimista nos apoyó para finalizar el presente Proyecto de Titulación, demostrando claramente que uno de sus propósitos de vida es ayudar a los demás por medio de la docencia.

A la familia de nuestro tutor de tesis, por permitir que el valioso tiempo familiar sea destinado para revisar proyectos de titulación, sabiendo que uno de los tesoros de esta vida es el tiempo, tiempo que jamás podrá ser recuperado.

A nuestra Codirector de Proyecto, Ing. Catalina Avilés, por su invaluable colaboración, predisposición y respaldo durante todo el proceso del presente Proyecto de Titulación.

Esteban Calvache

Santiago Flores

DEDICATORIA

A mis queridos padres por estar siempre a mi lado con su apoyo incondicional e infinito en todos los aspectos, son las personas que más amo en el mundo y es gracias a ellos el haber llegado a estas instancias de mi carrera universitaria.

A mis amigos y todas las personas que creyeron en mí por su aliento diario durante toda mi etapa estudiantil en esta gran universidad, fueron muy importantes para iniciar, continuar y finalizar una de las fases más hermosas de la vida.

Santiago Flores

DEDICATORIA

A mi familia, padres, hermanas, sobrinos y cuñado que son los principales pilares en mi vida personal, tengo la plena seguridad que siempre puedo contar con su apoyo y confianza incondicional.

En especial, a mi madre, Blanca Cunalata, y hermana, Natalie Guerra, que en cierto momento crítico durante el primer traspie en mi carrera universitaria supieron recargarme de energía, optimismo y valor para continuar con mis primeros pasos en esta gran universidad.

Esteban Calvache

CONTENIDO

DECLARACIÓN	II
CERTIFICACIÓN	III
AGRADECIMIENTOS	IV
DEDICATORIA.....	V
CONTENIDO.....	VII
ÍNDICE DE FIGURAS	XII
ÍNDICE DE TABLAS	XIV
RESUMEN	XV
PRESENTACIÓN	XVII
 CAPÍTULO I	
MARCO TEÓRICO.....	1
1.1. INTRODUCCIÓN	1
1.1.1. PERCEPCIÓN	2
1.1.2. MEMORIA.....	4
1.1.3. PROCESAMIENTO	6
1.1.4. LÓGICA	7
1.1.5. CONEXIÓN.....	11
1.1.6. ARQUITECTURA	14
1.2. VIRTUALIZACIÓN.....	17
1.2.1. RAÍCES DE LA VIRTUALIZACIÓN	17
1.2.2. CONCEPTO	18
1.2.3. TIPOS DE VIRTUALIZACIÓN	18
1.2.4. VIRTUALIZACIÓN DE SERVIDORES Y DE ESCRITORIOS.....	21
1.2.5. VIRTUALIZACIÓN PARA EL DESARROLLO DE APLICACIONES ..	24
1.3. COMPUTACIÓN EN LA NUBE, CLOUD COMPUTING.....	26
1.3.1. MODELOS DE CLOUD COMPUTING	29
1.3.1.1. IaaS	29
1.3.1.2. PaaS	30
1.3.1.3. SaaS	31
1.3.2. IMPLEMENTACIÓN DE CLOUD COMPUTING	32
1.3.2.1. La Nube Pública	33

1.3.2.2.	La Nube Privada	34
1.3.2.3.	La Nube Híbrida	35
1.3.3.	PASOS A SEGUIR HACIA IAAS	36
1.3.3.1.	Conocer el Ambiente Computacional Actual de la Empresa	36
1.3.3.2.	Educarse acerca del Valor de IaaS	36
1.3.3.3.	Experimentar con las opciones	37
1.3.3.4.	Investigar cuáles son los Requisitos de la Organización.....	37
1.3.3.5.	Preparar un Equipo de Evaluación.....	37
1.3.3.6.	Llevar a cabo un Proyecto Piloto.....	38
1.3.3.7.	Plan de una Estructura Organizacional bien Diseñada	38
1.3.3.8.	Evaluar constantemente los Estándares Emergentes.....	38
1.3.3.9.	Abordar el Continuo Ciclo de Desarrollo/Despliegue	38
1.3.3.10.	Convertir la Experiencia en una Hoja de Ruta Estratégica.....	39
1.4.	ANÁLISIS DE UN CASO DE ÉXITO DE IMPLEMENTACIÓN EN LA UNIVERSIDAD ESTATAL DE CAROLINA DEL NORTE	39
1.4.1.	CARACTERÍSTICAS Y FUNCIONES CLAVE DE VCL PARA LA EDUCACIÓN E INVESTIGACIÓN	42
1.4.2.	VCL EN APOYO DE LAS APLICACIONES DE ADMINISTRACIÓN UNIVERSITARIA Y DE NEGOCIO	48
1.4.3.	VCL, UNA PLATAFORMA DE GESTIÓN DE CLOUD COMPUTING DE PROPÓSITO GENERAL	49
1.4.4.	ALCANCE A LOS COLEGIOS DE LA COMUNIDAD Y K12	50
1.4.5.	OPORTUNIDADES DE DESARROLLO ECONÓMICO.....	51
1.5.	PROTOCOLOS Y MODELO DE RED UTILIZADOS.....	52
1.5.1.	MODELO ISO-OSI.....	53
1.5.1.1.	Capa Física	53
1.5.1.2.	Capa Enlace de datos	54
1.5.1.3.	Capa de Red	54
1.5.1.4.	Capa de Transporte	54
1.5.1.5.	Capa de Sesión.....	54
1.5.1.6.	Capa de Presentación.....	55
1.5.1.7.	Capa de Aplicación	55
1.5.2.	IP – INTERNET PROTOCOL	55
1.5.3.	TCP – TRANSMISSION CONTROL PROTOCOL.....	56

1.5.4.	UDP – USER DATAGRAM PROTOCOL	57
1.5.5.	HTTP – HYPERTEXT TRANSFER PROTOCOL (SECURE)	57
1.5.6.	RDP – REMOTE DESKTOP PROTOCOL	58
1.5.7.	SSH – SECURE SHELL	58
1.5.8.	NFS –NETWORK FILE SYSTEM	59

CAPÍTULO II

SOLUCIONES DE <i>SOFTWARE</i> Y REQUERIMIENTOS DE <i>SOFTWARE</i> Y		
<i>HARDWARE</i>		60
2.1.	SOLUCIONES DE APROVISIONAMIENTO	60
2.1.1.	HP CLOUDSYSTEM MATRIX	62
2.1.2.	BMC CLOUD LIFECYCLE MANAGEMENT	63
2.1.3.	VCE VBLOCK SYSTEMS	64
2.1.4.	IBM SERVICE DELIVERY MANAGER – ISDM	65
2.2.	DIFERENCIAS ENTRE LA SOLUCIÓN DE IBM Y LAS OTRAS	
MOSTRADAS		67
2.2.1.	IBM SERVICE DELIVERY MANAGER VS VCE VBLOCK	67
2.2.2.	IBM SERVICE DELIVERY MANAGER VS BMC CLOUD LIFECYCLE	
MANAGEMENT		69
2.2.3.	IBM SERVICE DELIVERY MANAGER VS HP CLOUDSYSTEM	
MATRIX		71
2.3.	ARQUITECTURA DE <i>SOFTWARE</i> Y <i>HARDWARE</i>	72
2.3.1.	COMPONENTES DEL ENTORNO ADMINISTRADO	73
2.3.1.1.	VMware ESXi	73
2.3.1.2.	Vmware VCenter Server	74
2.3.2.	COMPONENTES DEL ENTORNO DE ADMINISTRACIÓN	74
2.3.2.1.	Tivoli Service Automation Manager – TSAM	75
2.3.2.2.	Tivoli Monitoring – ITM	75
2.3.2.3.	Tivoli Usage and Accounting Manager – TUAM	76
2.3.2.4.	Network File System – NFS	76
2.3.3.	CONFIGURACIÓN DE LA SOLUCIÓN	77
2.3.3.1.	Configuración en alta disponibilidad	77
2.3.3.2.	Configuración <i>stand-alone</i>	82
2.4.	REQUERIMIENTOS Y DEFINICIÓN DEL PROBLEMA	83
2.4.1.	REQUERIMIENTOS DE INSTALACIÓN	86

2.4.1.1. Requerimientos de <i>Hardware</i>	86
2.4.1.2. Requerimientos de <i>Software</i>	88

CAPÍTULO III

DISEÑO, IMPLEMENTACIÓN DEL SISTEMA DE APROVISIONAMIENTO	90
3.1. DISEÑO	91
3.1.1. ARQUITECTURA	91
3.1.2. GESTIÓN DE SEGURIDAD	92
3.1.3. FLUJO DE TRABAJO	95
3.2. INSTALACIÓN DE LA SOLUCIÓN	97
3.2.1. COMPONENTE DE <i>HARDWARE</i>	98
3.2.2. COMPONENTE DE <i>SOFTWARE</i>	101
3.3. CONFIGURACIÓN DE LA SOLUCIÓN.....	105
3.3.1. IMPORTAR EL CERTIFICADO DE SEGURIDAD	106
3.3.2. CONFIGURACIÓN DE CORREO ELECTRÓNICO.....	107
3.3.3. CONFIGURACIÓN DEL ENTORNO ADMINISTRADO	108
3.3.4. DEFINICIÓN DE RECURSOS DE NUBE	112
3.3.5. CREACIÓN DE PLANTILLAS.....	113
3.3.6. CREACIÓN DE CATÁLOGO DE APLICACIONES.....	113

CAPÍTULO IV

PRUEBAS DEL SISTEMA DE APROVISIONAMIENTO Y COSTOS DE LA REALIZACIÓN DEL PROYECTO	115
4.1. ESTADO DE LOS SERVICIOS	115
4.1.1. INICIAR LOS SERVICIOS.....	115
4.1.2. VERIFICAR LOS SERVICIOS	117
4.2. PROCESO PARA PREPARAR UN DESPLIEGUE	126
4.3. REALIZAR UNA SOLICITUD DE APROVISIONAMIENTO.....	128
4.4. ANÁLISIS DE COSTOS DE REALIZACIÓN DEL PROYECTO	135
4.4.1. INGRESO DE INFORMACIÓN	135
4.4.1.1. Análisis de Costos.....	135
4.4.1.2. Análisis del Costo-Beneficio.....	137
4.4.2. ANÁLISIS DE RESULTADOS	139

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES	141
--------------------------------------	-----

5.1. CONCLUSIONES.....	141
5.2. RECOMENDACIONES	142
REFERENCIAS BIBLIOGRÁFICAS	145
ANEXOS	158

ÍNDICE DE FIGURAS

Figura 1.1: Soluciones VCL provistas por la IBM <i>SmartCloud for Education</i>	52
Figura 1.2: Representación del modelo ISO-OSI	53
Figura 1.3: Ejemplo de cabecera IP	56
Figura 1.4: Ejemplo de un segmento TCP.....	56
Figura 1.5: Ejemplo de un segmento UDP	57
Figura 2.1: Comparación desde el punto de vista del stack de cloud computing. .	69
Figura 2.2: Gestión de servicios de valor del cliente en la industria	71
Figura 2.3: Entornos de trabajo	73
Figura 2.4: Configuración en Alta Disponibilidad.....	81
Figura 2.5: Configuración <i>stand-alone</i>	82
Figura 3.1: Flujo de trabajo.....	96
Figura 3.2: Servidor IBM System x3500 M4	99
Figura 3.3: Disposición de entornos de trabajo	106
Figura 3.4: Sitio Web de la aplicación Thunderbird	108
Figura 3.5: Aplicación para importar los objetos DCM	112
Figura 3.6: Aplicación para importar los recursos virtuales de la nube	112
Figura 4.1: Ingreso de credenciales VMware vSphere Client.....	116
Figura 4.2: Encender el servidor virtual ITM.....	116
Figura 4.3: Activación de las máquinas virtuales.....	117
Figura 4.4: Servicio de la base de datos de ITM	118
Figura 4.5: Servicios del motor de monitoreo ITM.....	118
Figura 4.6: Estado del servidor web	118
Figura 4.7: Servicios del servidor NFS	119
Figura 4.8: Servicio web operativo	119
Figura 4.9: Servicio de la base de datos de TUAM	120
Figura 4.10: Servicio del servidor de aplicaciones de TUAM	120

Figura 4.11: Servicio del agente de monitoreo para S.O. Linux de TUAM	121
Figura 4.12: Portal web de la aplicación TUAM.....	121
Figura 4.13: Servicios de LDAP en ejecución	121
Figura 4.14: Servicio de la base de datos de TSAM	122
Figura 4.15: Servicio de la base de datos de LDAP	123
Figura 4.16: Servicio del servidor de aplicaciones de TSAM.....	123
Figura 4.17: Servicio del agente de monitoreo para S.O. Linux de TSAM	124
Figura 4.18: Portal de la aplicación TSAM	124
Figura 4.19: Servidores DNS y vCenter en ejecución	128
Figura 4.20: Utilización de recursos de la solución	128
Figura 4.21: Login - Portal de auto-servicio TSAM.....	128
Figura 4.22: Inicio - Portal de auto-servicio TSAM	129
Figura 4.23: Detalles del proyecto - Portal de auto-servicio TSAM	129
Figura 4.24: Plantilla - Portal de auto-servicio TSAM	130
Figura 4.25: Configuración de <i>hardware</i> - Portal de auto-servicio TSAM.....	130
Figura 4.26: <i>Software</i> Adicional - Portal de auto-servicio TSAM.....	131
Figura 4.27: Configuración de red - Portal de auto-servicio TSAM	131
Figura 4.28: Configuración adicional - Portal de auto-servicio TSAM	132
Figura 4.29: Resumen - Portal de auto-servicio TSAM	132
Figura 4.30: Estado de solicitud - Portal de auto-servicio TSAM.....	133
Figura 4.31: Mis proyectos - Portal de auto-servicio TSAM.....	133
Figura 4.32: Notificación de aprovisionamiento - Portal de auto-servicio TSAM	134
Figura 4.33: Proceso de aprovisionamiento en ejecución	134

ÍNDICE DE TABLAS

Tabla 2.1: ISDM vs VBLOCK	68
Tabla 2.2: ISDM vs BCM CLM	71
Tabla 2.3: Requerimientos de estación de trabajo	87
Tabla 2.4: Requerimientos de <i>hardware</i>	88
Tabla 3.1: Roles en ISDM	94
Tabla 4.1: Costos referenciales (No incluyen I.V.A.)	136

RESUMEN

En este proyecto se realiza la implementación de un sistema piloto en la nube para el aprovisionamiento de máquinas virtuales. Inicialmente, producto de este proyecto, el Laboratorio de Redes de la Carrera de Ingeniería en Electrónica y Redes de Información de la Escuela Politécnica Nacional – EPN, puede hacer uso de este sistema, logrando automatizar e incrementar sus capacidades para brindar infraestructura de TI a sus estudiantes.

En el primer capítulo, se definen los conceptos principales para una mejor comprensión de las tecnologías a utilizar. Adicionalmente, se menciona la evolución de dichas tecnologías desde sus inicios hasta las últimas tendencias. Se estudian tecnologías utilizadas por universidades extranjeras para la creación de un *Virtual Computing Lab* – VCL, para tomar buenas prácticas y evitar problemas; todo esto con el objetivo de implementar una solución informática en la nube, optimizada para la educación e investigación de la comunidad académica.

En el segundo capítulo, se establecen los requerimientos del Sistema Piloto de Aprovisionamiento a partir del dimensionamiento de los servidores y estaciones de trabajo. En lo relacionado con el *hardware* para el servidor, la primera alternativa considerada fue que la EPN adquiera o arriende *hardware*; como segunda alternativa y la que realmente se utilizó, es que los autores del Proyecto de Titulación dispusieron de *hardware* mínimo que fue utilizado de forma temporal durante el tiempo de la demostración y sustentación del mismo.

En el tercer capítulo, se implementa el Sistema Piloto de Aprovisionamiento en la Nube, el cual permite el suministro de máquinas virtuales y la utilización de las mismas en tiempo real. Este sistema es una solución de *cloud computing* que hace uso de las características y funcionalidades de una nube privada.

En el cuarto capítulo, se verifican los resultados que evidencien el correcto funcionamiento del sistema de aprovisionamiento, se hace un estudio objetivo de los costos para la implementación del proyecto piloto y se define el proceso para el acceso de los usuarios a la solución implementada.

En el quinto y último capítulo, se presentan las conclusiones y recomendaciones que se obtuvieron al concluir el Proyecto de Titulación desarrollado.

Adicionalmente, en los anexos, se documentan los temas relacionados a la tabla de especificaciones del servidor físico utilizado, proceso (*“step-by-step”*) de instalación, procesos de configuración y personalización de las diferentes funcionalidades y características que tiene a disposición el sistema de aprovisionamiento.

PRESENTACIÓN

El presente Proyecto de Titulación busca implementar un sistema piloto en la nube para el aprovisionamiento de máquinas virtuales para el Laboratorio de Redes de la Carrera de Ingeniería en Electrónica y Redes de Información de la Escuela Politécnica Nacional – EPN. Con este Proyecto la Carrera busca estar a la vanguardia en el ámbito tecnológico, ya que es imprescindible mantener actualizados sus servicios de infraestructura de TI.

Una de las últimas tendencias tecnológicas a nivel mundial es la migración de sistemas, aplicaciones e infraestructura hacia la denominada nube, ya sea ésta privada, pública o híbrida. Como en toda tecnología en auge existen varias empresas que están trabajando agresivamente para producir una solución que tenga la mejor aceptación en el mercado mundial; por lo cual, existen varios conceptos alrededor de esta tecnología que deberán ser necesariamente estandarizados.

Las universidades de primer mundo han migrado o están migrando en algunos casos, sus laboratorios a la nube, teniendo mucho éxito en sus implementaciones; con ello aumentan la accesibilidad a los recursos, y así despiertan mayor interés en la comunidad académica. Con esta implementación se pretende contar con tecnología de vanguardia así como servir a la comunidad politécnica con el fin de ser los pioneros entre las universidades nacionales en utilizar soluciones en la nube.

Sin duda actualmente la Escuela Politécnica Nacional – EPN dispone de soluciones tecnológicas de virtualización para la administración de sus laboratorios, pero no dispone de soluciones sobre la nube, que es la tendencia actual del mundo de tecnología; este proyecto facilitará y permitirá la interacción con este tipo de soluciones a autoridades, administradores y estudiantes.

Es de gran importancia la realización de este Proyecto de Titulación, ya que servirá como proyecto piloto para comprobar las ventajas de disponer de un sistema de aprovisionamiento, el mismo que abrirá las puertas para la investigación y utilización de más soluciones en la nube. También permitirá que

los estudiantes conozcan de soluciones que ayuden la fácil integración con el mundo de la nube.

La implementación del sistema de aprovisionamiento permitirá una fácil administración de los laboratorios, optimizando tiempo, dinero y recursos.

El catálogo de aplicaciones que se pueden implementar en el sistema de aprovisionamiento depende de las necesidades de la comunidad politécnica y los estándares requeridos por la solución.

El presente trabajo está orientado hacia profesionales en el ámbito de las Tecnologías de la Información y de la Comunicación - TIC, el mismo que puede ser utilizado como punto de partida para ofrecer el servicio de aprovisionamiento a toda la EPN e inclusive a otras universidades, colegios del país; todo esto con el fin de mejorar la educación y la investigación mancomunada.

CAPÍTULO 1

MARCO TEÓRICO

1.1. INTRODUCCIÓN

Las tecnologías de la información cambian el modo de pensar. Avanzan, mostrando lo que es posible lograr y liberan la mente para soñar que más se puede hacer. Los sueños inspiran a construir mejores tecnologías, que a su vez disparan nuevas ideas sobre más innovaciones tecnológicas, que vayan haciendo realidad los más impensables deseos.

No obstante, hay una parte que no se llegó a considerar como parte de la revolución informática, o quizá jamás se la pensó tan siquiera. Todos los componentes electrónicos – informáticos como sensores, dispositivos de almacenamiento, procesadores, *software* y redes, se han combinado para abstraer información y procesos mentales del cerebro humano. Así lo que se conoce e incluso las decisiones que se sabe cómo tomar, se capturan y se comparten. Todos estos hechos se combinan con datos sobre la naturaleza y la actividad humana, información que se ha empezado a recabar recientemente, en cantidades y velocidades sin precedentes, por medio de sensores y transacciones electrónicas.

Hoy la tecnología recopila el conocimiento mundial y la deposita en una fuente global para así formar un plano más amplio; la búsqueda en pos de lograr que las máquinas piensen, lejos de generar una parodia del pensamiento humano, ha producido un nuevo tipo de pensamiento.

Varios individuos de diferentes partes del mundo pueden tener acceso a la misma fuente de información, casi al mismo tiempo y trabajar conjuntamente gracias a la ayuda de las máquinas y redes. De esta manera se crea un enjambre de información, seres humanos y computadoras nutriéndose mutuamente. [1]

El objetivo de esta relación simbiótica es construir un mundo que trabaje mejor, creando permanentemente sistemas que mejoran la calidad de vida de los seres humanos, y el nivel de existencia del planeta.

Pero la historia de cómo se llegó hasta aquí nos lleva a 1911, con la invención de las rudimentarias máquinas de tarjetas perforadas. Se proyecta hacia el futuro con tecnologías que confieren a dispositivos manuales de fácil acceso, simulaciones de supercomputadoras mediante redes de cómputo en la nube, que permiten incorporar la información, la informática y las redes a la estructura de la vida y de los negocios. [2], [7], [11]

Para comprender los detalles de este proceso, se pueden examinar los avances que se dieron en seis pilares fundamentales de la evolución de las tecnologías de información.

- Percepción
- Memoria
- Procesamiento
- Lógica
- Conexión
- Arquitectura

Estos pilares se combinan y componen el entorno computacional que hoy se conoce. Las historias detrás del desarrollo de estos pilares destacan grandes investigadores de todo el mundo, que trabajaron en Bell Labs, Machines Bull, Cray Research, Intel, Xerox PARC, Sony, Apple, IBM y muchas otras compañías y entidades.

1.1.1. PERCEPCIÓN

Mucho antes de la palabra escrita, los seres humanos se comunicaban oralmente, alguien hablaba, y el otro percibía las ondas sonoras, las traducía en palabras e ideas entendiendo así el mensaje. Pero ¿por qué entonces ha sido tan complejo lograr que las computadoras hagan lo mismo?

Si bien el deseo de hacer que las máquinas comprendan el habla se remonta al siglo XX, durante décadas los científicos no tuvieron éxito en la materia. Hasta que en la feria Mundial de Seattle de 1963, IBM exhibió la máquina más avanzada de reconocimiento de voz del mundo, denominada Shoebox. Podía comprender palabras: cero, uno, dos, tres, cuatro, cinco, seis, siete, ocho, nueve, menos, más,

subtotal, total, falso y apagado. Esta máquina imprimía los valores de los cálculos elementales. [3]

Al principio las máquinas no podían detectar cuándo empezaba y cuándo terminaba una palabra, hasta que un día revisando las ondas sonoras en el osciloscopio, el investigador William Dersch y su equipo, descubrieron que las formas de onda sonora solían desfasarse unas con otras. Este fenómeno es imperceptible para el oído humano, pero este desfase se pudo traducir de tal forma que las computadoras empezaran a distinguir entre los diferentes sonidos del habla humana. Hacia 1980, se había alcanzado el hito de que las computadoras comprendieran 1000 palabras con una precisión del 91% y en 1984 ya podían reconocer hasta 5000 palabras. [83]

Hoy en día el reconocimiento de voz se ha vuelto omnipresente y totalmente preciso con un amplio vocabulario. Muchas líneas de atención al cliente suelen depender de sistemas de reconocimiento de voz. Los teléfonos inteligentes permiten a los usuarios utilizar términos de búsqueda, o ingresar una dirección que deseen ubicar en un mapa.

Los nuevos automóviles vienen incorporados con sistemas de reconocimiento de voz permitiéndole al conductor realizar llamadas, obtener ayuda de navegación, o cambiar la estación de radio sin necesidad de tocar tecla alguna. Aun así, a pesar de lo mucho que ha evolucionado el reconocimiento de voz, continúa siendo un gran desafío conseguir que las máquinas comprendan cabalmente el habla humana.

La tecnología introducida en la década de los 70 para manejar inventarios ofrecía una solución muy adecuada para los supermercados, ya que los sistemas *mainframe* y las bases de datos podían almacenar y clasificar gran cantidad de información sobre existencias y precios. Si hubiera un método más rápido de ingresar los datos en una computadora sobre cada producto que pasaba por las cajas registradoras, las tiendas podrían tener datos exactos sobre su inventario; así podrían modificar precios con tan solo ingresarlos en la base de datos. El principal problema era que no existía un sistema que pudiera leer los precios de

los miles de productos y trasladarlos a una computadora, después de todo, las computadoras no podían ver.

El sistema *Universal Product Code* (UPC) se convirtió en uno de los aportes más trascendentales a la tecnología industrial. Para los comerciantes, el código de barras se tradujo en un mayor ahorro, una mejor atención al cliente, un control de inventario preciso y un gran almacenamiento de datos comerciales. [4]

En la actualidad el sistema UPC es omnipresente y se utiliza para identificar absolutamente todo. El código de barras les confirió a las computadoras un modo de registrar datos, un sentido de la vista, aunque no similar al de los seres humanos, sino otro, propicio para las computadoras. Las computadoras no podían mirar un paquete de azúcar y reconocerlo como azúcar, pero sí podían leer una serie de líneas del paquete y saber que contenía azúcar.

Hoy en día, la tecnología de reconocimiento facial (sensores 3D, análisis de textura de piel, técnicas de correspondencia de plantillas) puede cotejar fotos de sospechosos con una base de fotos o videos e identificar las mismas caras mejor que cualquier persona.

En el pasado las máquinas contaban con una capacidad limitada para obtener información del mundo que las rodeaba, pero han ido adquiriendo una capacidad cada vez mayor para monitorear el funcionamiento del mundo a detalle, y analizar todos esos datos.

Su uso como interfaz informática aparece en celulares, tabletas, kioskos; en todos estos casos el ser humano depende de una tecnología multitáctil sofisticada e intuitiva para dar al dispositivo una orden. El modo de percepción más afectivo es aquel vínculo ente las personas y las máquinas que ofrezca esta conexión física, que le permita al ser humano “tocar” la información.

1.1.2. MEMORIA

El cerebro humano es el procesador más poderoso y grande sobre la tierra, pero no puede procesar nada sin memoria. De hecho necesita dos tipos de memoria a procesar, a largo y a corto plazo para funcionar eficientemente.

La memoria a corto plazo permite recordar por ejemplo, lo que se dijo en una conversación en curso, luego almacena los detalles más significativos en la memoria a largo plazo y desecha el resto. Se trata de la memoria activa, que recupera porciones de conocimiento para efectuar una tarea, y luego reestructura el nuevo conocimiento para la nueva tarea.

La memoria a largo plazo almacena todos los conocimientos, generalmente los reserva en segundo plano y los recupera cuando se los necesita, incluso cuando no.

En la Informática la unión perfecta entre el procesamiento y la memoria fue una larga lucha. Recién en el año 2000, cuando el costo de la tecnología de unidades de disco bajó notablemente, a la vez que aumentaba su capacidad, esto se hizo realidad. Desde fines de los 50 hasta 2010, la cantidad de información que se podía almacenar en una unidad de disco se multiplicó por 17 millones. [5]

Durante este mismo periodo la velocidad de lectura de datos aumentó 8 mil veces, mientras el costo de almacenamiento disminuyó a la millonésima parte. En los años 80 para almacenar un Gigabyte de información se requería de una máquina del tamaño de un refrigerador de 230 Kilogramos, en 2012 un Gigabyte entra en un disco más pequeño que una moneda. [84], [85]

Se ha hecho del almacenamiento una tarea simple para las máquinas y se ha consolidado al procesamiento de tal modo que se han podido crear bases de datos comerciales, simulacros de explosiones nucleares y películas generadas por computadora como Avatar. Y aun así, todas las unidades de disco necesitan mover un brazo para localizar los datos. Un terabyte de datos en disco contiene un único punto de acceso, y todo está canalizado a través de ese cabezal de escritura/lectura. Los procesadores se han vuelto tan veloces que por lo general deben esperar que se recuperen los datos del disco.

En el almacenamiento de estado sólido (Stuart Parking), cada bit puede leerse de manera instantánea sin tener que esperar al brazo de lectura/escritura. Los tiempos de arranque son de una fracción de segundo. Esta tecnología se la puede

evidenciar en un iPhone o en cualquier dispositivo pequeño que utilice una unidad *flash*. [6]

Parking descubrió cómo crear una columna alta (escala atómica) de material magnético dentro de un transistor, donde cada piso de este rascacielos microscópico representaría un bit de datos. La técnica utiliza el giro de los electrones para manipular esos bits; en concreto, los dispara sobre una pista de carrera hacia arriba y hacia abajo dentro de la columna. “Cada transistor podría almacenar no solo un bit, sino 100 bits” Stuart Parking (IBM Fellow), mayo 2010. De este modo puede tener el mismo costo bajo que en una unidad de disco, pero con un rendimiento 10 millones de veces mejor. [86], [87]

La memoria de vía rápida, o *racetrack*, en laboratorio funciona. Parking estima que se tardará algunos años en poder comercializarse. El tipo de almacenamiento masivo que hoy precisa una unidad de disco, para ese entonces cabrá en un chip del tamaño de una uña que apenas consuma energía.

1.1.3. PROCESAMIENTO

En esencia la computación no es más que el procesamiento de unos y ceros. Entonces el poder de la computación radica en acelerar este procesamiento.

Existen dos formas de aumentar significativamente la velocidad de las computadoras. Una consiste en reducir el tamaño de los componentes para que las señales eléctricas tarden menos en recorrer la máquina y, además, para poder compactar en un mismo sistema más chips que puedan efectuar más tareas.

Por otro lado se pueden utilizar (descubrir) nuevos materiales para construir los componentes. Muchas veces, un cambio de material ha permitido transportar pulsos de información a través de cables más delgados, en menos espacio o con menos energía, lo cual aporta mayor velocidad sistémica global.

Durante la década de 1990, por ejemplo, IBM construyó los primeros chips de cobre, que fueron fundamentales para mantener el ritmo de la Ley de Moore, justo cuando las tecnologías anteriores comenzaban a perder ímpetu. [88]

La labor incansable con superconductores, con los que, hacia los años 80, ya se venía trabajando desde décadas, se tradujo en un importante descubrimiento en el campo de la ciencia de los materiales. Los superconductores son conductores que no oponen resistencia, es decir se los podría considerar perfectos. Tienen características magnéticas que no se hallan en otro conductor; pero los científicos se encontraron con el problema de que los materiales demostraban superconductividad en el cero absoluto (-237°C), la cual sólo puede alcanzarse en el laboratorio. [6], [8], [9], [10]

Después de varios años de pruebas se encontraron materiales que demostraban superconductividad a temperaturas más altas (-203°C), las cuales eran más fáciles de conseguir debido a que lo podían hacer con nitrógeno líquido. Con esto se abrió un camino importantísimo que abrió el espacio para la fabricación de superconductores de alta temperatura que podían incorporarse en productos más prácticos. [14]

El funcionamiento del tren de levitación magnética de Shanghai, que supera los 480 Km por hora, depende de superconductores. Los superconductores aún no se han empleado en la computación, pero el debate sobre la velocidad de la década venidera se centrará en la computación a exaescala. Estos sistemas operarían a una velocidad de un trillón de cálculos por segundo, unas 500 veces más que la velocidad de las supercomputadoras más rápidas del año 2010. La necesidad cada vez mayor de información ha impulsado la demanda de velocidad, y esta velocidad nos deja cada vez más “sedientos” de información. Este ciclo no muestra signos de querer acabar, lo que siembra muchas incógnitas interesantes.

1.1.4. LÓGICA

Las computadoras revelan patrones –dado x, entonces y- al traducir el lenguaje binario de las máquinas a los lenguajes del pensamiento humano. Las primeras computadoras se programaban conectando y desconectando cables. Si un interruptor realizaba un cálculo determinado, el interruptor del otro extremo del cable realizaba otro cálculo relacionado. El lenguaje de la computación, en ese momento, era el equivalente a los gruñidos de un cavernícola.

En los primeros equipos, los programas consistían en tableros de conexiones, con cables distribuidos de tal manera que al conectar a los interruptores, realizaban los cálculos. Los programadores debían reordenar físicamente los alambres, para modificar las funciones de una computadora. Esta práctica cambió (evolució) y, con el tiempo, los programadores empezaron a ingresar códigos alfanuméricos, para decirle al computador qué hacer y dónde guardar la información. [9]

Pero este método de programación, tenía un problema muy simple: los programadores cometían muchos errores cuando reingresaban datos que ya se habían utilizado. Por ejemplo, si se utilizaba la letra delta para el espacio, y si un programador escribía la letra delta desprolija, entonces se podía confundir con un 4; también podía suceder entre la letra B y el 13, y así se pueden citar muchos ejemplos más.

Es así como nace la idea de un compilador: un programa que pudiera tomar las instrucciones comprensibles para el ser humano y traducirlas a un código que las computadoras entendieran. Esto permitió al programador escribir $x=7$, y no el código de máquina, para ingresar un 7 en el registro y luego enviarlo a una ubicación de memoria dada.

El traductor era muy ineficiente, de modo que el código resultante se ejecutaba mucho más lento de lo que habría tardado una persona capacitada en escribirlo. Aun así, esta labor en lo que se denominó “programación automática” marcó el inicio del progreso real en la materia de *software*, haciendo que las computadoras sean más asequibles, dándoles aún más valor.

A la par se creó otro “traductor” llamado *speedcoding*. Este lenguaje tomó en cuenta, las operaciones y cálculos más comunes, entre los programas; creando nuevas instrucciones para operaciones de más alto nivel, como el cálculo de la raíz cuadrada de un número, y permitió al programador ingresar código en lugar de lenguaje de máquina. [15]

En 1957, un grupo de investigadores de IBM, liderados por John Backus, generaron el primer lenguaje de alto nivel verdadero, Fortran (por sus siglas en inglés *Formula Translating System*). [78], [79]

En un principio se asemejaba al “compilador” inicial, pero generaba un código de máquina que resultaba tan eficiente como el de cualquier programador. Si bien para ejecutar un código en Fortran, se debía conocer bastante sobre computadoras, era la primera vez que un código era comprensible para todo aquel cuya experiencia estaba fuera de la computación, lo cual abrió la puerta de la programación a matemáticos, científicos, entre otros.

Por ejemplo, cualquier persona que supiera álgebra básica, pero nada de computación, podía probablemente leer enunciados en Fortran, como los siguientes:

Cálculo de las raíces de una ecuación de segundo grado:

$$R1 = (-B + \text{SQRT}(B^2 - 4*A*C)) / (2*A)$$

$$R2 = (-B - \text{SQRT}(B^2 - 4*A*C)) / (2*A)$$

Fortran inició el proceso de abstracción del *software* con respecto al *hardware* en el que se ejecutaba. Antes los programas debían escribirse específicamente para el computador en el que se ejecutaban, pero con Fortran podía ejecutarse en cualquier computadora que tuviese un compilador Fortran.

Dos años después, apareció la versión mejorada Fortran II, que incorporaba una herramienta maravillosa, conocida como la subrutina. Gran parte de la programación consiste en repetir lo que ya se ha hecho, lo cual se debe seguir repitiendo.

Una subrutina no es más que un subprograma independiente que se ejecuta desde dentro de otro programa. Las subrutinas tienen dos grandes ventajas: la primera radica en que el programador que la utiliza no necesita saber cómo funciona, sino más bien que resultados espera; la segunda es que una subrutina predeterminada puede incorporarse a un programa sin mayores esfuerzos.

Fortran, era muy similar a fórmulas algebraicas que usan científicos e ingenieros todos los días. Pero esta capacidad de escribir programas con símbolos matemáticos no impresionó a los encargados de las operaciones de

procesamiento de datos, quienes preferían palabras. Es así como nace COBOL (***Common Business-Oriented Language***) [90], un lenguaje abundado en palabras. Cualquier cálculo simple podía tener una instrucción como la siguiente:

MULTIPLY NUMBERONE BY NUMBERTWO GIVING ANSWER

Estas instrucciones tranquilizaban a administradores y gerentes y les daban la ilusión de que podían leer y entender los programas que sus empleados escribían.

Los años 70 y 80 aportaron dos tendencias muy contradictorias en materia de *software*. Por un lado las computadoras se volvían más estandarizadas y simples de programar, pero por otro lado el usuario promedio tenía cada vez menos necesidad de programar.

En los laboratorios Bell crearon el sistema operativo Unix y el lenguaje de programación C. Unix se afianzó en la informática académica y terminó siendo el sistema operativo estándar de las empresas medianas. C y su lenguaje derivado, C++, se volvieron el lenguaje oficial para la mayoría de la programación compleja.

En el Instituto Federal Suizo de Tecnología (ETH), se escribió el lenguaje Pascal específicamente como herramienta para enseñar programación. Paralelamente en USA se inventó BASIC (***Beginner's All-purpose Symbolic Instruction Code***), un lenguaje de programación simple diseñado para no especialistas. A medida que el acceso a las computadoras se extendía, la instrucción en BASIC formó parte habitual de los programas curriculares de las universidades y, poco a poco, de las escuelas secundarias y hasta primarias. [13]

Mientras tanto, los programadores experimentados, creaban aplicaciones para computadoras personales, modificando así los términos en que las personas consideraban el *software* y la programación. El programa de hojas de cálculo VisiCalc, publicado en 1979 para la Apple II, [91] permitió a los analistas crear modelos financieros sofisticados sin necesidad de escribir ni una sola línea de código. Así como también, programas de procesamiento de texto, como Word Perfect, dejaron obsoletos a las máquinas de escribir y sistemas de texto dedicados.

1.1.5. CONEXIÓN

En los años 60, las compañías de computadoras buscaban mantener la fidelidad de los usuarios a sus propios sistemas integrales. De hecho, la tendencia de la industria era tener 360 grados de funcionalidad independiente.

Pero ARPA (Agencia de Proyectos para la Investigación Avanzada), ansiaba poner a disposición de la mayor cantidad de científicos las pocas supercomputadoras existentes, con la esperanza de colocar a Estados Unidos en posición de ventaja con respecto a la Unión Soviética durante la Guerra Fría y la carrera espacial. Para esto la única estrategia razonable era juntar a los investigadores y a las escasas computadoras en red. [92]

Se adoptó una nueva tecnología radical en telecomunicaciones, conocida como conmutación de paquetes. Los datos se dividían en muchos pedacitos electrónicos denominados paquetes; y cada paquete contenía una dirección electrónica que podía ser leída por computadoras y conmutadores electrónicos.

A diferencia de las conversaciones telefónicas, la conmutación de paquetes no tenía que ocupar toda la línea de transmisión, ya que los paquetes podían dividirse y pasar por diferentes líneas y computadoras intermedias hasta llegar al destino, donde la computadora receptora reordenaba rápidamente los paquetes.

Ese sistema hacía uso eficiente de los recursos de la red y era resistente. Si un enemigo como la Unión Soviética lograba interceptar parte de la red, los paquetes eludían el camino afectado y llegaban a su destino sin problemas. Cualquier red de conmutación de paquetes podría sobrevivir a un ataque nuclear como no lo haría una red telefónica.

En 1970, la red ARPANET, salió a la luz. Se incorporaron más instrucciones y computadoras *mainframes*. Los científicos e ingenieros a lo largo y a lo ancho de los Estados Unidos tuvieron acceso a las computadoras más potentes de la Tierra. Así se instauró la idea de que la computación no era algo que debía apropiarse, sino compartirse.

Cuando se introdujeron las redes informáticas, cada red se comunicaba con un propio protocolo. Cada compañía que incursionaba en las redes, lo hacía con su propio protocolo, que desde ya era incompatible con el protocolo de la competencia.

Pero con el apoyo de ARPA, un grupo de científicos, elaboraron un conjunto de reglas conocido como protocolo de control de transmisión/protocolo de Internet, o TCP/IP. En 1983, su labor con TCP/IP permitió que la red ARPANET se conectara con una red académica llamada CS/NET, y así los investigadores de ambas redes compartan información.

El gobierno de USA por su parte trabajó junto a la Fundación Nacional de la Ciencia (NSF) de los Estados Unidos, para financiar la creación de centros de investigación, interconectados entre sí. A esta red se la llamo NSFNET y empleaba el conjunto de protocolos TCP/IP.

Pero Internet tenía varios adversarios, por ejemplo las compañías de telecomunicaciones veían a las redes informáticas como un competidor amenazante y se opusieron al financiamiento del gobierno.

IBM tampoco aceptaba demasiado a TCP/IP, ya que en 1974 la compañía había presentado la arquitectura SNA (Arquitectura de Sistemas de Red); y dada la prominencia de IBM en el sector, la arquitectura SNA transportaba más tráfico de datos que cualquier otra tecnología de redes de la época. Fue la tecnología SNA la manera en que las computadoras de IBM intercambiaban información durante más de 15 años, hasta que apareciera TCP/IP. [16]

A medida que más y más personas se sumaban al desarrollo de Internet, empezaron a encontrar formas de perfeccionarlo. Tim Berners-Lee, científico informático del CERN, laboratorio europeo de física de partículas, creó un modo de acceder a los documentos de Internet a través de un método de hipertexto, que conectó todas las páginas en una *World Wide Web*. [18]

En el centro de supercomputadoras de la Universidad de Illinois, Marc Andreessen y sus compañeros montaron el primer navegador gráfico, Mosaic, que

permitió a los usuarios navegar por la web haciendo clic sobre íconos en lugar de escribir cadenas de comandos.

