

ESTUDIO COMPARATIVO DE PLATAFORMAS CLOUD COMPUTING PARA ARQUITECTURAS SOA

Alumno

Ing. Franco BOCCHIO

Directores

Mg. Hernán MERLINO (UNLa-UTN) y Mg. Darío RODRIGUEZ (UNLa)

TRABAJO FINAL PRESENTADO PARA OBTENER EL GRADO
DE
ESPECIALISTA EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN

**ESCUELA DE POSGRADO
FACULTAD REGIONAL DE BUENOS AIRES
UNIVERSIDAD TECNÓLOGICA NACIONAL**

Julio, 2013

RESUMEN

Las plataformas Cloud Computing están fundadas en un paradigma tecnológico moderno que ofrece nuevas alternativas a empresas de diversas envergaduras para implementar modelos de negocios innovadores. Con estos nuevos modelos de negocio las empresas pequeñas pueden hacer uso de las plataformas Cloud Computing disponiendo de la posibilidad de incrementar, tanto progresiva como abruptamente, su capacidad de cómputo y almacenamiento de datos en función de las necesidades y en tiempo real, implicando una oportunidad singular para la competencia de mercado.

En adición, las arquitecturas orientadas a servicios otorgan características de grandes beneficios para los sistemas modernos, permitiendo altos niveles de reutilización de funcionalidades, encapsulamiento y nuevas oportunidades para sociedades entre proveedores y consumidores de servicios. En este trabajo se propone, entonces, analizar y comparar en este trabajo las plataformas de los principales proveedores de servicios Cloud Computing, alineados a los distintos modelos arquitectónicos SOA que de las plataformas antedichas se desprenden con el objetivo de encontrar similitudes y diferencias, así como también faltantes.

ABSTRACT

Cloud Computing platforms are based on a modern technological paradigm that offers new alternatives to companies to implement various innovative business models. With these new business models small companies can use cloud computing platforms providing the possibility of increasing both steeply as progressive their computing power and data storage as needed in real time, implying a unique opportunity to market competition.

In addition, the service-oriented architecture features provide great benefits for modern systems allowing high reutilization levels of functionality, encapsulation and new opportunities for partnerships between service providers and consumers. Then, the purpose of this paper is to analyse and compare in platforms leading providers of cloud computing services, aligned to the SOA architectural models that from these platforms emerge.

ÍNDICE

1. INTRODUCCION	1
1.1 ¿QUÉ ES UNA ARQUITECTURA SOA?	1
1.2 EJEMPLO DE ARQUITECTURA GENÉRICA ALINEADA A SOA	3
2. ESTADO DE LA CUESTIÓN	7
2.1 CARACTERÍSTICAS DE SOA	7
2.1.1 Bajo Acoplamiento	7
2.1.2 Transparencia de Red	8
2.1.3 Reusabilidad y Granularidad	9
2.1.4 Sistemas distribuidos	10
2.1.5 Diferentes Propietarios	11
2.1.6 Heterogeneidad	12
2.2 Modelos usuales de SOA	12
2.2.1 Arquitectura de Servicios	13
2.2.2 Arquitectura de Composición de Servicios	14
2.2.3 Arquitectura de Inventario de Servicios	14
2.2.4 Arquitectura Empresarial Orientada a Servicios	15
2.3. Amazon Elastic Compute Cloud (EC2)	15
2.3.1. Descripción	16
2.3.2. Características Principales	17
2.3.2.1. Elastic	17
2.3.2.2. Totalmente Controlado	17
2.3.2.3. Flexible	17
2.3.2.4. Diseño pensado para su uso con otros Amazon Web Services	18
2.3.2.5. Fiable	18
2.3.2.6. Seguro	18
2.3.2.7. Interfaces de seguridad	18
2.3.2.8. Instancias Aisladas	18

2.3.2.9. Económico	19
2.3.2.10. Instancias en demanda	19
2.3.2.11. Instancias reservadas	19
2.3.2.12. Instancias puntuales	19
2.3.3. Amazon CloudWatch (autoescalabilidad)	20
2.3.4. Blueprints / Imágenes para acelerar el aprovisionamiento	20
2.3.5. Amazon EC2 con Microsoft Windows Server y SQL Server	21
2.3.6. Soporte para Sistemas operativos Linux	22
2.3.7. Soporte para almacenamiento de datos	22
2.3.7.1. Amazon Simple Storage Service (Amazon S3)	22
2.3.7.2. Amazon Relational Database Service (Amazon RDS) (beta)	23
2.3.7.3. Amazon SimpleDB (beta)	23
2.3.8. Soporte para Colas	23
2.3.9. Alternativas de Hipervisor	24
2.4. WINDOWS AZURE	24
2.4.1. Descripción	25
2.4.2. Características Principales	25
2.4.2.1. Siempre Disponible	25
2.4.2.2. Alto nivel de servicio	25
2.4.2.3. Abierto	26
2.4.2.4. Servidores ilimitados, almacenamiento ilimitado	26
2.4.2.5. Gran capacidad	26
2.4.2.6. Cache distribuida y CDN	26
2.4.3. Autoescalabilidad	27
2.4.4. Blueprints / Imágenes para acelerar el aprovisionamiento	27
2.4.5. Soporte para Sistemas operativos Microsoft Windows	28
2.4.6. Soporte para Sistemas operativos Linux	28
2.4.7. Soporte para almacenamiento de datos	29
2.4.7.1. SQL Server en máquinas virtuales de Windows Azure	29

2.4.7.2. Base de datos SQL	30
2.4.7.3. Tablas	31
2.4.7.4. Blob no estructurado	31
2.4.8. Soporte para colas	31
2.4.9. Alternativas de Hipervisor	32
2.4.9.1. Rápido	32
2.4.9.2. Pequeño	32
2.4.9.3. Estrechamente integrado con el núcleo	32
2.5. GOOGLE APP ENGINE	33
2.5.1. Descripción	33
2.5.2. Características Principales	33
2.5.2.1. Rápido para iniciar	34
2.5.2.2. Fácil de usar	34
2.5.2.3. Conjunto rico de APIs	34
2.5.2.3. Escalabilidad Inmediata	34
2.5.2.4. Modelo de negocio de pago por uso	34
2.5.2.5. Infraestructura probada	35
2.5.3. Autoescalabilidad	35
2.5.4. Soporte para Sistemas operativos Linux	35
2.5.5. Soporte para almacenamiento de datos	36
2.5.6. Soporte para colas	37
2.5.7. Alternativas de Hipervisor	37
2.6. RED HAT OPENSIFT	38
2.6.1. Descripción	38
2.6.2. Características Principales	38
2.6.2.1. Prestación de servicios de aplicación acelerada	39
2.6.2.2. Dependencia minimizada con el proveedor	39
2.6.2.3. Pilas de aplicaciones de autoservicio y en demanda	39
2.6.2.4. Flujos de trabajo estandarizados para Desarrolladores	40

2.6.2.5. Elección de los lenguajes de programación y los softwares de base	40
2.6.2.6. Aplicaciones empresariales con Java EE6	40
2.6.2.7. Servicios de base de datos incorporados	40
2.6.2.8. Sistema de cartucho Extensible para agregar servicios	40
2.6.2.9. Soporte para múltiples entornos (desarrollo / Pruebas / Producción)	41
2.6.2.10. Dependencia y Gestión de la Construcción	41
2.6.2.11. Integración Continua y Gestión de Versiones	42
2.6.2.12. Gestión de versiones de código fuente	42
2.6.2.13. Inicio de sesión Remoto por SSH al contenedor de aplicaciones	42
2.6.2.14. Integración con IDE	42
2.6.2.15. Línea de comandos enriquecida	42
2.6.2.16. Desarrollo de aplicaciones móviles	43
2.6.2.17. Redundancia de componentes del sistema para alta disponibilidad	43
2.6.2.18. Aprovisionamiento automático de la pila de aplicaciones	43
2.6.2.19. Escalabilidad automática de aplicación	43
2.6.2.20. Elección de la infraestructura de nube	44
2.6.3. Autoescalabilidad	44
2.6.4. Blueprints / Imágenes para acelerar el aprovisionamiento	45
2.6.6. Soporte para Sistemas operativos Linux	45
2.6.7. Soporte para almacenamiento de datos	46
2.6.8. Soporte para colas	46
2.6.9. Alternativas de Hipervisor	47
2.7. IBM SMARTCLOUD	47
2.7.1. Descripción	47
2.7.2. Características Principales	48
2.7.3. Autoescalabilidad	49
2.7.4. Blueprints / Imágenes para acelerar el aprovisionamiento	50
2.7.5. Soporte para Sistemas operativos Microsoft Windows y Linux	51
2.7.6. Soporte para almacenamiento de datos	51

2.7.7. Soporte para colas	53
2.7.8. Alternativas de Hipervisor	54
2.7.9. Soporte para Datagrids	56
2.8. VMWARE VCLOUD SUITE	56
2.8.1. Descripción	57
2.8.2. Características Principales	58
2.8.2.1. Agilice el acceso a los recursos y a las aplicaciones	59
2.8.2.2. Simplifique y automatice la gestión de operaciones	60
2.8.2.3. Mejores acuerdos de nivel de servicio para todas las aplicaciones	60
2.8.2.4. Infraestructura de la nube	60
2.8.3. Autoescalabilidad	62
2.8.4. Blueprints / Imágenes para acelerar el aprovisionamiento	62
2.8.4.1. Infraestructura Virtual entregada con confianza	63
2.8.5. Soporte para Sistemas operativos Microsoft Windows y Linux	63
2.8.6. Soporte para almacenamiento de datos	63
2.8.7. Soporte para colas	64
2.8.8. Alternativas de Hipervisor	65
2.9. OPENSTACK	66
2.9.1. Descripción	66
2.9.2. Características Principales	67
2.9.2.1. Infraestructura de Cómputo (Nova)	67
2.9.2.2. Infraestructura de Almacenamiento (Swift)	68
2.9.2.3. Servicios de Imagen (Glance)	68
2.9.3. Autoescalabilidad	69
2.9.4. Blueprints / Imágenes para acelerar el aprovisionamiento	69
2.9.3.1. Oz	70
2.9.3.2. vmbuilder	70
2.9.3.3. BoxGrinder	70
2.9.3.4. VeeWee	70

2.9.3.5. Imagefactory	70
2.9.5. Soporte para Sistemas operativos Microsoft Windows	71
2.9.6. Soporte para Sistemas operativos Linux	71
2.9.7. Soporte para almacenamiento de datos	72
2.9.8. Soporte para colas	73
2.9.9. Alternativas de Hipervisor	74
3. ESTUDIO COMPARADO DE LAS ARQUITECTURAS RELEVADAS	75
3.1. DESCRIPCIÓN DEL PROBLEMA	75
3.2. CARACTERÍSTICAS CONSIDERADAS	75
3.1.1. Definición de las características evaluadas	76
3.3. MATRIZ COMPARATIVA	79
3.4. ESTUDIO DE LA TABLA COMPARATIVA	80
3.4.1 Escalabilidad automática (AutoScaling)	80
3.4.2 BluePrints/Imágenes para acelerar el aprovisionamiento	81
3.4.3 Soporte para lenguajes	83
3.4.4 Soporte para almacenamiento de datos	84
3.4.5 Soporte para Colas y Servidores Web	86
3.4.6 Alternativas de Hipervisores	87
3.4.7 Cache In-Memory distribuido / Datagrid	88
4. REFERENCIAS	91

ÍNDICE DE FIGURAS

Figura 1.1	Arquitectura de 3 Capas Alineada a SOA	15
	[Microsoft Corp., 2003]	
Figura 2.1	Bajo Acoplamiento del Servicio	18
	Adaptado de [Pulier & Tylor, 2006]	
Figura 2.2	Sistemas Distribuidos con Diferentes Propietarios	21
	Adaptado de [Josuttis., 2007]	
Figura 2.3	Solución completa e integrada en la nube	68
	[VMware Inc., 2013e]	
Figura 2.4	Responsabilidades de Productos	69
	[VMware Inc., 2013e]	