Al mismo tiempo que esto sucedía, las computadoras personales causaban furor en las compañías. Por extraño que parezca, mientras había razones para conectar unas supercomputadoras; a nadie se le había ocurrido conectar muchas computadoras pequeñas dentro de un único edificio de oficinas. Se asumió la tarea de elaborar redes de área local, o LAN.

En el laboratorio de Zúrich de IBM, se desarrolló una tecnología llamada Token Ring. Se trataba de una forma de dirigir el tráfico de una red de área local, de modo tal que los mensajes no colisionaran para evitar el tráfico del sistema.

La tecnología de Token Ring introdujo las LAN al mundo corporativo –como los primeros sistemas de intranet, antes que se conociera la palabra-, pero la tecnología de IBM no prevaleció. [93]

En el famoso centro de investigación de Palo Alto, California, de Xerox, el ingeniero Robert Metcalfe perfeccionó una tecnología LAN denominada Ethernet, que terminó siendo más económica y veloz que la Token Ring. Enseguida Ethernet se convirtió en la red de área local estándar. [94]

Internet fue la clave para liberar la información de sus limitaciones físicas. Las personas podían estar en donde quisieran y encontrar lo que quisieran, de un modo automático e instantáneo. De todas formas, a finales de los 90, todavía quedaba una última limitación física sobre Internet: los cables.

La computación estaba destinada a volverse omnipresente, siempre activa y disponible en todo sistema que pudiera conectarse y comunicarse con cualquier otro. En los inicios del nuevo siglo, el acceso inalámbrico a Internet de alta velocidad -WiFi- llegó a los hogares, oficinas, aeropuertos, cafés y en algunas ciudades hasta a los parques. Los proveedores de telefonía móvil en el mundo, construyeron redes móviles inalámbricas de última generación, capaces de transmitir datos a altas velocidades y a bajo costo.

Este avance impulsó al desarrollo de teléfonos inteligentes como iPhone, Samsung (Galaxy, Sx), Blackberry, dispositivos que son básicamente computadoras conectadas al Internet, y que también pueden realizar llamadas telefónicas. Con esta simple conexión se tenía acceso a información, aplicaciones y capacidad de informática en cualquier sitio del planeta. [17]

La computación en todo momento y lugar se ha convertido en mucho más que una predicción hecha realidad. La tecnología de redes ha hecho que la computación sea parte de nuestro entorno, esta es la computación en nube, o *cloud computing*.

Hoy en día en esta nube, las máquinas conversan cada vez más con otras máquinas, no con seres humanos. En la próxima década, la conversación entre máquinas crecerá exponencialmente y superará la comunicación de datos que depende de la intervención del ser humano en uno o ambos extremos.

1.1.6. ARQUITECTURA

La arquitectura relata la historia de cómo los progresos tecnológicos se combinaron para crear nuevos sistemas, que a su vez hacen progresar nuestra percepción sobre la tecnología.

En un inicio las computadoras, eran de gran tamaño, y esto representaba ganancias suficientes para que las compañías estuvieran en la cúspide de los negocios. Las ventas se concentraban en grandes compañías, que necesitaban tener sus bases de datos, o manejar sus sistemas dentro de una computadora. Nadie avizoraba, la necesidad de las personas, o empresas pequeñas de una computadora, para sus cálculos, bases de datos, etc.

Primero se empezó con la idea del uso compartido de sistemas, sin mucho éxito. A medida que el uso compartido mejoró y se extendió, los terminales, similares a una máquina de escribir, hicieron a la Informática más ampliamente accesible. Hacia fines de 1968, unas 36 compañías prestaban servicios de tiempo compartido y operaban sistemas informáticos conectados a 10000 terminales ubicados en las instalaciones de los clientes. [79]

Aquellos que nunca antes habían tocado una computadora ahora podían utilizar una. Los terminales colmaron las fábricas, empresas pequeñas, entidades gubernamentales e incluso algunas escuelas. [20]

A comienzos de los 70, el club de madres de la escuela de Lakeside en Seattle, Washington, empleó lo recaudado de una feria para financiar un terminal y adquirir tiempo para que los estudiantes pudieran acceder a un sistema de tiempo compartido.

Este terminal atrajo a un alumno de octavo grado, muy delgado y con abundante cabello revuelto, llamado Bill Gates. Aprendió a programar en BASIC y, al poco tiempo, utilizó la máquina para escribir su primer programa de computadora: un juego de tres en raya. [95]

El término tiempo compartido finalmente dio lugar a la virtualización; en la década del 2000, el sector de la virtualización levantó vuelo y se volvió más sofisticado. Permite ahorrar tiempo y dinero porque posibilita el funcionamiento de una computadora prácticamente a su capacidad máxima, por lo cual las entidades pueden lograr más, con menor cantidad de máquinas.

La virtualización facilitó la aparición de *cloud computing*, al permitir que un conjunto de computadoras puedan ser compartidas entre millones de usuarios alrededor del planeta.

En 1974, Intel había preparado el camino hacia la miniaturización de las computadoras con su microprocesador 8080. Dos máquinas irrumpieron el mercado: TRS-80 de RadioShack y Apple II de Apple. [96]

VisiCalc, el primer programa de hojas de cálculo, apareció en 1979 y demostró que las computadoras personales podían ser útiles a los negocios. En 1981, IBM presentó su PC, hecha con partes de proveedores, empleaba el microprocesador 8088 de Intel y el sistema operativo PC-DOS de Microsoft. [98]

Un hecho importante, con la salida al mercado de la PC de IBM, dado que funcionaba con tecnología fabricada por proveedores externos, fue que otras

compañías podían clonar estas PC y hacer que sus copias funcionaran exactamente igual. Aparecieron cientos de desarrolladores de *software*: Ashton-Tate, Lotus Development, Satellite Systems International, etc. Prosperaron los fabricantes de clones, como Compaq Computer. [97]

Con esto, una oficina de ventas pequeña podía almacenar información sobre sus clientes en una base de datos. Todo resultaba novedoso y omnipotente. Estas máquinas no podían realizar las tareas complejas de una computadora *mainframe*, pero las personas que no precisaban realizar estas tareas atribuibles a un *mainframe* superaban ampliamente a las que sí lo hacían.

Rápidamente la computación se estaba volviendo parte de la vida cotidiana y personal, un cambio tan profundo como la primera ola de teléfonos, radios y televisores que inundó los hogares.

Esta arquitectura –esta idea de llevar la computación a los individuos- guió a la industria durante casi 30 años. Las computadoras personales continuaron ganando poder. Cuando surgió Internet, se conectaron a la web. Se descubrían actividades personales e interesantes para hacer con la computadora, como la creación de videos y música, lo cual hacía que se requieran máquinas más potentes y con mayor capacidad.

Entonces estas tendencias convergentes impulsaron la creación de *cloud computing*, y una verdadera avalancha de unidades remotas: dispositivos, sensores y aparatos.

A medida que la computación invadía cada uno de los aspectos de los negocios y la sociedad, el concepto de la información adoptó un nuevo significado y relevancia. La información ya no era algo que simplemente se podía ensamblar y comunicar, sino toda una entidad que debía estudiarse, diferenciarse y comprenderse.

La información contenía un valor que nunca antes se había percibido, y ahora una ciencia evolucionaba para descubrirlo. En la información se encuentran maneras

de que el mundo físico funcione mejor, de ahorrar dinero en los negocios, de construir cohetes espaciales y de comprimir un video dentro de un cable delgado.

La computación liberó el valor de la información dada su capacidad de realizar cálculos a gran velocidad, necesarios para revelar los secretos escondidos de la información. Pero al final, se cerró el ciclo, y la información se convirtió en la razón principal por la que resultaba tan valioso poseer y manejar una computadora.

1.2. VIRTUALIZACIÓN

A continuación, se presenta una breve evolución de las tecnologías de virtualización y los principales conceptos utilizados.

1.2.1. RAÍCES DE LA VIRTUALIZACIÓN

La evolución tecnológica siempre está incrementando el nivel de abstracción de arquitecturas de *hardware* y *software*. Lenguajes de programación de alto nivel como Fortran, COBOL, BASIC, C y Java han permitido a los programadores implementar *software* sin tener que prestar mucha atención a los sistemas operativos en los que van a ser implementados. Uno de los trabajos de los sistemas operativos, es proveer las abstracciones a los programas de los complejos y variados detalles necesarios para manejar elementos de memoria de entrada/salida (I/O).

Las aplicaciones de *software* contemporáneas, deben trabajar sobre el tiempo para determinar las características físicas del *hardware* donde se están ejecutando.

Aunque los paquetes de aplicaciones y *middleware* se han convertido intencionalmente en ignorantes de los detalles y configuraciones de implementaciones físicas específicas, los sistemas operativos que proveen este aislamiento deben estar totalmente conscientes del *hardware* en el que residen. Detalles como las direcciones IP, MAC, configuraciones de memoria física, procesadores y números de serie, están inmersos en el estado del sistema operativo en el momento de su instalación.

1.2.2. CONCEPTO

La virtualización es una tecnología probada de software que permite ejecutar múltiples sistemas operativos y aplicaciones simultáneamente en un mismo servidor.

En simples palabras, es el proceso mediante el que se crea una versión virtual, en lugar de una física. La virtualización se puede aplicar a computadoras, sistemas operativos, dispositivos de almacenamiento de información, aplicaciones o redes. Sin embargo, la virtualización de servidor es la parte más importante de este concepto

Gracias a la virtualización, se utiliza el *software* para simular la existencia del *hardware* y crear un sistema informático virtual. Esto permite que las empresas ejecuten más de un sistema virtual, además de múltiples sistemas operativos y aplicaciones, en un único servidor. De esta manera, se pueden ofrecer economías de escala y una mayor eficiencia. [22], [112]

1.2.3. TIPOS DE VIRTUALIZACIÓN

Existen diferentes enfoques frente a la virtualización, o tipos de virtualización, cada uno es adecuado a situaciones específicas. A continuación se describen los tipos de virtualización haciendo enfoque en: virtualización de *hardware*, virtualización de escritorios, virtualización de aplicaciones, virtualización de almacenamiento y virtualización de red.

1.2.3.1. Virtualización de *Hardware*

La Virtualización de *Hardware*, es el tipo de virtualización más común debido a las ventajas que ofrece en cuanto al uso del *hardware* y el tiempo de funcionamiento de una aplicación. Según VMWare, la mayoría de servidores funcionan a menos del 15% de su capacidad real, lo que causa la expansión de servidores y aumenta la complejidad. [112]

Normalmente un servidor dedica toda su capacidad de *hardware* (memoria, procesamiento, almacenamiento) a un solo sistema operativo. Cuando se

virtualiza el *hardware*, un *hypervisor* gestiona los recursos del *hardware* y los divide entre los diferentes sistemas operativos, conocidos como máquinas virtuales.

Otra de las ventajas de la virtualización de *hardware*, es que gracias al uso de tecnologías de virtualización, las máquinas virtuales en ejecución pueden ser migradas de un servidor a otro por actualizaciones, mantenimiento, etc., reduciendo los tiempos de inactividad de las aplicaciones.

Mediante esta “división” de *hardware* para múltiples sistemas operativos, se pueden tener muchas aplicaciones corriendo en el mismo servidor, sin encontrar problemas de compatibilidad. Además evita que cualquier fallo en una aplicación afecte a todas las aplicaciones que se ejecutan en el mismo servidor. [112], [113], [114], [115], [116]

1.2.3.2. Virtualización de Escritorios

La virtualización de escritorio separa el entorno de escritorio del dispositivo físico que se utiliza para acceder a él.

Esto se denomina más a menudo como “Infraestructura de Escritorio Virtual” *VDI*, donde muchos escritorios virtuales (Windows 10, RHEL, SUSE, etc.) se ejecutan como máquinas virtuales en un servidor, pero los usuarios pueden acceder a ellas haciendo uso de un cliente que normalmente es un computador personal.

La ventaja de la virtualización de escritorios es la comodidad de trabajo, seguridad de la información y la portabilidad. Debido a que a los escritorios se los puede acceder de forma remota, un usuario es capaz de trabajar desde cualquier lugar y desde cualquier computador personal.

Esto crea una gran flexibilidad para que los usuarios puedan trabajar desde casa, carretera, universidad, etc., sin necesidad de usar el mismo equipo físico en cada uno de estos lugares. También se protege, los datos importantes de una empresa, universidad de pérdida o robo, por mantenerla controlada en los servidores centrales. [112], [113], [114], [115], [116]

1.2.3.3. Virtualización de Aplicaciones

Es similar a la virtualización de escritorios, mencionada en el punto anterior, la virtualización de aplicaciones se diferencia en que sólo entrega una aplicación específica del servidor a un usuario o varios usuarios finales.

En lugar de iniciar sesión en un escritorio, el usuario interactúa únicamente con una aplicación como si fuese una aplicación nativa del dispositivo desde donde la accede. Esto hace a la virtualización de aplicaciones preferible para el trabajo desde tabletas o teléfonos inteligentes haciendo el trabajo mucho más fácil.

La ventaja de la virtualización de aplicaciones es la eficiencia. En muchas situaciones aplicaciones son muy útiles y además son redundantes entre los empleados, estudiantes, etc. Con esto en mente, ejecutar un sistema operativo para cada uno de los usuarios se convierte en innecesario cuando se puede ejecutar únicamente la aplicación a la que deben acceder. [112], [113], [114], [115], [116]

1.2.3.4. Virtualización de Almacenamiento

La virtualización de almacenamiento mejora la flexibilidad del almacenamiento, mediante la creación de un conjunto virtual de almacenamiento con los dispositivos de almacenamiento de la red. Lo que hace es presentar todo el almacenamiento físico en un clúster como un solo grupo compartido con todos los servidores.

La virtualización de almacenamiento permite la portabilidad de las máquinas virtuales, sin la necesidad de que estén conectadas a una matriz de almacenamiento compartido por lo general NAS o SAN (Network-Attached Storage, Storage-Area Network) si no que cada máquina conectada a la red contribuye a la agrupación de almacenamiento virtual como si fuese una NAS o SAN. [112], [113], [114], [115], [116], [117], [118]

1.2.3.5. Virtualización de Red

A veces referidas como redes definidas por *software*, al igual que en la virtualización de almacenamiento, la red reúne recursos de virtualización de todos

los equipos de red físicos y los presenta a las máquinas virtuales como una única red virtual.

Esto incrementa la agilidad de la red y reduce los tiempos de aprovisionamiento para las nuevas arquitecturas de red. Una vez que los equipos están conectados, configurar la comunicación de cada uno de ellos para que se comuniquen entre sí, se vuelve mucho más fácil, tanto manual como de manera automática usando plantillas.

Además, la virtualización de red mejora la visibilidad de la infraestructura de red para los equipos de TI ya que todo se maneja desde una plataforma virtual. También permite la mejor portabilidad de las máquinas virtuales (la arquitectura de red se conserva) y para implementar nuevas características de seguridad, como por ejemplo un cortafuegos distribuido (uno por máquina virtual). [112], [113], [114], [115], [116], [119], [120]

1.2.4. VIRTUALIZACIÓN DE SERVIDORES Y DE ESCRITORIOS

La virtualización es una forma de ejecutar varios sistemas operativos en una sola máquina, ya sea que las máquinas virtuales sean de diferentes sistemas operativos en conjunto, versiones del mismo sistema operativo a ensayar, o cualquier otra configuración que pueda imaginar.

Entre otras cosas la virtualización permite:

- Consolidar cargas de trabajo para reducir requerimientos de *hardware*, energía y espacio.
- Ejecutar varios sistemas operativos simultáneamente, como una forma de actualización de la empresa, o para aprovechar las ventajas de los sistemas operativos específicos, o por cualquier otra razón.
- Ejecutar nuevo *software*, más fiable, y *hardware* energéticamente eficiente.
- Migrar cargas de trabajo para brindar tolerancia a fallos.

- Proporcionar redundancias para soportar recuperación a desastres.
- Mejorar la eficiencia y rendimiento de los sistemas cliente servidor por igual.

El mercado de servidores x86 creció exponencialmente en la última década, impulsada en gran medida por una filosofía de "una aplicación, un servidor". Este enfoque llenó los Centros de Datos con una gran cantidad de racks de sistemas subutilizados, muchos operando a menos del 15 por ciento de su capacidad, pero el consumo de energía y generación de calor sobre una base 24/7. [111]

Incluso con bajas tasas de utilización, los administradores de TI a menudo tuvieron que dedicar tres sistemas separados para cada aplicación:

- Uno para ejecutar la aplicación.
- Uno para copia de seguridad del primer sistema, en caso de falla de *hardware*.
- Uno que sirva como plataforma de desarrollo continuo y análisis de problemas.

Estos sistemas suelen operar en una variedad de entornos de sistemas operativos actuales y anteriores, incluyendo Microsoft Windows (Vista, 7, 10, NT, Server 2000, Server 2008, Server 2012, etc.), Solaris, UNIX y Linux. A la mayoría de los gerentes de TI les encantaría consolidar cargas de trabajo diferentes en un número menor de sistemas de *hardware*, pero son comprensiblemente cautos de los problemas de *software* que pueden surgir, cuando se hace que varias aplicaciones independientes compartan una única instancia de un sistema operativo.

La virtualización proporciona un mecanismo para consolidar las aplicaciones, junto con los actuales SO, *middleware* y entornos de comunicaciones, en un único sistema compartido.

La virtualización toma los límites físicos de los servidores y los convierte en límites lógicos, en base a los recursos requeridos. Desde luego, se pueden tener servidores de uso específico, pero se puede hacer del tamaño que necesitan y asignar el resto de recursos a otras exigencias del Centro de Datos.

Al virtualizar los servidores, se pueden mover máquinas virtuales entre máquinas físicas y reconfigurar dinámicamente el Centro de Datos a medida que van cambiando las necesidades. En un entorno virtualizado, los recursos están localizados según la carga de trabajo o los horarios de mantenimiento, y no en cuartos de racks en los que están instalados.

Cuando se ejecuta una máquina virtual, cada carga de trabajo continúa ejecutándose en un entorno informático que corresponde exactamente a un entorno físico de su sistema dedicado, eliminando la necesidad de cambiar los parámetros del sistema operativo o de comunicaciones. En lo que se refiere al sistema operativo, solo ve el *hardware* con el que debe trabajar, entran cálculos y salen resultados.

El ahorro en las inversiones de *hardware*, energía y refrigeración se deben al hecho de que las máquinas virtuales están preparadas para responder a solicitudes externas, pero casi no consumen nada de recursos adicionales en ausencia de tales peticiones, muy lejos de los recursos consumidos en un ambiente real o no virtualizado.

Además de consolidar las cargas de trabajo existentes, la virtualización también facilita la introducción de nuevas aplicaciones al Centro de Datos. Después de que TI da luz verde a un nuevo proyecto, el desarrollo puede empezar inmediatamente en “flamantes” máquinas virtuales añadidas a un servidor físico existente. La virtualización esencialmente permite a la empresa basar sus adquisiciones de *hardware* sobre la demanda agregada, mas no en los caprichos de un programa determinado.

Aunque no es estrictamente el dominio de las tecnologías de virtualización en sí, la mayoría de entornos de *software* virtualizados incluyen utilidades que facilitan las tareas operativas como el aprovisionamiento de nuevos servidores virtuales, la asignación (reasignación) de los recursos virtuales, y la asignación de las máquinas virtuales en los sistemas físicos. Estas utilidades simplifican la programación de las interrupciones de servicio de *hardware* planificadas, así como la recuperación a partir de interrupciones no planificadas, mediante la

migración de máquinas virtuales a partir de un servidor físico a otro, sin incurrir en ninguna interrupción del servicio.

Esta capacidad se denomina a menudo como la migración en vivo. Con la migración en vivo, una máquina virtual se mueve instantáneamente de un servidor a otro, presentando una experiencia transparente para el usuario final y manteniendo las garantías de disponibilidad. Sin embargo, si los procesadores que se ejecutan en los ordenadores en el entorno previo y posterior a la migración no son iguales, la migración en vivo puede producir un comportamiento inesperado en el *software* del cliente.

Para dar cabida a las interrupciones planificadas, el departamento de TI puede trasladar las máquinas virtuales para que se ejecuten en una configuración de *hardware* similar sobre un sistema alternativo; esto permitirá que el *hardware* original pueda ser puesto fuera de línea para el servicio. Con las herramientas de migración en vivo, disponibles en muchos de los principales proveedores de virtualización, este traslado puede llevarse a cabo sin interrupción del servicio al usuario final, es decir esta migración es transparente para el usuario final.

Para cortes no planificados, el operador del sistema puede reiniciar las máquinas virtuales correspondientes en un sistema virtualizado diferente, hasta que en el sistema fallido se pueda restaurar el servicio. Además, algunas ofertas de virtualización, cuando se ejecutan junto con otras herramientas de gestión de sistemas, pueden permitir a un administrador del sistema identificar una falla de *hardware* (no planificada) y usar herramientas de migración en vivo; así se podrá migrar las máquinas virtuales del sistema que eminentemente fallará, hacia un sistema estable. En este caso, a pesar de que la interrupción es no planificada, habrá poca o ninguna perturbación al usuario final.

1.2.5. VIRTUALIZACIÓN PARA EL DESARROLLO DE APLICACIONES

La virtualización también está cambiando la forma en la que trabajan los desarrolladores de aplicaciones. Los desarrolladores a menudo deben adaptar su código de acuerdo al ambiente donde se va a ejecutar la aplicación. Para lograr

esto, usualmente se dedican máquinas de desarrollo específico, para diferentes versiones de Microsoft Windows, Linux, etc.

En el ejercicio de las anomalías de *software*, se encontraría con la máquina con el entorno operativo en el que se había observado el error y se trataría de llegar a una corrección. Por supuesto, si el sistema en particular con el entorno requerido no se ha utilizado durante algún tiempo, no habría la seguridad de que estaría en buenas condiciones de trabajo, cuando sea necesario.

La virtualización permite a las organizaciones de desarrollo, mantener una pila de máquinas virtuales correspondientes a todos los entornos de *hardware* y *software* específicos, en los que el diseño debe funcionar. Entonces, si necesitan hacer frente a una anomalía de *software*, cargan la máquina virtual apropiada, y se disponen a seguir el código problemático.

Esto vale para algo más que el desarrollo de *software*; muchos departamentos de TI soportan múltiples aplicaciones en diversas plataformas. Con la ayuda de la virtualización, es posible que un solo servidor haga el trabajo de múltiples servidores.

Si, por ejemplo, tienen que probar nuevas funcionalidades de aplicaciones o actualizaciones a través de una amplia gama de sistemas, los entornos virtuales pueden ahorrar *hardware*, espacio y otros recursos. A través de la virtualización, literalmente, miles de máquinas virtuales se pueden archivar, lo que permite probar rápidamente cualquier solicitud de revisión en cualquier ambiente que se necesite, en lugar de tener que reconstruir un conjunto específico de parámetros de *software* manualmente.

La virtualización facilita enormemente la tarea de migración de aplicaciones a las nuevas versiones de los sistemas operativos de cliente y aplicaciones de servidor. TI puede instalar varias máquinas virtuales, cada una ejecutando diferentes versiones de sistema operativo, y migrar las aplicaciones específicas al “nuevo y mejorado” sistema operativo, a un ritmo cómodo para todos, en lugar de la base todo o nada, que ha sido característica en las transiciones de *software* en el pasado.

La virtualización no se limita al servidor. La evolución del concepto de un escritorio virtualizado se evidencia por el *Virtual Desktop Infrastructure Alliance*, que ayuda a los administradores de TI a crear, administrar máquinas virtuales “de escritorio” en los servidores de Centros de Datos. [101]

Los usuarios finales pueden acceder a estos entornos de escritorio desde cualquier momento y en cualquier lugar, utilizando dispositivos de cliente ligero. Incluso los viejos sistemas de baja potencia pueden utilizar el Protocolo de Escritorio Remoto (RDP) para acceder a equipos más potentes. [102]

Este enfoque de la implementación del cliente puede reducir los gastos de apoyo, así como los costos de adquisición de *hardware*, ya que los PC virtuales residen en servidores centralizados en un entorno de TI administrado, eliminando la necesidad de visitar el sistema de escritorio real para la mayoría de las actividades de mantenimiento.

Por último, pero no menos importante, la virtualización desempeñará un papel cada vez mayor en la creación de entornos de TI, más seguros y robustos. En el lado del cliente, por ejemplo se pueden instalar dos máquinas virtuales, en un sistema cliente: una que trata los datos confidenciales de la empresa, y una segunda para las tareas del usuario final.

Se podría bloquear la primera máquina virtual para descargar aplicaciones no autorizadas, así como protectores de pantalla y otras amenazas a la seguridad, al tiempo que en el segundo acceso a la máquina virtual menos segura, se pueda descargar material y aplicaciones. Es importante tener en cuenta que, en este escenario, no hay manera de “extender” el acceso de una máquina virtual a otra.

Cuando se trata de seguridad del lado del servidor, la virtualización podría ser utilizada para hacer cumplir los requisitos de las normas de seguridad de la compañía, sin necesidad de *hardware* adicional, para mantener las piezas de su infraestructura de TI adecuadamente aisladas unas de otras.

1.3. COMPUTACIÓN EN LA NUBE, CLOUD COMPUTING

Cloud Computing, se ha convertido en una de las tendencias de transformación tecnológica más importante de la década en los negocios y la informática. Se está

presenciando el comienzo de la transición de una era, en que las empresas asumen que van a tener una infraestructura completa, para una época que está diseñada como un conjunto de servicios que son consumidos por los desarrolladores, socios y clientes.

La computación en la nube permite a las empresas responder a las iniciativas de negocio rápidamente, a menudo a un costo mucho más bajo, y sin las complejidades asociadas con los modelos computacionales que han estado en vigor desde hace décadas. [28]

También se puede innovar y transformar en una serie de servicios flexibles, que permiten un enfoque más ágil de lo que es posible, con los modelos de computación tradicionales. “La tecnología de la información se convierte en un servicio, que se consume de la misma manera que consumimos la electricidad o el agua”. [80]

La computación en nube (metáfora de la nube utilizada para representar Internet en los diagramas de red), es un sistema basado en Internet y Centros de Datos remotos, para proporcionar un conjunto de recursos informáticos compartidos, que incluyen las aplicaciones, plataformas de *software*, procesos de negocio, y demás servicios de información.

La gestión de estos servicios, se la hace sin necesidad de instalar aplicaciones en el medio que se está utilizando; es fácil de usar mediante el apoyo en un modelo de “auto-servicio” para que los recursos puedan ser gestionados, adquiridos o provisionados, según las necesidades o las reglas de negocio.

Una empresa, puede utilizar solo los recursos que necesita para completar una tarea, y luego de que esta tarea esté cumplida, los recursos vuelven al “pool” original.

De esta manera, esta tecnología ofrece un uso más eficiente de los recursos. Pero para esto, existen algunas tareas fundamentales necesarias para apoyar este modelo de computación, que incluyen las siguientes características:

- **Auto-aprovisionamiento y des-aprovisionamiento automático:** Forma automatizada para los usuarios de sumar y restar servicios basados en las necesidades del negocio.
- **Elasticidad y capacidad de escalar (mayor o menor recursos):** El usuario es capaz de adquirir más recursos, según lo demande, y pagar por la capacidad que necesite. Cuando ya no se necesitan esos recursos, el usuario puede devolver esa capacidad.
- **Agrupación de recursos:** La arquitectura *cloud*, permite la creación eficiente de grupos de recursos compartidos, que hacen de la nube una opción económicamente viable.
- **Medición del uso de servicio y facturación:** Dado que los clientes solo pagan por los recursos que compran o adquieren, los entornos *cloud* brindan una manera fácil de medir el uso de los recursos.

Igualmente importante es que la nube proporcione un nuevo modelo económico de computación. En lugar de comprar, administración y mantenimiento de un entorno de *software* y servidores, una empresa es capaz de comprar recursos de computación de manera circunstancial, evitando gastos de capital innecesarios.

Y si una empresa ya ha invertido en un entorno informático interno que se ajusta a los requerimientos del negocio, las partes de este entorno pueden transformarse en lo que se llama un entorno de nube privada, con las características de auto-servicio, como una nube pública.

Si bien en ambos casos la nube proporciona elasticidad, que permite al conjunto de recursos expandirse o contraerse, según la necesidad, existen algunas diferencias clave como estas:

- Una nube pública hace una fuente ilimitada de recursos, tales como aplicaciones y almacenamiento a disposición del público en general a través de Internet. Los servicios en la nube pública pueden ser gratuitos o se ofrecen en un modelo de pago por uso.

- Una nube privada, tiene un tamaño restringido al tamaño del entorno informático interno, construido por la empresa y que puede ser creado, gestionado y administrado por la propia empresa. El acceso a una nube privada está restringido a aquellos usuarios que tienen concedido el permiso.

Un factor importante de los servicios que proporciona una nube pública, es que el departamento de TI descarga la responsabilidad de mantener las aplicaciones o el servicio.

El proveedor de la nube pública se hace cargo de esa responsabilidad. Además, la organización puede agregar servicios cuando son necesarios. Este método ahorra tiempo, dinero y esfuerzo.

1.3.1. MODELOS DE CLOUD COMPUTING

Para comprender los beneficios de la computación en nube, es necesario comprender los servicios fundamentales. El servicio se define como una tarea que ha sido empaquetada, por lo que puede ser automatizada; además se puede entregar a los clientes de una manera consistente y repetible.

Cualquier servicio en la nube tiene un conjunto de características: un cliente puede acceder a la interfaz de auto-servicio y añadir más cómputo, *middleware*, base de datos, o en un servicio u otra aplicación. Cuando ya no es necesario el servicio, se suspende su uso, y solo se cobra por lo que se usó. Existen tres capas fundamentales de servicios en la nube: IaaS, PaaS, SaaS.

1.3.1.1. IaaS [104], [105]

Infraestructura como Servicio (IaaS), es uno de los servicios más directos de *Cloud Computing*. IaaS es la prestación de servicios de computación, incluyendo servidores, redes y almacenamiento, a partir de un modelo de alquiler. Esto difiere de los modelos de alojamiento tradicionales debido a que un servicio puede ser adquirido por un período de tiempo establecido. En otras palabras, el consumidor final termina pagando solo por el uso, en lugar de tener que firmar un contrato a largo plazo.

1.3.1.2. PaaS [104], [105]

No todo servicio necesita estar siempre disponible como *Software* como Servicio (SaaS). A veces, la organización de desarrollo tiene que construir soluciones personalizadas para satisfacer la demanda del negocio. Si bien los desarrolladores pueden utilizar una plataforma IaaS, para crear este tipo de aplicación, el enfoque no es muy productivo.

En IaaS, los desarrolladores son responsables de traer su propio *middleware* y herramientas de gestión de ciclo de vida de la plataforma, así como la creación de los sistemas operativos, *middleware* y de enrutamiento; además del mantenimiento y la aplicación de parches sobre el sistema personalizado.

Plataforma como Servicio (PaaS) se ha desarrollado para resolver el problema de tener que tratar con tanta complejidad, siendo un entorno integrado que apoya el desarrollo, ejecución y gestión de las aplicaciones basadas en la nube.

Proveedores de PaaS, crean un entorno gestionado, que reúne *middleware* integrado y servicios de desarrollo para apoyar a las organizaciones. Estas plataformas crean un ambiente abstraído que apoya la creación, implementación y administración de un entorno en nube. PaaS tiene las siguientes ventajas:

- PaaS es una infraestructura empaquetada completa, que puede ser utilizada para diseñar, implementar y desplegar aplicaciones y servicios en un entorno de nube pública o privada.
- PaaS permite a las empresas aprovechar los servicios de *middleware* clave, sin tener que lidiar con las complejidades de la gestión de los elementos individuales de *hardware* y *software*.
- PaaS requiere una pila completa de herramientas de desarrollo, que puede ser accedida a través de un servidor web (o cualquier herramienta especializada, tales como IDE – *Integrated Development Environment*) de una forma de auto-servicio.

Las organizaciones de operaciones y desarrollo, ahora tienen una plataforma bien organizada que apoya un enfoque de ciclo de vida coherente, con el seguimiento de los componentes necesarios para realizar un seguimiento de las configuraciones de *software* y actualizaciones de versiones.

Este enfoque ayuda a las organizaciones a gestionar, de manera constante, el proceso de desarrollo e implementación. A fin de proporcionar este nivel de abstracción, un número de componentes deben ser incluidos en una plataforma PaaS:

- Servicios de ejecución y desarrollo de aplicaciones.
- Servicios integrados de ciclo de vida.
- Servicios de gestión de carga de trabajo.
- Servicios de gestión de datos.

1.3.1.3. SaaS [104], [105]

Software como Servicio (SaaS), incluye aplicaciones de negocios especialmente diseñadas, que se ofrecen a través de la nube. Existen cientos de diferentes tipos de servicios, que van desde sistemas de gestión de relaciones con clientes (CRM), hasta aplicaciones de pago.

Pero, ¿cuáles son las características de un entorno SaaS? Una plataforma SaaS es lo suficientemente generalizada por lo que es útil para una amplia variedad de clientes. A diferencia de las aplicaciones que una empresa implementa internamente, un entorno SaaS se encarga de todos los detalles de desarrollo e implementación para el cliente.

La organización de TI no tiene que manejar el sistema operativo, bases de datos, *software* o actualizaciones de versión de la aplicación. Esta complejidad es manejada por el vendedor. Por lo tanto, el proveedor de SaaS incluye los IaaS, así como la Plataforma como Servicio, como parte del ambiente. Existen muchos

tipos de aplicaciones disponibles en el modelo SaaS. Entre las típicas, se incluyen las siguientes:

- Contabilidad
- Colaboración
- Gestión de proyectos
- Pruebas
- Análisis
- Gestión de contenidos
- *Marketing* en Internet
- Gestión de riesgos
- *Customer Relationship Management (CRM), Enterprise Asset Management (EAM), Enterprise Resource Planning (ERP), Enterprise Content Management (ECM).*

¿Qué no funciona como SaaS? Una aplicación especializada uno a uno, que apoya las necesidades de un usuario, no es económicamente viable, como un entorno SaaS. Debido a que los proveedores de SaaS suelen vender en un modelo de alquiler mensual o anual, sus productos deben ser fáciles de usar y de gestionar para el cliente final. Las aplicaciones SaaS deben ser diseñadas de manera modular, orientada al servicio.

1.3.2. IMPLEMENTACIÓN DE CLOUD COMPUTING

Después de revisar IaaS, PaaS y SaaS, es necesario entender cómo se puede utilizar uno de estos servicios para satisfacer las necesidades del negocio de una empresa. Los tres tipos de modelos de implementación son: Públicos, Privados e Híbridos (combinación de los dos).

Lo que separa a los tres modelos de nube, es quien posee en realidad los hosts, y opera los recursos de la nube. La ubicación también es otro factor importante. El tipo de despliegue a utilizar, depende de muchos factores, que van desde la reglamentación de la industria, a las necesidades específicas. A continuación se detallan los modelos de nube, para tener un mejor manejo en las opciones.

1.3.2.1. La Nube Pública [106], [107]

Cuando la gente empezó a hablar de la noción de la nube, hace varios años, la mayoría se refería a lo que ahora se conoce como nube pública. La nube pública es en realidad un conjunto de servidores, redes, almacenamiento, servicios e interfaces, operadas y de propiedad de una tercera parte, para ser utilizada por otras empresas que son accesibles desde cualquier lugar a través de Internet.

Además, debido a las economías de escala, los proveedores de administración de servicios en la nube, pueden proveer de personal con profundas habilidades de TI, en áreas especializadas tales como seguridad.

Por ejemplo, una empresa puede utilizar un servicio de nube pública para una carga de trabajo, tales como el correo electrónico (*e-mail*). El correo electrónico es un buen candidato para un servicio público, porque es una aplicación relativamente genérica, con un modelo de trabajo bien entendido.

Las empresas que se especializan en servicios de correo electrónico de nube pública, pueden optimizar su entorno de *hardware* y *software* para soportar este tipo de carga de trabajo y proporcionar diferentes niveles de seguridad por un precio. En realidad, estas empresas pueden ofrecer servicios de correo electrónico por una parte del precio que cuesta mantener y apoyar un servicio de correo interno.

Las empresas proveedoras de servicios en la nube manejan un concepto conocido como multiusuario, que significa que diferentes compañías comparten todos o algunos de los mismos recursos. Uno de los beneficios de esta plataforma es que añadir un nuevo usuario o cliente, se lo pueda hacer de una manera rápida, fácil y rentable, porque la nube no proporciona un conjunto fijo de recursos sólo para ese cliente.

Por supuesto, que el correo electrónico no es el único servicio que se puede poner en la nube. Otros ejemplos incluyen servicios de video, *streaming*, recursos humanos, servicios de copia de seguridad (por nombrar algunos).

Las organizaciones de desarrollo, pruebas o despliegue pueden tener períodos de alta demanda, seguidos por períodos menos activos; además que estos equipos a menudo se encuentran en diferentes geografías. Tener una plataforma optimizada, que sea escalable, basada en la demanda, es un modelo económico lógico para muchas organizaciones.

El desarrollo, pruebas y ejecución de las aplicaciones en la nube, puede resultar más económico que comprar y configurar servidores. Siempre y cuando el proveedor sea de confianza, las empresas verán los beneficios de la utilización de este tipo de servicios en la nube pública.

En el modelo de nube pública, el usuario final realmente no tiene que saber nada acerca de la tecnología subyacente. El proveedor de la nube está a cargo de todo el desarrollo, el mantenimiento del entorno y solución de problemas.

1.3.2.2. La Nube Privada [106], [107]

En algunas ocasiones, una plataforma pública no puede ser el medio más apropiado para una organización. La empresa puede ser parte de una industria altamente regulada y tiene que demostrar que su TI puede cumplir con las regulaciones.

Mientras algunas empresas usan la nube pública, puede ser necesario también un mayor control sobre una parte determinada de sus entornos de TI. La empresa necesitaría un servicio personalizado y especializado de seguridad, tener el ambiente de *hardware* listo, y un centro de “hospedaje” para la infraestructura y la solución; o simplemente utilizar los datos y aplicaciones en un entorno de datos existente.

Las empresas podrían adoptar lo que se llama una nube privada; pero, qué hace una nube privada diferente a un Centro de Datos que incluye algunos servidores de virtualización? A continuación algunas diferencias clave:

- Una nube privada, permite a las operaciones internas de TI asignar los recursos bajo demanda.

- La nube privada ofrece recursos internos de TI, en un modelo de prestación de servicios.
- La nube privada está diseñada para añadir la automatización de datos y aplicaciones existentes en un Centro de Datos.
- La nube privada administra las agrupaciones de recursos, incluyendo desde la capacidad de cómputo para el almacenamiento, análisis, gestión de procesos y *middleware*.
- Una nube privada implementa seguridad personalizada y la capacidad de gobierno.

1.3.2.3. La Nube Híbrida [106], [107]

La mayoría de las organizaciones que adoptan una estrategia de *Cloud Computing*, descubren que un enfoque híbrido se adapta bien a su estrategia de TI. Una nube híbrida es un entorno que utiliza tanto los servicios de una nube pública y privada para satisfacer una necesidad de la empresa.

Este entorno permite a un cliente hacer uso de una combinación de recursos, al mismo tiempo, a través de un conjunto consistente de servicios. Por ejemplo, un proceso de negocio puede copiar los resultados de una transacción comercial, calculados en un servicio público, hacia un ambiente privado.

Gracias a la automatización, este proceso puede ser ejecutado de acuerdo a las mejores prácticas y en cumplimiento con las normas requeridas. En algunas situaciones, un proceso puede ser establecido para que requiera datos desde una aplicación SaaS, que está físicamente alojada en una nube privada de otro país.

La nube híbrida es un servicio único, que se despliega de acuerdo a las necesidades de la empresa. Las herramientas de gestión deben estar bien diseñadas, y en el lugar correcto para asegurar que el entorno híbrido funcione bien, para de esta manera apoyar los objetivos de negocio de la empresa.

1.3.3. PASOS A SEGUIR HACIA IAAS [126], [127], [128], [129]

Parte de este proceso, para aprovechar la Infraestructura como Servicio (IaaS), requiere ir hacia atrás, y visualizar la planificación. A continuación se desarrollan diez pasos, para comenzar con IaaS.

1.3.3.1. Conocer el Ambiente Computacional Actual de la Empresa

Antes de planificar el futuro, se debe entender primero las situaciones pasadas y actuales, y hacer estas preguntas:

- ¿Qué tipo de entorno informático tiene actualmente?
- ¿Qué tan bien apoya su infraestructura a los negocios actuales, y lo bien que aportará el cambio?
- ¿Está en condiciones de vender servicios a los clientes o socios?
- ¿Tiene una estructura que le permite moverse rápidamente?

1.3.3.2. Educarse acerca del Valor de IaaS

Si se piensa en IaaS, es probable que tenga un poco de experiencia con *Cloud Computing*. De ser así, se puede abordar el siguiente paso a este “viaje” a través de la nube. Se quiere crear nuevas innovaciones que ayuden a la organización a ser más competitiva.

Aunque suena tentador y fácil entrar y empezar a comprar las herramientas y la tecnología para usar Infraestructura como un Servicio (IaaS), antes es necesario revisar estos consejos:

- Informarse sobre las opciones que existen.
- Analizar las diferentes opciones de los proveedores de IaaS.
- Entender cómo funciona cada plataforma, y las habilidades que va a tener que aprender.

1.3.3.3. Experimentar con las opciones

Muchas empresas no tienen experiencia directa en IaaS y podrían tomar decisiones sin los conocimientos necesarios. Una de las ventajas del modelo de la nube es que se puede experimentar con ofertas comerciales en el mercado, sin tener que gastar dinero en efectivo. Muchos vendedores, permiten que los desarrolladores puedan probar antes de comprar. Por lo tanto, se debe tomar el tiempo para experimentar.

Por ejemplo, algunas ofertas de IaaS están vinculadas a un *software* específico de modelo de servicio (SaaS). En algunas ocasiones, es posible que desee mantener IaaS separado del entorno SaaS. Con un poco de experimentación se tendrá una buena idea de lo que significa usar IaaS.

1.3.3.4. Investigar cuáles son los Requisitos de la Organización

Se debería crear un conjunto de requisitos, considerando las preocupaciones emergentes:

- ¿Qué idioma prefiere en su solución?
- ¿Cuánta abstracción se puede pedir?
- ¿Necesita una plataforma que automatice muchas de las tareas de rutina en el desarrollo e implementación?
- ¿Qué tipo de *middleware* ya está en uso y qué servicios *middleware* necesita?
- ¿Cuánta integración es lo que tiene que hacer con otros entornos públicos o privados?

Obtener las respuestas a estas preguntas, hace su proceso de selección mucho más fácil.

1.3.3.5. Preparar un Equipo de Evaluación

Un entorno IaaS, puede generar un cambio dramático en la forma en que la organización desarrolla *software*, por lo que no se debe tomar una decisión por

cuenta propia. Un entorno IaaS toca muchas partes de la empresa, por lo que se debería armar un equipo representativo de todos los componentes y los principales interesados, incluyendo a representantes de desarrollo, control de calidad, y operaciones.

1.3.3.6. Llevar a cabo un Proyecto Piloto

Identificar un proyecto que esté bien definido y que pueda mostrar a los equipos de operaciones y de desarrollo el apoyo de IaaS durante el desarrollo de un proyecto empresarial. Es posible que se desee probar varios entornos IaaS para que se pueda experimentar diferentes enfoques.

1.3.3.7. Plan de una Estructura Organizacional bien Diseñada

La tecnología es tan buena, como lo bien que la organización está preparada para utilizarla. El cambio es difícil y muchos desarrolladores y personal de operaciones pueden ser resistentes a probar un nuevo enfoque. Capacitar al equipo en la conmutación en nube y los beneficios que conlleva un entorno IaaS permite abrir un mundo diferente de hacer las cosas.

Muchos profesionales de TI están preocupados de que los entornos de nube, puedan hacer que sus habilidades y conocimientos queden obsoletos. Sin embargo, IaaS en realidad hace que exista un apoyo más eficaz en la empresa.

1.3.3.8. Evaluar constantemente los Estándares Emergentes

La computación en la nube, como una práctica comercial para las empresas está evolucionando, por lo que se debe conocer los estándares emergentes y seguir las mejores prácticas. Siguiendo estos enfoques, se seguirá siendo competitivo.

1.3.3.9. Abordar el Continuo Ciclo de Desarrollo/Despliegue

Las empresas exitosas entienden que después de desarrollar e implementar una aplicación a través de un entorno IaaS, ésta no ha terminado. Las aplicaciones se están tornando cada vez más dinámicas y más las aplicaciones en la nube, que están siendo actualizadas continuamente, con las últimas innovaciones.

1.3.3.10. Convertir la Experiencia en una Hoja de Ruta Estratégica

Trazar un plan de trabajo, con las cosas que se deben poner en práctica en los próximos años. Si se mira los pasos que se detallaron, se los debe utilizar como al comienzo de un viaje. No se deben tratar de hacer planes para diez años o más; hacer un *roadmap* de 3 a 5 años, ayuda a construir un plan significativo y realista.

1.4. ANÁLISIS DE UN CASO DE ÉXITO DE IMPLEMENTACIÓN EN LA UNIVERSIDAD ESTATAL DE CAROLINA DEL NORTE

A continuación se describe un caso de éxito de implementación de un Laboratorio de Computación Virtual, en la Universidad de Carolina del Norte, Estados Unidos de América. El mismo fue tomado del artículo “The Virtual Computing Lab (VCL): An Open Source Cloud Computing Solution Designed Specifically for Education and Research”. [109], [110], [111]

Con este ejemplo de implementación exitosa, la Facultad de Ingeniería Eléctrica y Electrónica y en especial la Carrera de Electrónica y Redes de Información pueden validar todos los beneficios que una de estas implementaciones traería a la carrera, a la facultad y a la universidad.

Esta sección se basa en describir las necesidades que tenía la universidad de Carolina del Norte, lo que implica la implementación de un VCL -*Virtual Computing Lab*, las características y funciones claves para la educación y la investigación, el apoyo que da un VCL a las aplicaciones de administración y de negocio, el alcance que se puede tener (colegios, otras universidades) y como se puede tener un desarrollo económico para la carrera.

El Laboratorio de Computación Virtual (VCL por sus siglas en inglés) representa una verdadera solución de computación en la nube (*Cloud Computing*), que ha sido optimizada para las necesidades educativas y de investigación de la comunidad académica.

Inició en 2002, con el primer sistema de producción puesto en marcha en 2004 por la Universidad Estatal de Carolina del Norte con el apoyo del Centro de

Estudios Avanzados de IBM, relaciones corporativas universitarias de IBM, y el equipo de desarrollo de *hardware* de IBM. [30]

Se ha mejorado de forma continua durante más de ocho años por lo tanto es extremadamente estable en entornos de producción. En 2012, la nube VCL de Carolina del Norte (NC VCL) entregó varios cientos de miles de sesiones de usuarios a más de un cuarto de millón de estudiantes y profesores de Carolina del Norte.

Originalmente desarrollada como el sistema en nube para la educación e investigación de la Universidad Estatal de Carolina del Norte, ahora soporta a estudiantes de otras universidades con el sistema de la Universidad de Carolina del Norte y en algunas escuelas mediante el sistema "*North Carolina Community College System*".

También hay varios pilotos de extensión K-12 en todo el estado. Nubes de educación similares se pueden encontrar en California, Virginia, Georgia, Tennessee y otros estados, así como dentro de la comunidad HBCU (*Historically Black Colleges and Universities*) como parte de la iniciativa Nube HBCU.

También se pueden encontrar nubes de educación basadas en VCL en Europa, Oriente Medio, India, China y Japón. Su difusión se aceleró cuando VCL se convirtió en un proyecto de código abierto Apache en 2009.

VCL ha estado en operación confiable en Carolina del Norte desde 2004, sin fallos de seguridad importantes para ese tiempo. Hoy ha escalado a más de 2000 servidores, que soportan más de 1000 imágenes de *software* en su biblioteca.

Ha escalado fácilmente a este tamaño a través de los años, llegando a ser más eficiente conforme el número de usuarios crece, lo que permite alinear juiciosamente los picos y valles de las necesidades de recursos de una larga y heterogénea población de usuarios, mientras que se maximiza la utilización y a la vez se reduce al mínimo los costes de adquisición de recursos de *hardware* y *software*.

Todo esto se ha logrado con no más de dos equivalentes de tiempo completo (*Full Time Equivalents FTE*) que gestionan el sistema. Teniendo en cuenta la eficiencia energética de sus servidores elegidos, la nube del estado de Carolina del Norte está siendo operada a solo unos centavos por hora de computación. La mayoría de universidades han adoptado VCL, experimentando una reducción total a sus costos de un 50 a un 80 por ciento o más.

VCL puede desplegar un amplio rango de soluciones, desde *clusters* de servidores complejos, a bloques de máquinas instaladas con todo el *software*, aplicaciones o *middleware* necesario para una clase regular de estudiantes, a una máquina física y escritorios virtuales.

Como tal, ha sido utilizada para apoyar la computación de alto rendimiento (HPC) de los investigadores, estudiantes e instructores de clase, tanto dentro como fuera del aula; administradores universitarios que soportan aplicaciones de negocio, e incluso la creación de empresas con el apoyo de programas y consultores de la universidad, de valor incalculable para el desarrollo económico regional.

Se puede apoyar todo esto porque VCL es una verdadera pila de gestión de la nube, con un portal de auto-servicio que actualmente soporta un gran número de soluciones de nube común, comercial, o nubes relacionadas a servicios. Esto incluye soluciones de *hardware* e Infraestructura como un servicio (IaaS), Plataforma como servicio (PaaS), *Software* como servicio (SaaS), e incluso nube como servicio (CaaS) capaz de ejecutar una VCL completa.

Puede apoyar la implementación de servidores "*bare-metal*" usando una solución como *eXtreme Cloud Administration Toolkit* (xCAT) que puede, a su vez, ser usada para crear un gran grupo de servidores para procesar tareas, con el apropiado almacenamiento. Servidores "*bare-metal*" es un sistema de virtualización que no requiere de un sistema operativo anfitrión, y se instala directamente sobre el *hardware*, con lo cual se crea una interfaz directa hacia los recursos de *hardware* aprovechándolos al máximo.

También puede soportar una gran cantidad de *hypervisors* estándar para la implementación de máquinas virtuales, incluyendo el *hypervisor* KVM (*Kernel-*

based Virtual Machine), como también muchas de las soluciones de Infraestructura de Escritorio Virtual (VDI).

VCL puede desplegar varias soluciones IaaS disponibles en el mercado, como la solución *IBM SmartCloud Provisioning, Open Stack*, y así sucesivamente, o PaaS como las soluciones *IBM Workload Deployer (IWD)*. También permite acceder fácilmente a los recursos de nubes comerciales existentes, como *SmartCloud*, creando una solución de nube híbrida público-privada.

VCL fue diseñado de manera flexible desde el principio para dar cabida a todos los nuevos paradigmas y soluciones que podrían surgir, y modularmente construido para aceptar nuevos *plug-ins* que pueden orquestar varias soluciones a través del estándar definido para las interfaces de programación de aplicaciones (API), típicamente definido para todo.

1.4.1. CARACTERÍSTICAS Y FUNCIONES CLAVE DE VCL PARA LA EDUCACIÓN E INVESTIGACIÓN

VCL tiene muchas características y funciones diseñadas específicamente para la comunidad educativa y de investigación, incluidas las reservas de bloque para regularmente repetir clases; capacidad para los docentes, asistentes de cátedra e investigación y otros para construir y almacenar investigaciones, material de clase que puede ser desplegado en un servidor físico o en máquinas virtuales.

VCL es también extremadamente estable, su costo de mantenimiento y operación en producción es muy bajo. Es libre como una solución de código abierto Apache, aunque requiere una inversión, ya sea mediante el desarrollo de conocimiento del personal de TI o mediante la contratación de servicios de apoyo.

Cualquier sistema de gestión de identidades de cualquier universidad, se puede simplemente conectar al VCL, quien automáticamente gestiona las licencias y proporciona una gran cantidad de estadísticas del sistema y las métricas que se pueden utilizar para la facturación y medición, capacidad de planificación (incluyendo los planes de compra de licencias), y así sucesivamente.

Dado que es de código abierto, VCL se puede utilizar por los educadores para enseñar acerca de la computación en nube, ya que su funcionamiento y su diseño se encuentran abiertos a todos.

Además el número de comunidades de apoyo han ido creciendo con los años, y se reúnen periódicamente para: compartir buenas prácticas en recursos como bibliotecas de imágenes, proporcionando asistencia mutua en la solución de problemas, aprovechando oportunidades de financiación conjunta, y otras funciones.

El Laboratorio de Computación Virtual administrado por la Universidad Estatal de Carolina del Norte, puede ser utilizado por cualquier usuario utilizando un nombre de usuario y contraseña válidos en <http://vcl.ncsu.edu>. [81]

Técnicamente todos los estudiantes de la Universidad de Carolina del Norte con credenciales Shibboleth (aproximadamente 250000) tienen acceso a ella. Un estudiante o educador puede acceder al sistema a solicitar que uno o más bienes (carga *bare-metal*) o máquinas virtuales sean entregadas inmediatamente, o en un tiempo posterior programado. [32]

Una imagen representa un sistema operativo, típicamente Windows o Linux, que ejecuta una o más aplicaciones de *software* o *middleware* y que puede seleccionarse para ser ejecutada en una máquina escogida.

La página web se puede acceder desde cualquier dispositivo o máquina que pueda ejecutar un navegador web, proporcionando un uso completo de todos los recursos solicitados. La interfaz es idéntica a la que los usuarios podrían experimentar si las máquinas estuvieran en su escritorio, incluso permitiendo a los usuarios instalar otro *software* en las máquinas entregadas.

Cuando el usuario haya terminado, los recursos pueden ser desmontados y devueltos al *pool* general para ser asignados de nuevo al siguiente usuario. Los recursos solo se utilizan cuando se necesitan, lo que maximiza la utilización de licencias de *software* y *hardware*, permitiendo reducir al mínimo los costos de energía y requisitos de compra. El acceso a esos recursos a través de Internet,

significa que estarán disponibles durante todo el tiempo, desde un aula, residencia o casa.

Al hacer una petición al NC VCL, cada institución educativa puede acceder a su propio “*portlet*” con un menú desplegable que está personalizado para que aparezca como una página web para esa escuela específica.

El control de acceso puede estar vinculado a LDAP (*Lightweight Directory Access Protocol*), Shibboleth u otro sistema de inicio de sesión. El VCL gestiona adecuadamente los acuerdos de licencia de *software*, limitando el acceso simultáneo a las licencias en caso de ser necesario.

El acceso de grupos al *software*, puede ser gestionado fácilmente con políticas, tales como lista de registro de clase. Se pueden recoger estadísticas de uso relevante, que se usan para la medición y facturación interna, así como la planificación de la capacidad, y otros.

El VCL soporta una amplia gama de usuarios como investigadores, docentes, estudiantes. Para apoyar el trabajo de cálculo intensivo, un investigador puede acceder a la página web de VCL y solicitar un clúster de máquinas, posiblemente con una *suite* de *software* diferente en cada máquina.

Usando la misma infraestructura, un profesor puede solicitar 20 o más máquinas para una clase de 20 estudiantes. Se puede ejecutar una imagen especial en cada una de las máquinas con el *software* necesario para esa clase. Las máquinas pueden ser entregadas por tiempos limitados, en días específicos (mientras dure una clase).

De la misma manera una maestra de colegio puede pre-programar máquinas para que funcionen durante algunas materias, donde el profesor pueda abrir sesiones que usen muy pocos recursos para acceder al material de la clase.

VCL brinda apoyo a la comunidad educativa proporcionando:

- Una o más máquinas, ya sea como físicas o virtuales, individuales o como *clusters* de servidores con el *software* necesario para laboratorios en clase o demostraciones educativas.

- Cualquier número de máquinas, para cualquier número de estudiantes en una clase regular, con cualquier *software* necesario para esa clase instalado en cada máquina, todos entregados cada vez que se reúne la clase; esto puede ser programado por adelantado por el profesor para que se lleve a cabo durante todo el semestre.
- Una *suite* de *software* educativo en una máquina física o virtual en casa, en el dormitorio o en el salón de clase, que se puede utilizar para la enseñanza dentro y fuera de clase, proyectos y tareas.
- Telepresencia y aplicaciones de educación a distancia, incluyendo educación continua y educación a personal militar estacionado en sus bases.

VCL puede también proporcionar simultáneamente soporte HPC – *High Performance Computing* para la comunidad de investigadores. Esta capacidad permite a la nube VCL soportar las necesidades de todas las comunidades en un campus, con la misma infraestructura *cloud* simple, con un enorme ahorro de costes. Esto puede incluir:

- Grupos de máquinas ya sean débilmente o fuertemente acoplados, servidores *blade* totalmente poblados, con o sin el *software* pre-instalado en el clúster de las máquinas. Un ejemplo del primer caso es un clúster de servidores web de alta disponibilidad, con cada máquina instalada correctamente con el servidor web y el *software* de base de datos, para la investigación relacionada. Un ejemplo de esto último es HaaS (*Hardware-as-a-Service*)
- Cualquier número de máquinas individuales, con una gama de especificaciones de *hardware*, tales como la capacidad de cómputo, memoria, almacenamiento, redes, etc.
- Mini-nubes reales: por ejemplo, nubes dentro de una nube, sandbox, etc.

Con las especificaciones mencionadas los servidores de la nube de NC VCL suelen dividirse en dos grandes *pools*, uno de apoyo a las necesidades de la

comunidad de investigación de HPC, y otro de apoyo a la educación, la investigación, y otras necesidades de la población universitaria general. [34]

Los servicios en cada uno de los grupos tienen pequeñas diferencias en configuraciones de red y físicas, junto con diferentes políticas de usuarios y diferentes conjuntos de aplicaciones y servicios disponibles. Sin embargo, con “un poco de movimiento”, VCL permite que el administrador del sistema mueva los recursos de un *pool* a otro.

Esto proporciona una gran flexibilidad, maximiza el uso eficaz de recursos y el rendimiento aumenta la economía de la operación. Por ejemplo, más de la mitad de los recursos se encuentra en el *pool* de HPC. Pero en el inicio de clases de cada semestre, los servidores son transferidos del lado de HPC al lado de propósito general, invirtiendo la tendencia durante el verano y los periodos vacacionales.

El *pool* HPC proporciona a los investigadores los servidores y el almacenamiento necesarios para los cálculos a gran escala. También proporciona el estándar HPC (asíncrono) de procesamiento por lotes con las aplicaciones HPC más estándar, colas de trabajo, y así sucesivamente.

Los servidores en el lado de propósito general (VCL 24x7) proporcionan entornos de escritorio o servidores de aplicaciones típicos para el uso dentro de la clase, tareas de clase, proyecto, etc., que son generalmente reservados por solo unas horas a la vez.

Sin embargo, el VCL 24x7 puede proporcionar reservas a largo plazo para los servidores y puede aprovisionar nubes de investigación. Los profesores, asistentes de cátedra u otros estudiantes con permiso, tienen la autoridad para crear y guardar sus propias imágenes para fines especiales.

Por ejemplo, un profesor puede seleccionar una imagen autorizada de Windows que se ejecuta en un servidor VCL, instalar cualquier *software* necesario para una clase y luego colocar la imagen de nuevo en el repositorio de imágenes de la

universidad. Los estudiantes de esta clase pueden acceder simultáneamente a múltiples copias de la imagen.

El VCL creció muy rápidamente debido a las políticas puestas en marcha por la oficina de TI, por los departamentos de la institución e institutos, investigadores del departamento de HPC, Ciencias de la Vida y otros departamentos, recibieron regularmente becas de la NSF (*National Science Foundation*), el Departamento de Defensa de EE.UU., los Institutos Nacionales de Salud (NIH) y otros, que incluyó financiamiento de los servidores.

Sin embargo, el financiamiento generalmente no cubría el espacio, la energía, la refrigeración o el apoyo personal, todos los costos típicamente asociados a la operación de un servidor durante su vida útil. Estos costos debían ser asumidos por los departamentos o institutos.

Como resultado, estos grupos apoyaron con entusiasmo el “trato” ofrecido por la OIT *Office of Information Technology*: soporte operativo gratuito por tres años, garantizando la entrega y el uso exclusivo de los recursos de sus servidores cada vez que fueran solicitados por los investigadores financiados.

A cambio, la OIT define las especificaciones del servidor, lo que consigue un *pool* estandarizado que proporciona el mejor costo de propiedad. Además, se podría “prestar” los mismos equipos de investigación a la población en general, cuando no sean utilizados en los ciclos del plantel, lo que podría significar el 50% o más de las veces.

Al combinar adecuadamente los trabajos de HPC y de los usuarios VCL 24x7, el estado de Carolina del Norte ha sido capaz de hacer funcionar la investigación y la educación en su nube VCL a un costo de 3 a 20 centavos de dólar por hora de cómputo.

Esto coincide con las actuales tasas subsidiadas proporcionadas por algunos proveedores de nube pública sin la correspondiente reducción en la calidad de servicio. El dueño de los recursos de la nube tiene un mejor control sobre la satisfacción de los Acuerdos de Nivel de Servicio (SLA) o expectativas de usuario.

Durante el semestre escolar regular, por su población de 30000 usuarios, el VCL ve constantemente un pico diario de cerca de 500 usuarios de propósito general (no-HPC) simultáneos de 10 a.m. – 10 p.m., este uso requiere un mínimo de 500 servidores reales o virtuales asignados para apoyar a los usuarios VCL 24x7.

Este requisito varía según la escuela, dependiendo de patrones, tamaño y uso. Sin la flexibilidad proporcionada por la nube basada en VCL, la mayoría de servidores reales estarían ociosos de 10 p.m. – 10 a.m.

Sin embargo, con el VCL, en momentos de inactividad, estos recursos pueden o bien ser entregados a los trabajos asignados por lotes, o apagados para ahorrar energía. Mediante el uso de este enfoque, sobre todo durante los descansos vacacionales, se puede maximizar la utilización de los recursos, conduciendo así a un bajo coste por CPU.

1.4.2. VCL EN APOYO DE LAS APLICACIONES DE ADMINISTRACIÓN UNIVERSITARIA Y DE NEGOCIO

VCL no distingue entre servicios o aplicaciones académicas o de negocio. Por lo tanto, una serie de esfuerzos dentro del sistema pueden estar destinados a apoyar la parte administrativa de la universidad.

Estos esfuerzos muestran una nube basada en VCL que puede entregar fácilmente las aplicaciones y los servicios de apoyo a los sistemas típicos de planificación de recursos empresariales (ERP), para la facturación, nómina de estudiantes, programación de clases y calificaciones.

El Director de Informática de la Universidad Estatal de Georgia (GSU) se ha embarcado en una importante consolidación y virtualización de la computación administrativa, así como en el interés de construcción de una nube de estado para abordar los desafíos presupuestarios actuales.

Se está trabajando también, en Carolina del Norte en los proyectos orientados a la salud en la que el VCL proporcionará aplicaciones y recursos inmediatos, dinámicamente flexibles para responder a las necesidades y urgencias del día a día.

Para soportar el lado administrativo en una institución educativa, el VCL puede adicionalmente proporcionar:

- Cualquier aplicación de negocio que pueda ser entregada como servidor físico, virtual o de escritorio, con multipropósito, y largo tiempo de préstamo del servidor.
- Capacidad dinámica de escalabilidad horizontal, para aplicaciones de servidor normalmente estáticas. Se pueden aumentar dinámicamente el número de instancias del servidor web, para soportar un sitio web, que se ha convertido transitoriamente popular.

Dada la capacidad de VCL para apoyar al negocio y al aspecto académico de un campus, se pueden encontrar algunas empresas comerciales que utilizan este modelo para apoyar a sus empresas. Un ejemplo de esto es una gran empresa de telecomunicaciones en China, donde las soluciones de código abierto son bien recibidas.

1.4.3. VCL, UNA PLATAFORMA DE GESTIÓN DE CLOUD COMPUTING DE PROPÓSITO GENERAL

VCL es a menudo erróneamente comparada con otras soluciones existentes basadas en la nube como IaaS, PaaS, SaaS y, que proporcionan individualmente conjuntos muy específicos y limitados de servicios en la nube.

VCL es, de hecho, una plataforma de gestión de *cloud computing* de propósito general o *Stack* de Gestión de la Nube (CMS) con un portal de auto-servicio que los profesores y estudiantes pueden usar fácilmente a través de Internet para acceder a cualquier número de IaaS, PaaS, SaaS, y cualquier otro servicio en la nube que pueda soportar.

El código VCL crea una nube totalmente funcional que ofrecerá todo, desde *Hardware-as-a-Service* hasta escritorios virtuales. Es capaz de ofrecer versiones gratuitas de *hypervisor* como VMware ESXi, KVM o Xen sin el gran gasto que se requiere en la actualidad para adquirir soluciones comerciales proporcionadas por VMware, Citrix u otros.

Puede ofrecer todos estos servicios de la manera más rentable, gestionando las licencias de *software*, monitoreando el *hardware* y *software* de uso real, gestionando el acceso a usuarios/grupos en base a las políticas definidas por el administrador, y otros.

El poder de VCL es su alcance, versatilidad y apertura a ser modificado o ampliado para apoyar e interactuar con una amplia variedad de nubes, plataformas de *hardware*, sistemas operativos y *hypervisors*.

Antes de que el código VCL fuera donado a Apache, fue modularizado y mejorado con un rico conjunto de API, lo que permite la integración de una amplia variedad de *plug-ins* para la interoperabilidad y la mejora funcional.

El VCL no se limita sólo a los despliegues de un *hypervisor*, o a una limitación de todos los VDI y soluciones en la nube que se proporcionan con más frecuencia en la actualidad. Con su integración con xCat o *Tivoli Provisioning Manager* (TPM), los cuales pueden hacer despliegues de máquinas individuales o grupos, puede trabajar con todas las soluciones principales de nube privada y pública. Esto permite la creación de una nube privada con *Tivoli Service Automation Manager* (TSAM), o una nube híbrida privada-pública con *SmartCloud*. [35]

1.4.4. ALCANCE A LOS COLEGIOS DE LA COMUNIDAD Y K12 [38]

Además de apoyar a los estudiantes de la Universidad Estatal de Carolina del Norte y de otras universidades del sistema UNC, el VCL ahora también apoya a los estudiantes en varios colegios de la comunidad de Carolina del Norte.

La Universidad Estatal de Carolina del Norte, a través de su Instituto de Viernes y otros programas K-12 de divulgación, así como de otras universidades, incluyendo la Universidad Central de Carolina del Norte, la Universidad de Duke, la Universidad Estatal de Morgan y otras, han estado explorando ampliamente el uso de VCL en apoyo de la educación K-12.

Miembros de la facultad de Duke están estudiando el uso de VCL para ofrecer el entorno de programación Alice y el *software* “*Project Lead the Way*” para

educadores y estudiantes K-12 como parte de los programas STEM (*Science, Technology, Engineering and Math*) [40] con todo el estado. Los instructores en Hillside New Technology High School en Durham están enseñando con el *software* de Alice entregado por el VCL de Carolina del Norte como parte de un proyecto de extensión dirigido a los profesores del Departamento de CEI (*Certified E-Learning Instructor Specializing in Online Instruction*) e NCCU (*North Carolina Central University*).

Un proyecto similar para los estudiantes del centro de la ciudad de Baltimore, está siendo estudiado en la Universidad Estatal Morgan, con su instalación VCL. En todos los casos, VCL es capaz de hacer frente a la Ley de Protección de la Infancia de la Información (CIPA) requisitos definidos por los distritos escolares locales.

Con un paradigma de la nube, la asistencia técnica debe residir dentro de los centros de *hosting* de los recursos de computación de la nube. Desde estos centros ahora se pueden configurar fuera de una escuela K-12 o del distrito escolar, lo que hace que el personal de TI en sitio se reduzca.

Sin embargo, la nube todavía requiere conectividad a Internet y la tecnología de navegadores en el lugar de usuario, por lo que la inversión en la conectividad de la red de educación a nivel estatal es estratégica.

El modelo de implementación frecuentemente citado para VCL en la clase K-12 es uno en el que un profesor lanza un carrito de baratas, clientes que permiten a los estudiantes acceder a Internet y obtener *software* educativo avanzado y otros recursos y servicios, independientemente de la situación económica, infraestructura técnica o la ubicación geográfica del distrito escolar.

1.4.5. OPORTUNIDADES DE DESARROLLO ECONÓMICO

Varios proyectos basados en la nube que impliquen extensión universitaria a la comunidad empresarial, sugieren formas en que VCL puede convertirse en un motor para el desarrollo económico más allá de ser un recurso que apoya la formación continua.

Algunas universidades cuentan con instalaciones o incluso recintos destinados al apoyo a la comunidad empresarial, en especial la creación de empresas y *spin-offs* universitarias. Además del acceso a los recursos por parte de los estudiantes, la consultoría de negocios, un sistema universitario o una nube universitaria basada en VCL también puede proporcionar una infraestructura de bajo costo de TI, *software* y herramientas para las empresas regionales.

Tales servicios pueden atraer y nutrir la creación de empresas que puedan, a su vez, proporcionar puestos de trabajo a los estudiantes universitarios y otros residentes de estado, investigación y financiamiento para la universidad, y contribuir al desarrollo económico global de la región.

En la Figura 1.1 se puede evidenciar una arquitectura IBM para un VCL para educación.

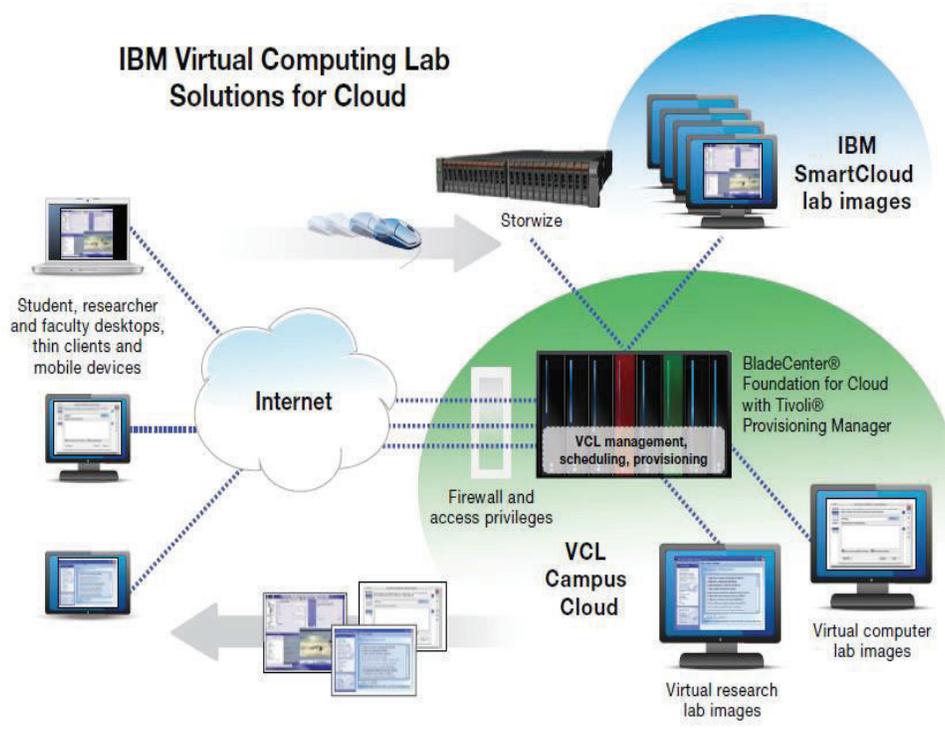


Figura 1.1: Soluciones VCL provistas por la IBM *SmartCloud for Education* [30]

1.5. PROTOCOLOS Y MODELO DE RED UTILIZADOS

A continuación se revisarán los conceptos básicos, de los principales protocolos que se utilizarán en el despliegue de la solución de aprovisionamiento.

1.5.1. MODELO ISO-OSI [45]

Virtualmente, todas las redes que se usan diariamente, están basadas en el modelo ISO-OSI (*International Organization for Standardization- Open Systems Interconnection*). El modelo OSI fue desarrollado por la ISO en 1984. Este estándar se basa en 7 capas que proveen un modelo abstracto de las redes de comunicaciones, definiendo los diferentes pasos por los cuales deben pasar los datos, para viajar desde un dispositivo a otro dentro de la red de comunicaciones.

Este modelo es realmente importante en el mundo de las comunicaciones, marcando una nueva era, desde su aparición; donde sin importar el fabricante, tecnología utilizada, ubicación geográfica, todos los miembros de la red de comunicaciones se entienden perfectamente.



Figura 1.2: Representación del modelo ISO-OSI

1.5.1.1. Capa Física

Transmite y recibe una secuencia de bits no estructurados, sin procesar a través de un medio físico, que puede ser eléctrico u óptico. Lleva también las señales a la capa superior. Proporciona codificación de datos, conexión al medio físico, técnica de transmisión, transmisión de medio físico.

1.5.1.2. Capa Enlace de datos

Ofrece una transferencia sin errores de datos desde un nodo a otro, a través de la capa Física. Permite a la capa de Red “asumir” una transmisión sin errores. Para esto la capa de Enlace proporciona, establecimiento y finalización de enlaces, control de tráfico de tramas, secuenciación de tramas, confirmación de tramas (ACKs), delimitación de trama, comprobación de errores, administrando el acceso al medio.

1.5.1.3. Capa de Red

Controla el funcionamiento de la subred, decide qué ruta deberían tomar los datos, en función de las condiciones de la red, prioridad de servicio, calidad de servicio y otros factores. Para esto la capa de Red proporciona, control de tráfico de la red, fragmentación de paquetes, asignación de direcciones lógico físicas, cuentas de uso de red.

1.5.1.4. Capa de Transporte

Garantiza que los datos contenidos en los mensajes se entreguen sin errores, en la secuencia adecuada y sin pérdidas o duplicaciones. Libera a los protocolos de las capas superiores de cualquier responsabilidad relacionada con la transferencia de datos entre ellos y sus pares.

La capa de Transporte proporciona segmentación de mensajes que recibe de la capa sesión, ensambla mensajes que le llegan de la capa de Red, confirmación de mensaje, control de tráfico de mensajes, multiplexación de sesión.

Comúnmente la capa de Transporte acepta mensajes relativamente grandes de la capa de Sesión, pero existen limitaciones muy estrictas de la capa de Red, por lo que debe dividir los mensajes de la capa de Sesión en mensajes más pequeños.

1.5.1.5. Capa de Sesión

Establece sesiones entre procesos que se ejecutan en diferentes nodos u estaciones. Proporciona establecimiento, mantenimiento y finalización de sesiones, así como también soporte para la sesión.

1.5.1.6. Capa de Presentación

Da formato a los datos que luego serán enviados a la capa de Aplicación. Se le considera el traductor de la red, debido a que traduce los datos utilizados por la capa de Aplicación a un formato común en la estación que enviará los datos y viceversa. La capa de Presentación proporciona, conversión de código de caracteres, conversión de datos, comprensión de datos, cifrado de datos.

1.5.1.7. Capa de Aplicación

Ésta es la capa que ve el usuario; actúa como ventana para los usuarios y procesos de aplicaciones para tener servicios de red.

La capa de Aplicación tiene varias funciones que se utilizan con frecuencia tales como, el uso compartido de recursos, acceso a archivos remotos, acceso a impresora remota, comunicación entre procesos, administración de la red, servicios de directorio, mensajería electrónica, terminales virtuales de red, entre otros.

Las capas superiores a la capa física sólo se pueden comunicar con su capa inferior y superior respectivamente. En el nodo emisor, dado que los datos se transmiten desde la capa superior hasta la inferior, el protocolo de cada capa añade información de control propia de la capa a los datos.

En el nodo receptor, dado que los datos se transmiten desde la capa inferior hacia la superior, el protocolo de cada capa retira y analiza la información de control para esa capa, antes de enviar la trama restante a la capa superior correspondiente.

1.5.2. IP – INTERNET PROTOCOL

Proporciona los medios para la transmisión de datagramas (bloques de bits de datos) desde el emisor (origen) al receptor (destino), donde emisor y receptor son hosts que se identifican por direcciones fijas. También se encarga, de la fragmentación y el rearmado de grandes datagramas, para que puedan ser enviados a través de la red de comunicaciones. [46], [47]

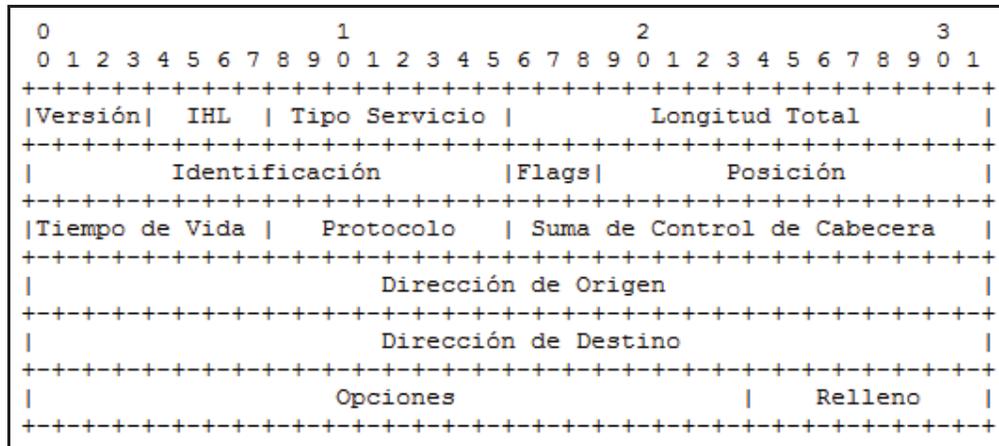


Figura 1.3: Ejemplo de cabecera IP [46]

1.5.3. TCP – TRANSMISSION CONTROL PROTOCOL

El protocolo de control de transmisión se utiliza como un protocolo “host” a “host” muy fiable entre los miembros de la red de comunicación. El propósito general es proporcionar un servicio de conexión confiable entre pares de procesos. Para proporcionar este servicio sobre un entorno de Internet no fiable, se requieren de mecanismos relacionados con las siguientes áreas: transferencia de datos, fiabilidad, multiplexación de datos, conexiones, control de flujo, etc. [48], [49]

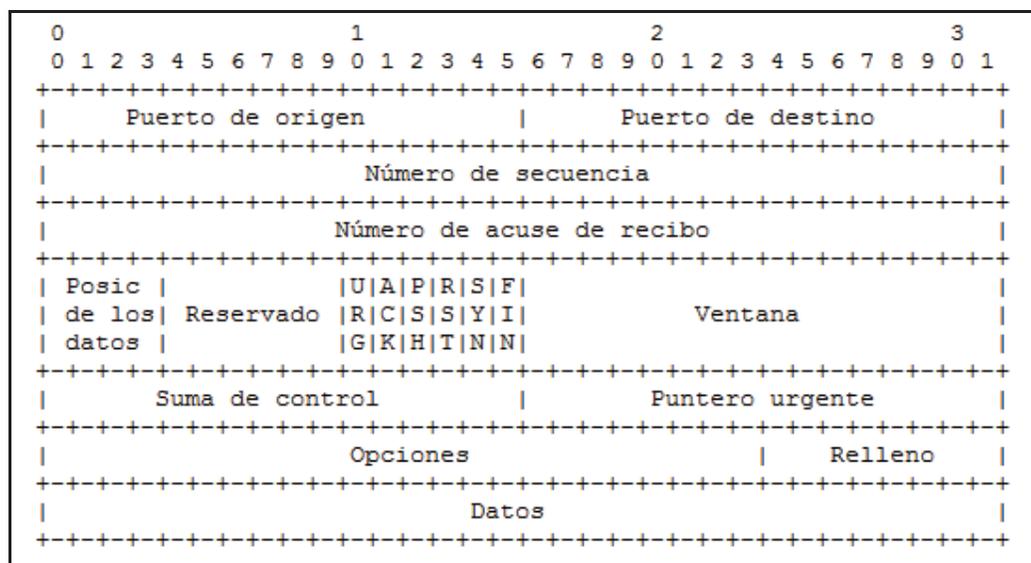


Figura 1.4: Ejemplo de un segmento TCP [49]

1.5.4. UDP – USER DATAGRAM PROTOCOL

El protocolo de datagramas de usuario, está definido para hacer disponible un tipo de datagramas para la comunicación por intercambio de paquetes entre hosts en el entorno de la red de comunicaciones.

UDP asume que el Protocolo Internet (IP) se utiliza como protocolo subyacente. UDP está orientado a transacciones, y tanto la entrega como la protección ante duplicados no se garantizan. [50]

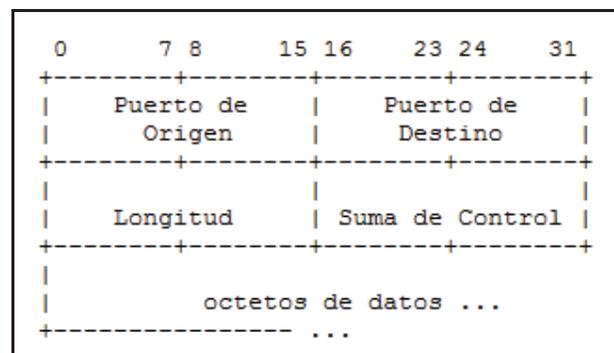


Figura 1.5: Ejemplo de un segmento UDP [51]

1.5.5. HTTP – HYPERTEXT TRANSFER PROTOCOL (SECURE)

HTTP – HTTPS es uno de los protocolos que más se utiliza en la actualidad, significa Protocolo de Transferencia de Hipertexto. Este protocolo pertenece al grupo TCP/IP y fue creado con la finalidad de publicar páginas HTML.

Su funcionamiento consiste en que, un navegador envía una solicitud GET al servidor pidiendo un archivo, el servidor responde al navegador el código de ese archivo, que luego será descifrado por el navegador.

HTTP utiliza tres tipos de mensajes para enviar y recibir la información del navegador: GET, POST, PUT.

GET es un mensaje de solicitud por parte del cliente. Este mensaje es enviado por un navegador para solicitar la página a un servidor.

POST y PUT son mensajes utilizados por el servidor para responder a las solicitudes que hace el navegador, así como también para enviarle información. POST incluye la información en el mensaje y PUT carga el contenido en el servidor.

Este protocolo es poco seguro, debido a que la información puede ser interceptada y leída sin problema.