ÍNDICE DE TABLAS

Tabla 1.1	Componentes de la arquitectura genérica	13
Tabla 2.1	Arquitecturas tecnológicas más comunes	23
Tabla 3.1	Caract. consideradas para el análisis de las plataformas en la nube	88
Tabla 3.2	Matriz Comparativa	91

NOMENCLATURA

AMI	Imagen de máquina Amazon (por sus siglas en inglés “Amazon Machine Image”)
Backend	Parte del software que procesa la entrada desde el front-end
CDN	Red de entrega de contenido (por sus siglas en inglés “Content delivery Network”)
EBS	Bloque de almacenamiento elástico (por sus siglas en inglés “Elastic Block Store”)
EC2	Computación elastic en la nube (por sus siglas en inglés “Elastic Compute Cloud”)
GIT	Repositorio
Host	Computadora conectada a una red que provee y utiliza servicios de ella
IaaS	Infraestructura como servicio (por sus siglas en inglés “Infras"tructure As A Service”)
Middleware	Middleware es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos
OASIS	Organización para la estandarización avanzada de la información estructurada (por sus siglas en inglés “Organization for the Advancement of Structured Information Standards”)
PaaS	Plataforma como servicio (por sus siglas en inglés “Platform As A Service”)
Proxy	Un proxy, en una red informática, es un programa o dispositivo que realiza una acción en representación de otro, esto es, si una hipotética máquina A solicita un recurso a una C, lo hará mediante una petición a B; C entonces no sabrá que la petición procedió originalmente de A.
Sandbox	Red de entrega de contenido (por sus siglas en inglés “Content delivery Network”)
SLA	Nivel de Acuerdo de servicio (por sus siglas en inglés “Service Level Agreement”)
SOA	Arquitectura Orientada a Servicios (por sus siglas en inglés “Service Oriented Architecture).
Stub	porción de código utilizada para sustituir a alguna funcionalidad de programación, simulando el comportamiento del código existente

Snapshot

En informática, una copia instantánea de volumen o Snapshot es una instantánea del estado de un sistema en un momento determinado

1. INTRODUCCIÓN

En este Capítulo se plantea qué es una arquitectura SOA (sección 1.1), y se presenta un ejemplo de una arquitectura genérica alineada a SOA (sección 1.2).

1.1 ¿QUÉ ES UNA ARQUITECTURA SOA?

Erl [2005] define a SOA como una representación de una arquitectura abierta, extensible y federada basada en composición, que promueve la orientación a los servicios interoperables e independientes de los proveedores, los cuales pueden ser identificados en catálogos con gran potencial de reutilización e implementados como servicios Web. SOA puede establecer una abstracción de la lógica del negocio y la tecnología, resultando en un bajo acoplamiento entre dominios. Newcomer & Lomow [2005] dicen que SOA es el producto de la evolución de plataformas de tecnología que se solían utilizar con frecuencia, preservando las características exitosas de las arquitecturas tradicionales. Podemos entender a las arquitecturas SOA como un estilo de diseño que guía los aspectos de creación y uso de servicios de negocio a través de su correspondiente ciclo de vida. Además define y aprovisiona la infraestructura de tecnologías de la información que permite a diferentes aplicaciones intercambiar datos y participar en los procesos de negocio, independientemente del sistema operativo o de los lenguajes de programación con los cuales estos servicios (y aplicaciones) fueron desarrollados e implementados.

Josuttis [2007] observa que muchas definiciones de SOA incluyen el término “servicios web”, sin embargo, es necesario hacer la distinción de estos conceptos y aclarar que SOA no es lo mismo que servicios Web puesto que SOA, a diferencia de los servicios web, define y trata un paradigma, en tanto que los servicios web son solo una forma posible de consumir la infraestructura utilizando una estrategia de implementación específica.

Podemos decir entonces que SOA es un paradigma arquitectónico que permite el tratamiento de procesos de negocio distribuidos de nuevos sistemas heterogéneos que se encuentran bajo el control o responsabilidad de diferentes propietarios, siendo sus conceptos clave principales los servicios, la interoperabilidad entre lenguajes, y el bajo acoplamiento. En tanto que los ingredientes principales de SOA son la infraestructura, la arquitectura y los procesos.

Es preciso también aclarar que SOA no es una “bala de plata” (silver bullet) ni una tecnología específica, por lo cual, hay casos en los cuales SOA puede ser un paradigma apropiado y otros en los cuales pueden no ser la solución más adecuada o conveniente.

SOA puede ser entendido también como un modelo de diseño cuyas raíces conceptuales establecen los principios de encapsulamiento de la lógica de la aplicación por medio de servicios, los cuales pueden interactuar a través de protocolos de comunicación estandarizados.

La arquitectura orientada a servicios representa la evolución a un nuevo modelo que permite la construcción de aplicaciones distribuidas. Los servicios implementados en esta arquitectura son distribuidos en componentes que proveen interfaces bien definidas (las cuales funcionan como contratos), que para el caso de los servicios web procesan y distribuyen mensajes basados en XML. Las soluciones basadas en servicios encuentran sentido cuando se trata la construcción de aplicaciones y sistemas que resuelven problemas organizacionales, departamentales y corporativos. Un negocio con múltiples sistemas y aplicaciones desarrolladas en diferentes plataformas pueden utilizar SOA para construir soluciones integradas de bajo acoplamiento que implementan flujos de trabajos unificados y cooperativos.

Hasan y Duran [2006] presentan el siguiente ejemplo para esclarecer el marco teórico de SOA: el concepto de servicios puede resultar familiar para cualquiera que realice compras en línea utilizando aplicaciones web de tipo “e-commerce” (comercio electrónico; algunos ejemplos de estas aplicaciones pueden ser Ebay, Amazon, etc). Una vez que se realiza un pedido, se debe proporcionar al sistema los datos de una tarjeta de crédito, la cual es típicamente autorizada y actualizada (gasto) por un proveedor de servicios externo. Una vez que la orden ha sido consumada, la compañía de comercio electrónico coordina la entrega con un proveedor de servicios de envíos para entregar el producto que el cliente adquirió. SOA no es la primera arquitectura distribuida que permite utilizar componentes a través de interfaces de dominios independientes, por lo cual su aporte diferencial no reside precisamente en esta característica. SOA usa los servicios web como puntos de entrada (entry points) de las aplicaciones, los cuales conceptualmente son equivalentes a los componentes proxies y stubs tradicionales utilizados por años en arquitecturas distribuidas, con excepción de que la interacción entre los proveedores de servicios web y los consumidores se caracterizan por presentar mucho menor acoplamiento.

SOA también es un paradigma único en tanto que incorpora factores que son críticamente importantes para el negocio, tales como fiabilidad de servicios, integridad de mensajes, integridad transaccional y protocolos de seguridad para los mensajes.

Muchas aplicaciones implementan modelos de comunicación sincrónicos rígidos bajo un flujo de trabajo lineal el cual es altamente susceptible a fallas en algún punto. SOA asume que los errores ocurren y en efecto ocurrirán, para lo cual ofrece estrategias para una eficiente administración de estos. Por ejemplo, si un servicio falla al recibir una petición de mensaje en su primer intento, la arquitectura SOA propone que su implementación reintente el envío de este mensaje, y si el servicio se encontrara por completo no disponible (lo cual nunca debería ocurrir en una SOA robusta), la

arquitectura debe estar diseñada para evitar posibles fallas catastróficas que podrían interrumpir por completo la recepción de peticiones (requests).

En conclusión, y en un sentido más amplio, podemos decir que SOA representa un proceso de maduración de la tecnología como el incremento de uso de los servicios web y tecnologías de integración en general. SOA reconoce que los sistemas de misión crítica basadas en tecnologías distribuidas deben proporcionar ciertas garantías. Debe asegurar entonces que las solicitudes de servicio se entreguen correctamente, que serán respondidas en el momento oportuno y que se publicarán sus políticas de comunicación e interfaces.

1.2 EJEMPLO DE ARQUITECTURA GENÉRICA ALINEADA A SOA

En la figura 1.1 se presenta un diagrama que modela una arquitectura genérica de 3 capas alineada a SOA. A continuación se presenta la tabla 1.1 (fragmentada en 1.1a, 1.1b y 1.1c), que describe cada uno de los componentes que integran esta arquitectura genérica:

Término	Definición
Agentes de Servicio	Este componente (Service Agents) permite a las implementaciones de los servicios de negocio acceder a otras instancias de servicios, con el propósito de reutilizar funcionalidades, testear minimizar las pruebas funcionales (las funcionalidades reutilizables se encuentran encapsuladas en componentes y se implementan una única vez), minimizar el mantenimiento (cuando se implementan mejoras sobre un componente de servicios reutilizables, todos los servicios que refieran a este servicio podrán beneficiarse con las mejoras implementadas).

Tabla 1.1a. Componentes de la arquitectura genérica. Adaptado de [Microsoft Corp., 2003].

Término	Definición
Componentes de IU	Se refiere a los componentes de interfaz de usuario que serán implementados y desplegados en esta arquitectura
Componentes de Lógica de Acceso a Datos	Estos componentes contienen la lógica de acceso a datos (Data Access Logic Components), abstrayendo y separando concretamente la capa de lógica del negocio de la capa de acceso a datos (separación y asignación de responsabilidades por capas).
Componentes de Negocio	Corresponden a la implementación de los servicios de negocio, comprendiendo fundamentalmente las funcionalidades y operaciones de negocio requeridas. Implementan las operaciones definidas en las interfaces del negocio (Service Interfaces).
Componentes de Procesos de IU	Hace referencia a los componentes que implementan procesos competentes a la capa de interfaz de usuario.
Entidades de Negocio	Corresponden a las entidades del modelo de dominio (Business Entities). Son utilizadas tanto por los servicios como por los componentes de acceso a datos.
Flujo de Trabajo del Negocio	Este componente encapsula e implementa mecanismos de flujos de trabajo (Business Workflow) que permiten formalizar y organizar los procesos del negocio para el modelo de dominio en cuestión.
Fuentes de datos	Algunas de las fuentes de datos posibles (Data Sources) podrían ser una base de datos relacional, archivos de datos, un data grid, bases de datos NoSQL, etc.
Interfaces de Servicios	Las interfaces de servicios funcionan como contratos, definiendo formalmente las operaciones (capacidades) que brindarán los servicios basados en esta arquitectura.
Seguridad, gestión operativa, Comunicaciones	Estos componentes (Security, Operational Management, Communication) operan de manera transversal porque su funcionamiento debe considerarse en todas las capas. La aplicación podría utilizar también componentes para realizar la administración de excepciones, autorizar a los usuarios a realizar ciertas tareas y comunicarse con otros servicios y aplicaciones.

Tabla 1.1b. Componentes de la arquitectura genérica. Adaptado de [Microsoft Corp., 2003].

Término	Definición
Usuarios	Se refiere a los usuarios que harán uso de la aplicación.

Tabla 1.1c. Componentes de la arquitectura genérica. Adaptado de [Microsoft Corp., 2003].