De esta necesidad nace el protocolo HTTPS, que no es más que el mismo protocolo HTTP, pero con la adición de la característica de encriptación. Es decir que envía la información encriptada, para que si es interceptada, ésta no pueda ser utilizada.

1.5.6. RDP – REMOTE DESKTOP PROTOCOL [130]

Se basa en, y constituye una extensión de, la familia de protocolos ITU T.120. RDP es un protocolo multicanal que permite por canales virtuales separados transportar datos de comunicación desde un dispositivo y datos de presentación desde el servidor, así como también datos cifrados del mouse y teclado del cliente. RDP soporta hasta 64000 canales independientes para la transmisión de datos y provisiones multipunto.

1.5.7. SSH – SECURE SHELL [131]

SSH es un protocolo de acceso remoto seguro y otros servicios de red seguros a través de una red insegura. Provee una encriptación muy fuerte, autenticación de host y protección de integridad.

El protocolo ha sido diseñado para ser simple y flexible a la hora de permitir la negociación de parámetros, y para reducir al mínimo el número de viajes de ida y vuelta. El método de intercambio de claves, el algoritmo de clave pública, algoritmo de cifrado simétrico, el algoritmo de autenticación de mensajes son todos negociados. Se espera que en la mayoría de los entornos, sólo se necesitará 2 viajes de ida y vuelta para el intercambio completo de clave, la autenticación del servidor, solicitud de servicio, y la notificación de aceptación de la solicitud de servicio. El peor caso es de 3 viajes de ida y vuelta

1.5.8. NFS –NETWORK FILE SYSTEM [132]

El protocolo NFS proporciona acceso remoto transparente a los archivos compartidos a través de redes. El protocolo está diseñado para ser portable a través de las diferentes máquinas sistemas operativos, arquitecturas de red y protocolos de transporte.

Esta portabilidad se consigue a través del uso de primitivas RPC *Remote Procedure Call* construidos sobre una representación de datos externa (XDR).

CAPÍTULO 2

SOLUCIONES DE *SOFTWARE* Y REQUERIMIENTOS DE *SOFTWARE Y HARDWARE*

2.1. SOLUCIONES DE APROVISIONAMIENTO

La computación en la nube tiene el potencial de transformar radicalmente TI, aumentando la capacidad de respuesta a las necesidades del negocio, y al mismo tiempo reduciendo el costo de la infraestructura, plataformas y aplicaciones.

De igual forma, la propia naturaleza de la operación de un entorno de nube tiene el potencial de aumentar la escala de TI y su complejidad. Más máquinas virtuales, más dependencias, y un rápido cambio, la infraestructura dinámica con todos los requisitos de gestión de infraestructuras complejas.

La implantación y el mantenimiento de la infraestructura de TI suponen un reto importante ya que deben dar soporte a las necesidades de un entorno educativo cambiante. La velocidad de cambios en tecnología es extraordinariamente rápida e impredecible, lo que hoy es la última moda en tecnología, para mañana ya está obsoleto. [133], [134]

Los gastos de capital al iniciar un proyecto y los costes de operación son elevados y frecuentemente la infraestructura es subutilizada y es muy difícil asignar la infraestructura a nuevos proyectos o aplicaciones. Es por eso que la optimización de los recursos económicos, tecnológicos y de personal de técnico se torna imperativamente vital para la sostenibilidad y crecimiento del negocio o servicio.

Las tareas de adquisición, configuración, soporte y mantenimiento de los entornos de *hardware* y *software* suelen efectuarse de manera manual, lo que se puede traducir en altos costes de operación, mano de obra y en un mayor riesgo de configuraciones inadecuadas que producirán errores.

Por todo esto, es necesario contar con una solución que permita implementar estándares y protocolos que permitan una conexión segura, la cual, dé soporte a usuarios finales (profesores o alumnos) y equipos de trabajo, independientemente

de si se encuentran en el campus universitario, hogares o en cualquier punto del planeta.

La computación en nube ayuda a contrarrestar estos retos; de igual manera, puede ayudar a responder a cargas de trabajo inesperadas e impredecibles y a reducir los ciclos de desarrollo y despliegue de aplicativos. [133], [134]

Se puede configurar nuevos entornos de servidores en minutos, en lugar de tener que invertir semanas en ello. Con esto a más de reducir el tiempo de puesta en producción del ambiente, se consigue mejorar la calidad gracias a la disminución de errores de configuración.

Asimismo, se pueden crear configuraciones e imágenes de máquinas virtuales que pueden ser compartidas con los estudiantes, profesores, asistentes de cátedra, etc., reduciendo costos operativos y mejorando la calidad y el uso de recursos.

La división de desarrollo de *software* de la empresa IBM ha logrado un ahorro muy significativo, llegando a reducir a aproximadamente el 50% en costes operativos (administración, mantenimiento) y una mejora de calidad del 30% aplicando un enfoque de computación en la nube.¹

La solución de aprovisionamiento debe proporcionar a los profesionales de TI, profesores y estudiantes, acceso a entornos de colaboración en equipo, con independencia de donde se encuentren. Además debe contar con un catálogo de aplicaciones o herramientas que son utilizadas en las asignaturas de la carrera, para el presente piloto se tomaron como referencia las asignaturas de Base de Datos, Sistemas Operativos y Redes de Área Local.

A continuación se describen las principales soluciones de aprovisionamiento que se pueden encontrar en el mercado ecuatoriano, las cuales tienen presencia

¹ Basado en los resultados del Programa de Adopción Tecnológica de IBM, en cuyo marco se desarrolló una nube interna de "Innovación colaborativa" utilizando tecnología de IBM. La solución cuenta con más de 100.000 participantes. Los resultados varían en función del entorno actual del cliente. Los resultados finales solo se podrán determinar tras efectuar un análisis de la rentabilidad de la inversión.

mundial y una gran acogida en Latinoamérica, países que tienen economías muy similares a la de Ecuador.

2.1.1. HP CLOUDSYSTEM MATRIX

HP CloudSystem Matrix es una solución que se basa en un modelo de servicios compartidos, utilizando un grupo de recursos de cómputo, almacenamiento y red.

Matrix integra tecnologías probadas para proporcionar una plataforma completa sobre la que los servicios de infraestructura se puedan aprovisionar y ajustar fácilmente a las cambiantes demandas del negocio. [58]

La solución es compatible con la infraestructura física **HP CloudSystem c-Class** y la infraestructura virtual con tecnologías de *hypervisor* de proveedores como VMware y Microsoft. [59]

HP CloudSystem Matrix está diseñado para enfrentar los problemas más difíciles que se encuentran en las organizaciones de TI: costo, tiempo, energía y cambio. Dispone de un *software* flexible e integrado que unifica la gestión de los recursos de *hardware*, sin importar el tamaño de la empresa, **CloudSystem Matrix** permite:

- Minimizar el tiempo en la administración de la infraestructura física y virtual.
- Manejar el crecimiento del negocio a través de un despliegue y optimización de infraestructura flexibles.
- Bajar costos a través de un uso más eficiente del personal e instalaciones

Un componente clave de **HP CloudSystem Matrix** es la capacidad de proporcionar una gestión del ciclo de vida de los grupos de servidores físicos y virtuales, incluyendo:

- Diseño, gestión de recursos y herramientas de auto-servicio que soportan papeles clave de roles de arquitectura, administración y usuarios de servicios de TI.

- Un entorno de diseño de la plantilla que se utiliza para definir los planos para el aprovisionamiento de servidores en un servicio.
- Utilización eficaz de los recursos habilitados por la asignación automática de recursos del *pool* de recursos asignados a los usuarios.
- Aprovisionamiento automatizado de servidores virtuales y físicos, incluyendo el *hardware HP CloudSystem* y máquinas virtuales, implementación del sistema operativo, personalización del sistema operativo, y la implementación de aplicaciones.
- Flujos de trabajo personalizables para permitir la integración con los actuales procesos de TI para la aprobación, implementación del sistema operativo, y la provisión de almacenamiento utilizando una versión integrada de **HP Operations Orchestration**.
- Gestión continua de la infraestructura aprovisionada para poder crecer, activar y desactivar los servicios de infraestructura.
- Seguimiento y mantenimiento de servicios de infraestructura aprovisionados y disponibles.

2.1.2. BMC CLOUD LIFECYCLE MANAGEMENT

Las organizaciones de TI deben tomar decisiones clave sobre las plataformas compatibles, grados de flexibilidad y escalabilidad utilizando una estrategia de gestión unificada.

La respuesta está en la implementación de una sola plataforma unificada que puede gestionar con éxito todo el ciclo de vida de la nube a través de ambos recursos internos y externos - desde la solicitud, mediante el aprovisionamiento de auto-servicio, al mantenimiento y desmantelamiento.

El ciclo de vida debe tener la capacidad de adaptarse a las necesidades de la empresa con la flexibilidad para entregar diferentes niveles de servicios en la nube

configurados por el usuario y el rigor de la gestión para garantizar la integridad operativa de la nube, desde la solicitud hasta el retiro.

También se debe emplear un proceso de aprovisionamiento con política manejada a través de un portal de auto-servicio usando un catálogo de servicios. Con la gestión de ciclo de vida de la nube en su lugar, puede lograr los objetivos fundamentales de un entorno de nube: agilidad, ahorro de costes y un uso óptimo de los recursos.

Muchas organizaciones se acercan a la computación en nube y hoy ya han tenido alguna experiencia en la implementación de virtualización en sus Centros de Datos. Todo el conocimiento de conceptos de virtualización es útil y fácilmente aplicado durante la migración a sistemas de computación en la nube.

Al ampliar el entorno virtualizado tradicional, **BMC Cloud Lifecycle Management** ofrece un modelo operativo para el ciclo de vida de servicios en la nube y la utilización de las nubes públicas en un modelo híbrido. Todos los recursos en el ambiente pasan por un ciclo de vida que, cuando se define y automatiza apropiadamente, ofrece una nube transparente y entendible para TI. [52]

BMC Cloud Lifecycle Management proporciona:

- Portal de auto-servicio y de administración de la nube, fácil de usar.
- Diseño de servicios flexible y aprovisionamiento con los modelos de servicio.
- Operaciones establecidas y en curso basadas en políticas inteligentes.
- Las operaciones en curso con los procesos de ITIL integradas, incluyendo el rendimiento y la gestión de la capacidad.
- Soporte a la heterogeneidad, incluyendo integraciones físicas, virtuales multi-*hypervisor*, integraciones de nube pública.

2.1.3. VCE VBLOCK SYSTEMS

Vblock™ Systems de VCE dan rienda suelta a la simplicidad mediante la entrega de eficiencia y agilidad de negocios de virtualización y *cloud computing*.

Constituye la perfecta integración de las mejores tecnologías en su clase de computación, red y de almacenamiento de los líderes de la industria Cisco, EMC y VMware, Vblock ofrece *pools* dinámicos de recursos que pueden ser inteligentemente aprovisionados y administrados para atender las cambiantes y fugaces demandas y oportunidades de negocio. [56]

Cada sistema **Vblock** tiene una configuración base, que consta de un conjunto mínimo de componentes informáticos y de almacenamiento, así como los recursos de red fijos. Dentro de la configuración base, ciertos aspectos de *hardware* se pueden personalizar. [57]

El Sistema **Vblock 300** es un Centro de Datos de clases ágil y eficiente, proporcionando un rendimiento modular y escalable. Cuenta con una alta densidad, conmutador de estructura compacta, servidores *blade* fuertemente integrados a base de tejido, y el almacenamiento unificado de las mejores en su clase.

El Sistema **Vblock 700** se adapta a las necesidades de las empresas más grandes que ejecutan miles de máquinas virtuales y está diseñado para satisfacer los requisitos más exigentes para la planificación empresarial crítica de recursos empresariales (ERP), gestión de relaciones con clientes (CRM), bases de datos, mensajería y servicios de colaboración.

Las organizaciones de TI deben tomar decisiones clave sobre las plataformas compatibles, grados de flexibilidad y escalabilidad utilizando una estrategia de gestión unificada.

2.1.4. IBM SERVICE DELIVERY MANAGER – ISDM

IBM® Service Delivery Manager es una solución que tiene un conjunto de productos de *software* pre-configurados y pre-integrados. Esta solución se implementa sobre un ambiente virtual, con lo cual, permite la reutilización de la infraestructura de virtualización que las organizaciones tienen instalada. [54]

Los administradores del Centro de Datos se ven beneficiados gracias a las bondades que la solución ISDM provee durante la creación de plataformas de

servicios que están destinados para un amplio espectro de tipos de carga de trabajo con un alto grado de integración, flexibilidad y optimización de recursos.

ISDM es una solución fácil de usar y entender, por lo que representa una buena opción para dar los primeros pasos en el trabajo con modelos privados de *cloud computing*. El producto permite implementar rápidamente una solución de *software* completa para automatizar la gestión de servicios en un entorno de Centro de Datos virtual, que a su vez puede ayudar a que las organizaciones avancen hacia una infraestructura más dinámica y flexible. [55], [60]

ISDM es una solución única que ofrece todos los componentes de *software* necesarios para implementar la computación en la nube. La computación en la nube permite entregar recursos de TI por medio de un servicio, lo que ayuda a mejorar el rendimiento del negocio y controlar los costos de entrega de los recursos de TI de una organización.

Los beneficios potenciales que el modelo de *cloud computing* que ISDM otorga a las organizaciones se listan a continuación:

- Reducción de gastos operativos y capital.
- Disminución del tiempo de salida en vivo o a producción de características o proyectos empresariales que aumentan la competitividad de una organización en el mercado local o internacional.
- Estándares y servicios de TI consolidados que impulsan a la mejor utilización de recursos económicos, tecnológicos y humanos.
- Aumenta la capacidad de recuperación ante la demanda del mercado local o internacional.
- Mejora en la calidad y cantidad del servicio para los usuarios o consumidores de TI.

ISDM proporciona capacidades preinstaladas esenciales para un sistema en la nube, incluyendo:

- Una interfaz de portal de auto-servicio para la solicitud de un ambiente virtual, incluyendo recursos de procesamiento, red, memoria, almacenamiento y aplicaciones.
- Permite la gestión de aprovisionamiento y des-aprovisionamiento automatizado de los recursos virtuales de un centro de datos.
- Plantillas de automatización y flujos de trabajo pre-configuradas para la administración de imágenes virtuales de VMware.
- Gestión de servicios de un modelo privado de *cloud computing*.
- Monitoreo en tiempo real de la elasticidad y utilización de la infraestructura del centro de datos virtual.
- Copia de seguridad y recuperación de información sensible.

2.2. DIFERENCIAS ENTRE LA SOLUCIÓN DE IBM Y LAS OTRAS MOSTRADAS

En la presente sección, se comparan las principales características de las soluciones de aprovisionamiento, enfatizando en las capacidades de integración con otras soluciones disponibles en el mercado.

2.2.1. IBM SERVICE DELIVERY MANAGER VS VCE VBLOCK

En la Tabla 2.1 se especifican las principales diferencias entre las soluciones **IBM Service Delivery Manager** y **VCE VBLOCK**.

VCE VBLOCK	ISDM
Diseñado para las ventas de campo verde, orientado a <i>hardware</i> Cisco y EMC, no pueden aprovechar el <i>hardware</i> y la infraestructura existente.	Está diseñado para ejecutarse en infraestructura de <i>hardware</i> existente , reduciendo los costos CAPEX por adelantado.

<p>Sólo es compatible con la gestión del hypervisor VMware en plataformas x86 y se integra únicamente con nubes VMware, estrechando la elección del cliente y causando bloqueos.</p>	<p>Apoya gestionar múltiples hypervisors - VMware, KVM, Xen, System p, y el apoyo de System z.</p> <p>Utiliza extensión HCI para mover la carga de trabajo a través de Amazon EC2, nube IBM y utiliza como base Cast Iron para integrar la nube pública como un nivel de servicio (ejemplo: SAP ERP conectado a Salesforce.com).</p>
<p>No tiene gestión de solicitudes out-of-the-box (sólo auto-servicio de aprovisionamiento), los usuarios deben integrar <i>vCloud Director</i> con <i>VMware Service Manager</i> para la gestión de la solicitud.</p>	<p><i>vCloud</i> no puede realizar un seguimiento de las solicitudes, manejar aprobación y denegación de solicitudes de utilización de recursos de la nube que ISDM hace out-of-the-box.</p>
<p>No incluye el seguimiento y la devolución de cargo con <i>vCloud Director</i>; VMware carece de monitoreo de la infraestructura física.</p> <p>Es una suite de productos separados y aumenta las ventas para el seguimiento y devolución de cargo vCloud y requiere tercera parte para el monitoreo físico.</p>	<p>ISDM incluye ITM y TUAM para el seguimiento y devolución de cargo, ITM puede monitorear tanto la infraestructura física y virtual.</p>
<p>Carece de aprovisionamiento fuera de la caja de aplicaciones y gestión de imágenes en vCloud Director.</p>	<p>AppDirectordoes-vFabric aprovisionamiento de aplicaciones, VMware se limita a la búsqueda de imágenes por nombre o atributo en lugar de darles seguimiento en una biblioteca de imágenes como TPM hace dentro ISDM.</p>
<p>Requiere el uso de un servidor de instalación tradicional de vCloud Director en un servidor físico RHEL.</p>	<p>Cuenta con un conjunto de imágenes virtuales que se puede implementar rápidamente en un entorno virtual para dar rápidamente valor.</p>

Tabla 2.1: ISDM vs VBLOCK

En la Figura 2.1 se detalla la comparación entre las soluciones VCE e ISDM desde el punto de vista del *stack* del modelo de *cloud computing*. Se puede

determinar una mayor cantidad de características en la solución ISDM en contraste con VCE.

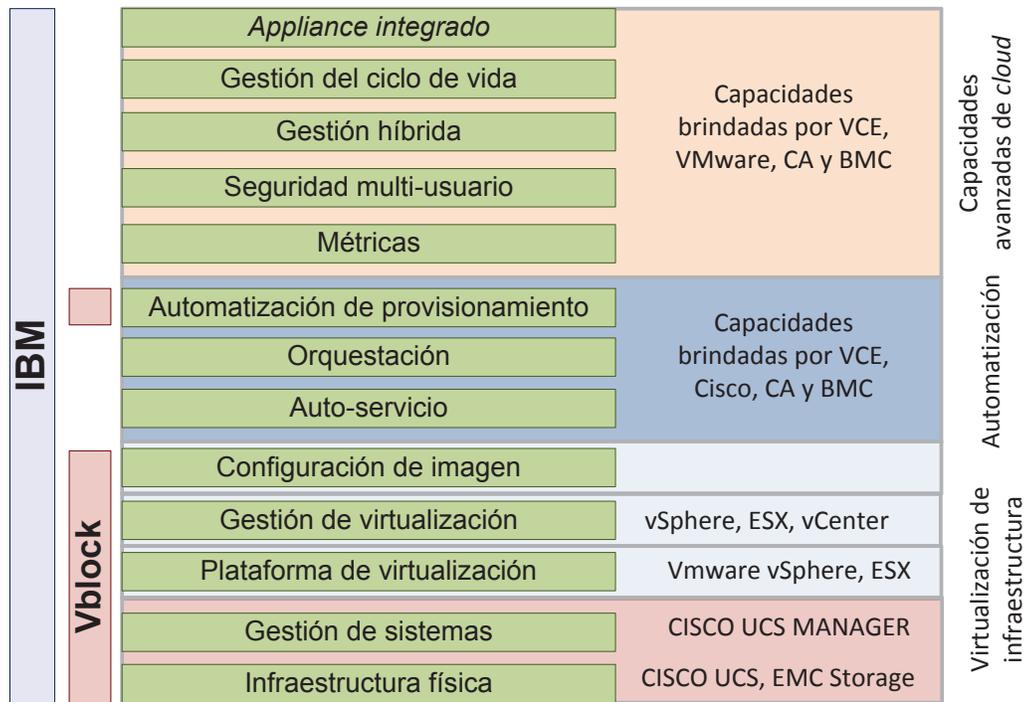


Figura 2.1: Comparación desde el punto de vista del *stack* de *cloud computing*. [133]

2.2.2. IBM SERVICE DELIVERY MANAGER VS BMC CLOUD LIFECYCLE MANAGEMENT

En la Tabla 2.2 se especifican las principales diferencias entre las soluciones **IBM Service Delivery Manager** y **BMC CLM**.

DIFERENCIADOR	BMC CLM	ISDM
Optimización de carga de trabajo	No hace la optimización de la carga de trabajo para aplicaciones. Se basa en los flujos de trabajo predefinidos, los cuales son difíciles de personalizar y son estáticas después de la implementación (no elástico).	Ofrece implementaciones repetibles basada en patrones flexibles de aplicaciones, <i>middleware</i> y máquinas virtuales.

Tiempo de valor y consumo	Va a través de nueve productos diferentes. Es difícil de configurar y muy complicado de mantener sin nivel de entrada que ofrece la nube.	Proporciona una instalación basada en imágenes pre-configuradas para la gestión del servicio. IBM ofrece un nivel de entrada de la computación en nube con SCP <i>IBM SmartCloud Provisioning</i> .
Medición y contracargo	Proporciona capacidad de carga detrás de base, y no hay una medición del uso de recursos compartidos.	Ofrece de uso de capacidades para la medición de profundidad y carga automática de vuelta para servicios en la nube aprovisionadas. <i>Tivoli Usage and Accounting Manager (TUAM)</i> proporciona granularidad de medición para procesamiento, red, almacenamiento y servidores.
Monitoreo fuera de la caja	Requiere que los clientes configuren la supervisión de forma manual. La vigilancia no incluye la aplicación, <i>middleware</i> y monitoreo de base de datos. Sólo proporciona métricas de CPU y memoria para las máquinas y servidores virtuales.	Está integrado con monitoreo inteligente de la nube, que incluye tanto el seguimiento físico como el virtual.
Aprovisionamiento de <i>stack</i> completo	No puede orquestar o provisionar aplicaciones complejas con servidores de seguridad, balanceo de carga y almacenamiento compartido.	Proporciona la capacidad de aprovisionamiento- <i>stack</i> completo con ISDM <i>IBM Service Delivery Manager / SCO IBM SmartCloud Orchestrator</i> , TPM <i>Tivoli Provisioning Manager</i> , ITNCM <i>IBM Tivoli Netcool Configuration Manager</i> y TPC <i>Tivoli Storage Productivity Center</i> .

Gestión de imágenes	Ofrece integraciones con CMDB para hacer el seguimiento de la imagen y el control de versiones, pero no proporciona a los clientes la capacidad de rastrear y componer imágenes en middleware, aplicaciones y nivel de la máquina.	Proporcionan gestión avanzada de imágenes, que permite a los clientes controlar los componentes individuales para proporcionar una manera más ágil y flexible las imágenes.
Configuración flexible	Ofrece modelos para el diseño y definición de despliegue de aplicaciones de varios niveles. Servicio configura colocación de recursos basada en políticas.	Puede hacer un patrón basado en la implementación de aplicaciones en diferentes plataformas, junto con las capacidades de escalamiento y de gestión automática.

Tabla 2.2: ISDM vs BCM CLM

2.2.3. IBM SERVICE DELIVERY MANAGER VS HP CLOUDSYSTEM MATRIX



Figura 2.2: Gestión de servicios de valor del cliente en la industria [77]

- **HP CloudSystem Matrix**, no se puede integrar con plataformas no HP existentes en el Centro de Datos.
- **HP CloudSystem Matrix**, no gestiona el riesgo de los entornos físicos ni virtuales.
- **HP CloudSystem Matrix**, no asegura el correcto funcionamiento para las redes de siguiente generación.
- **HP CloudSystem Matrix**, no tiene soporte para plataformas basadas en System Z.

La solución de IBM no tiene los inconvenientes mencionados y ofrece las ventajas adicionales que se listan a continuación:

- Los clientes pueden escalar rápidamente en entornos de arriba abajo, disfrutando así de una mayor flexibilidad de la infraestructura.
- Los clientes pueden obtener IaaS en marcha y funcionando en un día en lugar de semanas.
- Escalar a cientos de miles de máquinas físicas que ejecutan miles de máquinas virtuales.
- Proporcionar una nube sin parar fiable, capaz de tolerar de forma automática y recuperación de fallos de *software* y *hardware*.

2.3. ARQUITECTURA DE *SOFTWARE* Y *HARDWARE*

En el presente proyecto se definió como alcance la implementación de la solución ISDM para el aprovisionamiento de máquinas virtuales. Este proyecto piloto servirá para demostrar las capacidades de ISDM aplicado a un posible uso en la Carrera, permitiendo conocer de la instalación, utilización y administración de *software* en una nube privada.

Esta solución contempla dos entornos de trabajo. El primer entorno es el administrado desde el cual se utilizarán los recursos para las máquinas virtuales aprovisionadas.

El segundo entorno es el de administración desde el cual se gestiona el sistema de aprovisionamiento; este entorno incluye los componentes que forman la solución: TSAM – Tivoli Service Automation Manager, ITM – Tivoli Monitoring, TUAM – Tivoli Usage and Accounting Manager y NFS – Network File System. En los siguientes ítems se explicarán estos entornos detalladamente. En la Figura 2.3 se muestra la disposición lógica de los dos entornos.

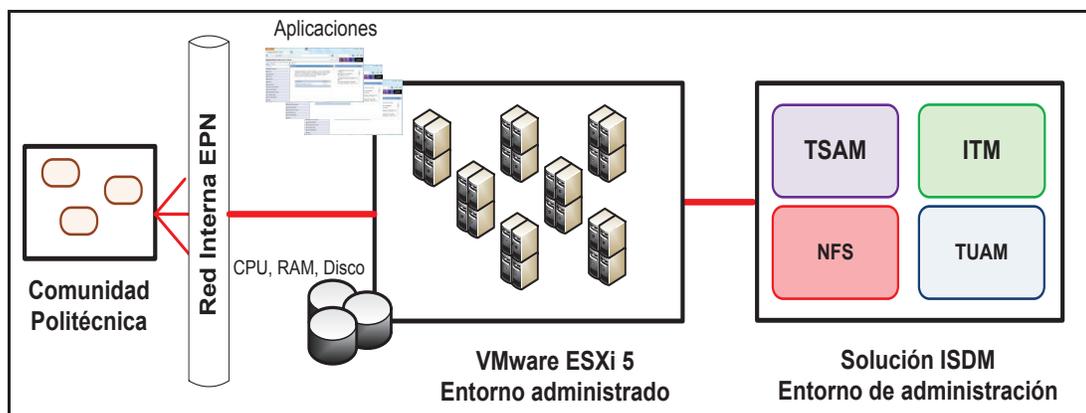


Figura 2.3: Entornos de trabajo

Como pasos previos a la implementación de la solución se requiere describir los componentes de cada entorno, definir la arquitectura a utilizar y sus respectivos requerimientos.

2.3.1. COMPONENTES DEL ENTORNO ADMINISTRADO

Esta sección describe los componentes que forman el entorno administrado. A continuación, se especifica la función de cada uno y la relación entre los mismos: VMware ESXi y VMware VCenter.

2.3.1.1. VMware ESXi

VMware ESXi pertenece a la *suite* de productos de la familia vSphere. Es un *hypervisor* que permite trabajar y experimentar con ambientes virtuales orientados a realizar pruebas piloto e inclusive en ambientes productivos. Es independiente del sistema operativo y permite la ejecución de máquinas virtuales en tiempos reducidos. Su versión básica puede ser descargada del sitio web oficial de

VMware y dada la necesidad de disponer de mejores y avanzadas capacidades existe la posibilidad de realizar la migración de forma transparente y sencilla adquiriendo y actualizando el archivo de licenciamiento. [68], [69]

Al ser una plataforma virtual permite la consolidación efectiva de los recursos económicos, energéticos y administrativos. La instalación es muy sencilla utilizando una imagen en formato ISO. De igual forma la configuración es mínima y se realiza utilizando un cliente de escritorio vSphere Client o vía WEB.

2.3.1.2. VMware VCenter Server

VMware VCenter Server también es un producto de la familia vSphere. Este componente permite gestionar de forma centralizada todos los *hypervisors* ESXi. Además, permite la creación de Centro de Datos mediante la agrupación en *cluster* de uno o más ESXi.

Adicionalmente, permite tener capacidades de *templates*, *clusters*, alta disponibilidad, entre otros. Este producto tiene costo, sin embargo, tiene un periodo de utilización de hasta 90 días en modo de prueba, para tener un mayor tiempo de utilización del producto será necesario adquirir la licencia. [68], [70]

2.3.2. COMPONENTES DEL ENTORNO DE ADMINISTRACIÓN

Esta sección describe los componentes que forman la solución *IBM Service Delivery Manager*, estos corresponden al entorno de administración. Cada uno de los componentes cumple con una funcionalidad específica y se relacionan entre sí para dar como resultado una solución integral de aprovisionamiento de máquinas virtuales sobre una nube privada.

Los componentes que forman la solución ISDM son: *Tivoli Service Automation Manager* (TSAM), *Tivoli Monitoring* (ITM) y *Tivoli Usage and Accounting Manager* (TUAM) y *Network File System* (NFS). Cada uno de los componentes está implementado sobre una máquina virtual, la cual, tiene como sistema operativo base SUSE Linux Enterprise 11 SP3.

2.3.2.1. Tivoli Service Automation Manager – TSAM [72]

TSAM es el componente principal de la pila de productos que forman la solución ISDM. Este permite controlar, gestionar y automatizar el aprovisionamiento y desaprovisionamiento de servicios de TI. Los servicios que se pueden ofrecer comprenden *hardware*, red, sistema operativo y aplicaciones.

Los servicios ofrecidos forman un catálogo de servicios que pueden ser expuestos a uno o varios clientes de la nube. En el caso específico de la Escuela Politécnica Nacional, se puede pensar que una facultad o carrera es un cliente y la administración se puede centralizar en la Dirección de Gestión de la Infraestructura y Procesos - DGIP, por ejemplo.

Dentro de la estructura de objetos que forman el *software* TSAM existen aquellos que pueden ser compartidos y otros que son de uso exclusivo para un único cliente. Los objetos que pueden ser compartidos son asignados de forma manual utilizando el portal de administración, por ejemplo, recursos de red, *hardware*, aplicaciones e imágenes primarias. Los objetos de un solo cliente son asignados de forma automática, por ejemplo, grupos, usuarios y proyectos.

Cada uno de los clientes tiene la capacidad de emitir solicitudes usando el portal de auto-servicio. Este portal expone el catálogo de servicios disponible para cada cliente y estará asociado a los recursos asignados. Las solicitudes siguen un flujo de trabajo, el cual, incluye emisión del requerimiento, verificación de disponibilidad de recursos, aprobadores y despliegue.

2.3.2.2. Tivoli Monitoring – ITM [73]

La solución ISDM dispone de un sistema de monitoreo que se denomina *Tivoli Monitoring*. Con este monitoreo se obtiene reportes de utilización de recursos, tendencias de crecimiento, solución de problemas y seguimiento de alertas o eventos detectados.

La información se muestra en tiempo real o de forma histórica. Estos reportes pueden ser personalizados de acuerdo a las necesidades del negocio o servicio.

El sistema ISDM tiene la capacidad de desplegar agentes en las máquinas virtuales aprovisionadas de forma no atendida. Esta instalación se la puede hacer sobre las plataformas soportadas por la solución. Los sistemas operativos incluidos son Windows, Linux, y AIX – Advanced Interactive eXecutive.

2.3.2.3. Tivoli Usage and Accounting Manager – TUAM [74]

TUAM permite conocer el costo de administración de la infraestructura y crea reportes mediante métricas que genera el producto TSAM. En el momento que TSAM genera, modifica o elimina un proyecto hace uso de los recursos de infraestructura, este seguimiento es utilizado por TUAM para la generación de dichos reportes.

Cada proyecto es asignado a un grupo de trabajo, cliente u organización. Esta variable permite obtener tendencias de utilización por usuario o grupo de usuarios. Con este tipo de métricas se pueden generar proyecciones de crecimiento o facturación por el uso de la infraestructura.

2.3.2.4. Network File System – NFS [75], [76]

Este componente tiene varias funcionalidades. La principal es ser un repositorio de archivos para el almacenamiento de los binarios de instalación que forman el catálogo de *software*, por ejemplo, el binario de instalación del agente de monitoreo de ITM.

La segunda funcionalidad implementada en el mismo sistema es ser un servidor de correo, el cual se utiliza para el envío de notificaciones durante la creación, modificación o eliminación de proyectos.

Finalmente, tiene la funcionalidad de servidor web, es decir, por medio de esta dirección URL se podrán acceder a todas las interfaces o portales de todos los productos que forman la solución ISDM. El servidor web está basado en las funcionalidades otorgadas por el producto IBM HTTP Server. Este componente también dispone de un producto adicional que es *Tivoli System Automation for Multiplatforms* que se utiliza para ambientes configurados en modalidad de alta disponibilidad.

2.3.3. CONFIGURACIÓN DE LA SOLUCIÓN

La solución ISDM está integrada por cuatro máquinas virtuales, cada una corresponde a uno de los componentes de *software* que forman la solución. Los componentes de *software* están preinstalados dentro de su respectiva imagen y se encuentran totalmente integrados entre sí.

Esta disposición facilita el despliegue de la solución, sin embargo, existen pasos que se deben cumplir antes y después de la instalación, por ejemplo, definir la configuración de la solución, ya sea en modalidad *stand-alone* (un solo equipo para cada rol) o en alta disponibilidad.

En las dos modalidades se mantiene el objetivo principal por el cual se utiliza esta solución; sin embargo, la configuración cambia y sus diferencias se revisarán a detalle en los siguientes ítems.

2.3.3.1. Configuración en alta disponibilidad

Una configuración en modalidad de alta disponibilidad considera que los principales componentes de una solución no deben ser un punto de falla, esto con el objetivo de asegurar la continuidad del servicio.

En la solución ISDM existen dos componentes críticos TSAM y NFS, los cuales, deben ser instalados en esta modalidad. Si uno de los dos componentes falla o deja de cumplir con su función, el servicio se verá afectado en su disponibilidad.

No considerar un único punto de falla implica disponer de dos o más nodos físicos o virtuales que cumplan con la misma funcionalidad. Estos nodos están configurados en modo activo-activo de tal forma que se encuentren sincronizados y disponibles de forma simultánea.

Esta configuración en alta disponibilidad permite aprovechar la capacidad de los recursos de todos los nodos que forman el clúster. Se implementa haciendo uso del *software Tivoli System Automation for Multiplatforms*, el cual, también está incluido en el componente NFS.

En la Figura 2.4 se muestra la arquitectura de la solución en modalidad de alta disponibilidad. Existe una primera capa, la cual, da la cara al usuario final, esta capa corresponde a los balanceadores de carga que pueden ser instalados en *software* o *hardware*. En *hardware* existen fabricantes con gran reconocimiento y trayectoria como son Citrix y F5. En *software* existen productos *open source* y licenciados.

La lógica de funcionamiento de los balanceadores de carga es recibir peticiones HTTP o HTTPS de los usuarios y enviarlas hacia dos o más servidores de aplicación. La distribución de solicitudes se realiza utilizando los algoritmos *Round Robin*², *Weighted Round Robin*³, *Least Connections*⁴, *Least Response Time*⁵.

IBM tiene su propio balanceador en *software* que se denomina *Load Balancer* y que forma parte de la *suite* de productos *WebSphere Edge Components*. Este balanceador puede ser configurado en alta disponibilidad en modalidad activo-pasivo.

El balanceador utiliza una dirección IP virtual que se configura a nivel de aplicación, no está asociada a un dispositivo de red, no debe ser utilizada en otro dispositivo y en el caso de usar DHCP se la debe excluir del rango de direcciones disponibles.

La dirección IP Virtual debe ser configurada a nivel de *loopback* en los servidores a balancear. Para la presente arquitectura, los servidores a balancear son los servidores web; pueden configurarse más de dos servidores web que brinden el servicio.

La segunda capa corresponde al componente NFS. Este componente es configurado en alta disponibilidad en modalidad activo-activo utilizando el *software Tivoli Automation for Multiplatforms*.

² Round Robin, selecciona los elementos de un grupo de forma equitativa y ordenada, sin importar la disponibilidad del elemento.

³ Weighted Round Robin, selecciona los elementos de un grupo dando prioridad a los elementos que tienen asignado un mayor peso.

⁴ Least Connections, selecciona los elementos de un grupo de acuerdo a la disponibilidad del mismo, basado en la menor cantidad de conexiones utilizadas.

⁵ Least Response Time, selecciona los elementos de un grupo de acuerdo a la disponibilidad del mismo, basado en el menor tiempo de respuesta.

La dirección virtual que se configuró en el balanceador debe ser configurada a nivel de *loopback* en estos nodos NFS. Con esta configuración el balanceador puede enviar las solicitudes hacia el *cluster* de servidores balanceados.

Los servidores NFS, enviarán las solicitudes hacia el *cluster* de servidores TSAM usando el algoritmo de pesos denominado *Weighted Round Robin*, en este punto también existe un segundo nivel de balanceo de carga.

En los servidores denominados NFS también se tienen implementado la funcionalidad de un servidor de correo y repositorio de archivos que igualmente están configurados en alta disponibilidad para lo cual también se usa *Tivoli Automation for Multiplatforms*. Esta alta disponibilidad se la realiza a nivel de servicios.

La tercera capa muestra los servidores TSAM en los cuales se concentra todo el procesamiento de aprovisionamiento de máquinas virtuales. *Tivoli Automation for Multiplatforms* permite la configuración en alta disponibilidad en modo activo-activo de los servidores TSAM.

Todas las solicitudes de los usuarios llegan a TSAM, punto final que realizará el procesamiento de las mismas. Los servidores TSAM son el punto principal de la solución ISDM, en los cuales, se tienen los portales web de administración y auto-servicio.

TSAM requiere del registro, procesamiento y administración de *templates* de máquinas virtuales. La funcionalidad de *templates* se la obtiene del componente de *software VMware vCenter Server*. Este componente se instala en un servidor adicional en modalidad *stand-alone*.

Para la instalación del componente *VMware vCenter* se requiere la disposición de un servidor DNS, el cual, se instala en otro servidor adicional en modalidad *stand-alone*. *VMware vCenter Server* administra y controla uno o varios hosts ESXi. El servidor DNS cumple con la función de servidor de nombres de dominio. Una de las mejores prácticas en la implementación de proyectos de tecnología es utilizar nombres de dominio, esto debido a que generalmente la dirección IP se

expone a frecuentes cambios por motivos varios, mientras que el nombre del host se mantiene.

Se tiene a disposición un servidor físico, por lo cual, es el único que se administra y que pertenece al *cluster* de hosts. En el posible escenario que se desee crecer en capacidad de recursos se puede añadir uno o varios hosts físicos y registrarlos en el *cluster*; esta configuración es relativamente sencilla de ejecutarla. En el componente TSAM es necesario realizar el descubrimiento y registro de este nuevo host para hacer uso de sus recursos.

La cuarta capa corresponde a los componentes ITM y TUAM. Estos servidores son instalados en modalidad *stand-alone*, ya que no representan un punto crítico de funcionamiento para la continuidad del servicio. Es decir, si estos servidores dejan de funcionar, el aprovisionamiento de máquinas virtuales no se verá interrumpido o afectado.

Adicionalmente, el costo de instalación, administración y mantenimiento no justifica la instalación de este componente en modalidad de alta disponibilidad. Sin embargo, para justificar la característica de alta disponibilidad de estos dos componentes, se puede hacer uso de la característica de vMotion de VMware, la cual, permite a las máquinas virtuales moverse entre hosts físicos, los cuales están asociados a un *cluster*.

Todos los componentes anteriormente mencionados se instalan sobre el host ESXi. Este componente es el *hypervisor* que se utiliza como sistema operativo base del host principal, sobre este se instala toda la estructura de máquinas virtuales que forman la solución ISDM incluyendo las máquinas virtuales que serán aprovisionadas a los usuarios finales del servicio.

Finalmente, en la arquitectura planteada existen cuatro componentes o servidores que se disponen en modalidad *stand-alone*, estos son: *VMware vCenter*, DNS, ITM y TUAM.

Para asegurar la alta disponibilidad de estos componentes se plantea hacer uso de la funcionalidad *vMotion* de *VMware*. *vMotion* permite mover una o varias máquinas virtuales desde un servidor físico a otro, sin detener el servicio.

La máquina virtual mantiene la configuración de red y conexiones, asegurando un proceso de migración sin problemas. De esta manera todos los componentes instalados planteados disponen de una estrategia de alta disponibilidad, inclusive los instalados en modalidad de alta disponibilidad, tendrán una estrategia adicional a nivel de VMware con vMotion.

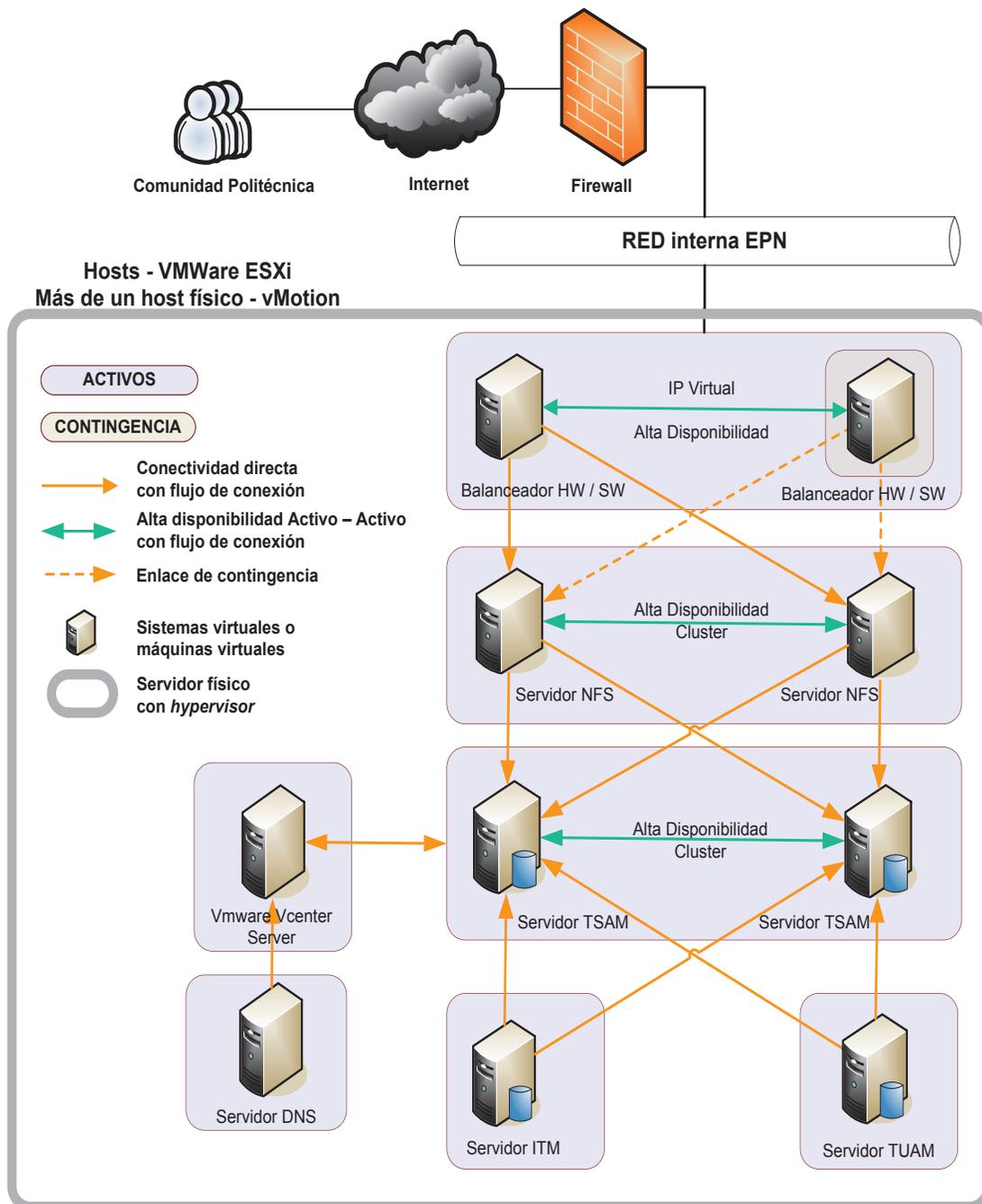


Figura 2.4: Configuración en Alta Disponibilidad

2.3.3.2. Configuración *stand-alone*

La configuración en modalidad *stand-alone* plantea un solo servidor virtual o físico que cumpla con una determinada función. Esta configuración es adecuada para ambientes de desarrollo, pruebas, demo o sistemas pilotos como es el caso del presente Proyecto de Titulación.

Esta configuración incluye todos los componentes necesarios y requeridos para la implementación del proyecto. El presente Proyecto de Titulación se limita a realizar una implementación piloto que puede servir como punto de partida para realizar una instalación real en ambiente de producción, al tratarse de una prueba piloto esta es la configuración a utilizar durante la fase de implementación. La inversión de recursos económicos, administrativos y mantenimiento no justifican la implementación de un sistema en alta disponibilidad.

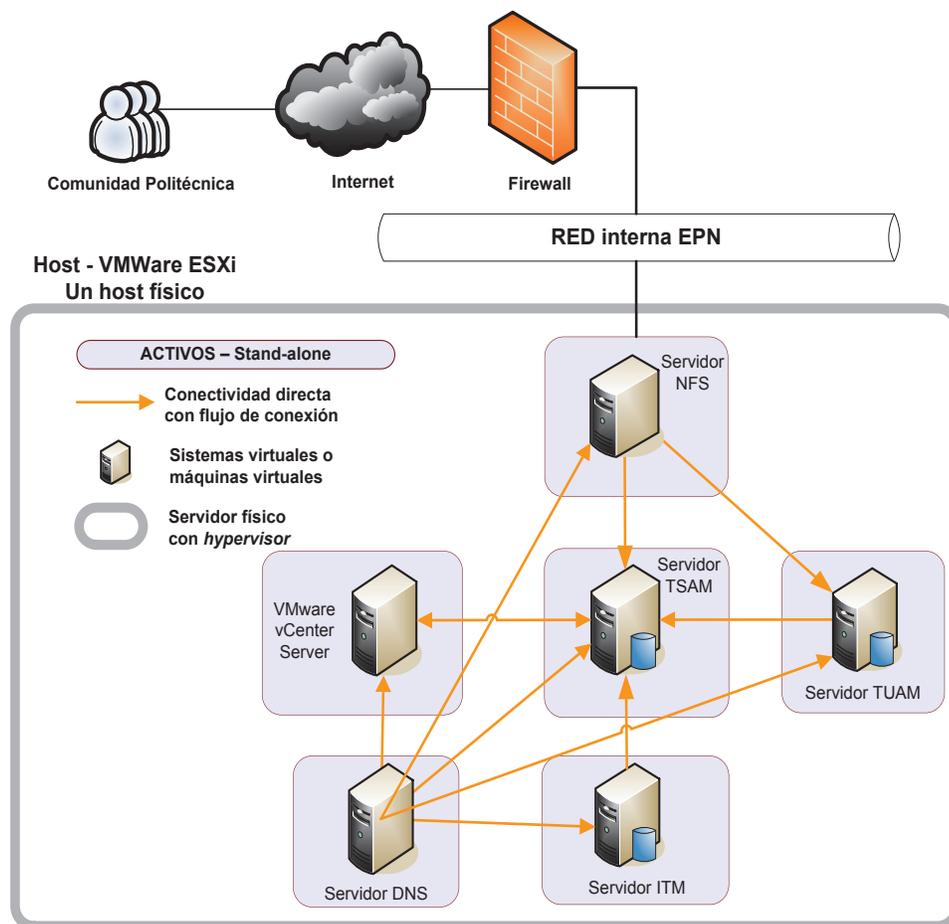


Figura 2.5: Configuración *stand-alone*

Las funcionalidades de cada uno de los componentes son exactamente las mismas que se explicaron en el ítem anterior. Como resumen, se dispone de un servidor NFS que da la cara al usuario y envía todas las solicitudes al servidor TSAM para su correspondiente procesamiento. Este procesa todos los requerimientos siguiendo un flujo de trabajo. TSAM requiere de un servidor *VMware VCenter Server* para la gestión de *templates* y *clusters*. El servidor DNS es requerido por el componente *VMware VCenter Server*. Finalmente, se tiene un servidor ITM para el monitoreo y un servidor TUAM para las estadísticas de utilización. En la Figura 2.5 se muestra la arquitectura en modalidad *stand-alone*.

2.4. REQUERIMIENTOS Y DEFINICIÓN DEL PROBLEMA

La carrera de Electrónica y Redes de Información, de la Escuela Politécnica Nacional, al momento cuenta con dos laboratorios equipados con alrededor de 20 computadores cada uno.

Los laboratorios están destinados a atender los requerimientos de algunas cátedras de la carrera de Electrónica y Redes de Información, por lo que cada computador cuenta con programas específicos por asignatura. Las aplicaciones más utilizadas por los estudiantes de la carrera son: Microsoft SQL Server 2012, Visual Studio, Matlab, NS-3, Cisco Packet Tracer y compiladores.

Se prepara una imagen virtual con las aplicaciones requeridas por las asignaturas, estas aplicaciones son instaladas de manera manual. En el caso de instalar aplicaciones adicionales a lo preparado en la imagen virtual, la instalación se realiza por red en el caso que sea posible o de forma manual utilizando un servidor FTP (*File Transfer Protocol*) como repositorio de los binarios de instalación. La instalación desatendida está limitada por las características del *software* que se desea utilizar.

En el caso de requerir dos distribuciones de sistema operativo, las estaciones de trabajo están particionadas con el objetivo de contar con un “ambiente” (Sistema operativo, aplicaciones) para cada asignatura que utiliza diferente sistema operativo. La configuración de *hardware* es: 4-8 GB de memoria RAM, 250-500

GB de espacio en disco, procesador Intel core i5-i7. Cuentan con un usuario y contraseña genérico por asignatura, para controlar el acceso a los diferentes “ambientes”.

Estas estaciones de trabajo son constantemente formateadas y se mantiene su estado de configuración, mediante el método de congelamiento o *freezer*. Una vez que una máquina tenga todas las aplicaciones instaladas, se procede a clonarla para que de esta manera todas las estaciones del laboratorio tengan el mismo estado de instalación y configuración. El proceso de clonación toma alrededor de 45 minutos por máquina, haciendo que el proceso completo pueda llegar a durar hasta 8 horas, considerando tareas en paralelo.

Los estudiantes deben compartir las estaciones de trabajo en la mayoría de clases. Si un estudiante decide llevar su máquina personal, es él el encargado y responsable de instalar las aplicaciones que se requieran durante la clase. De la misma manera deberá buscar la forma de adquirir los binarios de instalación de las aplicaciones y de sistema operativo, inclusive.

La solución ISDM tiene la funcionalidad de instalar aplicaciones de forma automática. Esta instalación se configura por única vez y desde un portal web de auto-servicio los usuarios pueden configurar la máquina virtual que deseen utilizar.

En este proceso se selecciona el sistema operativo, los recursos de *hardware* y las aplicaciones. Las aplicaciones van a formar parte de un catálogo de *software*. El catálogo de *software*, los recursos máximos de memoria RAM, disco y procesador estarán disponibles durante el proceso de aprovisionamiento.

Al disponer de una cantidad limitada de estaciones de trabajo, se reduce la posibilidad de acceso a las mismas dependiendo de los horarios y cantidad de asignaturas. Los recursos de *hardware* no pueden ser incrementados con facilidad, ya que se depende de una inversión económica y de la capacidad total del *hardware*. Con la solución ISDM se puede hacer uso de máquinas virtuales desde cualquier sitio físico dentro y fuera de la Universidad, es decir, inclusive desde la casa de cada uno de los estudiantes o docentes.

La configuración de los *hosts* físicos es altamente escalable dependiendo de las necesidades reales que demanden las aplicaciones, el sistema de aprovisionamiento ISDM debe ser configurado de acuerdo al *hardware* disponible.

El estado de una máquina virtual aprovisionada se puede almacenar en modo de plantilla o *template*. Esta plantilla puede ser utilizada para un próximo aprovisionamiento.

El aprovisionamiento de las máquinas virtuales tiene un tiempo promedio de 60 minutos. Dentro de este proceso se incluye la aprobación del administrador de la solución ISDM. Esta aprobación permite validar solicitudes viables, reales y que se ajusten a las necesidades de una determinada asignatura o proyecto de titulación.

Un docente o administrador tiene la posibilidad de generar un aprovisionamiento masivo para todos los estudiantes de su asignatura, es decir, él puede generar 30 máquinas virtuales con la configuración que sea requerida a nivel de *hardware* y *software*. Una máquina virtual aprovisionada puede ser replicada en modo de plantilla.

El aprovisionamiento es personalizado para cada uno de los estudiantes. Este aprovisionamiento no solo puede ser realizado para las asignaturas, en su lugar, pueden hacer uso de este servicio para utilizarlo en proyectos de titulación.

En cada aprovisionamiento se le asignará un usuario administrador y su contraseña. Estas credenciales serán enviadas por medio de correo electrónico al usuario que solicita el aprovisionamiento.

En caso de ser rechazada la solicitud, el usuario también será notificado vía correo electrónico. Los binarios de instalación correspondientes al catálogo de aplicaciones estarán disponibles en el servidor de aprovisionamiento, desde este servidor serán copiados y utilizados de forma transparente en cada aprovisionamiento realizado por los usuarios de la plataforma implementada. El usuario no tiene necesidad de adquirir los instaladores de los productos de *software*.

2.4.1. REQUERIMIENTOS DE INSTALACIÓN

A continuación se indican los requerimientos previos a la instalación del sistema de aprovisionamiento. Los requerimientos están orientados a la arquitectura de un servidor *System x 3500 M4*. Este servidor será utilizado durante las fases de implementación, configuración y pruebas.

2.4.1.1. Requerimientos de *Hardware*

ISDM tiene la capacidad de aprovisionar y gestionar máquinas virtuales en diferentes plataformas, sin embargo, se acotará a la plataforma System X debido a que ésta será utilizada en el desarrollo de este proyecto.

El dimensionamiento es determinante para estimar los recursos requeridos durante la implementación de un sistema informático.

Este es un proceso que se basa en la concurrencia de los usuarios finales, experiencia de proyectos ejecutados y utilización de herramientas de estrés o carga de trabajo.

De esta forma los fabricantes de productos de *software* publican en sus sitios web los requerimientos estimados, los cuales, pueden tener ciertas variaciones dependiendo del proyecto.

Los requerimientos de *hardware* para el entorno administrado dependen del número de máquinas virtuales que se proyecta desplegar y también está asociado a la cantidad máxima de recursos que se asignarán a cada máquina virtual.

Para el dimensionamiento del entorno administrado se incluyeron tres asignaturas con diez estudiantes en cada una, dando un total de treinta usuarios.

Estos usuarios podrán acceder a servidores virtuales con recursos de *hardware* más que suficientes para las actividades que se realizan en los laboratorios. Las máquinas virtuales aprovisionadas pueden disponer de 8 GB en RAM, 80 GB en disco y una milésima parte de procesador físico. Los requerimientos para las estaciones de trabajo se muestran en la Tabla 2.3.

Esta configuración de *hardware* corresponde al conjunto de máquinas virtuales que pueden ser asignadas para la asignatura Base de Datos.

Aplicaciones	% usado de procesador	Memoria [GB]	Disco [GB]
Sistema Operativo	20%	1	30
Base de Datos SQL	35%	4	25
Utilitarios (Editores de texto, imagen, video)	25%	1	15
Adicionales	10%	1	5
Espacio libre	10%	1	5

Tabla 2.3: Requerimientos de estación de trabajo

Para poder soportar los requerimientos para la demostración del sistema piloto implementado en el presente Proyecto de Titulación es necesario de un servidor físico con las siguientes características. Estas características están estimadas considerando que el servicio se brindará de forma simultánea a diez usuarios de tres asignaturas:

- 272 GB en RAM
- 2.4 TB en disco
- 9 *cores* en procesador

El servidor *System X 3500 M4* que se utilizará para el sistema piloto cumple con estas capacidades. El número de usuarios puede crecer en la proporción que la asignación de recursos de las máquinas virtuales disminuya, debido a que quedarán recursos de *hardware* disponibles para más máquinas virtuales.

Los requerimientos de *hardware* para el entorno de administración se muestran en la Tabla 2.4. Esta configuración de *hardware* corresponde al conjunto de máquinas virtuales que forman la solución ISDM. Este dimensionamiento viene establecido por el fabricante de la solución, documentado en el sitio web.

Servidor	Tipo de procesador	Número de cores	Memoria [GB]	Disco [GB]
TSAM	Xeon E5-2609 (2.4GHz 80W)	4	16	130
ITM		2	6	100
TUAM		2	6	25
NFS		2	4	25

Tabla 2.4: Requerimientos de *hardware* [112]

2.4.1.2. Requerimientos de *Software*

Para el entorno administrado se utilizará el *hypervisor* VMware vSphere 5.1. Esta versión se utiliza debido a que fue la última versión disponible al inicio del presente Proyecto de Titulación. Sobre el servidor físico System X 3500 M4 se instalará como sistema operativo base el servidor VMware ESXi.

Adicionalmente, es necesario disponer de un servidor VMware VCenter, el cual, será instalado como una máquina virtual ejecutándose sobre el servidor ESXi. También, se requiere de un usuario con privilegios de administrador para la conexión con el servidor VMware VCenter. Finalmente, se requiere la creación de un Centro de Datos, en el cual, se desplegarán las máquinas virtuales aprovisionadas.

Para el entorno de administración se requiere una estación de trabajo con sistema operativo Windows 7 y que tenga instalado el explorador Firefox o Internet Explorer. En este equipo se ejecutará el ayudante de instalación, el cual, permite

la creación de los archivos de configuración de los cuatro componentes de la solución. La solución ISDM es un arreglo de máquinas virtuales empaquetadas que es implementado directamente sobre un ambiente virtual.

El presente Proyecto de Titulación se plantea aplicar para tres asignaturas: Base de Datos, Sistemas Operativos y Redes de Área Local. Las aplicaciones a utilizar son: Microsoft SQL Server 2012, Microsoft SQL Server 2014, Cisco Packet Tracer, Wireshark y compiladores. Estas asignaturas fueron coordinadas y definidas en conjunto con el coordinador encargado de la carrera de Electrónica y Redes de Información de ese entonces.

CAPÍTULO 3

DISEÑO, IMPLEMENTACIÓN DEL SISTEMA DE APROVISIONAMIENTO

Existen múltiples factores que influyen en la correcta implementación y funcionamiento de un sistema informático. Cada uno de los factores tiene un grado de importancia relevante a las diferentes fases de un proyecto. Durante la ejecución de una implementación, un proyecto se ve dividido en cuatro grandes y complejas fases claramente identificadas: análisis, diseño, implementación y soporte.

En la primera fase se especifican los requerimientos, necesidades, expectativas y se establece el alcance del proyecto a desarrollar. Esta información puede ser extraída mediante talleres técnicos, para lo cual se requiere la participación de todos los responsables involucrados en el proyecto. Dada la disponibilidad de varias herramientas de teleconferencia estos talleres no necesariamente deben ser realizados en forma presencial, también pueden ser ejecutados de manera remota.

En la fase de diseño se definen los componentes, arquitectura, roles de usuarios y las integraciones a desarrollar. Estas definiciones dan como resultado un diagrama de arquitectura, el mismo que será utilizado como punto de partida para la implementación del proyecto. Una arquitectura de tecnología se compone de *hardware* y *software*, es necesario realizar una definición acertada y precisa, ya que tienen un impacto directo en las actividades de implementación del sistema.

En la fase de implementación se valida el alcance, la arquitectura, los requerimientos y se da luz verde para el inicio del despliegue del proyecto. Estas validaciones son realmente necesarias, dado a que generalmente existe una considerable diferencia de tiempo y recursos entre las fases de diseño y ejecución de la implementación. Esta fase es la más compleja de todas, ya que, todo lo establecido en las fases previas debe encajar, engranarse y funcionar como reloj para lograr los objetivos planteados.

Finalmente, existe una fase de soporte. Esta fase principalmente se utiliza para dar seguimiento, continuidad al proyecto, aplicar actualizaciones, realizar ajustes, monitorear el rendimiento y recomendar mejores prácticas para una correcta administración.

La correcta administración de un sistema de tecnología incide en el uso, desarrollo de la solución, la percepción de la calidad del servicio que el usuario recibe, e inclusive permite el crecimiento a escalas mayores del sistema implementado, permitiendo extender el servicio brindado a otras áreas del negocio.

Para el presente proyecto, las principales definiciones que se realizaron para la implementación de la solución son el diseño de arquitectura, roles o permisos, configuración de componentes y lógica de funcionamiento.

3.1. DISEÑO

En esta sección se utilizan las definiciones de arquitectura y requerimientos de la solución a implementar que se documentaron en los numerales 2.3 y 2.4 del capítulo 2. Es necesario validar estas definiciones antes de realizar la implementación del cualquier sistema.

Adicionalmente, se establece el modelo de gestión de seguridad y flujo de trabajo que se acopla a las actuales necesidades del laboratorio de la Carrera de Ingeniería en Electrónica y Redes de Información de la Escuela Politécnica Nacional. Finalmente, se implementa y configura el sistema aplicando todas las definiciones previamente realizadas.

3.1.1. ARQUITECTURA

La Figura 2.5 muestra la arquitectura en modalidad *stand-alone*. Este diagrama de arquitectura se utiliza durante esta fase de implementación. En el acápite 2.3.2 se detalla los componentes que forman parte de la solución y en el 2.3.3.2 se define la arquitectura en modalidad *stand-alone*.

3.1.2. GESTIÓN DE SEGURIDAD

Un modelo de gestión de seguridad tiene dos conceptos claramente definidos: autenticación y autorización. Estos dos conceptos son comúnmente utilizados para gestionar la seguridad de aplicaciones de sistemas informáticos, siendo utilizados en ambientes empresariales, académicos o de cualquier índole que se requiera el control de acceso.

Autenticación es el proceso que permite validar las credenciales, que generalmente son la combinación de un nombre de usuario y contraseña. Estos datos son ingresados en campos editables dentro de un portal previo al ingreso de una determinada aplicación, que brinda uno o varios servicios.

Para el presente proyecto, los docentes de cátedra tendrán el acceso correspondiente a la plataforma de aprovisionamiento y serán ellos los encargados de distribuir y otorgar acceso a sus respectivos estudiantes.

Autorización es la definición de roles con los privilegios o permisos que un usuario tendrá al ingresar al portal de servicios. Esencialmente, permite el control de las acciones que un usuario tiene al disponer de permisos de ingreso al portal de auto-servicio, por ejemplo, permisos de lectura o escritura.

Los dos procesos se los pueden federar⁶ a un LDAP (*Lightweight Directory Access Protocol*); el producto de *software* utilizado en el presente Proyecto de Titulación es *Microsoft Active Directory*. Los dos procesos serán ejecutados de forma interna por el componente TSAM; sin embargo, existe la posibilidad de realizar la integración con un directorio activo. Esta integración se recomienda realizarla después de la instalación y configuración de todos los componentes de la solución para aislar un posible punto de falla o conflicto.

Cada docente de cátedra que acceda al portal de auto-servicio debe estar registrado en la aplicación TSAM, con lo cual dispondrá de un identificador único en la solución de aprovisionamiento. Este portal de auto-servicio es el que principalmente se hará uso para el despliegue de máquinas virtuales, por lo cual,

⁶ **FEDERAR** es un proceso que permite otorgar los privilegios de gestión a un tercero.

es muy importante detallar los privilegios que serán otorgados a los grupos de seguridad.

Existen siete (7) grupos de seguridad creados por defecto: administrador de la nube, administrador de la nube del cliente, gerente de la nube, aprobador, administrador de seguridad, administrador de equipo y usuario final. Estos grupos se ajustan a las necesidades de la mayoría de clientes, sin embargo, existe la posibilidad de modificarlos o crear nuevos grupos.

Administrador de la nube: Los usuarios que estén asignados a este rol son los administradores de la nube. Este rol permite la gestión de clientes, imágenes virtuales, aprobaciones, proyectos y recursos de *hardware* y *software*. El administrador de la solución debe pertenecer a este rol. Un proyecto es la creación de un requerimiento de aprovisionamiento, por ejemplo, crear 20 máquinas virtuales para la asignatura denominada Redes de Área Local.

Administrador del cliente: Son administradores de los servicios en la nube dedicados específicamente a un determinado cliente. El presente Proyecto de Titulación se orienta a un sistema piloto para tres asignaturas, cada una con diez estudiantes, sin embargo, esta solución de aprovisionamiento puede ser implementada para brindar el servicio a más de una carrera o inclusive otra universidad pueden acceder a los servicios brindados por la solución, para lo cual, se deben considerar otros aspectos que no forman parte del alcance de este proyecto. Los usuarios que pertenecen a este rol pueden gestionar usuarios, grupos de usuarios, proyectos, recursos de *hardware* y *software* que fueron creados para el cliente. El Coordinador de Carrera que utiliza el servicio puede pertenecer a este rol.

Gerente de la nube: Este rol es de tipo gerencial y tendrá permisos de lectura para monitorear el desarrollo de los proyectos. El Administrador del sistema y Coordinador de Carrera pueden pertenecer a este rol. Existen gerentes que también les gusta jugar con las soluciones que su empresa adquiere y sus empleados utilizan, por lo cual existe la posibilidad de incrementar los privilegios en la aplicación.

Aprobador: Los usuarios asignados a este rol tienen la capacidad para aprobar, cancelar y monitorear proyectos que se desarrollan en el entorno de un determinado cliente. No tiene acceso a proyectos de otros clientes.

El Administrador del sistema y Coordinador de Carrera deben pertenecer a este rol debido a que tienen todo el conocimiento de los proyectos que se desarrollan en las cátedras y también de los proyectos de investigación en la Facultad.

Administrador de seguridad: Este rol permite la gestión de los usuarios que pertenecen a un cliente. El administrador de una Facultad y de un laboratorio debe pertenecer a este rol. Este rol no tiene el control de usuarios de otros clientes.

Administrador de equipo: Los usuarios en este rol pueden emitir solicitudes de aprovisionamiento de máquinas virtuales, monitorear los proyectos, gestionar servidores y hacer uso del aprovisionamiento. Tiene permisos para cambiar contraseñas y el estado de los proyectos. El Docente debe pertenecer a este rol.

Usuarios de un equipo: Los usuarios en este rol pueden hacer uso de las máquinas virtuales aprovisionadas a su equipo de trabajo. Los estudiantes de una determinada cátedra deben pertenecer a este rol.

En la Tabla 3.1 se muestran los roles que pueden ser asignados de acuerdo a la función que cumple un usuario. Los privilegios de los roles pueden ser modificadas o puede crearse roles adicionales.

	Administrador de la nube	Administrador del cliente	Gerente	Aprobador	Administrador de seguridad	Administrador de equipo	Usuarios de un equipo
Administrador	x	x	x	x	x	x	x
Coordinador			x	x			
Docentes						x	x
Estudiantes							x

Tabla 3.1: Roles en ISDM

3.1.3. FLUJO DE TRABAJO

El aprovisionamiento de máquinas virtuales a implementar dispone de un portal de auto-servicio de fácil e intuitiva utilización. Este portal permite realizar varias acciones sobre el sistema, que dependerán de los roles que el usuario tenga asignado.

El flujo de trabajo que se muestra en la Figura 3.1 permite entender de mejor manera el procedimiento que un usuario final (docente) debe realizar para solicitar el aprovisionamiento de máquinas virtuales. En esta figura se muestran cuatro fases: Autenticación, Generación de solicitud, Aprobación y Despliegue. A continuación, se explica cada una de las fases.

La Autenticación es el proceso en el que un usuario ingresa las credenciales para acceder al portal de auto-servicio. Debe existir un procedimiento previo para el registro de los usuarios en el sistema, este proceso debe ser ejecutado por el administrador del sistema. Los docentes serán los usuarios del sistema.

El proceso de registro se lo puede incluir como parte del proceso de matriculación en una determinada cátedra, la cual necesariamente demande recursos de este sistema de aprovisionamiento.

En el proceso de Generación de Solicitud se establece el requerimiento inicial para el aprovisionamiento de máquinas virtuales que será utilizado en un determinado proyecto. En esta fase se debe ingresar el nombre del proyecto, una descripción, la cantidad de máquinas virtuales, validar la disponibilidad de recursos del entorno administrado, establecer la configuración de *hardware* y *software*, y enviar el requerimiento, el mismo deberá pasar por la Aprobación del Coordinador de Carrera.

En la fase de Aprobación el aprobador del sistema valida el requerimiento en base a la información proporcionada por el usuario y aprueba o deniega la solicitud emitida. La notificación de que existe un nuevo requerimiento le llega al Aprobador por medio del portal de auto-servicio y también vía correo electrónico.

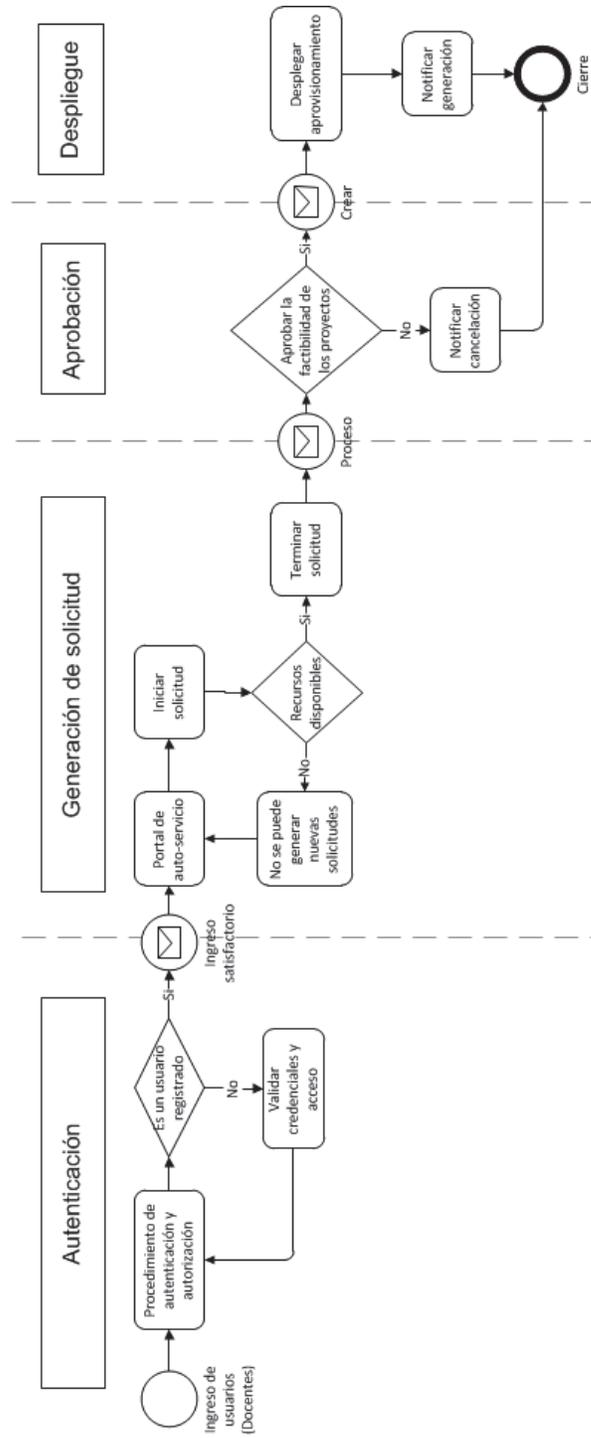


Figura 3.1: Flujo para generar una solicitud de aprovisionamiento

Finalmente, la fase de Despliegue es el proceso establecido para preparar, instalar y entregar las máquinas virtuales al usuario final que realizó el requerimiento. El usuario final es notificado por el portal o vía correo electrónico que el aprovisionamiento fue aprobado o cancelado. En caso de ser aprobado se incluye los datos de conexión como credenciales y direccionamiento de red.

3.2. INSTALACIÓN DE LA SOLUCIÓN

Luego de definir la arquitectura, requerimientos y establecer el alcance del proyecto ya se puede iniciar con la implementación de la solución de aprovisionamiento. A continuación, un resumen de las actividades realizadas que fueron documentadas previamente:

- Se realizó un comparativo de las soluciones de nube privada con el objetivo de diferenciar las funcionalidades técnicas, administrativas y económicas. Esto permite tener un abanico de alternativas para implementar soluciones informáticas que permitan la ejecución de un servicio de aprovisionamiento de máquinas virtuales sobre un entorno privado.
- Para la implementación del sistema de aprovisionamiento se consideró como punto de partida el documento de Proyecto de Titulación aprobado por la Carrera de Electrónica y Redes de Información, sin embargo, se realizaron ciertos ajustes y modificaciones de acuerdo a las condiciones reales durante el proceso de instalación.
- Se documentaron dos opciones de arquitectura y se definió utilizar la arquitectura en modalidad *stand-alone*, ésta arquitectura permite mostrar las funcionalidades de la solución de aprovisionamiento de máquinas virtuales. En la definición de la arquitectura también se detallaron las funciones de cada componente y la interacción entre los mismos.
- Se definieron los requerimientos técnicos de *hardware* y *software*. Los cuales incluyen dimensionamiento, definición de grupos de seguridad y flujo de trabajo. Estas definiciones serán utilizadas durante la implementación del sistema de aprovisionamiento.

La implementación del sistema de aprovisionamiento fue realizada desde cero. En primera instancia se instaló el servidor físico que contiene los recursos físicos de procesamiento, memoria y disco que permite demostrar el funcionamiento del servicio brindado.

Luego, se instaló el *software* de la solución de aprovisionamiento que corresponde a la parte lógica, no física, del servicio a implementar.

3.2.1. COMPONENTE DE *HARDWARE*

En la fase de definición de requerimientos de *hardware* se establecieron los respectivos requerimientos para la solución de aprovisionamiento ISDM. Para la presente fase de implementación, se dispone de un servidor físico *IBM® System x3500 M4*.

Este servidor dispone de los recursos físicos necesarios y requeridos para brindar el servicio durante las pruebas del sistema piloto de aprovisionamiento de máquinas virtuales.

La instalación del sistema se realizará sobre este servidor físico que será configurado en modalidad *stand-alone*. Las especificaciones técnicas de este equipo se encuentran en el Anexo A - Ficha técnica *IBM® System x3500 M4*.

Actualmente, el servidor utilizado no dispone de una configuración que emplee la capacidad total ofrecida por el servidor, teniendo la posibilidad de crecer en memoria, procesamiento y almacenamiento.

En la fase de armado físico del servidor se lo puede realizar siguiendo la guía de instalación del servidor que generalmente está documentado en el CD de instalación que viene con el servidor físico. Se debe tener en consideración los estándares que se deben mantener durante esta tarea, por ejemplo, mantener paridad en la instalación de las memorias RAM.

Este proceso permite poner a funcionar el *hardware*, la parte tangible de los sistemas de tecnología, que por sí solos no hacen nada. En la Figura 3.2 se muestra la imagen del servidor.



Figura3.2: Servidor IBM System x3500 M4

En el desarrollo de proyectos corporativos, la instalación del servidor físico se lo realiza de forma planificada y profesional. Existe toda una ciencia detrás de esta actividad en la que se deben realizar ciertas definiciones como: instalación eléctrica, sistema de enfriamiento, seguridad y disposición física dentro de un determinado *Datacenter* o Centro de Datos.

Para el presente proyecto piloto no se consideraron estos detalles técnicos, ya que el servidor es arrendado, la ubicación del servidor va a ser indefinida, en el sentido de que el servidor será movilizado por diferentes lugares y periodos dependiendo de las necesidades y demanda de las demostraciones de la solución implementada e inclusive de la defensa de este Proyecto de Titulación, sin embargo, estos detalles deberán ser considerados en una eventual puesta a producción del sistema.

Durante el proceso de armado físico se requiere la instalación de la memoria *RAM* (*Random-Access Memory*). En servidores con tecnología IBM se tiene una lógica de instalación de los DIMM (*Dual In-line Memory Module*) disponibles. Es recomendable seguir las instrucciones de instalación de memoria *RAM* documentadas en el CD de guía de instalación del servidor.

Otro de los componentes que se debe instalar durante el proceso de armado físico es el procesador. El servidor físico IBM que se utiliza en el proyecto dispone

de dos *sockets* para la instalación de dos procesadores físicos; sin embargo, se instalará un único procesador teniendo la posibilidad de incrementar su potencia de procesamiento en un futuro.

Luego del armado físico del servidor es necesario configurar el almacenamiento que será utilizado como repositorio de la solución de aprovisionamiento. La configuración de los discos se la puede realizar como un solo arreglo de discos o en varias particiones.

Para obtener un mejor rendimiento en ambientes que demanden alto procesamiento de escritura/lectura (I/O) se recomienda la configuración en partición de discos e inclusive se debe profundizar técnicamente en arquitectura de discos.

En el presente proyecto se configuró un arreglo de almacenamiento compuesto de cinco (5) discos de un (1) TB cada uno, dando un total de espacio disponible de aproximadamente cinco (5) TB.

Como se mencionó previamente, el *hardware* por sí solo no produce nada. Para aprovechar todo el potencial de la tecnología física se requiere de la parte lógica, o inteligencia de las soluciones tecnológicas. En el proyecto es el *hypervisor* que aporta con la parte lógica de la solución a implementar, que será el sistema operativo base del servidor físico.

Como sistema operativo base del servidor se instaló y configuró el *hypervisor* VMware ESXi 5.1. Este *hypervisor* está totalmente soportado por la solución ISDM. Sobre esta solución virtual se instala la pila de productos y máquinas virtuales que forman la solución en nube privada ISDM.

La imagen de instalación del *hypervisor* se descarga desde el sitio oficial web de VMware, para lo cual, es necesario crear una cuenta de usuario. Este archivo se copia o graba en un CD, dispositivo USB en formato de inicio de sistema para que pueda ser leído al encenderse el servidor físico. Durante el proceso de instalación del *hypervisor* ESXi se presentan pantallas solicitando la definición de los

parámetros de configuración como: almacenamiento, lenguaje, credenciales del usuario administrador y dirección IP.

VMware vCenter Server es el *software* utilizado para la gestión de uno o varios *hypervisors* ESXi. Este fue instalado en una máquina virtual sobre el mismo *hypervisor* VMware ESXi; sin embargo, como mejor práctica se recomienda que el *software* de administración y sistema administrado se instalen en servidores físicos independientes.

Durante el proceso de instalación de la aplicación VMware vCenter se presentan pantallas para configurar los requerimientos como: *software* requerido, lenguaje, base de datos, credenciales de usuario administrador, modalidad de instalación, puertos y direcciones URL de los servicios que utiliza. VMware vCenter se requiere para el manejo de plantillas en las máquinas virtuales.

VMware vSphere Client 5.1 es una aplicación cliente para la administración de la infraestructura VMware. Esta aplicación puede ser ejecutada desde una estación de trabajo que tenga conectividad a los servidores VMware, con lo cual se puede configurar el direccionamiento IP, tareas de administración de *Datacenters*, *clusters*, servidores físicos, máquinas virtuales, plantillas virtuales, entre otros.

Los pasos que se ejecutaron para realizar la instalación del sistema operativo y *software* de administración están especificados en el Anexo B – Instalación del componente de *hardware*.

3.2.2. COMPONENTE DE SOFTWARE

En la definición de arquitectura se definieron las funcionalidades e interacciones de los componentes de *software* que se requieren para la implementación de la solución de aprovisionamiento. En esta sección se especifican las actividades macro que se deben realizar para la instalación.

La configuración de las máquinas virtuales se realiza con archivos en formato XML (Extensible Markup Language); estos archivos pueden ser generados utilizando un ayudante de configuración o de forma manual.

La solución ISDM es el resultado de la integración de cuatro imágenes virtuales pre-instaladas sobre el sistema operativo SUSE Linux y distribuidas en formato OVF (Open Virtualization Format), las cuales deben ser re-configuradas de acuerdo al entorno de trabajo en el que va a ser utilizado.

En el presente Proyecto se utilizó el ayudante de configuración que presenta pantallas para definir las variables que se van a personalizar. El resultado de esta herramienta son cuatro archivos en formato XML. Cada archivo contiene los parámetros que se ingresaron; existen parámetros que son comunes para las cuatro imágenes virtuales que se ingresan una vez al inicio de la configuración, como por ejemplo, máscara de red, servidor de dominio, puerta de enlace, entre otros.

Por facilidad de interacción con una interfaz gráfica este procedimiento se ejecuta sobre cualquier plataforma Windows, ya que este sistema operativo tiene una interfaz amigable para el usuario final.

Las cuatro imágenes virtuales vienen en formato comprimido divididas en varios archivos en formato TAR⁷ (Tape Archive). Se recomienda utilizar una plataforma Linux para realizar el proceso de descompresión de cada imagen. Este procedimiento permite unir, descomprimir los archivos y obtener como resultado cuatro archivos en formato OVF.

OVF⁸ es un formato estándar de imágenes virtuales de las principales empresas que se dedican a sistemas virtuales, facilitando la creación y distribución de ambientes virtuales. Estos archivos OVF son importados en la infraestructura virtual de VMware, proceso mediante el cual se crean las cuatro máquinas virtuales dentro del clúster definido en el gestor virtual VMware vCenter Server.

Para realizar la configuración de las máquinas virtuales se debe crear una imagen en formato ISO⁹ que incluye el archivo de configuración y el certificado de

⁷ **TAR** es un formato de archivo usado en sistemas UNIX que permite almacenar archivos y carpetas en un único archivo.

⁸ **OVF** es un formato estándar de imágenes virtuales que permite la distribución de máquinas virtuales.