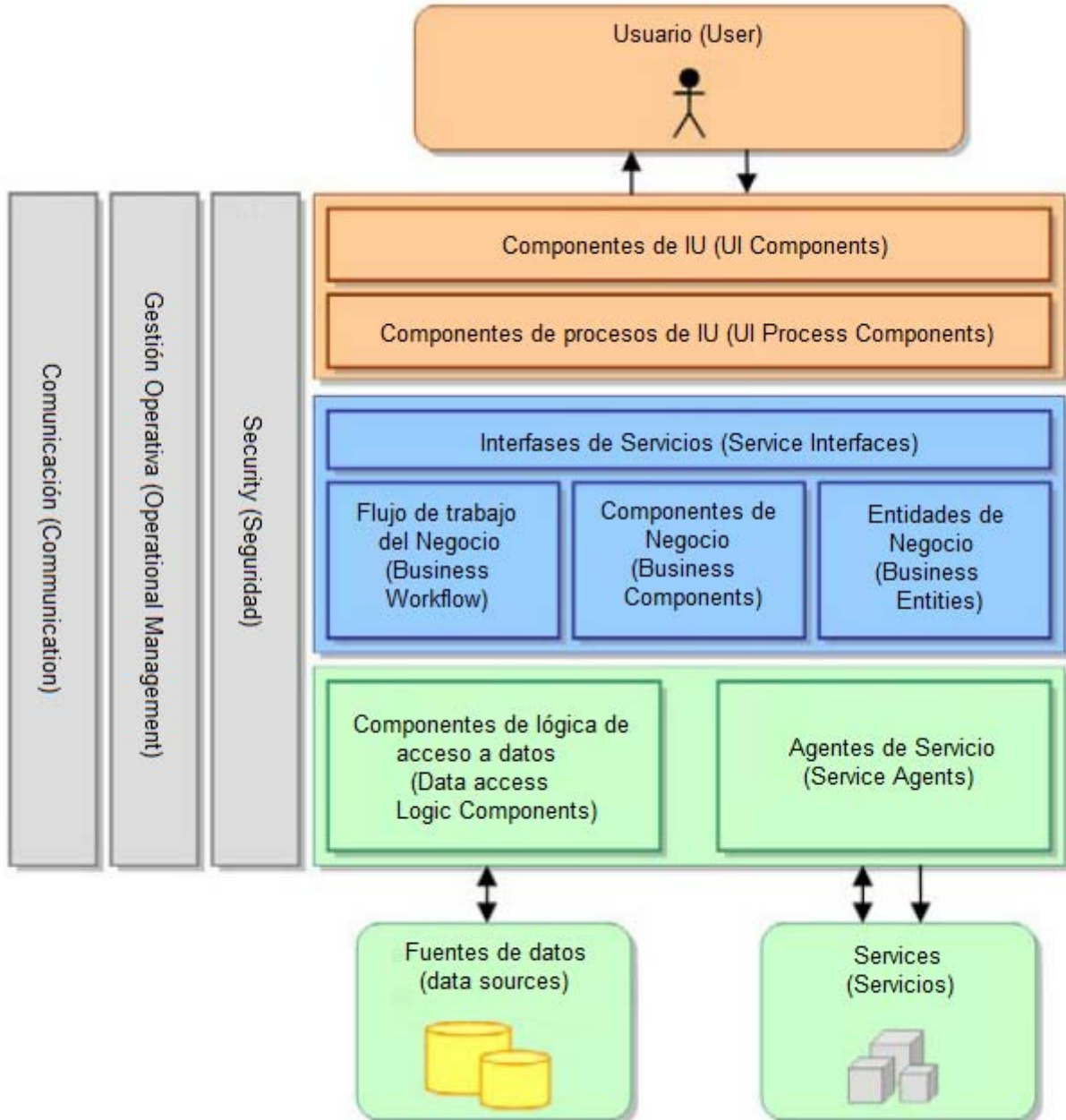


Figura 1.1. Arquitectura de 3 capas alineada a SOA. Adaptado de [Microsoft Corp., 2003].

2. ESTADO DE LA CUESTIÓN

En este capítulo se presenta el estado de la cuestión sobre las distintas plataformas de Cloud Computing ofrecidas por los más destacados proveedores. Se mencionarán las principales características de SOA (sección 2.1), algunos modelos usuales de SOA (sección 2.2), y luego se analizarán las plataformas Amazon Elastic Compute Cloud EC2 (sección 2.3), Windows Azure (sección 2.4), Google App Engine (sección 2.5), Red Hat OpenShift (sección 2.6), IBM SmartCloud (sección 2.7), VmWare VCloud Suite (sección 2.8) y OpenStack (sección 2.9).

2.1 CARACTERÍSTICAS DE SOA

En esta sección se presentan las principales características de SOA, tales como Bajo Acoplamiento (sección 2.1.1), transparencia de Red (sección 2.1.2), reusabilidad y granularidad (sección 2.1.3), sistemas distribuidos (sección 2.1.4), diferentes propietarios (sección 2.1.5) y heterogeneidad (sección 2.1.6).

2.1.1 Bajo Acoplamiento

Stevens y colaboradores [1974] señalan que el acoplamiento es el grado en que cada módulo de programa depende de cada uno de los otros módulos de programa. El bajo acoplamiento generalmente se correlaciona con la alta cohesión y viceversa.

Pulier y Tylor [2006] destacan que los servicios web poseen la característica de estar basados en una estructura flexible mediante la cual, una vez que una pieza de software ha sido expuesta como un servicio web, es relativamente fácil de mover a otro equipo, puesto que la funcionalidad se encuentra abstraída de la interfaz que define el contrato de sus operaciones.

La figura 2.1 ilustra el bajo acoplamiento del servicio. En la parte 1 del dibujo, un miniordenador accede a un servicio web que se ha expuesto en un mainframe. Digamos que, sin embargo, el propietario de la unidad central quiere remplazar la máquina antigua con un nuevo servidor Sun. Como vemos en la parte 2, la máquina Sun sustituye a la unidad central, pero la minicomputadora, que es el consumidor del servicio web no tiene conocimiento de esta sustitución. La minicomputadora se sigue comunicando por medio del protocolo SOAP. Es completamente transparente para el consumidor del servicio (cliente) que la interfaz del servicio esté implementada

en una computadora central, en un equipo Windows o en cualquier otra plataforma. Una vez que la unidad central ha sido sustituida por la máquina Sun, la minicomputadora continúa accediendo al servicio sin percibir este cambio. En las partes 3 y 4 de la figura antedicha el proceso de sustitución de ordenadores continúa. El propietario de la minicomputadora la reemplaza con un PC. El PC que está equipado con su propia interfaz SOAP puede acceder fácilmente a los servicios web expuestos por la máquina Sun. Si por cualquier razón el propietario del servicio decidiera sustituir la máquina Windows por otro servidor Sun, no habría problema alguno; una vez más se podría acceder al servicio sin necesidad de realizar ninguna modificación a éste. A continuación se presenta la figura 2.1, que ilustra el bajo acoplamiento de servicios.

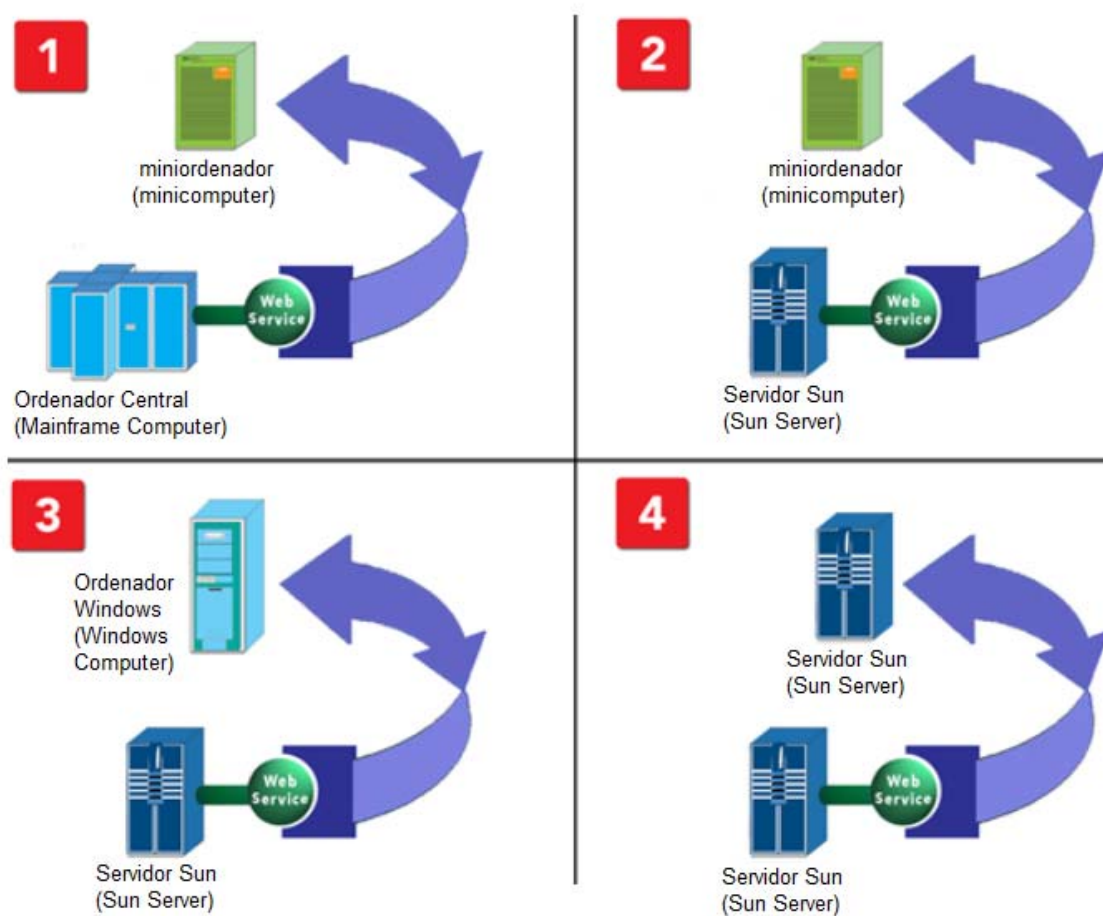


Figura 2.1. Bajo acoplamiento del servicio. Adaptado de [Pulier & Tylor, 2006].

2.1.2 Transparencia de Red

Pulier y Tylor [2006] argumentan que debido a que el acoplamiento entre los servicios web es "flojo" y que los consumidores y los proveedores de servicios de Internet envían mensajes entre sí mediante protocolos abiertos de Internet, los servicios web ofrecen transparencia total de la red a los

que los emplean. Transparencia de la red se refiere a la capacidad de un servicio web para estar activos en cualquier parte de una red, o grupo de redes, sin tener ningún impacto en su capacidad de funcionar. Debido a que cada servicio web tiene su propia URL, servicios web tienen una flexibilidad similar a sitios web en Internet. De la misma manera que no hay diferencia en qué parte del mundo un sitio web está alojado para poder ser navegado, un servicio web se puede encontrar en cualquier equipo que esté conectado a la red y se comunica con protocolos de Internet. Cuando navegamos por ejemplo Amazon.com para comprar un libro, no existe ninguna necesidad de conocer dónde residen las aplicaciones a las cuales estamos está accediendo con nuestro navegador, lo único que se necesita saber es la dirección web.

Un mismo servicio web puede estar situado en dos dominios diferentes. Si por alguna razón el dominio A se encontrara no disponible, entonces el consumidor del servicio podría acceder al servicio web desde el dominio B de manera alternativa. Lo único que tiene que ocurrir es la modificación de la dirección (URL) del servicio web en el documento WSDL, de manera que se establezca un vínculo entre el cliente (consumidor del servicio) y la nueva dirección del servicio web en el dominio B.

Dada nuestra amplia experiencia con la Internet en los últimos años, la propiedad de transparencia de la red puede no parecer tan importante, pero en realidad se trata de un aspecto fundamental para el futuro de la informática. La combinación de acoplamiento débil y la transparencia de red presentan nada menos que una revolución en la informática empresarial, no porque se trate de una idea novedosa, sino debido a que la infraestructura y los estándares han llegado por fin para que sea una realidad. Las empresas han gastado fortunas en los últimos años en la gestión de interfaces que hacen factible la interoperabilidad de los equipos en entornos distribuidos. Algunas empresas Norteamericanas gastan cientos de miles de millones de dólares por año en tecnologías de la información.

2.1.3 Reusabilidad y Granularidad

Carter [2007] manifiesta que un servicio es un software reutilizable y auto-contenido, independiente de las aplicaciones y las plataformas de computación en las cuales se ejecuta. Los servicios tienen interfaces bien definidas y permiten una correspondencia entre las tareas de negocio y los componentes exactos de TI necesarios para ejecutar la tarea. Los servicios SOA se centran en tareas de nivel de negocio, actividades e interacciones. La relación entre los servicios y los procesos de negocios es crítica. Un proceso de negocio es un conjunto de tareas de negocios relacionados que

abarcan personas, sistemas e información para producir un resultado o producto específico. Con SOA, un proceso se compone de un conjunto de servicios.

Antes de SOA, el foco estaba en sub-tareas técnicas. Puede haber oído que la gente llama a esto un buen "nivel de granularidad" o bajo "grado de abstracción." Por la simplicidad de SOA, sabemos que es más que una simple relación uno-a-uno entre los pasos de un proceso (como la comprobación de una calificación crediticia) y los servicios que están diseñados para apoyar ese proceso de negocio flexible.