⁹ **ISO** es un formato estándar de imagen para almacenar una copia de un sistema de archivos, generalmente un disco óptico.

conexión del componente VMware vCenter Server. El certificado de conexión del VMware vCenter Server se denomina *ru1.crt*, y permite crear una conexión de confianza entre las máquinas virtuales y el gestor de máquinas virtuales. El archivo de configuración fue generado previamente en formato XML y el certificado se encuentra en el *hypervisor* ESXi. A continuación, el código a manera de ejemplo del contenido del archivo de configuración en formato XML.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Environment xmlns="http://schemas.dmtf.org/ovf/environment/1" xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1"
xmlns:ovfenv="http://schemas.dmtf.org/ovf/environment/1" xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_ResourceAllocationSettingData" xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_VirtualSystemSettingData" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ovf:id="ISDM-
TIVSAM">
  <PropertySection>
    <Property ovfenv:key="ConfigSC.provider" ovfenv:value="vmware" />
    <Property ovfenv:key="ConfigSC.url" ovfenv:value="https://192.168.1.5/sdk" />
    <Property ovfenv:key="ConfigSC.username" ovfenv:value="root" />
    <Property ovfenv:key="ConfigSC.password" ovfenv:value="Tiv0li4dmin" />
    <Property ovfenv:key="ConfigSC.datacenter" ovfenv:value="ha-datacenter" />
    <Property ovfenv:key="ConfigSC.autostart" ovfenv:value="false" />
    <Property ovfenv:key="ConfigSC.environment" ovfenv:value="Cloud" />
    <Property ovfenv:key="ConfigVMWCert.certfile" ovfenv:value="ru1.crt" />
    <Property ovfenv:key="License.accept" ovfenv:value="yes" />
    <Property ovfenv:key="ConfigLocale.language" ovfenv:value="en" />
    <Property ovfenv:key="ConfigLocale.country" ovfenv:value="US" />
    <Property ovfenv:key="ConfigLocale.encoding" ovfenv:value="UTF-8" />
    <Property ovfenv:key="ConfigNET.bootproto" ovfenv:value="static" />
    <Property ovfenv:key="ConfigNET.hostname" ovfenv:value="tivsam01" />
    <Property ovfenv:key="ConfigNET.domain" ovfenv:value="edu.ec" />
    <Property ovfenv:key="ConfigNET.ipaddr" ovfenv:value="192.168.1.17" />
    <Property ovfenv:key="ConfigNET.netmask" ovfenv:value="255.255.255.0" />
    <Property ovfenv:key="ConfigPWD_ROOT.username" ovfenv:value="root" />
    <Property ovfenv:key="ConfigPWD_ROOT.password" ovfenv:value="Tiv0li4dmin" />
    <Property ovfenv:key="ConfigPWD_USER.username" ovfenv:value="virtuser" />
    <Property ovfenv:key="ConfigPWD_USER.password" ovfenv:value="Tiv0li4dmin" />
  </PropertySection>
</Environment>
```

La imagen ISO se la monta en el dispositivo óptico virtual CD/DVD y se establece como dispositivo de arranque (antes del disco). Al iniciarse por primera vez cada máquina virtual, se inicia el procedimiento de configuración no asistido. Una vez finalizada la ejecución de este procedimiento las máquinas virtuales se reinician y están preparadas para la fase de integración.

Estas cuatro máquinas virtuales se comunican entre sí utilizando la tecnología Service Connection Engine (SCE). Esta tecnología fue desarrollada por IBM para establecer la comunicación funcional entre las diferentes máquinas virtuales que forman parte de una determinada solución, es decir, no es exclusivo de ISDM. Se basa en la creación de *sockets* y la programación de *plug-ins* que permiten configurar la integración de la solución.

SCE permite intercambiar información de red para establecer una comunicación entre las diferentes máquinas virtuales. Por ejemplo, es necesario que la aplicación TUAM conozca la dirección IP o nombre de servidor TSAM para generar reportes de utilización y facturación de las máquinas virtuales aprovisionadas.

Este procedimiento de activación e integración debe ser ejecutado en las cuatro máquinas virtuales de forma secuencial y manual, el cual terminará cuando las cuatro máquinas virtuales se comuniquen una con otra. Este procedimiento debe seguir un orden: ITM, NFS, TUAM y TSAM.

En primer lugar se inicia el servidor virtual ITM. Este servidor inicia el *socket* ITM_SOCKET. Es el único *socket* inicializado en este servidor y expone la información del servidor ITM. Este servidor se queda a la espera de la comunicación de los tres (3) servidores restantes.

El servidor NFS utiliza el *socket* NFS_SOCKET para exponer su información. NFS se comunica con ITM utilizando el *socket* ITM_SOCKET e inicia la instalación y configuración del agente de monitoreo en el servidor NFS. Con este agente el servidor ITM tiene acceso a las métricas de monitoreo del servidor NFS. Esta configuración la realiza utilizando el *plug-in* ITMAGENT_PLUG.

En el servidor NFS se espera por la publicación de los *sockets* de los otros componentes de la solución, es decir, el servidor NFS no puede terminar su configuración de integración ya que debe establecer la configuración con los servidores TUAM y TSAM.

El servidor TUAM publica el *socket* TUAM_SOCKET. Nuevamente, se utiliza este *socket* para realizar la instalación y configuración del agente de monitoreo en el servidor TUAM. Esta configuración se la realiza utilizando el *plug-in* ITMAGENT_PLUG.

Adicionalmente, el servidor TUAM se comunica con el servidor NFS utilizando el *socket* NFS_SOCKET. Por medio de este *socket* se realiza la configuración del componente TUAM para que pueda recibir peticiones desde el servidor NFS; esta configuración se la realiza utilizando el *plug-in* TUAM-NFS_PLUG.

De igual forma, el servidor NFS censa y detecta el *socket* del servidor TUAM y realiza la configuración del servicio web para que pueda procesar solicitudes HTTP hacia el servidor TUAM; esta configuración la realiza utilizando el *plug-in* NFS-TUAM_PLUG. El servidor TUAM esperará por la publicación de la información del servidor TSAM.

El servidor TSAM publica el *socket* TSAM_SOCKET con su propia información. De igual forma que en los casos anteriores, el servidor TSAM se comunica con el servidor ITM mediante el *socket* ITM_SOCKET y también configura el agente de monitoreo utilizando el *plug-in* ITMAGENT_PLUG.

Adicionalmente, el servidor TSAM censa y detecta el *socket* NFS_SOCKET publicado por el servidor NFS, realiza la configuración del servidor web para procesar solicitudes hacia los portales de TSAM y configura los repositorios de archivos utilizando el *plug-in* TSAM-NFS_PLUG.

TSAM detecta el *socket* de TUAM con lo cual le permite realizar la configuración de TUAM para que pueda acceder a las estadísticas de TSAM; esto lo realiza utilizando el *plug-in* TSAM-TUAM_PLUG. NFS y TUAM censan y detectan el *socket* publicado por TSAM y realiza la configuración de sus componentes.

El componente NFS configura el servidor web y TUAM configura la transferencia de archivos. En este punto todos los componentes terminan el proceso de configuración con lo cual finaliza el proceso SCE.

Al completar el proceso de integración se concluye con la instalación del sistema de aprovisionamiento ISDM de forma satisfactoriamente. La documentación a detalle de la instalación de *software* base de la solución ISDM están documentados en el Anexo C – Instalación de *IBM Service Delivery Manager*.

3.3. CONFIGURACIÓN DE LA SOLUCIÓN

En la presente sección se presentan los pasos macro de configuración que se deben seguir para personalizar las funcionalidades de la solución de aprovisionamiento. Los siguientes subíndices están documentados detalladamente en el Anexo D – Configuración de *IBM Service Delivery Manager*.

3.3.1. IMPORTAR EL CERTIFICADO DE SEGURIDAD

TSAM utiliza los recursos del Entorno Administrado, el Entorno Administrado es la infraestructura virtual VMware, es decir, es la infraestructura física/virtual que se pondrá a disposición de la solución de aprovisionamiento. Para establecer una comunicación segura entre los dos componentes es necesario crear un enlace de confianza. Esto se realiza utilizando el certificado de seguridad *ru1.crt* correspondiente al componente VMware vCenter Server.

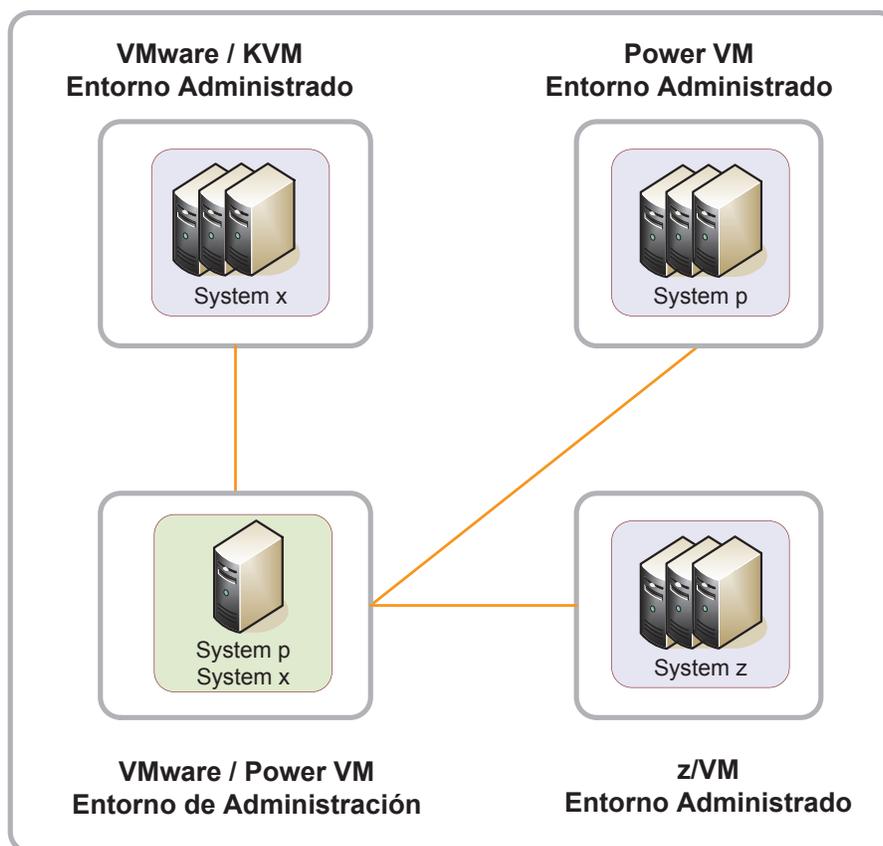


Figura 3.3: Disposición de entornos de trabajo

Dado a que el Entorno Administrado es el mismo que el Entorno de Administración se utiliza en el mismo certificado de seguridad del componente VMware vCenter Server. También existe la posibilidad de que la infraestructura virtual del Entorno Administrado sea diferente al utilizado en la instalación del Entorno de Administración. El Entorno de Administración corresponde al grupo de las cuatro máquinas virtuales que forman la solución ISDM.

En la Figura 3.3 se ilustran las opciones de disposición de los Entornos de Administración y Administrados. Los dos entornos pueden estar configurados en entornos de trabajos diferentes e inclusive dispuestos en localidades geográficamente distantes. Para el presente proyecto se disponen los dos entornos sobre un único servidor System x virtualizado con VMware.

3.3.2. CONFIGURACIÓN DE CORREO ELECTRÓNICO

El servidor de correo electrónico se instala y configura sobre el servidor virtual NFS. Esta funcionalidad es muy importante ya que el componente TSAM envía notificaciones del proceso de aprovisionamiento vía correo. Inclusive las credenciales de conexión serán enviadas por este medio.

Postfix es un producto de *software* dedicado al servicio de correo electrónico y es una aplicación de código abierto. Se configuró la integración entre las aplicaciones Postfix y TSAM para la gestión de notificaciones.

Esta aplicación de correo electrónico se implementó en el servidor NFS, se crearon cuentas para los usuarios con roles de administrador, coordinador y docentes.

La instalación de esta aplicación de correo electrónico se realiza sobre el servidor NFS, el mismo dispone de un sistema operativo SUSE Linux. Para la instalación del correo electrónico se necesita de como requerimientos de paquetes RPM, modificar los archivos de configuración de acuerdo al sistema que se está instalando, iniciar los servicios a nivel de sistema operativo y configurar el acceso a los usuarios del servicio de correo electrónico.

Adicionalmente, en las estaciones de trabajo de los usuarios se debe instalar una aplicación cliente para el acceso al correo electrónico. Para el proyecto se utilizó la aplicación de código abierto denominada “Thunderbird”. Esta aplicación fue instalada sobre el servidor VMware vCenter sobre una plataforma Windows.

La última versión de Thunderbird se encuentra disponible para todos los usuarios por medio del sitio web oficial de la aplicación. [114] En la Figura 3.4 se presenta

la pantalla de bienvenida del sitio web oficial, en la cual, se muestra que la versión disponible en ese entonces era la 31.5.0 para sistema operativo Windows, sin embargo, esta aplicación está en constante cambio y generalmente se tienen nuevas versiones liberadas.

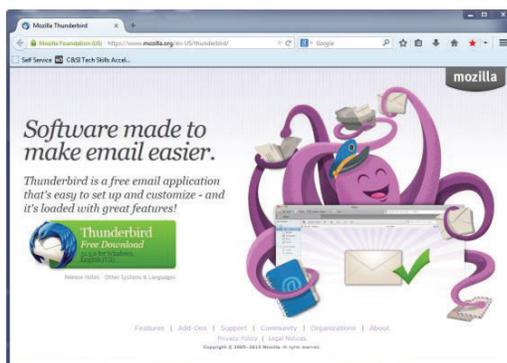


Figura 3.4: Sitio web de la aplicación Thunderbird

3.3.3. CONFIGURACIÓN DEL ENTORNO ADMINISTRADO

El Entorno Administrado es la fuente de recursos de *hardware* para las máquinas virtuales que serán aprovisionadas hacia los docentes de la comunidad politécnica. La cantidad, calidad y rendimiento de las máquinas dependerá de la disponibilidad de los recursos de este entorno. Los docentes serán los encargados de otorgar el correspondiente acceso a sus estudiantes.

Una vez realizada la instalación del *software* base de la solución ISDM se requiere el descubrimiento de los recursos del entorno administrado, para lo cual, se requiere configurar el servidor TSAM para que pueda reconocer y usar el entorno administrado.

El descubrimiento se lo realiza por medio del componente VMware vCenter Server, ya que es el componente que permite la gestión de toda la infraestructura virtual.

Para realizar esta actividad se requiere la creación de un archivo en formato XML con la definición de objetos que se desean descubrir y que serán importados dentro de la herramienta.

Este archivo XML denominado DCM_Objects.xml contiene los objetos del modelo de Centro de Datos. DCM significa Data Center Model. ISDM provee ejemplos de la definición de estos objetos DCM, estos objetos deben ser personalizados de acuerdo al ambiente de ejecución.

Existen dos secciones a modificar en la plantilla DCM: red de aprovisionamiento y parámetros del entorno administrado. En la red de aprovisionamiento se definen parámetros de red que serán utilizados durante el aprovisionamiento de máquinas virtuales. Los parámetros a definir son: nombre de VLAN, direcciones IP, *gateway*, rango de direcciones excluidas.

En la sección de los parámetros del entorno administrado se define los valores que corresponden al *hypervisor* VMware. Entre los parámetro que se definen están: nombre del *hypervisor*, identificador del repositorio virtual de almacenamiento, identificador de VLAN, dirección IP del servidor DNS, sufijo para las MV, arquitectura de la plataforma virtual, tipo de procesador, cantidad de memoria RAM, cantidad de disco, *switch* virtual. A continuación, se presenta el código correspondiente a los parámetros establecidos.

http://pic.dhe.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.tsam_7.2.4.1.doc/t_hypervisorindependentdcmitems.html

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE datacenter PUBLIC "-//Think Dynamics//DTD XML Import//EN" "xmlimport.dtd">
<!--
*****
* Licensed Materials - Property of IBM
* IBM Product Name (program number 5724-W78)
* (C) Copyright IBM Corp. 2011. All Rights Reserved.
*
* US Government Users Restricted Rights
* Use, duplication or disclosure restricted by GSA ADP Schedule
* Contract with IBM Corp.
*
* DISCLAIMER OF WARRANTIES :
*
* Permission is granted to copy and modify this Sample code, and to
* distribute modified versions provided that both the copyright
* notice,- and this permission notice and warranty disclaimer appear in all copies
* and modified versions.
*
* THIS SAMPLE CODE IS LICENSED TO YOU AS-IS.
* IBM AND ITS SUPPLIERS AND LICENSORS DISCLAIM ALL WARRANTIES, EITHER EXPRESS OR
* IMPLIED, IN SUCH SAMPLE CODE, INCLUDING THE WARRANTY OF NON-INFRINGEMENT AND THE
* IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO
* EVENT WILL IBM OR ITS LICENSORS OR SUPPLIERS BE LIABLE FOR ANY DAMAGES ARISING OUT
* OF THE USE OF OR INABILITY TO USE THE SAMPLE CODE, DISTRIBUTION OF THE SAMPLE CODE,
* OR COMBINATION OF THE SAMPLE CODE WITH ANY OTHER CODE. IN NO EVENT SHALL IBM OR ITS
* LICENSORS AND SUPPLIERS BE LIABLE FOR ANY LOST REVENUE, LOST PROFITS OR DATA, OR
* FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES,
* HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY,-, EVEN IF IBM OR ITS
* LICENSORS OR SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
*****
-->
<datacenter>
<kanaha-config>
<dcm-object id="1" type="managedSystem" name="KANAHA">
<property component="KANAHA" name="Cloud.IMAGE_FILE_REPOSITORY_ID3" value="VMwareFileRepository3"/>

```

```

</dcm-object>
</kanaha-config>
  <!-- Please note, before the xml can be imported the following data must be customized:
    For each subnetwork
      ipaddress, netmask, blocked-ranges,
      PMRDP.Net.Gateway, PMRDP.Net.Broadcast, PMRDP.Net.DomainName, PMRDP.Net.HostnamePrefix
      PMRDP.Net.VLANID (if needed)
      PMRDP.Net.VLAN_PVID (if needed)-->
<!-- data center network infrastructure template for TSAM
  For details on how to define DCM objects with XML files see TPM documentation-->
<!-- Define all involved subnets in the following section -->
<!-- Name is what you want to use to define the subnet -->
<!-- IP address is your subnet range of IP addresses to deploy -->
<!-- Netmask is the IP subnet mask address -->
<!-- Gateway value is your default route -->
<!-- This DCM object defines the subnet used for the Management VLAN -->
<subnetwork name="Cloud Management VLAN 300" ipaddress="192.168.1.0" netmask="255.255.255.0">
  <blocked-range from="192.168.1.1" to="192.168.1.150" />
  <!-- Required Parameter. Has always to be set to true -->
  <property component="KANAHA" name="PMRDP.Net.Cloud" value="true"/>
  <!-- Defines the Gateway for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.Gateway" value="192.168.1.1"/>
  <!-- Defines the broad cast address for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.Broadcast" value="192.168.1.255"/>
  <!-- Defines the VLAN id for this subnet. If external switch vlan tagging is active it is required to set this value to 0. -->
  <property component="KANAHA" name="PMRDP.Net.VLANID" value="0"/>
  <!-- Defines the domain name of the hostname for the provisioned virtual machine for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.DomainName" value="edu.ec"/>
  <!-- Defines the hostname prefix for the provisioned virtual machine for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.HostnamePrefix" value="vmw"/>
  <!-- references to the virtual switch templates. Multiple of these references must be set if this subnet is used with multiple hypervisors. -->
  <!-- Defines the Virtual switch DCM Name reference for VMware for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.VMware.SwitchTemplate" value="VM_SwitchTemplate3"/>
  <!-- Defines the destination ip address of the default route for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.DefaultRoute.Destination" value="192.168.1.1"/>
  <!-- Defines the netmask of the default route for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.DefaultRoute.NetMask" value="255.255.255.0"/>
  <!-- Defines the metric value (priority) of the default route for this subnet -->
  <property component="KANAHA" name="PMRDP.Net.DefaultRoute.Metric" value="2"/>
</subnetwork>
<switch name="VM_SwitchTemplate3" locale="en" failed="false" number-of-ports="0">
  <!-- Required Parameter which are platform independend -->
  <property component="KANAHA" name="PMRDP.Net.Cloud" value="true"/>
  <!-- Required Parameter. Has always to be set to Virtual Switch Template -->
  <property component="KANAHA" name="PMRDP.Net.SwitchType" value="Virtual Switch Template3"/>
  <!--The following properties are specific to the VMWare configuration -->
  <property component="KANAHA" name="PMRDP.Net.BridgePrefixName" value="VLAN"/>
  <!-- Specifies the virtualcenterid -->
  <property component="KANAHA" name="PMRDP.Net.HostGroupid" value="9999"/>
  <!-- Name of the VMWare virtual switch where the port group should be created on the ESX host.-->
  <property component="KANAHA" name="PMRDP.Net.VSwitchName" value="vSwitch0"/>
</switch>
<!-- Data Center Model template to define all TPM objects necessary to work with VMware backend -->
<!-- For details on how to define DCM objects with XML files see TPM documentation -->
<!-- VMware Pool Stack Definition -->
<!-- ===== -->
<!-- referenced in VMware Resource Pool Definition (please see below) -->
<software-stack name="ESXPoolStack3" vendor="IBM" stack-type="Declared">
  <software-resource-template name="Default Template3" software-resource-type="PLACEHOLDER" multiplicity-type="Zero_Or_More"
software-configuration-type="Regular" is-selected="true" is-default="true"/>
</software-stack>
<!-- VMware File Repository Definition -->
<!-- ===== -->
<!-- properties to be changed: -->
<!-- root-path of VMwareFileRepository -->
<!-- NIC name -->
<file-repository name="VMwareFileRepository3" is-device-model="Cloud File Repository" root-path="datastore1">
  <!-- To be changed: NIC name -->
  <nic name="Nic1" failed="false" managed="false" management="false" netboot-enabled="false"/>
  <property component="KANAHA" name="physicalLocation" value="9999"/>
  <property component="KANAHA" name="virtualcenterid" value="9999"/>
</file-repository>
<!-- VMware Resource Pool Definition -->
<!-- ===== -->
<!-- properties to be changed: -->
<!-- NetworkDeviceOnHost -->
<!-- SANStgPoolName -->
<spare-pool name="Esx Cloud Pool3">
  <server-template name="ESXPoolTemplate3">
    <template-software-module software-module-name="ESXPoolStack3"/>
  </server-template>
  <!-- To be changed: value -->
  <!-- Set to dcm id of imagelibrary repository used to save and restore multiple images -->
  <property component="KANAHA" name="ILRepold" value="9999"/>
  <!-- Set to true if IPs of saved images should be blocked if the server is deleted -->

```

```

<!-- Set to false if IPs of saved images should NOT be blocked if the server is deleted -->
<!-- Default: The IPs are blocked -->
<!-- property component="KANAHA" name="PMRDP.SMI.BlockIPAddressOnExit" value="true"/-->
<!-- Set to the number of images a user should be able to save -->
<!-- Default: The number of images is unlimited -->
<!-- property component="KANAHA" name="PMRDP.SMI.MaximumSavedImagesPerUser" value="3"/-->
<!-- To be changed: SANStgPoolName -->
<!-- Set to shared data stores. Multiple values are separated by semicolon. -->
<!-- e.g. value="datastore1,datastore2" -->
<property component="KANAHA" name="SANStgPoolName" value="datastore1"/>
<property component="KANAHA" name="cloud" value="true"/>
<property component="KANAHA" name="virtualcenterid" value="9999"/>
<!-- PMRDP.Net.VMware.NicAdapterType in new multiNIC Support, in legacy mode use VMware.NicAdapterType -->
<!-- This Property determines the type of adapter to be used while adding new NIC to a provisioned VM when your template has no nic -->
<!-- Permitted values are "E1000" , "Vmxnet" , "Flexible" -->
<property component="KANAHA" name="PMRDP.Net.VMware.NicAdapterType" value="E1000"/>
</spare-pool>
<!-- Virtual Center Definition -->
<!-- ===== -->
<!-- properties to be changed: -->
<!-- network-interface including IP address and net mask -->
<!-- ClusterName -->
<!-- DataCenterPath -->
<!-- HostBridgeName -->
<!-- properties which have to be adapted: -->
<!-- VC's username and password (better to change the credentials via Maximo UI after importing this dcm file) -->
<server name="visphere3" is-device-model="Cloud VMware Windows Installation" ignored-by-resource-broker="false" failed="true">
  <nic name="vNic" failed="false" macaddress="" managed="false" management="false" netboot-enabled="false">
    <!-- To be changed: VC's ipaddress and netmask -->
    <network-interface name="eth0" failed="false" managed="false" management="true" dynamic-ipaddress="false" ipaddress="192.168.1.20"
    netmask="255.255.255.0" address-space="DEFAULT" allocation="none"/>
  </nic>
  <!-- To be changed: ClusterName -->
  <!-- This is the default cluster name which TSAM will use if it cannot be discovered from the virtual center -->
  <property component="KANAHA" name="ClusterName" value="Cluster_Systemx"/>
  <!-- To be changed: DataCenterPath -->
  <property component="KANAHA" name="DataCenterPath" value="/DATACENTER"/>
  <!-- To be changed: HostBridgeName (this parameter is not used in the multi NIC network mode -->
  <property component="KANAHA" name="HostBridgeName" value="vSwitch0"/>
  <sap name="HTTPS" protocol-type="ipv4" app-protocol="HTTPS" context="VMware" port="443" auth-compulsory="false" role="host">
    <credentials search-key="VMware" is-default="false">
      <!-- Have to be changed: VC's username and password -->
      <!-- better to change the credentials via Maximo UI after importing this dcm file -->
      <password-credentials username="Administrator" password="Tivoli4dmin" is-encrypted="false"/>
    </credentials>
  </sap>
  <resource name="platform" resource-type="platform" managed="false" partitionable="false">
    <property component="KANAHA" name="platform.architecture"/>
  </resource>
  <resource name="computer" resource-type="generic" managed="false" partitionable="false">
    <property component="KANAHA" name="computer.manufacturer"/>
    <property component="KANAHA" name="computer.serialNumber"/>
    <property component="KANAHA" name="computer.type"/>
  </resource>
  <resource name="cpu" resource-type="cpu" managed="false" partitionable="false">
    <property component="KANAHA" name="cpu.family" value="amd"/>
    <property component="KANAHA" name="cpu.size" value="4"/>
    <property component="KANAHA" name="cpu.type" value="32-bit"/>
  </resource>
  <resource name="mem" resource-type="memory" managed="false" partitionable="false">
    <property component="KANAHA" name="memory.size" value="7919894528"/>
  </resource>
  <resource name="disk" resource-type="disk" managed="false" partitionable="false">
    <property component="KANAHA" name="disk.size" value="8589934592"/>
  </resource>
</server>
<storage-subsystem name="cloud-vmware-sc3" failed="false">
  <property component="KANAHA" name="vmware" value="true"/>
</storage-subsystem>
</datacenter>

```

Este archivo XML debe ser importado utilizando la interfaz de TSAM. En la aplicación TSAM se debe navegar a administración de recursos de la nube, importar el archivo y verificar que el proceso termine satisfactoriamente. En la Figura 3.5 se muestra la aplicación en la cual se importa el archivo de objetos DCM.

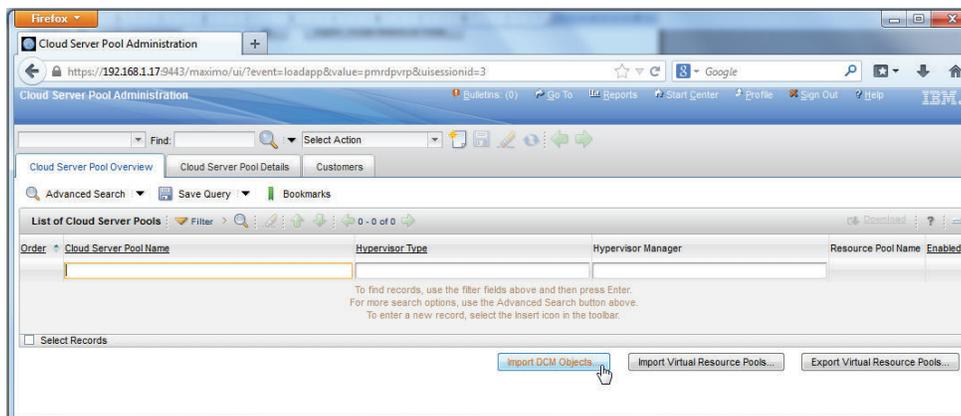


Figura 3.5: Aplicación para importar los objetos DCM

3.3.4. DEFINICIÓN DE RECURSOS DE NUBE

Es necesario definir el conjunto de recursos que estarán disponibles para el sistema de aprovisionamiento, para lo cual, se debe utilizar el archivo de propiedades *vrpool.properties*.

Este archivo permite definir el *hypervisor* que se utilizará durante el aprovisionamiento de máquinas virtuales. Los datos de este *hypervisor* son los establecidos en el ítem anterior.

Este archivo de configuración debe ser importado en TSAM navegando a administración de recursos de la nube, importar el archivo y verificar que el proceso termine satisfactoriamente. En la Figura 3.6 se muestra la aplicación en la cual se importa el archivo de configuración de recursos de red.

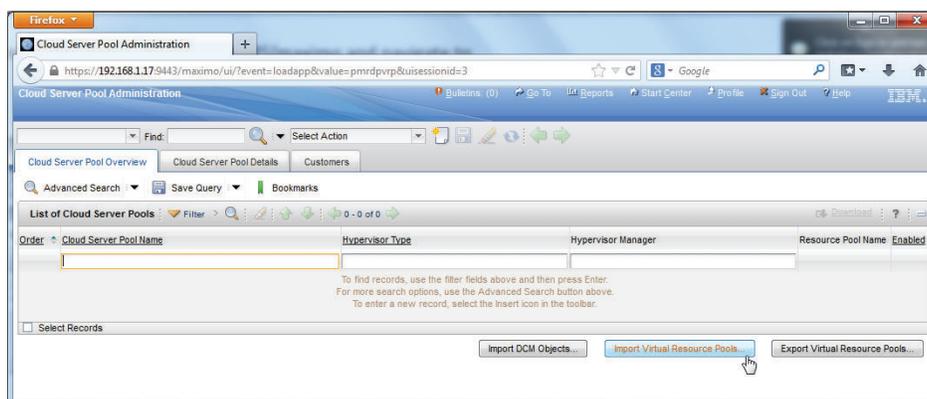


Figura 3.6: Aplicación para importar los recursos virtuales de la nube

3.3.5. CREACIÓN DE PLANTILLAS

Dentro del catálogo de *software* existen las plantillas de sistema operativo que estarán disponibles durante el aprovisionamiento de máquinas virtuales. Estas plantillas deben ser generadas e importadas en la solución ISDM.

Para el presente proyecto se crearon dos plantillas, una con sistema operativo Windows Server 2008 y otra con RedHat Enterprise Server 6.3. El usuario tendrá la posibilidad de seleccionar uno de las dos plataformas para su aprovisionamiento. Estos sistemas operativos fueron seleccionados bajo el criterio de que son los más conocidos por los autores del presente Proyecto de Titulación.

Las plantillas de los sistemas operativos deben ser creadas directamente en la aplicación VMware vCenter. Esta funcionalidad es nativa de este componente y el proceso de creación es sencillo. El procedimiento es crear una nueva máquina virtual, se puede utilizar una imagen ISO, establecer la configuración inicial de la máquina virtual y se crea una plantilla a partir de la máquina virtual creada.

Esta plantilla está creada en el Entorno Administrado y para que se registre en la solución ISDM se debe ejecutar el procedimiento de descubrimiento de recursos; este procedimiento actualizará los recursos existentes y creará los nuevos recursos. Adicionalmente, es necesario habilitar esta imagen virtual como un recurso disponible para un determinado cliente durante el proceso de solicitud de aprovisionamiento de los recursos virtuales.

3.3.6. CREACIÓN DE CATÁLOGO DE APLICACIONES

Dentro del Catálogo de Aplicaciones se definieron las aplicaciones SQL Server 2012, SQL Server 2014, Wireshark y Packet Tracer. Estas aplicaciones fueron seleccionadas a manera de prueba ya que generalmente son utilizadas en los laboratorios de la Carrera y corresponden a las tres asignaturas seleccionadas para el piloto: Base de Datos, Redes de Área Local y Sistemas Operativos.

Cada aplicación debe ser definida en la aplicación TSAM. Esta definición consta del binario de instalación, directorio de almacenamiento, flujo de trabajo para la

instalación de la aplicación, plataforma soportada, nombre y versión. Estas aplicaciones estarán disponibles durante el aprovisionamiento de un ambiente virtual y se desplegarán dependiendo de la plataforma seleccionada. Es decir, un usuario debe seleccionar la plantilla de sistema operativo, en el siguiente paso, solo se desplegarán las aplicaciones soportadas y compatibles sobre la plataforma previamente seleccionada.

Este procedimiento se puede replicar para incrementar la cantidad de aplicaciones disponibles en el catálogo, por ejemplo, se pueden incluir aplicaciones como Matlab, Visual Studio, Microsoft Office y demás aplicaciones que se consideren necesarias en el Laboratorio. Otra alternativa, es crear plantillas de sistema operativo que ya incluyan todas las aplicaciones de una determinada cátedra.

A continuación, el flujo de trabajo para instalar la aplicación SQL:

```
# -----
# Licensed Materials - Property of IBM
# 5724-F75
# (C) Copyright IBM Corp. 2003 - 2011
# All Rights Reserved
# US Government Users Restricted Rights -Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
# -----

@doc SQL-Server Windows implementation

workflow SQLServer2012_WIN_SI(in SoftwareID, in DeviceID, in SoftwareResourceTemplateID, out SoftwareResourceID) implements
SoftwareInstallable.Install LocaleInSensitive

var msicmd = "cmd /C msixec.exe -i "
var command
var Return_Code
var ReturnErrorString
var Return_Result
var TempDir = "C:\\"
log info "TempDir: " + TempDir

# Parameters
var softwareInstallableId = SoftwareID
log info "softwareInstallableId: " + softwareInstallableId

var deviceId = DeviceID
log info "deviceId: " + deviceId

var softwareResourceTemplateID = SoftwareResourceTemplateID
log info "softwareResourceTemplateID: " + softwareResourceTemplateID

var fileRepositoryId = DCMQuery(/SoftwareInstallable[@id=$softwareInstallableId]/fileRepository/@id)
log info "fileRepositoryId: " + fileRepositoryId

var softwareInstallableFilePath = DCMQuery(/SoftwareInstallable[@id=$softwareInstallableId]/file/@path)
log info "softwareInstallableFilePath: " + softwareInstallableFilePath

var softwareInstallableFileName = DCMQuery(/SoftwareInstallable[@id=$softwareInstallableId]/file/@name)
log info "softwareInstallableFileName: " + softwareInstallableFileName

Device.CopyFile("5678", "/opt/IBM/tivoli/tpm/repository/sql2012", softwareInstallableFileName, deviceId, TempDir,
softwareInstallableFileName, "default", "default", "3600")
log info "CopyFile: OK"
command = msicmd + "SqlLocalDB.msi /qn IACCEPTSSQLLOCALDBLICENSESETERMS=YES"
log info "command: " + command
Device.ExecuteCommand(deviceId, command, TempDir, "default", "3600", "ignore", Return_Code, ReturnErrorString, Return_Result)
log info "Instalación: OK"
```

CAPÍTULO 4

PRUEBAS DEL SISTEMA DE APROVISIONAMIENTO Y COSTOS DE LA REALIZACIÓN DEL PROYECTO

En el presente capítulo se muestran los resultados obtenidos durante la ejecución de las pruebas del sistema de aprovisionamiento implementado en este Proyecto de Titulación.

Las pruebas realizadas están enfocadas a verificar el estado de los servicios del motor de aprovisionamiento, mostrar el proceso para preparar un despliegue y realizar una solicitud de aprovisionamiento como usuario final.

4.1. ESTADO DE LOS SERVICIOS

A continuación, se presentan los procesos para iniciar y verificar los servicios del sistema de aprovisionamiento.

4.1.1. INICIAR LOS SERVICIOS

El servidor físico, también conocido como host, debe estar encendido y operativo. Hay que recordar que este servidor tiene instalado el *hypervisor* VMware ESXi, el cual es el sistema operativo base.

El servidor DNS y el servidor vCenter deben ser encendidos para disponer de la gestión del *hypervisor* VMware ESXi. El servidor DNS es un requerimiento para la instalación del servidor vCenter, por lo cual, los dos sistemas deben estar encendidos, el orden no es importante.

Para disponer de la solución ISDM, los servidores que forman la plataforma de aprovisionamiento deben ser encendidos, los servidores son: ITM, NFS, TUAM y TSAM.

Finalmente, se debe utilizar la aplicación cliente VMware vSphere Client e iniciar sesión usando las credenciales de administrador. En la Figura 4.1 se muestra el ingreso de credenciales acceder al servidor VCenter desde la aplicación cliente.

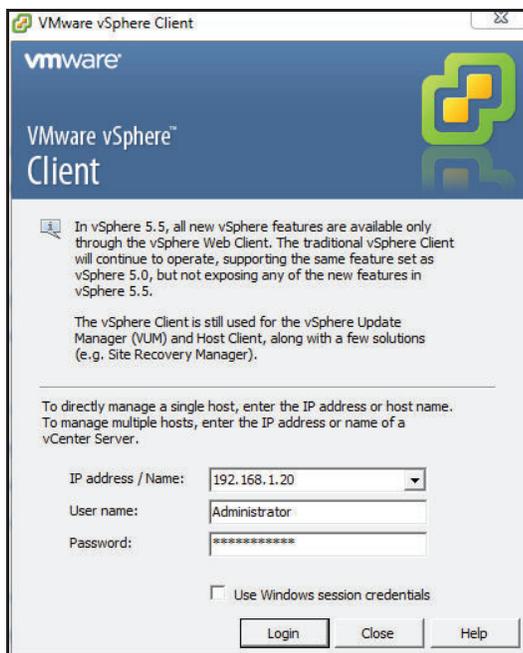


Figura 4.1: Ingreso de credenciales VMware vSphere Client

En la interfaz de la aplicación cliente se debe navegar en el panel izquierdo hacia: VcenterIBM.cloud.edu.ec > DATACENTER > ISDMITM > Power > Power on. En la Figura 4.2 se muestra la acción a ejecutar para iniciar una máquina virtual.

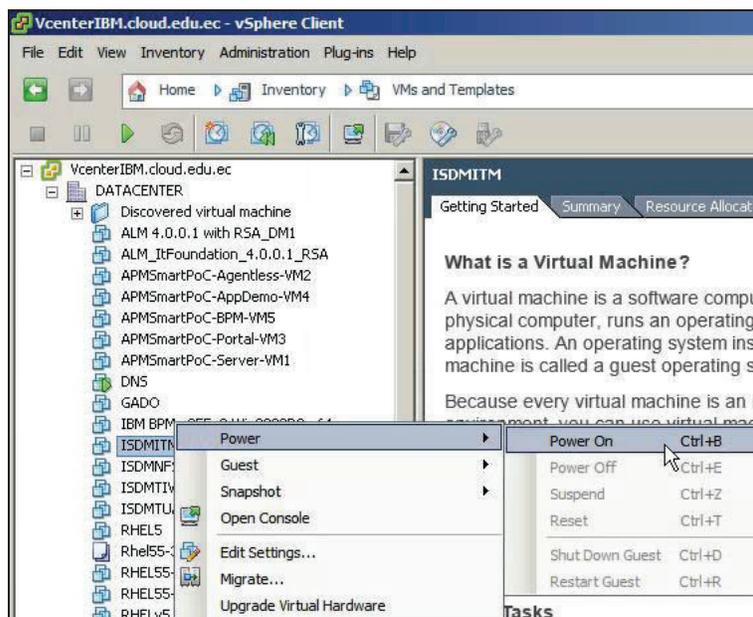


Figura 4.2: Encender el servidor virtual ITM

A continuación, se verifica la activación de las máquinas virtuales. Esto puede ser validado en la pestaña *Summary* de la sección de detalles. En el panel izquierdo, el ícono de la máquina virtual debe tener una flecha verde y los servicios en el detalle del panel derecho también deben tener íconos en color verde. En la Figura 4.3 se muestra el estado de activación de una máquina virtual.

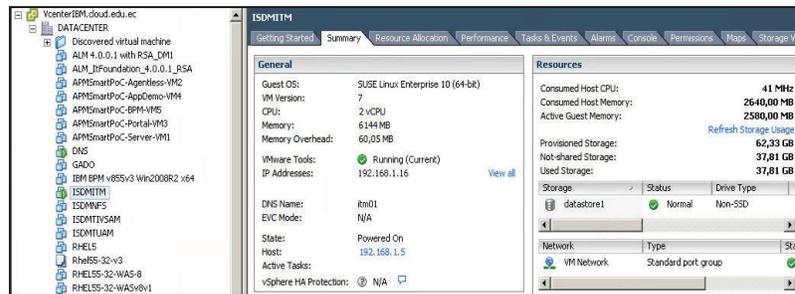


Figura 4.3: Activación de las máquinas virtuales

Se debe realizar el mismo procedimiento para las tres máquinas virtuales restantes. Es un procedimiento sencillo, el cual, conecta y sincroniza los componentes de la solución de aprovisionamiento.

4.1.2. VERIFICAR LOS SERVICIOS

Para realizar la verificación de los servicios se debe realizar un determinado procedimiento, considerando que cada máquina virtual / servidor virtual cumple con un rol específico; por lo tanto, la validación del estado de los servicios de los componentes de *software* difiere.

Para revisar el estado de los servicios en cada uno de los servidores se debe ejecutar una serie de comandos, es decir, es necesario establecer una sesión SSH hacia los servidores con sistema operativo Linux.