Cada empresa tiene una visión diferente respecto de la granularidad que requieren sus servicios en función de su diseño de negocios. En pocas palabras, la granularidad es la cantidad de función que un servicio expone. Por ejemplo, un servicio de granularidad fina proporciona unidades más pequeñas de un proceso de negocio, y un servicio de granularidad gruesa proporciona una tarea de negocio más amplia que contiene un mayor número de pasos.

Sandy Carter opina que los servicios no pueden ser demasiado grandes o demasiado pequeños, sino que deben tener la granularidad adecuada. Diseñar y decidir la granularidad de los servicios es una cuestión clave que conduce al éxito. Si el servicio es demasiado grande, será menos reutilizable. Si el servicio es demasiado pequeño, provoca una disminución en el rendimiento y mala asignación de tareas entre los negocios y los servicios que los apoyan.

Determinar qué tan grande o pequeño debe diseñarse un servicio es función qué tan atómica es la composición de su función.

Es importante observar que este concepto de servicios es una de las claves para hacer que SOA sea el lenguaje de los negocios. La mayoría de los líderes empresariales no comprenden el valor de SOA. En su lugar, se centran -y con razón- en el problema en cuestión. Debido a estos servicios de negocio, este lenguaje y la vinculación de los servicios de negocio y SOA se convierten en una pieza fundamental para la solución del problema, y constituyen la futura misión estratégica.

2.1.4 Sistemas distribuidos

Josuttis [2007] explica que a medida que las empresas crecen se vuelven más y más complejas, y cada vez se involucran más empresas y sistemas. Hay una integración y un cambio constantes. SOA es muy adecuado para tratar con sistemas distribuidos complejos. De acuerdo con el modelo de referencia SOA de OASIS, es un paradigma para "organizar y utilizar capacidades distribuidas.

NOTA:

Una definición más adecuada para "capacidades distribuidas" en IT sería "sistemas distribuidos", o como dice la definición de SOA de Wikipedia: "nodos de una red" o "recursos de una red."

SOA permite a las entidades que necesitan ciertas capacidades distribuidas, localizar y hacer uso de esas capacidades. En otras palabras, facilita las interacciones entre los proveedores de servicios y sus consumidores, lo que permite la realización de funcionalidades de negocio.

2.1.5 Diferentes Propietarios

Jossutis [2007] recuerda que la definición de SOA del modelo de Referencia de OASIS dice que las capacidades distribuidas "pueden estar bajo el control de diferentes dominios de propiedad." Este es un punto muy importante que a menudo es suprimido en definiciones de SOA. Esta es una de las claves para ciertas propiedades de SOA, y una razón importante por la que SOA no es sólo un concepto técnico.

SOA incluye prácticas y procesos que se basan en el hecho de que las redes de los sistemas distribuidos no son controladas por los propietarios individuales. Diferentes equipos, diferentes departamentos o incluso diferentes empresas pueden gestionar diferentes sistemas. Por lo tanto, diferentes plataformas, programas, prioridades, presupuestos, etc. deben ser tenidos en cuenta. Este concepto es clave para la comprensión de SOA y de grandes sistemas distribuidos en general.

A continuación se presenta la figura 2.2, que ilustra un conjunto de sistemas distribuidos con diferentes propietarios.

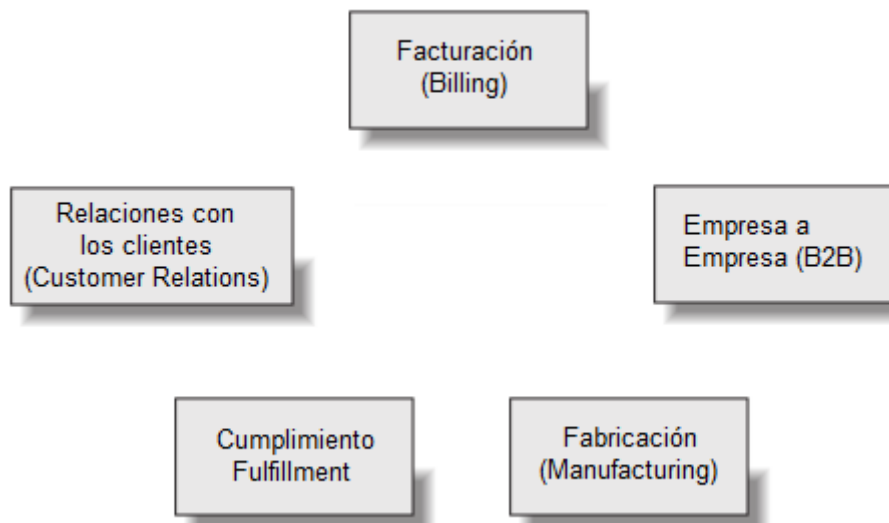


Figura 2.2. Sistemas distribuidos con diferentes propietarios. Adaptado de [Josuttis, 2007].

Las formas de lidiar con problemas y realizar modificaciones en los entornos con diferentes propietarios y en entornos donde se dispone de control total, pueden variar. No se puede implementar la funcionalidad y modificar el comportamiento de la misma manera en grandes

sistemas como en sistemas más pequeños. Una consideración importante es que "la política" entra en juego: hay que comprometerse con los demás, y hay que aceptar que existen diferentes prioridades y soluciones. Porque no se puede controlar todo, hay que aceptar que no siempre puede ser capaz de hacer las cosas a su manera.

2.1.6 Heterogeneidad

Josuttis [2007] explica que una diferencia muy importante entre los sistemas pequeños y grandes es la falta de armonía. Todos lo sabemos por experiencia (aunque podríamos no estar de acuerdo acerca de si se trata de un fenómeno natural o el resultado de un mal diseño). Los grandes sistemas utilizan distintas plataformas, diferentes lenguajes de programación (y paradigmas de programación), e incluso diferentes componentes middleware. Se trata de un lío de mainframes, ejércitos de SAP, bases de datos, aplicaciones J2EE, pequeños motores de reglas, etc. En otras palabras, son heterogéneos.

En el pasado, se han propuesto una gran cantidad de métodos para resolver el problema de la integración de sistemas distribuidos mediante la eliminación de la heterogeneidad: "Vamos a armonizar, y todos los problemas habrán desaparecido," "Vamos a sustituir todos los sistemas con aplicaciones J2EE", "Vamos a usar CORBA en todas partes", "Usemos serie MQ," y así sucesivamente. Pero todos sabemos que estos métodos no funcionan. Grandes sistemas distribuidos con diferentes propietarios son heterogéneos.

Esto es un hecho, y por lo tanto algo que debemos aceptar la hora de diseñar soluciones distribuidas de gran tamaño.

2.2 Modelos usuales de SOA

En esta sección se presentan algunos modelos usuales de SOA, entre ellos arquitectura de servicios (sección 2.2.1), arquitectura de composición de servicios (sección 2.2.2), arquitectura de inventario de servicios (sección 2.2.3) y arquitectura empresarial orientada a servicios (sección 2.2.4).

Erl [2009] observa que todos los programas de software acaban siendo compuestos, o bien residiendo en alguna forma de combinación arquitectónica de recursos, tecnologías y plataformas (relacionadas con la infraestructura o no). Si nos tomamos el tiempo para personalizar estos elementos arquitectónicos, podemos establecer un ambiente refinado y normalizado para la

aplicación de programas de software también personalizados. El diseño intencional de la tecnología de arquitectura es muy importante para la computación orientada a servicios. Es esencial crear un entorno en el que los servicios pueden ser recompuestos varias veces para maximizar la realización de las necesidades de negocio. El beneficio estratégico para personalizar el alcance, el contexto y los límites de una arquitectura es significativo. Para comprender mejor los mecanismos básicos de SOA, ahora tenemos que estudiar los tipos más comunes de las arquitecturas tecnológicas que existen.

Se presentan a continuación, en la tabla 2.1, estos tipos de arquitecturas tecnológicas más comunes, con sus correspondientes definiciones.

Término	Definición
Arquitectura de Composición de Servicios	La arquitectura de un conjunto de servicios reunidos en composición de servicios
Arquitectura de Inventario de Servicios	Arquitectura que soporta un conjunto de servicios relacionados que son independientemente estandarizados y gobernados
Arquitectura empresarial Orientada a Servicios	La arquitectura de la propia empresa (para cualquier tamaño de empresa) orientada a servicios
Arquitectura de Servicios	Se trata de la arquitectura de un único servicio

Tabla 2.1. Arquitecturas tecnológicas más comunes. Adaptado de [Microsoft Corp., 2003].

2.2.1 Arquitectura de Servicios

[Erl, 2009] La arquitectura de servicios es una arquitectura de tecnología limitada al diseño físico de un programa de software diseñado como un servicio. Esta forma de arquitectura de la tecnología es comparable en alcance a una arquitectura de componentes, excepto que se suele contar con una mayor cantidad de extensiones de infraestructura para apoyar su necesidad de aumentar la fiabilidad, el rendimiento, la escalabilidad, la previsibilidad del comportamiento, y sobre todo su necesidad de una mayor autonomía. El ámbito de aplicación de una arquitectura de servicios también tenderá a ser mayor debido a que un servicio puede, entre otras cosas, abarcar múltiples componentes.

Considerando que no siempre fue común documentar una arquitectura diferente para cada componente en aplicaciones distribuidas tradicionales, la importancia de la producción de servicios que deben existir como programas de software independiente, muy autosuficiente y autónomo, requiere que cada uno de ellos sea diseñado de forma individual.

2.2.2 Arquitectura de Composición de Servicios

[Erl, 2009] El propósito fundamental de entregar una serie de servicios independientes es que se puedan combinar en composiciones de servicios totalmente funcionales capaces de automatizar las tareas de negocio más grandes y más complejas.

Cada composición de servicios tiene una arquitectura de composición de servicios correspondiente. De la misma manera que una arquitectura de aplicación para un sistema distribuido incluye las definiciones de la arquitectura de sus componentes individuales, esta forma de arquitectura abarca las arquitecturas de servicio de todos los servicios que participan.

Una arquitectura de composición (especialmente una que compone servicios que encapsulan sistemas heredados dispares) puede ser comparado con una arquitectura de integración tradicional. Esta comparación por lo general sólo es válida parcialmente, ya que las consideraciones de diseño destacadas por la orientación a servicios aseguran que el diseño de una composición de servicios es muy diferente a la de las aplicaciones integradas.

Por ejemplo, una diferencia en cómo se documentan las arquitecturas de composición tiene que ver con la medida de detalle que estas incluyen acerca de los servicios reutilizables que participan en la composición. Debido a que estos tipos de especificaciones de arquitecturas de servicios están a menudo protegidas (según las necesidades planteadas por el principio de abstracción de Servicios), una arquitectura de composición sólo podrá hacer referencia a la interfaz técnica y acuerdo de nivel de servicio (SLA), publicado como parte del contrato público de servicios.

2.2.3 Arquitectura de Inventario de Servicios

[Erl, 2009] Un inventario de servicio es un conjunto de servicios estandarizados y gobernados de manera independiente entregados en arquitectura pre-definida.

Esta colección representa un alcance significativo que excede el límite de procesamiento de un único proceso de negocio e idealmente abarca numerosos procesos de negocio. El alcance y los límites de una arquitectura de inventario de servicio pueden variar.

Idealmente, el inventario de servicios es primero modelado conceptualmente, lo que lleva a la creación de un modelo de inventario de servicios. A menudo es este modelo el que termina por definir el alcance del tipo de arquitectura requerido, siendo referido como una arquitectura de inventario de servicio.

Desde una perspectiva de diseño de la empresa, el inventario de servicios puede representar un límite concreto para la implementación de la arquitectura estandarizada. Esto significa que debido a que los servicios dentro de un inventario están estandarizados, son las tecnologías y las extensiones proporcionadas por la arquitectura subyacente.

El alcance de un inventario de servicio puede ser de toda la empresa, o puede representar un dominio dentro de la empresa. Por esa razón, a este tipo de arquitectura no se le llama "arquitectura de dominio". Se relaciona con el alcance del inventario, que puede abarcar múltiples dominios.

Es difícil comparar una arquitectura de inventario de servicios con los tipos de arquitecturas tradicionales, porque el concepto de un inventario no ha sido común. El candidato más cercano sería una arquitectura de integración que representa algún segmento importante de una empresa. Sin embargo, esta comparación sólo sería relevante en su alcance, ya que las características del diseño orientado a servicios y los esfuerzos de estandarización relacionados intentan convertir un inventario de servicios en un entorno altamente homogéneo.

2.2.4 Arquitectura Empresarial Orientada a Servicios

[Erl, 2009] Esta forma de arquitectura tecnológica representa prácticamente la totalidad de los servicios, composición de servicios y arquitecturas de inventario de servicios que residen dentro de una empresa de IT específica.

Una arquitectura orientada a servicios empresariales, es comparable con una arquitectura técnica empresarial tradicional sólo cuando la mayoría o todos los entornos técnicos de la empresa están orientados a servicios. De lo contrario, puede ser simplemente una documentación de las partes de la empresa que han adoptado SOA, en cuyo caso existe como un subconjunto de la tecnología arquitectónica empresarial.

En entornos de múltiples inventarios o en entornos en los cuales los esfuerzos de estandarización no fueron un éxito total, una especificación de arquitectura empresarial orientada a servicios documentará todos los puntos de transformación y la disparidad de diseño que también pueda existir.

Además, la arquitectura orientada a servicios empresarial puede establecer estándares y convenciones de diseño que apliquen a la totalidad de empresa que todos los servicios, la composición y las implementaciones de la arquitectura del inventario deben cumplir.

2.3. Amazon Elastic Compute Cloud (EC2)

En esta sección se ven en detalle los siguientes puntos de la plataforma Amazon Elastic Compute Cloud: Descripción (sección 2.3.1), características principales (sección 2.3.2), Amazon CloudWatch (sección 2.3.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.3.4), Amazon EC2 con Microsoft Windows Server y SQL Server (sección 2.3.5), Soporte para Sistemas operativos Linux (sección 2.3.6), soporte para almacenamiento de datos (sección 2.3.7), soporte para colas (sección 2.3.8) y soporte para hipervisores (sección 2.3.9).

2.3.1. Descripción

[Amazon.com Inc., 2013a.] Amazon Elastic Compute Cloud (Amazon EC2) es un servicio web que proporciona capacidad informática con tamaño modificable en la nube. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables y basados en web.

Amazon EC2 reduce el tiempo necesario para obtener y arrancar nuevas instancias de servidor en minutos, lo que permite escalar rápidamente la capacidad, ya sea aumentándola o reduciéndola, según cambien sus necesidades. Amazon EC2 cambia el modelo económico de la informática, al permitir pagar sólo por la capacidad que utiliza realmente.

Amazon EC2 presenta un auténtico entorno informático virtual, que permite utilizar interfaces de servicio web para iniciar instancias con distintos sistemas operativos, cargarlas con su entorno de aplicaciones personalizadas, gestionar sus permisos de acceso a la red y ejecutar su imagen utilizando los sistemas que desee.

2.3.2. Características Principales

En este apartado se describen las principales características de la plataforma Amazon EC2. Las características fueron organizadas de la siguiente manera: Elastic (sección 2.3.2.1), Totalmente Controlado (sección 2.3.2.2), Flexible (sección 2.3.2.3), Diseño pensado para uso con otros Amazon web Services (sección 2.3.2.4), Fiable (sección 2.3.2.5), Seguro (sección 2.3.2.6), Interfases de seguridad (sección 2.3.2.7), Instancias Aisladas (sección 2.3.2.8), Económico (sección 2.3.2.9), Instancias en demanda (sección 2.3.2.10), Instancias reservadas (sección 2.3.2.11) e Instancias puntuales (sección 2.3.2.12).

2.3.2.1. Elastic

[Amazon.com Inc., 2013a.] Amazon EC2 permite aumentar o reducir la capacidad en cuestión de minutos, sin esperar horas ni días. Puede enviar una, cientos o incluso miles de instancias del servidor simultáneamente. Desde luego, como todo esta operación se controla con API de servicio Web, la aplicación se escalará (aumentará o disminuirá su capacidad) dependiendo de sus necesidades.

2.3.2.2. Totalmente Controlado

[Amazon.com Inc., 2013a.] Tendrá control total sobre sus instancias. Tiene acceso de usuario raíz (root) a todas ellas, y puede interactuar con ellas como con cualquier otra máquina. Puede detener su instancia y mantener los datos en su partición de arranque, para reiniciar a continuación la misma instancia a través de las API del servicio web. Las instancias se pueden reiniciar de forma remota mediante las API del servicio web. Asimismo, tiene acceso a la emisión de consola de sus instancias.

2.3.2.3. Flexible

[Amazon.com Inc., 2013a.] Tendrá la posibilidad de elegir entre varios tipos de instancia, sistemas operativos y paquetes de software. Amazon EC2 permite seleccionar una configuración de memoria, CPU, almacenamiento de instancias y el tamaño de la partición de arranque óptimo para

su sistema operativo y su aplicación. Por ejemplo, entre sus opciones de sistemas operativos se incluyen varias distribuciones de Linux y Microsoft Windows Server.

2.3.2.4. Diseño pensado para su uso con otros Amazon Web Services

[Amazon.com Inc., 2013a.] Amazon EC2 trabaja con Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), Amazon SimpleDB y Amazon Simple Queue Service (Amazon SQS) para proporcionar una solución informática completa, procesamiento de consultas y almacenamiento en una gran gama de aplicaciones.

2.3.2.5. Fiable

[Amazon.com Inc., 2013a.] Amazon EC2 ofrece un entorno muy fiable en el que las instancias de sustitución se pueden enviar con rapidez y anticipación. El servicio se ejecuta en los centros de datos y la infraestructura de red acreditados de Amazon. El compromiso del contrato a nivel de servicio de Amazon EC2 es de una disponibilidad del 99,95% en cada Región de Amazon EC2.

2.3.2.6. Seguro

[Amazon.com Inc., 2013a.] Amazon EC2 ofrece diversos mecanismos para proteger los recursos informáticos.

2.3.2.7. Interfaces de seguridad

[Amazon.com Inc., 2013a.] Amazon EC2 incluye interfaces de servicio web para configurar el cortafuegos (firewall) que controla el acceso de red a grupos de instancias, y el acceso entre estos.

2.3.2.8. Instancias Aisladas

[Amazon.com Inc., 2013a.] Al iniciar recursos de Amazon EC2 en Amazon Virtual Private Cloud (Amazon VPC), puede aislar sus instancias informáticas especificando el rango de IP que desea utilizar, así como conectarse a su infraestructura de IT existente mediante la red cifrada IPsec VPN

estándar del sector. También puede optar por lanzar instancias dedicadas en su VPC. Las instancias dedicadas son instancias de Amazon EC2 que se ejecutan en hardware dedicado a un único cliente para ofrecer más aislamiento.

2.3.2.9. Económico

[Amazon.com Inc., 2013a.] Amazon EC2 permite disfrutar de las ventajas financieras de Amazon. Pagará una tarifa muy baja por la capacidad informática que realmente utiliza. Consulte las Opciones de compra de instancias de Amazon EC2 para obtener una descripción más detallada.

2.3.2.10. Instancias en demanda

[Amazon.com Inc., 2013a.] Con On-Demand Instances puede pagar por la capacidad informática por hora, sin compromisos a largo plazo. Esto le liberará de los costes y las complejidades de la planificación, la compra y el mantenimiento del hardware y transformará lo que normalmente son grandes costes fijos en costes variables mucho más pequeños. Gracias a On-Demand Instances también se elimina la necesidad de comprar una "red de seguridad" de capacidad para gestionar picos de tráfico periódicos.

2.3.2.11. Instancias reservadas

[Amazon.com Inc., 2013a.] Las instancias reservadas ofrecen la opción de realizar un pago puntual reducido por cada instancia que desee reservar y recibir a cambio un descuento importante en el cargo de uso por horas de dicha instancia. Existen tres tipos de instancias reservadas (instancias reservadas de utilización ligera, mediana e intensa) que permiten equilibrar el importe del pago anticipado a realizar con su precio por hora efectivo. Está a su disposición el Marketplace de instancias reservadas, que le ofrece la oportunidad de vender instancias reservadas si cambian sus necesidades (p. ej., si desea transferir instancias a una nueva región de AWS, pasar a un tipo de instancia nuevo o vender capacidad para proyectos que finalizan antes de que expire el plazo de su instancia reservada).

2.3.2.12. Instancias puntuales

[Amazon.com Inc., 2013a.] Con las instancias puntuales, los clientes pueden ofertar la capacidad sin utilizar de Amazon EC2 y ejecutar dichas instancias mientras su oferta supere el precio puntual. El precio puntual cambia periódicamente según la oferta y la demanda, y los clientes cuyas ofertas alcancen o superen dicho precio tendrán acceso a las instancias puntuales disponibles. Si es flexible respecto a cuándo ejecutar sus aplicaciones, las instancias puntuales pueden reducir significativamente sus costes de Amazon EC2.

2.3.3. Amazon CloudWatch (autoescalabilidad)

[Amazon.com Inc., 2013h.] Amazon CloudWatch proporciona la supervisión de los recursos de la nube de AWS y de las aplicaciones que los clientes ejecutan en AWS. Los desarrolladores y administradores de sistema la utilizan para recopilar métricas y realizar su seguimiento, obtener conocimientos y reaccionar inmediatamente para que sus aplicaciones y empresas sigan funcionando sin problemas. Amazon CloudWatch supervisa recursos de AWS como las instancias de bases de datos de Amazon EC2 y Amazon RDS, y también puede supervisar métricas personalizadas generadas por las aplicaciones y los servicios de un cliente. Con Amazon CloudWatch, obtendrá una visibilidad para todo el sistema de la utilización de recursos, el rendimiento de las aplicaciones y el estado de funcionamiento.

Amazon CloudWatch le permite supervisar las instancias de Amazon EC2, volúmenes de Amazon EBS, Elastic Load Balancers y las instancias de bases de datos de Amazon RDS en tiempo real. Métricas como la utilización de la CPU, la latencia y los recuentos de solicitudes se proporcionan automáticamente para estos recursos de AWS.

2.3.4. Blueprints / Imágenes para acelerar el aprovisionamiento

[Amazon.com Inc., 2013i.] Amazon denomina “AMIs” a sus Blueprints o imágenes para acelerar y facilitar el aprovisionamiento de instancias en la nube. Una máquina de imagen Amazon (AMI) es un tipo especial de sistema operativo pre-configurado y software de aplicaciones virtualizadas que se utiliza para crear una máquina virtual en el Amazon Elastic Compute Cloud (EC2). Además, es la unidad básica de la implementación de los servicios prestados mediante EC2.

La plataforma Elastic Compute Cloud cuenta con más de 2200 imágenes de máquinas virtuales alternativas, con diferentes sistemas operativos, aplicativos y configuraciones. Una de las

configuraciones más populares cuenta con un sistema operativo “Ubuntu” y software de base “LAMP” (Linux, Apache, MySQL y PHP).

Además, las instancias de AMIs se pueden filtrar por Proveedor (Ej. IBM, Oracle, Amazon Web Services, Sun Microsystems, Novell, Microsoft o Community), por Región (física), por Arquitectura (i386, x86_64), por Root Device Type (Elastic block Store, instance-store) or Platform (Ubuntu, Red Hat, Fedora, Windows, Debian, etc.).

2.3.5. Amazon EC2 con Microsoft Windows Server y SQL Server

[Amazon.com Inc., 2013j.] Amazon EC2 con Microsoft Windows Server® (ediciones 2003 R2, 2008, 2008 R2 y 2012) es un entorno rápido y fiable para implementar aplicaciones con Microsoft Web Platform, incluidos ASP.NET, ASP.NET AJAX, Silverlight™ e Internet Information Server (IIS). Amazon EC2 permite ejecutar cualquier solución compatible basada en Windows en la plataforma informática en la nube de AWS, que ofrece alto rendimiento, fiabilidad y rentabilidad. Entre los casos prácticos de uso habitual con Windows se incluye el alojamiento de aplicaciones empresariales basadas en Windows, alojamiento de sitios web y de servicios web, procesamiento de datos, transcodificación de medios, pruebas distribuidas, alojamiento de aplicaciones en ASP.NET y cualquier otra aplicación que requiera software de Windows. Amazon EC2 también es compatible con las bases de datos SQL Server® Express, SQL Web y SQL Standard y, además, pone estas ofertas a disposición de sus clientes con tarifas por horas.