Primero se verifica que la base de datos de ITM se esté ejecutando correctamente. El comando `# su - db2inst1 -c "db2 list database directory"` permite listar las bases de datos creadas en la instancia. Para listar las bases de datos, la instancia de instancia de base de datos db2inst1 debe estar inicializada y totalmente operativa. En la Figura 4.4 se muestra el estado de la base de datos de ITM.

```
itm01:~ # su - db2inst1 -c "db2 list database directory"

System Database Directory

Number of entries in the directory = 1

Database 1 entry:

Database alias           = TEPS
Database name            = TEPS
Local database directory = /home/db2inst1
Database release level   = d.00
Comment                  =
Directory entry type     = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

itm01:~ # █
```

Figura 4.4: Servicio de la base de datos de ITM

En el mismo servidor ITM, se verifica que los servicios del motor de monitoreo estén inicializados y operativos, ejecutando el comando `# /opt/IBM/ITM/bin/cinfo -r`. En la Figura 4.5 se muestra el estado de los servicios de la aplicación de monitoreo ITM.

```
itm01:~ # /opt/IBM/ITM/bin/cinfo -r

***** Mon Jun 15 17:39:28 COT 2015 *****
User: root Groups: root
Host name : itm01          Installer Lvl:06.22.04.00
CandleHome: /opt/IBM/ITM
*****
Host  Prod  PID   Owner   Start  ID       ..Status
itm01 cq     6929  virtuser 16:56  None    ...running
itm01 lz     7470  virtuser 16:56  None    ...running
itm01 ms     15189 virtuser 16:57  TEMS    ...running
itm01 vm     18869 virtuser 16:58  ManageFrom ...running
itm01 kf     29375 root     17:39  None    ...running
itm01:~ # █
```

Figura 4.5: Servicios del motor de monitoreo ITM

Para verificar que el servidor web esté ejecutándose se debe utilizar el comando `# service ISDM-HTTPServers status`. En la Figura 4.6 se muestra el estado del servidor web.

```
itm01:~ # service ISDM-HTTPServers status
HTTP Server is online

itm01:~ # █
```

Figura 4.6: Estado del servidor web

Para verificar los servicios de los componentes que están instalados en el servidor NFS se debe ejecutar la siguiente lista de comandos para los servicios NFS, Samba, correo, web y autenticación. Cyrus es un servidor de correo electrónico.

```
# service nfsserver status
# service smb status
# service nmb status
# service postfix status
# service cyrus status
# service saslauthd status
```

En la Figura 4.7 se muestra la verificación de los servicios.

```
http01:~ # service nfsserver status
Checking for kernel based NFS server: running
http01:~ # service smb status
Checking for Samba SMB daemon running
http01:~ # service nmb status
Checking for Samba NMB daemon running
http01:~ # service postfix status
Checking for service Postfix: running
http01:~ # service cyrus status
Checking for service IMAP/POP3 (cyrus-imapd): running
http01:~ # service saslauthd status
Checking for service saslauthd: running
```

Figura 4.7: Servicios del servidor NFS

Para ingresar al portal del servidor web se debe abrir un explorador web e ingresar a la URL <http://192.168.1.15/> o <https://192.168.1.15/> En la Figura 4.8 se muestra el portal web.

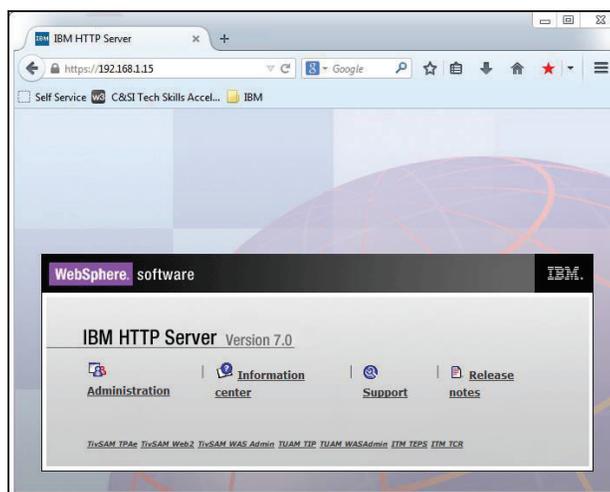


Figura 4.8: Servicio web operativo

Luego se verifica que la base de datos de TUAM se esté ejecutando correctamente. El comando **# su - db2inst1 -c "db2 list database directory"** permite listar las bases de datos creadas en la instancia db2inst1. Para listar las bases de datos la instancia debe estar inicializada y totalmente operativa. En la Figura 4.9 se muestra el estado de la base de datos de TUAM.

```

File Edit View Terminal Tabs Help
tuam01:~ # su - db2inst1 -c "db2 list database directory"

System Database Directory

Number of entries in the directory = 1

Database 1 entry:

Database alias           = TUAM
Database name           = TUAM
Local database directory = /home/db2inst1
Database release level  = d.00
Comment                 =
Directory entry type    = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

tuam01:~ #

```

Figura 4.9: Servicio de la base de datos de TUAM

Se verifica que el servidor de aplicaciones esté en ejecución, mediante el comando **# /opt/IBM/tivoli/tipv2/profiles/TIPProfile/bin/serverStatus.sh server1**. En la Figura 4.10 se muestra el estado del servidor de aplicación de TUAM.

```

tuam01:~ # /opt/IBM/tivoli/tipv2/profiles/TIPProfile/bin/serverStatus.sh server1
ADMU0116I: Tool information is being logged in file
           /opt/IBM/tivoli/tipv2/profiles/TIPProfile/logs/server1/serverStatus.log
ADMU0128I: Starting tool with the TIPProfile profile
ADMU0500I: Retrieving server status for server1
ADMU0508I: The Application Server "server1" is STARTED
tuam01:~ # █

```

Figura 4.10: Servicio del servidor de aplicaciones de TUAM

Para verificar la ejecución del agente de monitoreo ITM, se utiliza el comando **# /opt/IBM/ITM/bin/cinfo -r**. En la Figura 4.11 se muestra el estado del agente de monitoreo ITM.

```

tuam01:~ # /opt/IBM/ITM/bin/cinfo -r
***** Mon Jun 15 19:13:49 COT 2015 *****
User: root Groups: root
Host name : tuam01      Installer Lvl:06.22.04.00
CandleHome: /opt/IBM/ITM
*****
Host  Prod  PID  Owner  Start  ID  ..Status
tuam01 lz    5403 root   18:05 None  ..running
tuam01:~ #

```

Figura 4.11: Servicio del agente de monitoreo para S.O. Linux de TUAM

Para ingresar al portal web de la aplicación TUAM se debe ingresar a la URL <https://192.168.1.18:17311/ibm/console/logon.jsp> En la Figura 4.12 se muestra el portal web de TUAM.

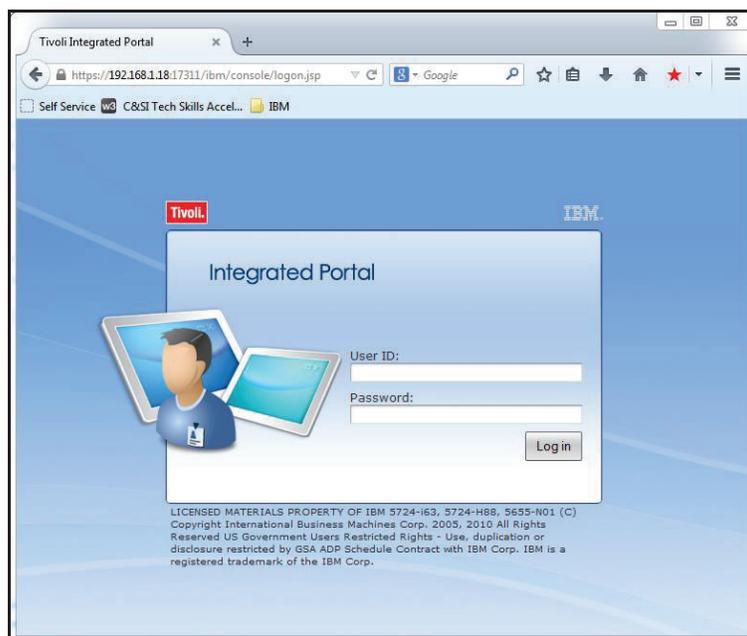


Figura 4.12: Portal web de la aplicación TUAM

Se verifica que los servicios del servidor LDAP estén siendo ejecutados, mediante el comando **# ps -ef | grep ibmdiradm**. En la Figura 4.13 se muestra la ejecución del servicio LDAP.

```

tiwsam01:~ # ps -ef | grep ibmdiradm
idscmdb 9466 1 0 19:22 ? 00:00:00 /opt/ibm/ldap/V6.2/sbin/64/ibmdiradm -I idscmdb
root 16300 15192 0 19:32 pts/0 00:00:00 grep ibmdiradm
tiwsam01:~ # ps -ef | grep ibmslapd
idscmdb 16183 1 0 19:32 pts/0 00:00:00 /opt/ibm/ldap/V6.2/sbin/64/ibmslapd -I idscmdb
root 16333 15192 0 19:33 pts/0 00:00:00 grep ibmslapd
tiwsam01:~ #

```

Figura 4.13: Servicios de LDAP en ejecución

Se verifica que la base de datos de TSAM se esté ejecutando correctamente. El comando `# su - ctginst1 -c "db2 list database directory"` permite listar las bases de datos creadas en la instancia de TSAM denominada ctginst1. En la Figura 4.14 se muestra el estado de la base de datos de TSAM.

```
tivsam01:~ # su - ctginst1 -c "db2 list database directory"

System Database Directory

Number of entries in the directory = 2

Database 1 entry:

Database alias           = TPMFOSD
Database name           = TPMFOSD
Local database directory = /home/ctginst1
Database release level  = d.00
Comment                 =
Directory entry type    = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

Database 2 entry:

Database alias           = MAXDB71
Database name           = MAXDB71
Local database directory = /home/ctginst1
Database release level  = d.00
Comment                 =
Directory entry type    = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

tivsam01:~ # █
```

Figura 4.14: Servicio de la base de datos de TSAM

Se verifica que las bases de datos LDAP se estén ejecutando correctamente. El comando `# su - idscmdb -c "db2 list database directory"` permite listar las bases de datos creadas en la instancia de LDAP denominada idscmdb. Para listar las bases de datos, la instancia debe estar inicializada y totalmente operativa. Existen dos bases de datos que brindan el servicio LDAP denominadas LDAPDB2B y SECURITY. Las dos bases de datos están creadas en la misma instancia, DB2 permite crear y configurar varias instancias en un mismo servidor físico/virtual y en cada instancia múltiples bases de datos. En la Figura 4.15 se muestra el estado de las bases de datos de LDAP.

```
tivsam01:~ # su - idscmdb -c "db2 list database directory"

System Database Directory

Number of entries in the directory = 2

Database 1 entry:

Database alias           = LDAPDB2B
Database name            = SECURITY
Local database directory = /home/idscmdb
Database release level   = d.00
Comment                  =
Directory entry type     = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

Database 2 entry:

Database alias           = SECURITY
Database name            = SECURITY
Local database directory = /home/idscmdb
Database release level   = d.00
Comment                  =
Directory entry type     = Indirect
Catalog database partition number = 0
Alternate server hostname =
Alternate server port number =

tivsam01:~ # █
```

Figura 4.15: Servicio de la base de datos de LDAP

Se verifica el estado de los servicios del servidor de aplicaciones TSAM. En la Figura 4.16 se muestra la ejecución del comando **# su - tioadmin -c "/opt/IBM/tivoli/tpm /tolos/tioStatus.sh wasadmin password"**. El nombre de usuario es wasadmin y la contraseña es password.

```
tivsam01:~ # su - tioadmin -c "/opt/IBM/tivoli/tpm/tools/tioStatus.sh wasadmin password"

Check status of server webserver1
ADMU0116I: Tool information is being logged in file
/opt/IBM/WebSphere/AppServer/profiles/ctgAppSrv01/logs/webserver1/serverStatus.log
ADMU0128I: Starting tool with the ctgAppSrv01 profile
ADMU0500I: Retrieving server status for webserver1
ADMU0508I: The Web server "webserver1" is STOPPED
Check status of server MXServer
ADMU0116I: Tool information is being logged in file
/opt/IBM/WebSphere/AppServer/profiles/ctgAppSrv01/logs/MXServer/serverStatus.log
ADMU0128I: Starting tool with the ctgAppSrv01 profile
ADMU0500I: Retrieving server status for MXServer
ADMU0508I: The Application Server "MXServer" is STARTED
Check status of server server1
ADMU0116I: Tool information is being logged in file
/opt/IBM/WebSphere/AppServer/profiles/casprofile/logs/server1/serverStatus.log
ADMU0128I: Starting tool with the casprofile profile
ADMU0500I: Retrieving server status for server1
ADMU0508I: The Application Server "server1" is STARTED
```

Figura 4.16: Servicio del servidor de aplicaciones de TSAM

Se verifica la ejecución del agente de monitoreo. En la Figura 4.17 se muestra el resultado de la ejecución del comando `# /opt/IBM/ITM/bin/cinfo -r`.

```
tivsam01:~ # /opt/IBM/ITM/bin/cinfo -r
***** Mon Jun 15 20:03:52 COT 2015 *****
User: root Groups: root db2iadmi idsldap dbsysadm
Host name : tivsam01   Installer Lvl:06.22.04.00
CandleHome: /opt/IBM/ITM
*****
Host      Prod  PID  Owner  Start  ID   ..Status
tivsam01 lz    7044 root   19:22  None ..running
tivsam01:~ #
```

Figura 4.17: Servicio del agente de monitoreo para S.O. Linux de TSAM

En la Figura 4.18 se muestra el portal de la aplicación TSAM. URL <https://192.168.1.7:9443/maximo>.

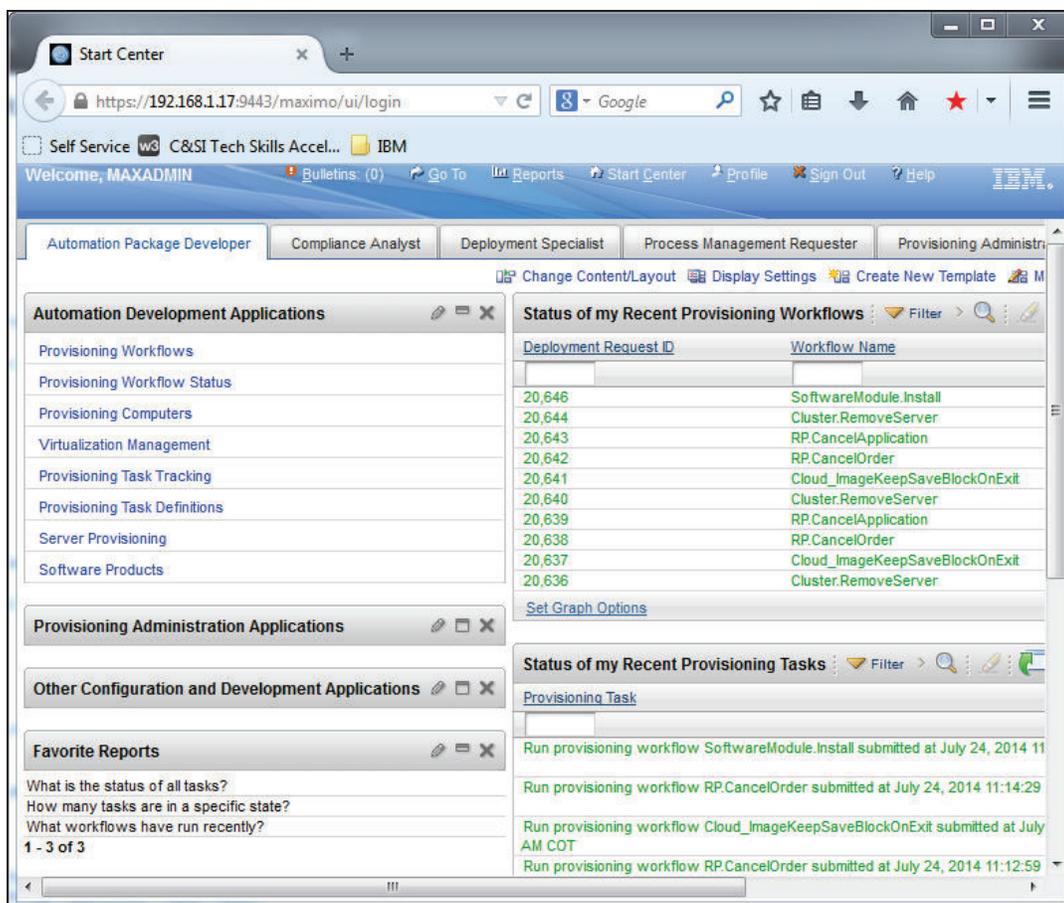


Figura 4.18: Portal de la aplicación TSAM

En la Figura 4.19 se muestra que los servidores DNS y vCenter se están ejecutando.

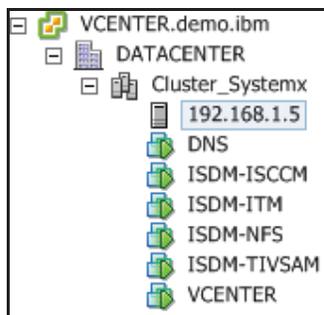


Figura 4.19: Servidores DNS y vCenter en ejecución

En la Figura 4.20 se muestra la utilización de recursos de todos los servidores que forman la solución de aprovisionamiento.

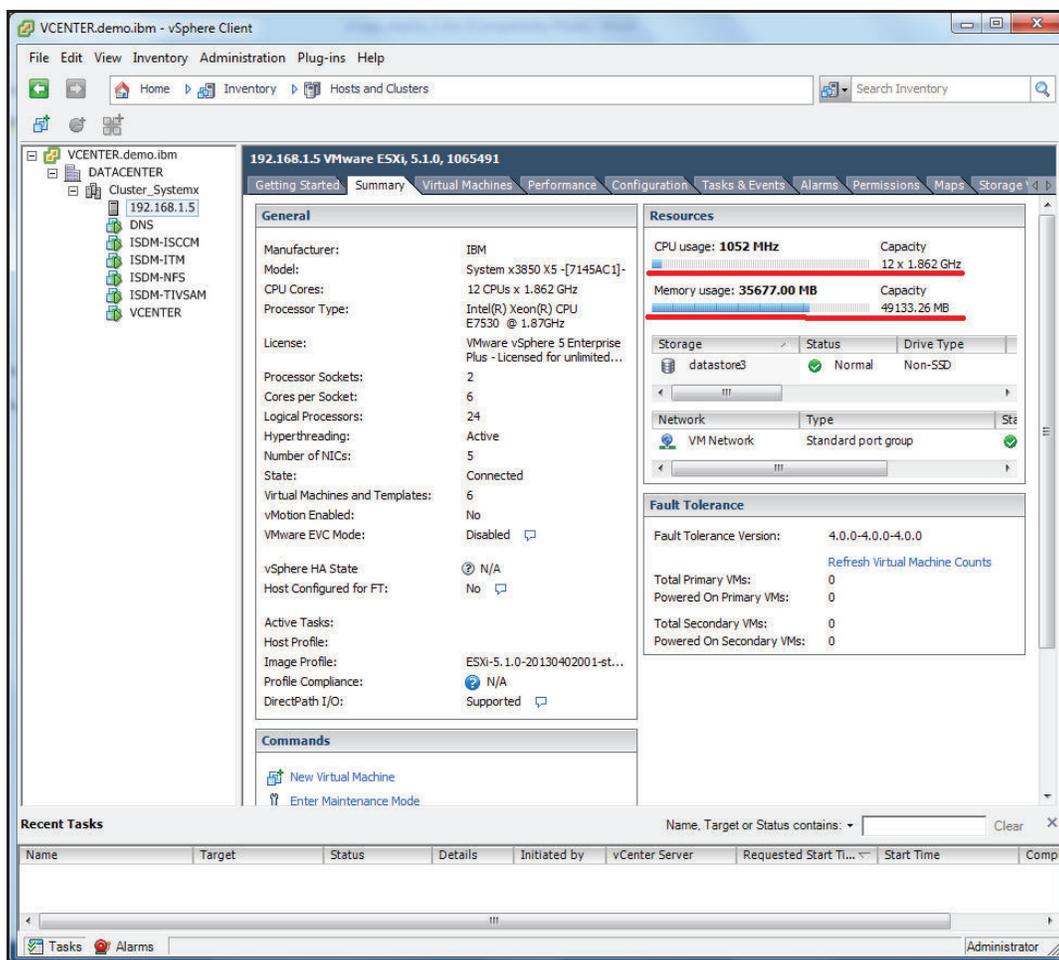


Figura 4.20: Utilización de recursos de la solución

4.2. PROCESO PARA PREPARAR UN DESPLIEGUE

A continuación, se especifican los pasos macro que se deben realizar con el objetivo de preparar un nuevo despliegue dependiendo de los requerimientos o necesidades que se tengan para un determinado proyecto o asignatura.

- Definir el proyecto o asignatura en la cual se desea prestar el servicio por medio de la solución de aprovisionamiento de sistemas virtuales. Por ejemplo, proyecto final de la asignatura denominada Sistemas Operativos; el mismo que incluirá la preparación de ambientes virtuales con bases de datos replicadas y sincronizadas en tiempo real. Como se podrá entender para trabajar con ambientes virtuales se tiene como primera premisa que los requerimientos de *hardware* deben ser lo suficientemente generosos para el normal desarrollo y funcionamiento.
- Del paso anterior se deben definir los requerimientos de *hardware* y *software* que se deben disponer en los sistemas virtuales aprovisionados. Por ejemplo, de este punto se determinan la cantidad de disco duro, memoria RAM, procesador y pila de *software* que se desea disponer.
- Dentro de la solución de aprovisionamiento se deben crear los usuarios con los correspondientes accesos que podrán disponer del servicio de aprovisionamiento. Un ejemplo en este caso, los estudiantes de la asignatura de Sistemas Operativos pueden acceder al despliegue del Sistema Operativo Linux, pero no tendrán acceso al Sistema Operativo Windows.
- Preparar la plantilla del sistema operativo sobre la cual se va a desarrollar el proyecto en mención; por ejemplo, en este punto se define que el proyecto deberá ser realizado sobre la plataforma Red Hat Enterprise Linux Server 6 Update 5.
- Preparar la pila de *software* que será desplegado durante el proceso de aprovisionamiento del ambiente virtual. Por ejemplo, para el caso del sistema operativo Windows, se determina que durante la creación y

solicitud de aprovisionamiento de máquina virtual se debe instalar la aplicación SQL Server SP1 Express 2014. Esta instalación será realizada de forma desatendida y automática; el usuario no tendrá que realizar la instalación, en su defecto el sistema de aprovisionamiento hará el trabajo.

- El docente al haber sido otorgado con los privilegios de usuario final, ingresará a la aplicación de aprovisionamiento mediante una página web, y realizará la solicitud de aprovisionamiento de acuerdo a las necesidades que tenga para un determinado proyecto.
- La solicitud realizada en el punto anterior, no se efectuará inmediatamente, debido a que previamente deberá ser validada y aprobada por el usuario administrador de la solución de aprovisionamiento; es decir, a pesar de que el estudiante tiene total control de solicitar el sistema con las características que desee, deberá pasar el filtro de aprobación.
- El usuario administrador recibirá un correo electrónico con todos los datos del usuario y sistema que se estableció durante el proceso de aprovisionamiento. El administrador del sistema tiene la potestad de aprobar o rechazar la solicitud del requerimiento.
- En cualquiera de los casos, aprobación o rechazo de la solicitud de aprovisionamiento, el estudiante o usuario final también será notificado por correo electrónico que su solicitud está lista para ser usada o fue rechazada por un determinado motivo. En el caso de que la solicitud haya sido aprobada, le llegará un segundo correo en el cual se incluyen las características del sistema solicitado y la contraseña de acceso al sistema operativo, credenciales correspondientes al usuario administrador.
- El usuario accederá al sistema virtual, ya sea por escritorio remoto para sistemas Windows o vía SSH para sistema Linux, en los dos casos ingresará con el usuario administrador del Sistema Operativo, lo cual, le otorgará total control sobre el ambiente virtual, siendo de entera responsabilidad los trabajos que se realicen en el mismo.

- Para tener un control sobre las acciones del estudiante, se puede realizar un proceso previo en el cual el usuario firme una carta de responsabilidad o inclusive firme un formulario o contrato en el cual el usuario se verá obligado a tener total responsabilidad sobre las acciones correctas o incorrectas que se realicen sobre el sistema operativo.

4.3. REALIZAR UNA SOLICITUD DE APROVISIONAMIENTO

A continuación, se establecen los pasos que debe seguir el usuario para realizar una solicitud de aprovisionamiento de máquina virtual.

Los pasos para el usuario final son demasiado intuitivos, a tal punto que no deberá tener conocimiento alguno de la complejidad de lo que implica tener un sistema de aprovisionamiento automatizado y funcional.

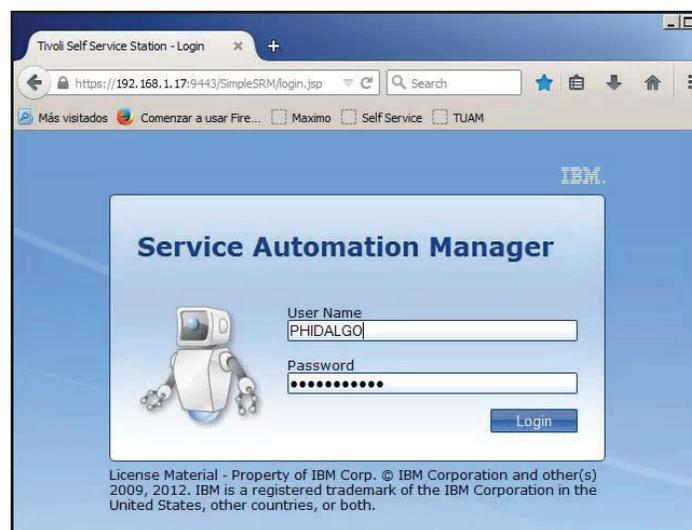


Figura 4.21: Login - Portal de auto-servicio TSAM

El usuario o estudiante debe ingresar al portal web de aprovisionamiento. Desde un explorador web soportado se debe ingresar a la aplicación en la dirección URL: *https://192.168.1.17:9443/SimpleSRM/login.jsp*. Los exploradores soportados son Internet Explorer en versión mínima 10 y Mozilla Firefox en versión mínima 30.

En la Figura 4.21 se muestra el portal de auto-servicio de la solución de aprovisionamiento.

En el portal de auto-servicio se debe ingresar a la sección denominada “Request a New Service” > Virtual Server Management > Create Project with VMware Servers. En la Figura 4.22 se muestra una captura en el portal TSAM.

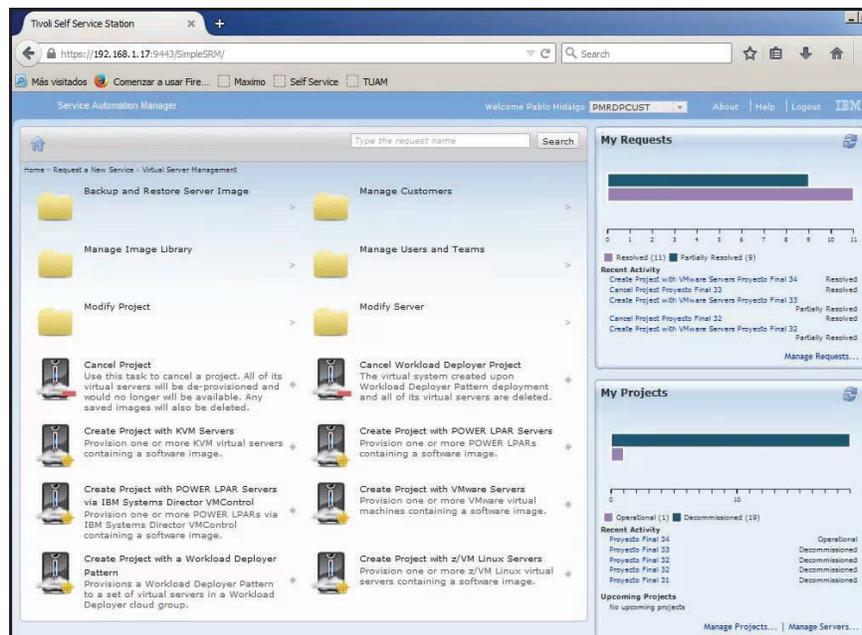


Figura 4.22: Inicio - Portal de auto-servicio TSAM

En la Figura 4.23 se muestra la sección en la cual se deben ingresar los detalles del proyecto de aprovisionamiento. Se define el nombre del proyecto, descripción, acceso a un determinado grupo de usuarios, fechas de vigencia del servicio de aprovisionamiento.

Figura 4.23: Detalles del proyecto - Portal de auto-servicio TSAM

En la Figura 4.24 se muestra el panel en el cual se define la fuente de recursos para el sistema de aprovisionamiento, sistema operativo y cantidad de servidores virtuales que serán desplegados.

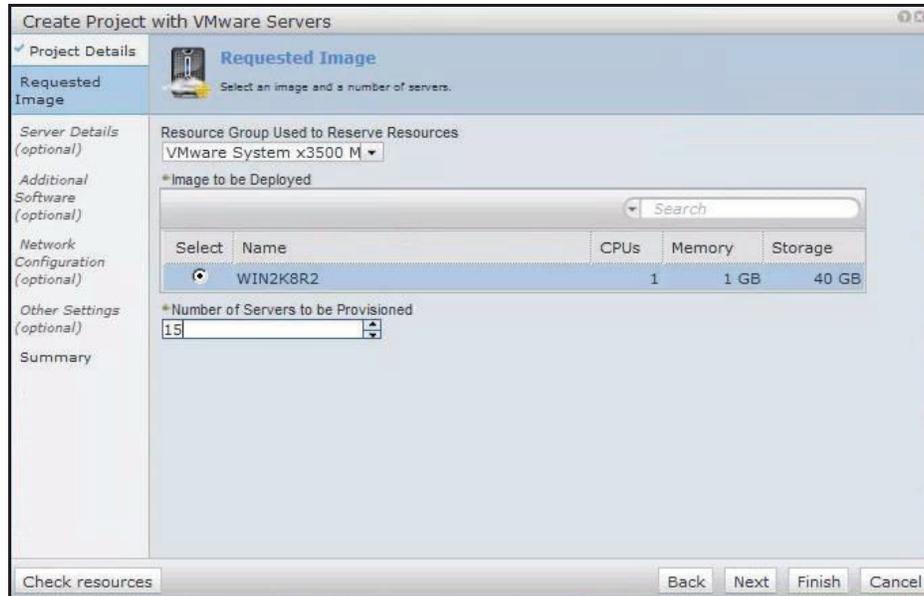


Figura 4.24: Plantilla - Portal de auto-servicio TSAM

En la Figura 4.25 se muestra la sección que permite definir la configuración de *hardware* de los “N” sistemas virtuales definidos en el paso anterior, es decir, todos los sistemas virtuales tendrán la misma configuración. Los parámetros configurables son: número de procesadores, memoria y disco.

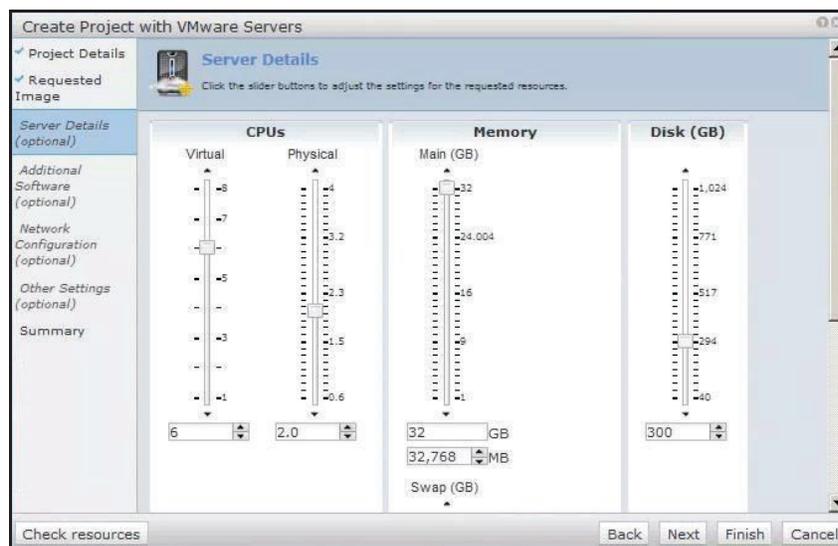


Figura 4.25: Configuración de *hardware* - Portal de auto-servicio TSAM

En la Figura 4.26 se muestra la lista de productos que pueden ser instalados de forma desatendida y a los que este usuario en particular tiene acceso.



Figura 4.26: Software Adicional - Portal de auto-servicio TSAM

En la Figura 4.27 se muestra la sección para configurar las interfaces de red que dispondrá el servidor aprovisionado.

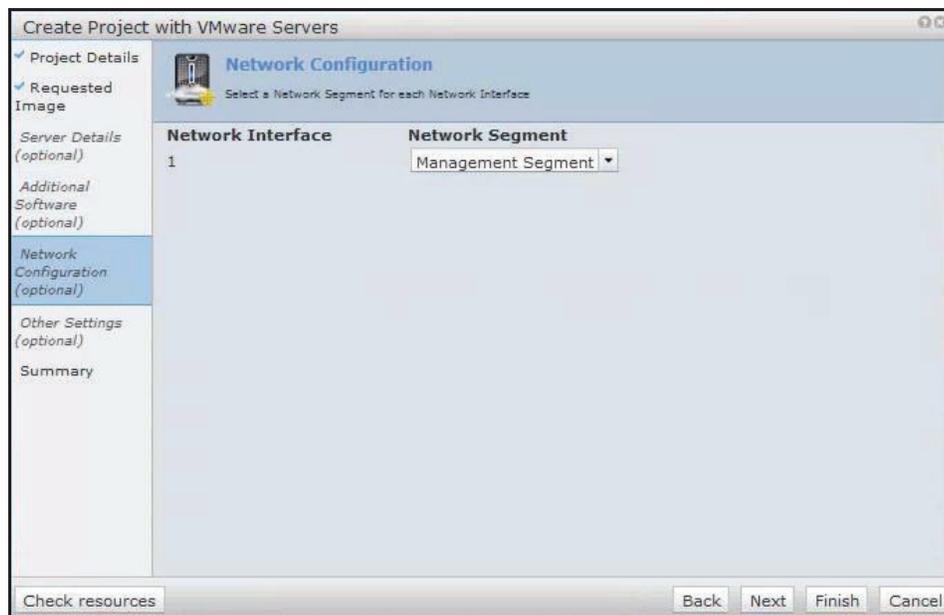


Figura 4.27: Configuración de red - Portal de auto-servicio TSAM

En la Figura 4.28 se muestran opciones adicionales de configuración como la instalación de un agente de monitoreo y opciones de respaldo de las imágenes aprovisionadas.

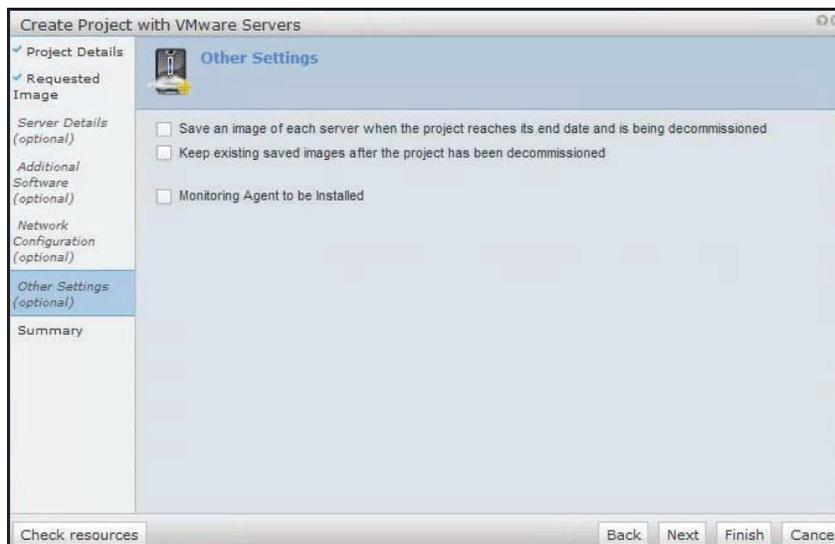


Figura 4.28: Configuración adicional - Portal de auto-servicio TSAM

En la Figura 4.29 se muestra un resumen de las opciones seleccionadas en los pasos previos, se debe revisar el mismo y se procede con el despliegue.

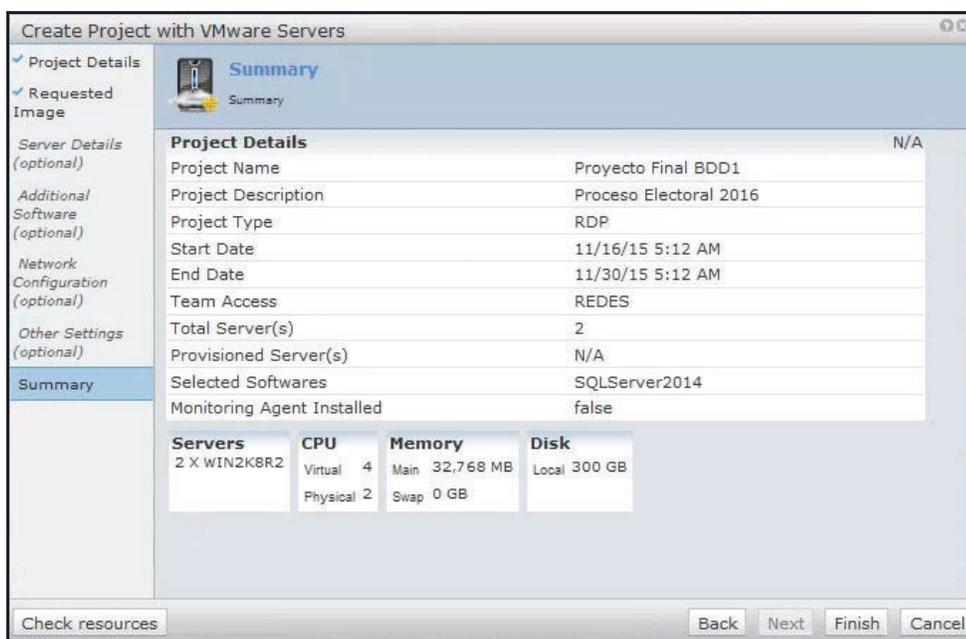


Figura 4.29: Resumen - Portal de auto-servicio TSAM

En la Figura 4.30 se puede monitorear el avance de la solicitud realizada, la misma que pasará por distintos estados hasta llegar al final.

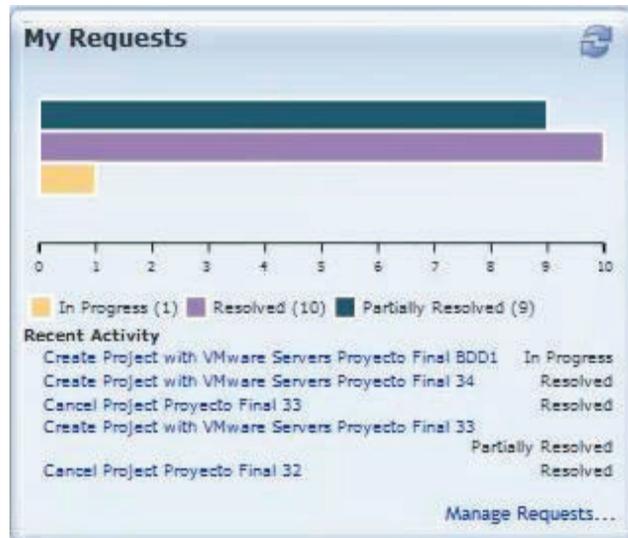


Figura 4.30: Estado de solicitud - Portal de auto-servicio TSAM

En la Figura 4.31 se muestra la sección en la que se aprueba o rechaza una solicitud de aprovisionamiento.

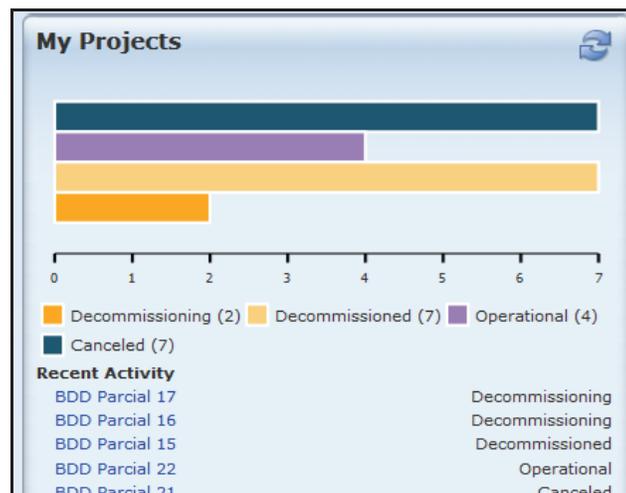


Figura 4.31: Mis proyectos - Portal de auto-servicio TSAM

Una vez que el proceso finalice correctamente, el usuario será notificado vía correo electrónico indicando que el sistema virtual está listo para ser usado. En la Figura 4.32 se muestra un ejemplo de notificación.