Amazon EC2 ejecutándose sobre Windows Server proporciona una integración perfecta con funciones existentes en Amazon EC2, como por ejemplo Amazon Elastic Block Store (EBS), Amazon CloudWatch, Elastic Load Balancing y Elastic IP. Las instancias de Windows están disponibles en varias zonas de disponibilidad en todas las regiones.

La capa de uso gratuito de AWS incluye instancias de Amazon EC2 que se ejecutan en Microsoft Windows Server. Los clientes que reúnen los requisitos para incluirse dentro de la capa de uso gratuito de AWS pueden utilizar hasta 750 horas al mes de instancias de t1.micro que se ejecutan en Microsoft Windows Server de manera gratuita.

2.3.6. Soporte para Sistemas operativos Linux

[Amazon.com Inc., 2013k.] La AMI de Amazon Linux es una imagen de Linux mantenida y soportada que ofrece Amazon Web Services para su uso en Amazon Elastic Compute Cloud (Amazon EC2). Está diseñada para proporcionar un entorno de ejecución estable, seguro y de alto rendimiento para aplicaciones que se ejecuten en Amazon EC2. También incluye paquetes que permiten una fácil integración con AWS, incluidas herramientas de configuración de inicio y muchas bibliotecas y herramientas populares de AWS. Amazon Web Services también proporciona actualizaciones continuas de seguridad y mantenimiento para todas las instancias ejecutadas en la AMI de Amazon Linux. La AMI de Amazon Linux se proporciona sin cargo adicional a los usuarios de Amazon EC2.

2.3.7. Soporte para almacenamiento de datos

En este apartado se provee información acerca de las alternativas ofrecidas por la plataforma EC2 para la persistencia de datos. Se analizarán las siguientes alternativas: Amazon Simple Storage Service “S3” (sección 2.3.7.1), Amazon Relational Database Service (sección 2.3.7.2) y Amazon SimpleDB (sección 2.3.7.3).

2.3.7.1. Amazon Simple Storage Service (Amazon S3)

[Amazon.com Inc., 2013c.] Amazon S3 es almacenamiento para Internet. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables basados en la Web.

Amazon S3 proporciona una sencilla interfaz de servicios web que puede utilizarse para almacenar y recuperar la cantidad de datos que desee, cuando desee y desde cualquier parte de la Web. Concede acceso a todos los desarrolladores a la misma infraestructura económica, altamente escalable, fiable, segura y rápida que utiliza Amazon para tener en funcionamiento su propia red internacional de sitios web. Este servicio tiene como fin maximizar las ventajas del escalado y trasladar estas ventajas a los desarrolladores.

2.3.7.2. Amazon Relational Database Service (Amazon RDS) ^(beta)

[Amazon.com Inc., 2013d.] Amazon Relational Database Service (Amazon RDS) es un servicio web que facilita las tareas de configuración, utilización y escalado de una base de datos relacional en la nube. Proporciona capacidad rentable y de tamaño modificable y, al mismo tiempo, gestiona las tediosas tareas de administración de la base de datos, lo que le permite centrarse en sus aplicaciones y en su negocio.

Amazon RDS le permite acceder a todas las funciones de un motor de base de datos MySQL, Oracle o Microsoft SQL Server conocido. Esto supone que el código, las aplicaciones y las herramientas que ya utiliza en la actualidad con sus bases de datos existentes funcionarán a la perfección con Amazon RDS.

2.3.7.3. Amazon SimpleDB ^(beta)

[Amazon.com Inc., 2013e.] Amazon SimpleDB es un almacén de datos no relacionales de alta disponibilidad y flexible que descarga el trabajo de administración de bases de datos. Los desarrolladores simplemente almacenan elementos de datos y los consultan mediante solicitudes de servicios Web; Amazon SimpleDB se encarga del resto.

Sin las limitaciones impuestas por las bases de datos relacionales, Amazon SimpleDB está optimizado para ofrecer alta disponibilidad y flexibilidad con poca o ninguna carga administrativa. La labor de Amazon SimpleDB pasa inadvertida: se encarga de crear y gestionar varias réplicas de sus datos y las distribuye geográficamente para permitir alta disponibilidad y capacidad de duración. El servicio sólo le cobra los recursos realmente consumidos en almacenamiento de los datos y en distribución de las solicitudes. Es posible cambiar el modelo de datos sobre la marcha, y el sistema indexa los datos automáticamente por usted.

2.3.8. Soporte para Colas

[Amazon.com Inc., 2013f.] Amazon Simple Queue Service (Amazon SQS) ofrece un sistema de gestión de colas fiable y ampliable para almacenar mensajes a medida que se transfieren entre sistemas. Mediante Amazon SQS, los desarrolladores pueden transferir datos entre componentes distribuidos de aplicaciones que realizan distintas tareas, sin perder mensajes y sin necesidad de que

cada componente esté siempre disponible. Amazon SQS facilita la tarea de creación de un flujo de trabajo automatizado, trabajando en estrecha conexión con Amazon Elastic Compute Cloud (Amazon EC2) y el resto de los servicios web de la infraestructura de AWS.

Amazon SQS funciona utilizando la infraestructura de gestión de mensajes a escala web de Amazon como un servicio web. Cualquier sistema de Internet puede añadir o leer mensajes sin tener instalado ningún software ni configuración de cortafuegos especial. Los componentes de las aplicaciones que utilizan Amazon SQS se pueden ejecutar independientemente y no es necesario que estén en la misma red, que se hayan desarrollado con las mismas tecnologías ni que se ejecuten a la vez.

2.3.9. Alternativas de Hipervisor

[Amazon.com Inc., 2013g.] Amazon EC2 actualmente utiliza una versión altamente personalizada del hipervisor Xen, aprovechando la paravirtualización. Como los huéspedes ("Guest") paravirtualizados se basan en el hipervisor para proporcionar apoyo a las operaciones que normalmente requieren un acceso privilegiado, es posible ejecutar el sistema operativo invitado sin acceso elevado a la CPU.

2.4. WINDOWS AZURE

En esta sección se ven en detalle los siguientes puntos de la plataforma Windows Azure: Descripción (sección 2.4.1), características principales (sección 2.4.2), autoescalabilidad con Autoscaling Application Block (sección 2.4.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.4.4), Soporte para Sistemas operativos Windows (sección 2.4.5), Soporte para Sistemas operativos Linux (sección 2.4.6), soporte para almacenamiento de datos (sección 2.4.7), soporte para colas (sección 2.4.8) y soporte para hipervisores (sección 2.4.9).

2.4.1. Descripción

[Amazon.com Inc., 2013a.] Windows Azure es una plataforma de nube abierta y flexible que permite compilar, implementar y administrar aplicaciones rápidamente en una red global de centros de datos administrados por Microsoft. Puede compilar aplicaciones en cualquier lenguaje, herramienta o marco, permitiendo además integrar sus aplicaciones de nube públicas con el entorno de TI existente.

2.4.2. Características Principales

En esta sección se definen las principales características de la plataforma Windows Azure, organizándolas de la siguiente manera: Siempre disponible (sección 2.4.2.1), Alto nivel de servicio (sección 2.4.2.2), Abierto (sección 2.4.2.3), Servidores ilimitados, almacenamiento ilimitado (sección 2.4.2.4), Gran capacidad (sección 2.4.2.5) y Cache distribuida y CDN (sección 2.4.2.6),

2.4.2.1. Siempre Disponible

[Microsoft Corp., 2013a.] Disponibilidad 24x7x365

2.4.2.2. Alto nivel de servicio

[Microsoft Corp., 2013a.] Windows Azure entrega un Contrato de nivel de servicio mensual del 99,95 % que permite compilar y ejecutar aplicaciones de alta disponibilidad sin importar la infraestructura. Proporciona revisiones automáticas del Sistema Operativo y de los servicios, equilibrio de carga de red integrado y resistencia ante errores de hardware. Admite un modelo de implementación con el que se puede actualizar una aplicación sin inactividad (downtime).

2.4.2.3. Abierto

[Microsoft Corp., 2013a.] Windows Azure permite utilizar cualquier lenguaje, marco o herramienta para crear aplicaciones. Las características y los servicios se exponen utilizando protocolos REST abiertos. Las bibliotecas de cliente de Windows Azure están disponibles para varios lenguajes de programación, se comercializan bajo una licencia de código abierto y se hospedan en GitHub.

2.4.2.4. Servidores ilimitados, almacenamiento ilimitado

[Microsoft Corp., 2013a.] Windows Azure permite escalar aplicaciones a cualquier tamaño con facilidad. Es una plataforma de autoservicio totalmente automatizada que permite el aprovisionamiento de recursos en cuestión de minutos. El uso de recursos aumenta o disminuye de manera flexible en función de las necesidades. Solo se pagan los recursos que usa la aplicación. Windows Azure está disponible en varios centros de datos del mundo, lo que permite implementar las aplicaciones cerca de los clientes.

2.4.2.5. Gran capacidad

[Microsoft Corp., 2013a.] Windows Azure proporciona una plataforma flexible en la nube que puede satisfacer los requisitos de cualquier aplicación. Permite hospedar y ampliar el código de aplicación dentro de roles de proceso de un modo totalmente confiable. Los datos se pueden almacenar en bases de datos SQL relacionales, almacenes de tablas NoSQL y almacenes de blobs no estructurados, y existe la opción de usar la funcionalidad de Hadoop e inteligencia empresarial para la minería de datos. Puede aprovechar la sólida funcionalidad de mensajería de Windows Azure para habilitar aplicaciones distribuidas escalables, así como para entregar soluciones híbridas que se ejecuten en la nube y en un entorno empresarial local.

2.4.2.6. Cache distribuida y CDN

[Microsoft Corp., 2013a.] Los servicios de caché distribuida y red de entrega de contenido (CDN) de Windows Azure permiten reducir la latencia y ofrecer aplicaciones con un gran rendimiento en cualquier lugar del mundo.

2.4.3. Autoescalabilidad

[Microsoft Corp., 2013b.] El "Bloque de Aplicación Autoescalable" puede escalar automáticamente su aplicación de Windows Azure basado en reglas definidas específicamente para su aplicación. Puede utilizar estas reglas para ayudar a la aplicación de Windows Azure a mantener su rendimiento en respuesta a los cambios en la carga de trabajo, y al mismo tiempo controlar los costos asociados con el alojamiento de su aplicación en Windows Azure. Junto con la escalabilidad, aumentando o disminuyendo el número de instancias de rol (working role) en su aplicación, el bloque también permite utilizar otras medidas de escalabilidad tales como funcionalidades determinadas de estrangulamiento ("throttling") dentro de la aplicación o el uso de las acciones definidas por el usuario.

Brinda además la posibilidad de optar por implementar el bloque en un rol de Windows Azure o en una aplicación interna.

El Bloque de Aplicación Autoescalable forma parte del Pack de la Enterprise Library 5.0 de Microsoft Integration para Windows Azure. El bloque utiliza dos tipos de reglas para definir el comportamiento automático de escalabilidad para su aplicación:

Reglas de restricciones: Para establecer los límites superior e inferior en el número de instancias, por ejemplo, digamos que 8:00-10:00 todos los días quiere un mínimo de cuatro y un máximo de seis instancias, se utiliza una regla de restricción.

Reglas Reactivas: Para permitir que el número de instancias de rol puedan cambiar en respuesta a los cambios impredecibles en la demanda, se utilizan reglas reactivas.

2.4.4. Blueprints / Imágenes para acelerar el aprovisionamiento

[Microsoft Corp., 2013d.] Las máquinas virtuales entregadas a demanda ofrecen una infraestructura de computación escalable cuando se necesita aprovisionar rápidamente recursos para satisfacer las necesidades de un negocio en crecimiento. Es posible obtener máquinas virtuales de los sistemas operativos Linux y Windows Server en múltiples configuraciones.

Los blueprints de Windows Azure ofrecen la posibilidad de desbloquear la cartera de TI y la infraestructura de suministro al ritmo que su negocio requiere. Para ello, simplemente hay que elegir la configuración deseada (instancias de memoria estándar o alta) y seleccionar una imagen de la galería de imágenes de máquinas virtuales.

Las Máquinas virtuales de Windows Azure ofrecen a los sistemas y aplicaciones la posibilidad de mover los discos duros virtuales (VHD) desde las instalaciones locales a la nube (y viceversa).

2.4.5. Soporte para Sistemas operativos Microsoft Windows

Krishnan [2010]. Si usted se está preguntando si el código va a ser diferente, ya que se ejecuta en la nube, o si vas a tener que aprender un nuevo marco (traducido del término en inglés "framework"), la respuesta es "no". Continuará escribiendo código que se ejecuta en Windows Server. El marco de .NET (3.5 SP1) se instala por defecto en todas las máquinas, y el código ASP.NET típico funcionará. Es también posible utilizar el soporte FastCGI para correr cualquier marco que soporte FastCGI (como PHP, Ruby on Rails, Python, etc). Si dispone de código nativo o binario, también puede ejecutarlo.

Krishnan [2010] Cada hipervisor administra varios sistemas operativos virtuales. Todos ellos corren el sistema operativo Windows Server compatible con 2008. En realidad, no se nota ninguna diferencia entre ejecutar un Windows Server 2008 y estas máquinas. Las únicas diferencias son algunas optimizaciones específicas para el hipervisor en el cual están corriendo.

Krishnan [2010] ¿Bajo qué sistema operativo estará corriendo su código en Windows Azure?

Incluso con toda esta discusión sobre el funcionamiento del sistema y las optimizaciones de integración con el hipervisor, la mayoría de los desarrolladores sólo se preocupan por el entorno en el cual correrán sus aplicaciones (el sistema operativo). La respuesta es simple: Las aplicaciones corren en Windows Server 2008 x64 Enterprise Edition. Bueno, casi. Microsoft lo llama "sistema operativo 2008 compatible con Windows Server", que se refiere al hecho de que se trata de Windows Server en casi todos los aspectos, excepto en algunos cambios de bajo nivel para optimizar el hipervisor. Sin embargo, las aplicaciones están abstraídas varias capas lejos de estos cambios, y no deben notar nada distinto respecto de su ejecución en una máquina Windows Server normal.

2.4.6. Soporte para Sistemas operativos Linux

[Microsoft Corp., 2013i.] Crear una máquina virtual que ejecute el sistema operativo Linux en Windows Azure es fácil por medio de la Galería de imágenes (blueprints) utilizando el Portal de administración. Es también posible acceder a las instancias de estas máquinas virtuales Linux para personalizarlas a gusto, por medio de un usuario con privilegios de administrador ("Root").

Es también factible implementar En Azure máquinas virtuales ya disponibles que corren sistemas operativos Linux, por ejemplo con instancias de máquinas virtuales vmWare. Para esto, solo es necesario convertir la imagen de la máquina virtual Linux al formato de Windows Azure (de .vmx a .vmdk), para luego subirla a nuestra cuenta Azure por medio del administrador de imágenes personalizadas de Windows Azure.

Algunas de las distribuciones del Sistema operativo Linux que soporta Windows Azure, inclusive proporcionando blueprints para acelerar el aprovisionamiento de las máquinas virtuales son las siguientes:

- openSUSE 12.3
- SUSE Linux Enterprise Server 11 Service Pack 2
- Ubuntu Server 12.04 LTS
- Ubuntu Server 12.10
- Ubuntu Server 13.04
- OpenLogic CentOS 6.3
- Ubuntu Server 12.10 DAILY

2.4.7. Soporte para almacenamiento de datos

En este apartado se provee información acerca de las alternativas ofrecidas por la plataforma Windows Azure para la persistencia de datos. Se analizarán las siguientes alternativas: SQL Server en máquinas virtuales de Windows Azure (sección 2.4.7.1), Base de datos SQL (sección 2.4.7.2), Almacenes de tablas NoSQL (sección 2.4.7.3), Blob no estructurado (sección 2.4.7.4).

2.4.7.1. SQL Server en máquinas virtuales de Windows Azure

[Microsoft Corp., 2013f.] Cuando las aplicaciones requieren funcionalidad completa de SQL Server, Máquinas virtuales es la solución ideal. Encontrará ofertas de imágenes de SQL Server 2012 y SQL Server 2008 R2, en sus ediciones Standard, Web y Enterprise. Si tiene una licencia de SQL Server con Software Assurance, como ventaja adicional puede trasladar la licencia existente a Windows Azure y pagar solo por proceso y almacenamiento. La ejecución de SQL Server en Máquinas virtuales es una solución adecuada en los escenarios siguientes:

- Para desarrollar y probar nuevas aplicaciones de SQL Server rápidamente No es necesario esperar semanas para el aprovisionamiento local de hardware, sino que basta con captar la imagen de SQL Server correcta en la galería de imágenes. Puede optar por realizar la implementación en un entorno de producción o local con poco esfuerzo.
- Para hospedar aplicaciones de SQL Server de nivel 2 y nivel 3 existentes Gracias a los distintos tamaños de VM entre los que elegir y dada la compatibilidad total con SQL Server, es posible trasladar las aplicaciones de SQL Server locales existentes y disfrutar de la eficacia de la computación en la nube.
- Para realizar copias de seguridad y restauraciones de bases de datos locales Es posible realizar la copia de seguridad de una base de datos local en un almacenamiento en blobs de Windows Azure y poder así restaurar la base de datos en una máquina virtual de Windows Azure en caso de que sea necesaria la recuperación ante desastres en el entorno local.
- Para ampliar aplicaciones locales Cree aplicaciones híbridas que utilicen activos locales y máquinas virtuales de Windows Azure para disfrutar de una mayor eficacia y alcance global.
- Para crear aplicaciones de varias capas en la nube Cree una aplicación de varias capas que utilice la funcionalidad de escalado única del servicio Base de datos SQL para la capa de aplicación y que aproveche la compatibilidad completa de SQL Server en Máquinas virtuales de Windows Azure para la capa de base de datos.

2.4.7.2. Base de datos SQL

[Microsoft Corp., 2013f.] Para aquellas aplicaciones que necesitan una base de datos relacional completamente funcional como servicio, Windows Azure ofrece la base de datos SQL, antes denominada Base de datos de SQL Azure. La base de datos SQL ofrece un alto nivel de interoperabilidad, lo que permite a los clientes crear aplicaciones en la mayoría de los principales marcos de desarrollo. Además, la base de datos SQL, basada en las tecnologías probadas de SQL Server, permite utilizar los conocimientos y la experiencia existente para reducir el tiempo de solución, así como crear o ampliar aplicaciones entre los sistemas locales y la nube.

Use la base de datos SQL para:

2.4.7.3. Tablas

[Microsoft Corp., 2013f.] Las tablas ofrecen funcionalidad NoSQL para las aplicaciones que requieren el almacenamiento de grandes cantidades de datos no estructurados. Las tablas son un servicio administrado con certificación ISO 27001 que se pueden escalar automáticamente para satisfacer un rendimiento y volumen masivos de hasta 100 terabytes, accesibles prácticamente desde cualquier lugar a través de REST y las API administradas.

2.4.7.4. Blob no estructurado

[Microsoft Corp., 2013f.] Los blobs son el modo más sencillo de almacenar grandes cantidades de texto no estructurado o datos binarios tales como vídeo, audio e imágenes. Los blobs son un servicio administrado con certificación ISO 27001 que se pueden escalar automáticamente para satisfacer un rendimiento y volumen masivos de hasta 100 terabytes, accesibles prácticamente desde cualquier lugar a través de REST y las API administradas.

2.4.8. Soporte para colas

[Microsoft Corp., 2013g.] El bus de servicio de Colas soporta un modelo de comunicación de mensajería negociado. Cuando se utilizan colas, los componentes de una aplicación distribuida no se comunican directamente entre sí, en cambio, intercambian mensajes via una cola, lo cual actúa como un intermediario. Un productor de mensajes (remitente) envía un mensaje a la cola y continua su procesamiento. Asíncronamente, un consumidor de mensajes (receptor) obtiene el mensaje de la cola y lo procesa. El productor no tiene que esperar una respuesta por parte del consumidor para poder continuar el proceso y enviar más mensajes. Las colas ofrecen la implementación de la técnica "primero que entra, primero que sale" (modelo conocido por sus siglas en inglés como "FIFO") enviando mensajes a uno o más consumidores que compiten por el tratamiento del mensaje. Es decir, los mensajes suelen ser recibidos y procesados por los receptores en el orden en que fueron añadidos a la cola, y cada mensaje es recibido y procesado por un único consumidor.

2.4.9. Alternativas de Hipervisor

En este apartado se tratan las alternativas de Hipervisor provistas por la plataforma Windows Azure. Se mencionarán asimismo las principales características del hipervisor provisto por la plataforma, organizándolas de la siguiente manera: Rápido (sección 2.4.9.1), pequeño (sección 2.4.9.2) y Estrechamente integrado con el núcleo (sección 2.4.9.3).

Krishnan [2010]. En 2006/2007, un equipo dirigido por Dave Cutler (el padre de Windows NT) comenzó a trabajar en un nuevo hipervisor pensado para ser optimizado para el centro de datos. Este hipervisor se basó en los siguientes tres principios:

2.4.9.1. Rápido

El hipervisor de Windows Azure ha sido diseñado para ser lo más eficiente posible. Gran parte de esto se consigue a través de optimizaciones de bajo nivel realizadas a la vieja usanza, tales como empujar cargas de trabajo al hardware siempre que sea posible. Puesto que Windows Azure controla el hardware en sus centros de datos, puede confiar en la presencia de características de hardware, a diferencia de hipervisores genéricos diseñados para un mercado más amplio.

2.4.9.2. Pequeño

El hipervisor es construido para ser claro y directo, y no incluye aquellas características que no están directamente relacionadas con la nube. Menor cantidad de código no sólo significa un mejor rendimiento, sino que también significa menos código para corregir o actualizar.

2.4.9.1. Estrechamente integrado con el núcleo

En Windows Azure, el kernel del sistema operativo que se ejecuta en el hipervisor está altamente optimizado para el hipervisor.

Con Windows Server 2008, Microsoft lanzó un hipervisor llamado Hyper-V. A menudo hay confusión sobre las diferencias entre Hyper-V y el hipervisor de Windows Azure, y algunos libros / artículos a menudo asumen que se trata del mismo hipervisor.

En realidad, ambos son diferentes y están contruidos también con diferentes propósitos. Hyper-V es entregado como parte de Windows, y está destinado para funcionar en una amplia variedad de hardware para una amplia variedad de propósitos. El hipervisor de Windows Azure se ejecuta sólo en los centros de datos de Microsoft, y se ha optimizado específicamente para el hardware que se ejecuta en Windows Azure.

Como era de esperar con dos productos similares de la misma empresa, no hay intercambio de código y diseño. En el futuro, las nuevas características del hipervisor de Windows Azure tendrán influencias en Hyper-V, y viceversa.

2.5. GOOGLE APP ENGINE

En esta sección se ven en detalle los siguientes puntos de la plataforma Google App Engine: Descripción (sección 2.5.1), características principales (sección 2.5.2), autoescalabilidad (sección 2.5.3), Soporte para Sistemas operativos Linux (sección 2.5.4), soporte para almacenamiento de datos (sección 2.5.5), soporte para colas (sección 2.5.6) y soporte para hipervisores (sección 2.5.7). Este apartado fue construido con información obtenida del sitio oficial de Google App Engine.

2.5.1. Descripción

[Google Inc., 2012a] Google App Engine permite crear y alojar aplicaciones web en los mismos sistemas escalables con los que funcionan las aplicaciones de Google. Google App Engine ofrece procesos de desarrollo y de implementación rápidos, y una administración sencilla, sin necesidad de preocuparse por el hardware, las revisiones o las copias de seguridad y una ampliación sin esfuerzos.