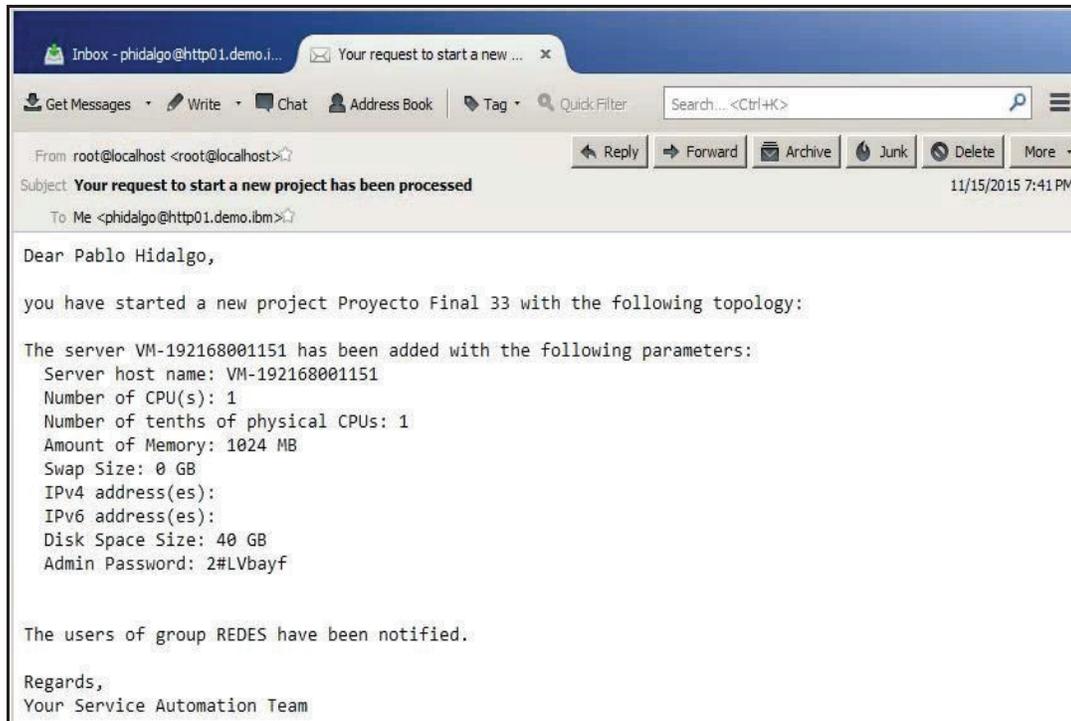


Figura 4.32: Notificación de aprovisionamiento - Portal de auto-servicio TSAM

En la Figura 4.33 se muestra el proceso de aprovisionamiento en ejecución.

VCENTER.demo.ibm

DATACENTER

- Cluster_Systemx
 - 192.168.1.5
 - DNS
 - ISDM-ISCCM
 - ISDM-ITM
 - ISDM-NFS
 - ISDM-TIVSAM
 - VCENTER
 - VM-192168001151
 - VM-192168001152
 - VM-192168001153
 - VM-192168001154
 - VM-192168001155

ISDM-ITM

Getting Started Summary Resource Allocation Performance Tasks & Events Alarms Console Permissions Maps Storage Views

close tab X

What is a Virtual Machine?

A virtual machine is a software computer that, like a physical computer, runs an operating system and applications. An operating system installed on a virtual machine is called a guest operating system.

Because every virtual machine is an isolated computing environment, you can use virtual machines as desktop or workstation environments, as testing environments, or to consolidate server applications.

In vCenter Server, virtual machines run on hosts or clusters. The same host can run many virtual machines.

Basic Tasks

- Shut down the virtual machine

Recent Tasks

Name	Target	Status	Details	Initiated by	vCenter Server	Requested Start Time	Start Time	Completed Time
Clone virtual machine	WIN7EN	73%	Copying Vir...	Administrator	VCENTER.dem...	2/25/2016 10:06:11 ...	2/25/2016 10:06:11 ...	
Clone virtual machine	WIN7EN	75%	Copying Vir...	Administrator	VCENTER.dem...	2/25/2016 10:05:45 ...	2/25/2016 10:05:45 ...	
Clone virtual machine	WIN7EN	77%	Copying Vir...	Administrator	VCENTER.dem...	2/25/2016 10:04:58 ...	2/25/2016 10:04:58 ...	
Clone virtual machine	WIN7EN	80%	Copying Vir...	Administrator	VCENTER.dem...	2/25/2016 10:04:44 ...	2/25/2016 10:04:44 ...	

Diagram illustrating the vCenter architecture: vSphere Client connects to vCenter Server, which manages a Cluster of Hosts. Virtual Machines are hosted on these Hosts within a Datacenter.

Figura 4.33: Proceso de aprovisionamiento en ejecución

4.4. ANÁLISIS DE COSTOS DE REALIZACIÓN DEL PROYECTO

En el marco del desarrollo de la medición de la viabilidad de implementar un sistema piloto para el aprovisionamiento de máquinas virtuales para el laboratorio de la Carrera de Ingeniería en Electrónica y Redes de Información de la Escuela Politécnica Nacional, se desarrolla un modelo de análisis de costos y beneficios conteniendo los siguientes puntos:

Análisis de Costos: Se considera el modelo de costos TCO (Total Cost of Ownership) en el cual se toman en cuenta todos los costos directos e indirectos para cada uno de los componentes que forman parte de la solución propuesta.

Análisis de Costo-Beneficio: Se consideran los beneficios sociales, económicos (universidad, estudiante), de gestión tomando en cuenta el resultado generado para los involucrados.

Análisis de retorno: Se considera el cálculo de retorno de inversión (ROI) que servirá para medir el impacto de las iniciativas para cada una de ellas, se realiza una comparación entre usar el sistema en la nube propuesto a usar un servidor por cada una de las aplicaciones.

Para efectos de modelo, se considera una parte del ingreso de información, donde fundamentamos conceptualmente sobre la información de costos y beneficios, análisis de resultados, descripción de objetivos, forma de funcionamiento y los resultados asociados a la implementación de la solución.

4.4.1. INGRESO DE INFORMACIÓN

4.4.1.1. Análisis de Costos

En esta sección se identificarán y listarán los costos asociados al proyecto, de acuerdo a las necesidades técnicas, profesionales y de mantenimiento. Se espera tener una estimación de costos del proyecto.

Para la construcción de este listado de costos a ser evaluados, se utilizó como base la metodología TCO.

TCO es una práctica internacional que se utiliza para el análisis y mejora de los procesos de adquisición de tecnología y costos adicionales a estos procesos. De acuerdo a la práctica, los costos asociados a una adquisición, tienen otros conceptos además del precio de adquirirlo, que se deben obligatoriamente considerar para de esta manera garantizar el correcto funcionamiento del proyecto, bien, solución, etc., durante la vida útil estimada. De acuerdo a la afirmación anterior, se pueden considerar los siguientes tipos de costos:

Costos Directos: *Hardware*, comunicaciones, *Software*, Migración de datos, Integraciones, etc.

Costos Indirectos: Capacitación a usuarios, respaldos, entrenamiento, solución de problemas, etc.

Rubros	Número de años		
	1	2	3
Costos Directos			
<i>Hardware</i> + mantenimiento	\$19,890.00	\$350.00	\$350.00
Comunicaciones *	\$0.00	\$0.00	\$0.00
Infraestructura Laboratorio**	\$0.00	\$0.00	\$0.00
VMWare Vsphere	\$1,268.00	\$265.00	\$265.00
IBM Smartcloud Monitoring *	\$0.00	\$0.00	\$0.00
IBM Tivoli Service Automation Manager *	\$0.00	\$0.00	\$0.00
IBM Tivoli Usage Accounting Manager *	\$0.00	\$0.00	\$0.00
IBM Service Delivery Manager *	\$0.00	\$0.00	\$0.00
Consultores Individuales ***	\$0.00	\$0.00	\$0.00
Costos Indirectos			
Servicios de instalación + soporte *****	\$10000.00	\$1000.00	\$1000.00
Administración del sistema *****	\$360.00	\$360.00	\$360.00
Respaldos ****	\$0.00	\$0.00	\$0.00
Capacitación a otros usuarios ***	\$0.00	\$0.00	\$0.00
Solucionar problemas ****	\$0.00	\$0.00	\$0.00
TOTAL:	\$31,518.00	\$1975.00	\$1975.00

Tabla 4.1: Costos referenciales (No incluyen I.V.A.)

* *Software* a costo cero, que puede ser utilizado solo para fines educativos, siempre y cuando el acuerdo de uso de *Software* se encuentre vigente entre la Escuela Politécnica Nacional e IBM.

** En este Proyecto de Titulación, se asume que el laboratorio de Redes cuenta con todas las prestaciones para la instalación física del servidor tales como energía eléctrica, redes de comunicaciones, etc.

*** El trabajo de instalación y configuración de la herramienta es realizado por los estudiantes que desarrollan este Proyecto de Titulación, así como también la capacitación a posibles administradores (ayudantes de laboratorio). Dado que en el laboratorio no se cuenta con el servidor para la instalación, se ha procedido a instalar en un ambiente demo, con un servidor alquilado.

**** Los respaldos y solución de problemas posteriores serán cubiertos por la gente que trabaje en el Laboratorio de Redes, teniendo el apoyo de IBM donde podrán abrir *tickets* de soporte sin costo. Al disponer de las licencias por convenio de educación, también se dispone del soporte de la aplicación.

***** En una posible instalación del sistema en producción, se deberá considerar el costo correspondiente a los servicios de implementación del sistema.

***** Se debe considerar el costo correspondiente a la administración del sistema, los administradores pueden ser los ayudantes de laboratorio.

4.4.1.2. Análisis del Costo-Beneficio

Se identifica a la Escuela Politécnica Nacional y a los estudiantes de la Carrera en Electrónica y Redes de Información como participantes de la implementación de este proyecto. Por ello se debe considerar los beneficios que este proyecto puede generar sobre cada uno de ellos.

Beneficios para los estudiantes de la carrera de Electrónica y Redes de Información: se analizarán tres puntos que se consideran los más relevantes para los estudiantes. Estos puntos se podrán cumplir siempre y cuando el sistema sea implementado en ambiente de producción real (el presente Proyecto de Titulación es un piloto).

- **Tiempo:** Este beneficio se traduce en la generación de mayor tiempo de acceso a las herramientas, que en muchas ocasiones están instaladas en el laboratorio solamente. Adicionalmente ya no será necesario ingresar al laboratorio, o esperar que, llegue la hora de clase para poder hacer pruebas en las herramienta de simulación que posee la institución; a partir de la configuración de cada máquina virtual (por administrador) se puede tener acceso 24/7.
- **Gastos:** Disminución de gastos para el estudiante asociados al acceso a las herramientas de *software* que se utilizan día a día, tales como *software* de simulación, bases de datos, desarrollo de *software*, etc. El no tener que contar con un equipo personal que soporte todas estas herramientas, hace que el acceso sea mucho más económico, pudiendo hacerlo desde un teléfono inteligente, *Tablet*, computador de escritorio, *laptop* con características básicas. El único requisito a tomar en cuenta será que cuente con un explorador web.
- **Planificación para optimizar el uso de recursos:** Este beneficio depende de la configuración de cada máquina virtual, donde se puede sencillamente calendarizar la disponibilidad del ambiente (horas de clase), por semestre, por semana, o por días específicos donde el acceso puede ser a cualquier hora, siempre y cuando el servidor se encuentre trabajando.
- **Iniciativas de Investigación:** Al contar con un servidor con una pila de aplicaciones y sistemas operativos, es más fácil aprovisionar una máquina con características específicas para proyectos de investigación. Sin necesidad de comprar nuevos servidores para cada uno de los proyectos.

Beneficios para la Carrera de Electrónica y Redes de Información: aquí se detallan los beneficios más significativos para la Carrera.

- **Tiempo:** Este beneficio se traduce como la reducción del tiempo necesario para tener disponible un ambiente para una materia específica, así como también el tiempo mínimo que se requiere para la gestión de cada uno de

estos ambientes generados. Además a todo esto, una vez que el servidor esté configurado, el aprovisionamiento de máquinas virtuales es automático así como también la gestión de recursos para cada una de los ambientes.

- **Gastos:** Se tendrá un ahorro de costos operativos, ya que se necesita mucho menos esfuerzo de gestión para el servidor, que para un laboratorio completo donde se instalan cada semestre las herramientas para cada asignatura en un usuario distinto.
- **Gestión e imagen:** Aumenta la calidad de gestión de los recursos del laboratorio así como también la prestación de servicios hacia los estudiantes, garantizando el acceso a las herramientas de *software* para cualquiera de los niveles, aplicando una política social e inclusiva.
- **Posibilidad de mejorar docencia:** Contando con infraestructura que se pueda configurar a medida, se abre la posibilidad de incluir nuevos tipos de conocimiento en virtualización, educación más personalizada en los laboratorios y fuera de ellos, proyectos de investigación que requieren alto procesamiento (puede ser calendarizado en las noches o madrugadas).

4.4.2. ANÁLISIS DE RESULTADOS

De lo que se revisó en la sección anterior, se puede ver que la universidad debe hacer una inversión de *hardware* y *software* de virtualización. Las herramientas que conforman la solución integralmente pueden ser usadas sin costo, siempre y cuando el uso que se les dé sea para fines educativos.

Las herramientas que se emplean para impartir las cátedras hacia los estudiantes, en su mayoría forman parte de convenios que tiene la universidad con diferentes proveedores de *software*, lo que no significa un costo adicional para la universidad.

En el caso de los estudiantes, podrán tener libre acceso a las herramientas de *software* que disponga la Carrera y la materia en particular sin tener que preocuparse del licenciamiento de estas herramientas. Además es muy

importante recalcar que la forma en la que accedan irá acorde a las posibilidades que cada uno de los estudiantes tenga, por lo que se garantiza el libre uso para los estudiantes.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Con el sistema piloto implementado se pudo evidenciar una opción viable que tiene la Carrera de Electrónica y Redes de Información para disponer de herramientas de *cloud computing* a disposición de los docentes y estudiantes que forman parte de la Carrera.
- La solución de aprovisionamiento de máquinas virtuales permiten optimizar el uso de recursos debido a que se puede planificar y calendarizar la disponibilidad del ambiente, estas definiciones pueden ser realizadas en base a las horas de clase, al semestre, a los días específicos en donde los usuarios accederán al sistema desplegado.
- El sistema de aprovisionamiento ISDM permite aprovechar los beneficios ofrecidos por la virtualización optimizando y centralizando la gestión de varios sistemas virtuales a través de un determinado servicio bajo demanda por medio de Internet; el servicio es presentado en la opción IaaS.
- El sistema piloto permite aprovisionar, gestionar y controlar los sistemas de computación virtuales para las tres asignaturas de prueba: Base de Datos, Redes de Área Local y Sistemas Operativos. El sistema piloto se ve limitado por los recursos disponibles del *hardware* en el cual se implementa el sistema.
- Producto de la implementación del sistema piloto de aprovisionamiento se ha determinado que en un escenario real de producción se podría garantizar el acceso libre, sin discriminación a todos los estudiantes de la Carrera de Electrónica y Redes de Información, a todas las herramientas de *software* con las que cuentan los laboratorios y tecnologías de vanguardia como las proporcionadas por International Business Machine - IBM.

- El desarrollo del presente Proyecto de Titulación puede utilizarse como documento base para la implementación en ambiente de producción de un sistema de aprovisionamiento de máquinas virtuales para la Carrera de Electrónica y Redes.
- La implementación del sistema piloto del presente Proyecto de Titulación permite evidenciar que el estudiante puede hacer uso de tecnologías de computación en la nube, modalidad nube privada, facilitando el acceso a todas las aplicaciones que requiere para el normal desarrollo de la carrera universitaria, esto siempre y cuando el sistema sea implementado en un ambiente real de producción.
- El aprovisionamiento de máquinas virtuales en la Carrera de Electrónica y Redes de la Información, garantiza al profesor de cada una de las asignaturas que los estudiantes disponen de las herramientas de *software* necesarias para realizar las actividades encomendadas, además que tiene un control total acerca del horario en el que está permitido ingresar, pudiendo también compartir material didáctico que considere necesario.
- La inversión de los estudiantes en sistemas portátiles de gran capacidad y en adquisición de *software* para el uso de las aplicaciones utilizadas en las asignaturas, ya no será necesario, debido a que con este sistema de aprovisionamiento se puede disponer de un verdadero servidor de cómputo con todos los utilitarios dentro, tan solo aplicando una solicitud vía web, la misma deberá ser aprobada.
- Actualmente, se desarrollan Proyectos de Titulación que demandan una gran capacidad de procesamiento, con este sistema de aprovisionamiento se puede apoyar de forma importante al desarrollo de estos proyectos implementados por los estudiantes en cuestión.

5.2. RECOMENDACIONES

- Se recomienda la implementación del sistema de aprovisionamiento a gran escala para toda la comunidad académica de la Escuela Politécnica

Nacional e incluir en el *roadmap* el brindar el sistema como un servicio a otras universidades del país. Para lo cual se debe invertir en la adquisición de *hardware* altamente robusto que permita crear una granja de servidores dispuestos a prestar el servicio de *cloud computing*.

- A partir de este Proyecto de Titulación se recomienda desarrollar los siguientes trabajos: Integración del sistema de aprovisionamiento con el sitio web público de la Escuela Politécnica Nacional – EPN e Implementación del sistema de aprovisionamiento para todas las Facultades de la Escuela Politécnica Nacional – EPN y universidades del país. Los gastos que se involucren deberán ser evaluados y financiados debidamente.
- Evaluar las últimas tecnologías que los distintos proveedores de *software* y *hardware* en el país para determinar la vigencia y validez de la solución o en su defecto migrar a las nuevas tendencias, el mundo tecnológico cambia de un día a otro, a un ritmo tan acelerado que es muy complicado seguirle la pista.
- Mantener el repositorio de *software* actualizado, sistemas operativos y herramientas propias que sirven para la enseñanza en la Carrera de Electrónica y Redes de Información.
- Proyectar esta solución, para dar servicios de *hosting* y aprovisionamiento de máquinas virtuales a otras universidades, entidades estatales y privadas, teniendo una gestión centralizada de las herramientas que cada uno de estos posibles clientes usa en su día a día.
- Usar el servidor para crear nuevas asignaturas, seminarios, garantizando la disponibilidad de la infraestructura por el tiempo que éstas duren. Se evita que los estudiantes o que la universidad adquieran equipos para estos fines. Se puede profundizar en materia de *Cloud Computing* y Virtualización que es parte del conocimiento que hoy en día el mercado busca.

- Se deben calendarizar el uso de los recursos del servidor, las máquinas virtuales que sean creadas y cualquier tipo de procesamiento específico que se requiera. Así, se consigue que el nivel de utilización de los recursos crezca y que la necesidad de tener nueva infraestructura decrezca. El número de máquinas virtuales con las que cuente el servidor, será igual al número de máquinas que el administrador decida, todo depende de una buena calendarización.
- Compartir el uso de los recursos del servidor entre los estudiantes y los docentes, para de esta manera aprovechar las características del mismo. Mientras se encuentren en clase, se puede asignar mayor recursos a las máquinas que los estudiantes usan, y en el tiempo que esto no suceda asignar los recursos a las máquinas que contienen proyectos de investigación. Por ejemplo, en tiempo de vacaciones de los estudiantes, se podrá contar con todo el poder del servidor para apoyo de proyectos de investigación.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Memorándum escrito por G. M. Amdahi, "Logical Equations for ANS Decoder", 13 de diciembre 1995, Museo de Historia de la Computadora, último acceso enero 2014, <http://archive.computerhistory.org/resources/text/IBM/Stretch/pdfs/06-06/102632244.pdf>
- [2] "Robert William Berner", The History of Computing Project, último acceso noviembre 2013, http://www.thocp.net/biographies/bemer_bob.htm
- [3] "IBM Shoebox", IBM, último acceso diciembre 2013, http://www03.ibm.com/ibm/history/exhibits/specialprod1/specialprod1_7.html
- [4] "Human Language Technologies", IBM Research, último acceso diciembre 2013, <http://www.research.ibm.com/hlt/html/history.html>
- [5] "A few facts about IBM Storage", IBM, último acceso enero 2014, http://www-03.ibm.com/ibm/history/exhibits/storage/storage_facts.html
- [6] "IBM Moves Closer to New Class of Memory", comunicado de prensa, 10 de abril de 2008, último acceso enero 2014, <http://www-03.ibm.com/press/us/en/pressrelease/23859.wss>
- [7] Instituto Marino, "SmartBay Environmental Monitoring System Installed in Galway Bay", comunicado de prensa, último acceso diciembre 2013, <http://www.marine.ie/home/aboutus/newsroom/news/smartbaymonitoringsysteminstalledingalwaybay.htm>.
- [8] Frank Da Cruz, "A Chronology of Computing at Columbia University", Universidad de Columbia, Laboratorio Watson, último acceso enero 2014, <http://www.columbia.edu/acis/history>
- [9] Transcripciones, John Backus (ref. X3715.2007), Charles Branscomb (ref. X5548.2010), Grace Hooper (ref. X5142.2009), Oral Histories Online,

- Museo de Historia de la Computadora, último acceso febrero 2014,
<http://www.computerhistory.org/collections/oralhistories/>
- [10] "The Chip That Jack Built", Texas Instruments, último acceso febrero 2014,
<http://www.ti.com/corp/docs/kilbyctr/jackbuilt.shtml>
- [11] "Tribute to Seymour Cray", IEEE Computer Society, último acceso febrero 2014, <http://www.computer.org/portal/web/awards/seymourbio>
- [12] "The Supercomputers at Los Alamos", IBM, último acceso diciembre 2013,
<http://www-03.ibm.com/deepcomputing/rr/>
- [13] "Blue Gene", IBM Research, último acceso enero 2014,
http://domino.research.ibm.com/comm/research_projects.nsf/pages/bluegene.idex.html
- [14] "The History of Superconductors", Superconductors.org, último acceso enero 2014, <http://www.superconductors.org/History.htm>
- [15] Lisa Hernandez, "The Exascale Supercomputer: 10 million cores in 2020", ConceivablyTech, último acceso diciembre 2013,
<http://www.conceivablytech.com/2997/products/the-exascale-supercomputer-10-million-cores-in-2020/>
- [16] Edward Hurley, "Q&A: An insider's view of CICS' development", SearchDataCenter.com, último acceso noviembre 2013,
<http://searchdatacenter.techtarget.com/news/1156274/QA-An-insiders-view-of-CICS-development>
- [17] Ray Kurzweil, "The significance of Watson", Kurzeil, Accelerating Intelligence (blog), último acceso diciembre 2013,
<http://www.kurzweilai.net/the-significance-of-watson>
- [18] "Data, data everywhere", Economist, último acceso enero 2014,
<http://www.economist.com/node/1555743>

- [19] "Columbia University Professor Ben Wood", Universidad de Columbia, último acceso enero 2014, <http://www.columbia.edu/acis/history/benwood.html>
- [20] John McCarthy, "Reminiscences on the History of Time Sharing", Universidad de Stanford, 1983, último acceso diciembre 2013, <http://www-formal.stanford.edu/jmc/history/timesharing/timesharing.html>
- [21] "¿Qué es la virtualización?", Tecnología Pyme, último acceso enero 2014, <http://www.tecnologiapyme.com/software/que-es-la-virtualizacion>
- [22] "Virtualización", Nosys, último acceso enero 2014, <http://www.nosys.es/es/virtualizacion.html>
- [23] "KVM", linux-KVM, último acceso enero 2014, http://www.linux-kvm.org/page/Main_Page
- [24] "The Xen Project", Xen Project, último acceso enero 2014, <http://www.xenproject.org/>
- [25] "Open VZ", OpenVZ Linux Containers, último acceso enero 2014, http://openvz.org/Main_Page
- [26] "Virtual Box", Virtual Box, último acceso enero 2014, <https://www.virtualbox.org/>
- [27] BladeCenter Foundation for Cloud, último acceso marzo 2014, <http://www.ibm.com/systems/bladecenter/solutions/infrastructure/virtualization/integratedcloudplatform/index.html>
- [28] "Nube híbrida", EMC, último acceso enero 2014, <http://mexico.emc.com/corporate/glosary/hybrid-cloud.htm>
- [29] "Computación en Nube", Computación en nube, último acceso febrero 2014, <http://www.computacionennube.org/computacion-en-nube/>

- [30] “Virtual Computing Lab”, Universidad Estatal de Carolina del Norte, último acceso marzo 2014, <http://vcl.ncsu.edu/>
- [31] “Computación en Nube”, Computación en Nube, último acceso febrero 2014, <http://www.computacionennube.org/computacion-en-nube/>
- [32] “What’s Shibboleth”, Shibboleth, último acceso marzo 2014, <https://shibboleth.net/about/>
- [33] “Mobile Cloud”, Computer.org, último acceso marzo 2014, http://www.computer.org/portal/c/document_library/get_file?uuid=b150980d-b8dc-445c-950c-25c0549b7982&groupId=889101b
- [34] “IBM Research Division cloud computing initiative”, IBM journal of research and development, último acceso abril 2014, <http://www.research.ibm.com/journal/rd53-4.html>
- [35] “Network Layout”, Apache VCL, último acceso abril 2014, <https://cwiki.apache.org/confluence/display/VCL/Network+Layout>
- [36] “VCL project incubation”, The Apache *Software* Foundation, último acceso mayo 2014, <http://incubator.apache.org/projects/vcl.html>
- [37] “xCAT”, xCAT, último acceso abril 2014, <http://xcat.sourceforge.net/>
- [38] “What is k12”, K12, último acceso abril 2014, <http://www.k12.com/what-is-k12#.VOpmQ2OZ5XU>
- [39] “What is STEM”, California STEM Learning Network, último acceso junio 2014, <http://www.cslnet.org/our-agenda/what-is-stem/>
- [40] “STEM”, Stem Education Coalition, <http://www.stemedcoalition.org/>
- [41] “About NCCU”, North Carolina Central University, último acceso agosto 2014, <http://www.nccu.edu/>

- [42] “How students learn in the cloud”, newsobserver.com, último acceso agosto 2014, <http://www.newsobserver.com/2011/05/23/1217291/howstudents-learn-in-the-cloud.html#storylink=misearch>
- [43] “Apache VCL”, VCL, último acceso septiembre 2014, <http://vcl.apache.org/>
- [44] “Hypervisors Bare Metal”, Virtualizamos, último acceso agosto 2014, <http://blog.virtualizamos.es/tag/bare-metal/>
- [45] “Definición de las siete capas del modelo OSI y explicación de sus funciones”, Microsoft support, último acceso noviembre 2014, <http://support.microsoft.com/kb/103884/es>
- [46] “Internet Protocol”, rfc-es.org, último acceso noviembre 2014, <http://www.rfc-es.org/rfc/rfc0791-es.txt>
- [47] “Internet Protocol”, ietf, último acceso noviembre 2014 <https://www.ietf.org/rfc/rfc791.txt>
- [48] “Protocolo de control de transmisión”, Microsoft, último acceso noviembre 2014, <https://msdn.microsoft.com/es-es/library/cc759700%28v=ws.10%29.aspx>
- [49] “Transmission Control Protocol”, IETF, último acceso noviembre 2014, <http://www.rfc-es.org/rfc/rfc0793-es.txt>
- [50] “Protocolo Internet”, Microsoft, último acceso noviembre 2014, <https://msdn.microsoft.com/es-es/library/cc757152%28v=ws.10%29.aspx>
- [51] “User Datagram Protocol”, IETF, último acceso noviembre 2014, <http://www.rfc-es.org/rfc/rfc0768-es.txt>
- [52] “BMC Cloud Lifecycle Management”, BMC Software, último acceso noviembre 2014, <http://documents.bmc.com/products/documents/67/49/206749/206749.pdf>

- [53] "Cloud Lifecycle Management", BMC *Software*, último acceso noviembre 2014, <http://www.bmc.com/it-solutions/cloud-lifecycle-management.html>
- [54] "IBM Service Delivery Manager", IBM Corporation, último acceso octubre 2014, <http://www-03.ibm.com/software/products/en/ibmservdelimana>
- [55] "IBM Service Delivery Manager", IBM Workcenter, último acceso noviembre 2014, <http://www-03.ibm.com/software/products/en/ibmservdelimana>
- [56] "The RSA Solution for Cloud Security and Compliance", VCE, último acceso noviembre 2014, http://www.rsa.com/solutions/technology/secure/sb/11065_CLDINF_SB_0810.pdf
- [57] "VBLOCK Solution for trusted multi-tenancy: technical overview, último acceso noviembre 2014, <http://www.ergogroup.com/images/usr/vblockwhitepaper.pdf>
- [58] "HP Cloud System Matrix", HP *Software*, último acceso diciembre 2014, <http://h10032.www1.hp.com/ctg/Manual/c03918683.pdf>
- [59] "Understanding HP Cloudsystem Matrix", HP *Software*, último acceso septiembre 2014, <http://h20195.www2.hp.com/v2/getpdf.aspx/4aa0-5550enw.pdf>
- [60] "IBM Cloud Computing: Overview", IBM Cloud, último acceso noviembre 2014, <http://www.ibm.com/cloud-computing/us/en/>
- [61] "UCL Virtual Computing Lab Total Cost of Ownership", IBM *Software*, último acceso noviembre 2014, <http://public.dhe.ibm.com/common/ssi/ecm/en/ebw03004usen/EBW03004USEN.PDF>
- [62] "IBM Tivoli Storage Productivity Center", IBM *Software*, último acceso noviembre 2014, <http://www.ibm.com/software/products/en/tivostorprodcent>

- [63] “IBM Tivoli Netcool Configuration Manager”, IBM *Software*, último acceso septiembre 2014,
<http://www.ibm.com/software/products/en/tivonetconfmana>
- [64] “IBM Tivoli Provisioning Manager”, IBM *Software*, último acceso septiembre 2014,
<http://www.ibm.com/software/products/en/provisioningmanagerforsoftware>
- [65] “IBM Cloud Orchestrator”, IBM *Software*, último acceso agosto 2014,
<http://www.ibm.com/software/products/en/ibm-cloud-orchestrator>
- [66] “IBM Announces Tivoli Service Automation Manager”, a Data Center Automation Solution for Internal Cloud Computing, EMA, último acceso diciembre 2014,
<http://www.enterprisemanagement.com/research/asset.php/1125/IBM-Announces-Tivoli-Service-Automation-Manager,-a-Data-Center-Automation-Solution-for-Internal-Cloud-Computing>
- [67] “Vendor Landscape: Server Provisioning and Configuration Management”, Gartner, último acceso noviembre 2014,
http://www.gartner.com/DisplayDocument?ref=g_search&id=671022&ref=g_SiteLink
- [68] “vSphere Hypervisor”, VMware, último acceso noviembre 2014,
<http://www.vmware.com/products/vsphere-hypervisor>
- [69] “VMware ESXi”, VMware, último acceso noviembre 2014,
<http://www.vmware.com/products/esxi-and-esx/overview>
- [70] “vCenter Server”, VMware, último acceso octubre 2014,
<http://www.vmware.com/products/vcenter-server>
- [71] “vCenter Server Support Center”, VMware, último acceso noviembre 2014,
<https://www.vmware.com/support/vcenter-server>

- [72] "IBM Tivoli Service Automation Manager", IBM *Software*, último acceso noviembre 2014, <http://www.ibm.com/software/products/en/serviceautomationmanager>
- [73] "IBM Tivoli Monitoring", IBM *Software*, último acceso octubre 2014 <http://www.ibm.com/software/products/en/tivomoni>
- [74] "Tivoli Usage and Accounting Manager", IBM *Software*, último acceso octubre 2014, <http://www.ibm.com/software/products/en/tivousagandacomanaenteedit>
- [75] "Network File Systems and Linux", Developerwork, último acceso agosto 2014, <http://www.ibm.com/developerworks/library/l-network-file-systems/>
- [76] "Network File System", IBM *Software*, último acceso septiembre 2014, <http://www.ibm.com/developerworks/library/l-network-file-systems/>
- [77] "Service Management", IBM *Software*, último acceso noviembre 2014, <http://www-01.ibm.com/software/uk/tivoli/solutions/vca/>
- [78] "THE HISTORY OF FORTRAN I, II, AND III", IBM Research Laboratory, último acceso febrero 2016, <http://www.softwarepreservation.org/projects/FORTRAN/paper/p165-backus.pdf>
- [79] "Economic Perspectives on the History of the Computer Time-Sharing Industry, 1965 -1985", IEEE Annals of the History of Computing, vol.30, no. 1, pp. 16-36, January-March 2008, doi:10.1109/MAHC.2008.3
- [80] "Computación en la Nube", último acceso febrero de 2015, <http://www.computacionennube.org/computacion-en-nube/>, última visita 23 de febrero de 2015.
- [81] "Virtual Computing Lab", último acceso febrero de 2016, <http://vcl.ncsu.edu>
- [82] "xCAT", último acceso febrero 2016, <http://xcat.org>

- [83] "Pioneering Speech Recognition", último acceso marzo 2015, <http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/speechreco/>
- [84] "Amazing facts and figures about the evolution of hard disk drives", último acceso abril 2015, <http://royal.pingdom.com/2010/02/18/amazing-facts-and-figures-about-the-evolution-of-hard-disk-drives/>
- [85] "First Gigabyte Hard Drive: The IBM 3380 HAD", último acceso diciembre 2014, <http://aphelis.net/first-gigabyte-hard-drive-ibm-3380/>
- [86] "IBM makes racetrack memory breakthrough, which could come in handy someday", último acceso mayo 2015, <http://www.engadget.com/2010/12/27/ibm-makes-racetrack-memory-breakthrough-which-could-come-in-han/>
- [87] "IBM-Stanford Spintronic Science and Applications Center Scientists", último acceso septiembre 2015, <http://www.almaden.ibm.com/spinaps/scientists/index.shtml?parkin>
- [88] "IBM HIGHLIGHTS, 1990 -1995", último acceso septiembre 2015, <https://www-03.ibm.com/ibm/history/documents/pdf/1990-1995.pdf>
- [89] "FORTRAN The Pioneering Programming Language", último acceso agosto 2015, <http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/fortran/>
- [90] "COBOL COmmon Business Oriented Language", último acceso agosto de 2015
<http://www.csee.umbc.edu/courses/graduate/631/Fall2002/COBOL.pdf>
- [91] "Early Days", Dan Bricklin's Web Site, último acceso junio de 2015, <http://bricklin.com/history/saiearly.htm>
- [92] "ARPA – The Advanced Research Projects Agency", último acceso mayo de 2015, <http://www.computermuseum.li/Testpage/99HISTORYCD-ARPA-History.HTM>

- [93] "Token Ring / IEEE 802.5", último acceso junio 2015, http://docwiki.cisco.com/wiki/Token_Ring/IEEE_802.5
- [94] "Ethernet turns 40", último acceso mayo de 2015, <http://theinstitute.ieee.org/technology-focus/technology-history/ethernet-turns-40>
- [95] "Bill Gates American computer programmer, businessman, and philanthropist", último acceso noviembre 2015, <http://www.britannica.com/biography/Bill-Gates>
- [96] "RADIO SHACK TRS-80 and TANDY COMPUTER CATALOGS", último acceso octubre 2015, <http://www.radioshackcatalogs.com/computer.html>
- [97] "The Birth of IBM PC", último acceso julio de 2015, https://www-03.ibm.com/ibm/history/exhibits/pc25/pc25_birth.html
- [98] "Microsoft MS-DOS early source code", último acceso julio de 2015, <http://www.computerhistory.org/atcm/microsoft-ms-dos-early-source-code/>
- [99] "The ATLAS Computer: The Technology, último acceso junio 2015, <http://www.ukuug.org/events/linux2001/papers/html/DAspinall.html>
- [100] "Virtual Memory", último acceso junio 2015, <https://inst.eecs.berkeley.edu/~cs152/sp05/lecnotes/lec8-2.pdf>
- [101] "VMware Announces Virtual Desktop Infrastructure Alliance", último acceso julio 2015, <http://www.vmware.com/company/news/releases/vdi>
- [102] "Remote Desktop Protocol", último acceso agosto 2015, <https://msdn.microsoft.com/en-us/library/windows/desktop/aa383015%28v=vs.85%29.aspx>
- [103] "Understanding Full Virtualization, Paravirtualization and Hardware Assist", último acceso Agosto 2015, https://www.vmware.com/files/pdf/VMware_paravirtualization.pdf

- [104] “SPI model (SaaS, PaaS, IaaS)”, último acceso julio 2015, <http://searchcloudcomputing.techtarget.com/definition/SPI-model>
- [105] “Cloud computing service models”, último acceso junio 2015, <http://www.ibm.com/developerworks/cloud/library/cl-cloudservicemodels/>
- [106] “Types of cloud computing”, último acceso mayo 2015, <http://thecloudtutorial.com/cloudtypes.html>
- [107] “Public, Private & Hybrid Clouds”, último acceso junio 2015, <http://www.eci.com/cloudforum/private-cloud-explained.html>
- [108] “Practical Guide to Platform as a Service”, último acceso mayo 2015, <http://www.cloud-council.org/CSCC-Practical-Guide-to-PaaS-BrightTALK-12-3-15.pdf>
- [109] “Virtual computing Lab”, último acceso mayo 2015, <https://vcl.unc.edu/index.php?mode=selectauth>
- [110] “Getting Started on VCL”, último acceso agosto 2015, <http://help.unc.edu/help/getting-started-on-vcl/>
- [111] “ITS Virtual Lab”, último acceso junio 2015, <http://sils.unc.edu/it-services/remote-access/its-virtual-lab>
- [112] “IBM Knowledge Center”, último acceso febrero 2016, http://www.ibm.com/support/knowledgecenter/SSBH2C_7.2.4/com.ibm.isdm_7.2.4.doc/c_hardwareprerequisites.html
- [113] “IBM Redbooks Product Guide - IBM System x3500 M4”, último acceso febrero 2015, <http://www.redbooks.ibm.com/technotes/tips0852.pdf>
- [114] “Thunderbird”, último acceso abril 2015, <https://www.mozilla.org/en-US/thunderbird/>

- [115] “Tecnología de Virtualización y Software de Máquina Virtual”, último acceso marzo 2016, <http://www.vmware.com/latam/virtualization/overview.html>
- [116] “Tecnología de Virtualización y Software de Máquina Virtual”, último acceso marzo 2016, <http://www.vmware.com/latam/virtualization/getting-started.html>
- [117] “Informática en nube: Clases de virtualización – Technet Magazine”, último acceso marzo 2016, <https://technet.microsoft.com/es-es/magazine/hh802393.aspx>
- [118] “Tipos de Virtualización y Características”, último acceso marzo 2016, <http://www.osandnet.com/tipos-de-virtualizacion/>
- [119] “5 Types of Virtualization You Should Know About”, último acceso marzo 2016, <http://www.sangfor.net/blog/5-types-virtualization-you-should-know.html>
- [120] “NAS – SAN comparison”, último acceso marzo 2015, <http://www.nas-san.com/differ.html>
- [121] “What is Network-Attached Storage”, último acceso marzo 2016, <http://searchstorage.techtarget.com/definition/network-attached-storage>
- [122] “Software-Defined Networking (SDN) & Network Virtualization”, último acceso marzo 2016, <http://www.vmware.com/software-defined-datacenter/networking-security>
- [123] “Network Virtualization Solutions – Cisco”, último acceso marzo 2016, <http://www.cisco.com/c/en/us/solutions/enterprise-networks/network-virtualization-solutions/index.html>
- [124] “Cloud – Hybrid Cloud Computing – Gartner IT Glossary”, último acceso marzo 2016, <http://www.gartner.com/it-glossary/hybrid-cloud-computing>

- [125] “What is a Hybrid Cloud”, último acceso marzo 2016, <http://www.emc.com/corporate/glossary/hybrid-cloud.htm>
- [126] “IBM Cloud Computing: Infrastructure as a Service (IaaS)”, último acceso marzo 2016, <http://www.ibm.com/cloud-computing/nl/nl/iaas.html>
- [127] “Implementing IaaS solution”, último acceso marzo 2016, <http://www.sitepoint.com/implementing-an-iaas-solution/>
- [128] “What is IaaS”, último acceso marzo 2016, <http://www.interoute.com/what-iaas>
- [129] “Implementing and using IaaS cloud within the Flexible Services for the Support of Research project”, Dr. David Wallom, Oxford e-Research Centre, págs 3-6, último acceso marzo 2016, https://www.ucisa.ac.uk/~media/groups/ng/events/Cloud_2011/FleSSR%20Flexible%20Services%20for%20the%20Support%20of%20Research.aspx
- [130] “Remote Desktop Protocol”, último acceso marzo 2016, <https://msdn.microsoft.com/en-us/library/windows/desktop/aa383015%28v=vs.85%29.aspx>
- [131] “RFC 4253 – The Secure Shell (SSH) Transport Layer Protocol”, último acceso marzo de 2016, <https://tools.ietf.org/html/rfc4253>
- [132] “RFC 1094 – NFS: Network File System Protocol specification”, último acceso marzo de 2016, <https://tools.ietf.org/html/rfc1094>
- [133] “Potencie las TI híbridas: Gestión de servicios de TI para la nube, último acceso marzo de 2016, <http://www8.hp.com/h20195/v2/GetPDF.aspx%2F4AA3-9208ESE.pdf>
- [134] “La nube, impulso para una mejor gestión de TI”, último acceso marzo de 2016, <http://searchdatacenter.techtarget.com/es/opinion/La-nube-impulso-de-una-mejor-gestion-de-TI>

ANEXOS

ANEXO A: FICHA TÉCNICA IBM® SYSTEM x3500 M4

ANEXO B: INSTALACIÓN DEL COMPONENTE DE HARDWARE

ANEXO C: INSTALACIÓN DE IBM SERVICE DELIVERY MANAGER

ANEXO D: CONFIGURACIÓN DE IBM SERVICE DELIVERY MANAGER