Las aplicaciones Google App Engine son fáciles de crear, fáciles de mantener y fáciles de escalar a medida que el tráfico y las necesidades de almacenamiento de datos crecen. Con App Engine no es necesario mantener ningún servidor. Basta con cargar su aplicación y esta ya se encontrará lista para servir a los usuarios.

2.5.2. Características Principales

En esta sección se describen las principales características de la plataforma Google App Engine. La información presentada en este apartado fue obtenida del sitio oficial de Google App Engine. Las características que se explicarán son las siguientes: Rápido para iniciar (sección 2.5.2.1), Fácil de usar (sección 2.5.2.2), Conjunto rico de APIs (sección 2.5.2.3), Escalabilidad Inmediata (sección 2.5.2.4), Modelo de negocio de pago por uso (sección 2.5.2.5) e Infraestructura probada (sección 2.5.2.6).

[Google Inc., 2012b]

2.5.2.1. Rápido para iniciar

Sin necesidad de comprar ni mantener ningún software o hardware, puede prototipar y desplegar aplicaciones disponibles para los usuarios en cuestión de horas.

2.5.2.2. Fácil de usar

Google App Engine incluye las herramientas necesarias para crear, probar, ejecutar y actualizar sus aplicaciones.

2.5.2.3. Conjunto rico de APIs

Google App Engine provee un conjunto rico de APIs de servicios de fácil uso, permitiendo crear servicios rápidamente.

2.5.2.3. Escalabilidad Inmediata

No hay prácticamente ningún límite respecto de qué tan alto o qué tan rápido su aplicación pueda escalar.

2.5.2.4. Modelo de negocio de pago por uso

Pague por lo que usa, comenzando sin ningún costo inicial. Pagará sólo por los recursos que su aplicación utilice a medida que crece.

2.5.2.5. Infraestructura probada

Google App Engine utiliza la misma tecnología probada que alberga otras aplicaciones de Google, ofreciendo automáticamente escalabilidad sin problemas. Preocuparse por las configuraciones de servidor y balanceo de carga se convierte en una cosa del pasado, ya que los expertos de Google manejan la supervisión del sistema.

2.5.3. Autoescalabilidad

Google App Engine está diseñado para alojar aplicaciones con muchos usuarios simultáneos. Cuando una aplicación puede servir a muchos usuarios simultáneos sin degradar su rendimiento, se dice que es escalable. Las aplicaciones escritas para App Engine escalan automáticamente. A medida que más personas utilizan la aplicación, App Engine asigna más recursos para la aplicación y administra el uso de esos recursos. La aplicación en sí no necesita saber nada acerca de los recursos que utiliza.

2.5.4. Soporte para Sistemas operativos Linux

[Google Inc., 2012c] Las aplicaciones se ejecutan en un entorno seguro que proporciona un acceso limitado al sistema operativo subyacente. Estas limitaciones permiten que App Engine distribuya peticiones web para la aplicación en varios servidores, así como también iniciar y detener los servidores para satisfacer las demandas de tráfico. La caja de arena (traducción literal del término en inglés "Sandbox") aísla la aplicación en su propio entorno seguro y confiable que es independiente del hardware, sistema operativo y la ubicación física del servidor web. Ejemplos de las limitaciones de acceso al sistema operativo incluyen:

- Una aplicación sólo puede acceder a otros ordenadores situados en internet a través de la URL proporcionada y/o por medio de servicios de correo electrónico. Otros equipos sólo se pueden conectar a la aplicación realizando peticiones HTTP (o HTTPS) en los puertos estándar.
- Las aplicaciones no pueden escribir en el sistema de archivos en ninguno de los entornos de ejecución. Una aplicación puede leer archivos, pero sólo los archivos subidos con el código de la aplicación. La aplicación debe utilizar el almacén de datos de App Engine, memcache u otros servicios para aquellos datos que se persisten entre las peticiones.
- El código de aplicación sólo se ejecuta en respuesta a una petición de web, una tarea en cola, o una tarea programada, y debe devolver datos de respuesta dentro de 60 segundos en cualquier caso. Un controlador de solicitudes no se puede generar un sub-proceso o ejecutar código después de que la respuesta haya sido enviada.

2.5.5. Soporte para almacenamiento de datos

[Google Inc., 2012d] El almacenamiento de datos de una aplicación web escalable puede ser complicado. Un usuario puede interactuar con cualquiera de las docenas de servidores web en un momento dado, y la siguiente petición del usuario podría ir a un servidor web diferente a la anterior solicitud. Todos los servidores web deben interactuar con los datos, que también se extienden a través de docenas de máquinas, posiblemente en diferentes lugares del mundo.

Con Google App Engine, no es necesario preocuparse por nada de eso. La infraestructura de App Engine se encarga de toda la distribución, la replicación y el equilibrio de carga de los datos detrás de una sencilla API-y se obtiene un motor de búsqueda de gran alcance, garantizando también las transacciones.

El repositorio de datos de App Engine y el almacén de datos de alta replicación (HRD) utilizan el algoritmo Paxos para replicar datos a través de múltiples centros de datos. Los datos se escriben en el almacén de datos en objetos conocidos como entidades. Cada entidad tiene una clave que identifica de manera única. Una entidad puede designar opcionalmente otra entidad como su matriz, la primera entidad es un hijo de la entidad matriz. Las entidades en el almacén de datos forman así un espacio de estructura jerárquica similar a la estructura de directorios de un sistema de archivos.

El almacén de datos es extremadamente resistente frente a fallos catastróficos, sin embargo, garantizar la consistencia puede ser bastante diferente de lo que usted está familiarizado. Las entidades que descienden de un antepasado común se dice que pertenecen a un mismo grupo de

entidades, las llaves de su ancestro común es la clave principal del grupo, que sirve para identificar a todo el grupo. Las consultas en una única entidad de su grupo, llamado consultas de antepasados, hacen referencia a la clave principal en lugar de la clave de una entidad específica. Los grupos de entidad son una unidad de consistencia y transaccionalidad: mientras que las consultas sobre múltiples grupos de entidades pueden devolver resultados viciados, eventualmente consistentes, aquellas limitadas a un único grupo de entidad siempre retornan actualizadas, produciendo resultados muy consistentes.

2.5.6. Soporte para colas

[Google Inc., 2012c] Una aplicación puede realizar tareas además de responder a solicitudes web. Las aplicaciones implementadas en Google App Engine pueden ejecutar estas tareas siguiendo la programación que se configure, por ejemplo, cada día o cada hora. Asimismo, es posible ejecutar tareas que ella misma ha añadido a una cola, como una tarea en segundo plano creada durante el procesamiento de una solicitud.

Las tareas programadas también se conocen como "tareas cron", administradas por el servicio cron. Las colas de tareas se incluyen actualmente como una función experimental. En este momento, solo el entorno de tiempo de ejecución Python puede utilizar colas de tareas. Se incluirá una interfaz de cola de tareas para aplicaciones Java en poco tiempo.

Con el API de la cola de tareas, las aplicaciones pueden realizar fuera de solicitudes de usuario trabajos que se han iniciado dentro de ellas. Si una aplicación necesita ejecutar algún trabajo en segundo plano, puede utilizar el API de la cola de tareas para organizarlo en pequeñas unidades discretas llamadas tareas. A continuación, la aplicación inserta estas tareas en una o más colas. App Engine detecta automáticamente nuevas tareas y las ejecuta cuando los recursos del sistema lo permiten.

2.5.7. Alternativas de Hipervisor

Google App Engine brinda muy escasa información acerca del hipervisor que utiliza, ya que no es posible cambiarlo o utilizar otro, dado que el servicio que brinda App Engine es PaaS (plataforma como servicio, por sus siglas en inglés "Platform as a Service"), motivo por el cual no es posible administrar la infraestructura, sino que esta se encuentra subyacente y transparente para el usuario de la plataforma.

2.6. RED HAT OPENSIFT

En esta sección se ven en detalle los siguientes puntos de la plataforma Red Hat Open Shift: Descripción (sección 2.5.1), características principales (sección 2.5.2), autoescalabilidad (sección 2.5.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.5.4), Soporte para Sistemas operativos Windows (sección 2.5.5), Soporte para Sistemas operativos Linux (sección 2.5.6), soporte para almacenamiento de datos (sección 2.5.7), soporte para colas (sección 2.5.8) y soporte para hipervisores (sección 2.5.9). Este apartado fue construido con información obtenida del sitio oficial de Google App Engine.

2.6.1. Descripción

[Red Hat Inc., 2013a.] OpenShift es la oferta de plataforma como servicio para Computación en la nube de Red Hat.

En esta plataforma los desarrolladores de aplicaciones pueden construir, desplegar, probar y correr sus aplicaciones. Proporciona espacio en disco, recursos de CPU, memoria, conectividad de red y un servidor Apache o JBoss. Dependiendo del tipo de aplicación que se está construyendo, también proporciona acceso a una plantilla de sistema de archivos para esos tipos (por ejemplo PHP, Python y Ruby/Rails).

También proporciona herramientas de desarrollo integradas para apoyar el ciclo de vida de las aplicaciones, incluyendo la integración de Eclipse, JBoss Developer Studio, Jenkins, Maven y GIT. OpenShift utiliza un ecosistema de código abierto para proporcionar servicios clave de la plataforma de aplicaciones móviles (Appcelerator), servicios NoSQL (MongoDB), servicios de SQL (PostgreSQL, MySQL), y más. JBoss proporciona una plataforma de middleware empresarial para aplicaciones Java, proporcionando apoyo para Java EE6 y servicios integrados tales como transacciones y mensajes, que son fundamentales para las aplicaciones empresariales.

2.6.2. Características Principales

En este apartado se presentan las características principales de OpenShift. Se organizan de la siguiente manera: Prestación de servicios de aplicación acelerada (sección 2.6.2.1), Dependencia minimizada con el proveedor (sección 2.6.2.2), pilas de aplicaciones de autoservicio y en demanda (sección 2.6.2.3), flujos de trabajo estandarizados para desarrolladores (sección 2.6.2.4), Políglota:

Elección de los lenguajes de programación y los softwares de base (sección 2.6.2.5), Aplicaciones empresariales con Java EE6 (sección 2.6.2.6), Servicios de base de datos incorporados (sección 2.6.2.7), Sistema de cartucho Extensible para agregar servicios (sección 2.6.2.8), soporte para múltiples entornos (sección 2.6.2.9), Dependencia y Gestión de la Construcción (sección 2.6.2.10), Integración Continua y Gestión de Versiones (sección 2.6.2.11), Gestión de versiones de código fuente (sección 2.6.2.12), Inicio de sesión Remoto por SSH al contenedor de aplicaciones (sección 2.6.2.13), Integración con IDE (sección 2.6.2.14), Línea de comandos enriquecida (sección 2.6.2.15), Desarrollo de aplicaciones móviles (sección 2.6.2.16), Redundancia de componentes del sistema para alta disponibilidad (sección 2.6.2.17), Aprovisionamiento automático de la pila de aplicaciones (sección 2.6.2.18), Escalabilidad automática de aplicación (sección 2.6.2.19) y Elección de la infraestructura de nube (sección 2.6.2.20).

2.6.2.1. Prestación de servicios de aplicación acelerada

[Red Hat Inc., 2013b.] La plataforma OpenShift mejora la productividad y la agilidad de los desarrolladores de aplicaciones vinculada con el retraso relacionado con los servidores, sistemas operativos, middleware y aprovisionamiento mediante aplicaciones de auto-servicio y a la carta. Este aumento de la productividad, junto con una normalización del desarrollo del ciclo de vida de aplicación de flujo de trabajo, permitirá la aceleración de la entrega de servicios de aplicaciones. Esto aumenta eficazmente la velocidad de las TI.

2.6.2.2. Dependencia minimizada con el proveedor

[Red Hat Inc., 2013b.] Al ser construida sobre una pila de tecnologías de código abierto, la plataforma OpenShift está diseñada para proporcionar libertad de elección, incluyendo la libertad de elegir alejarse de las PaaS. Para lograr esto, dentro de la plataforma OpenShift solo se utilizan lenguajes de código abierto y frameworks sin alteraciones. No se utilizan APIs, tecnologías o recursos privados. Esto asegura la portabilidad de aplicaciones de la plataforma OpenShift, evitando así las dependencias con los proveedores de tecnología utilizada por la plataforma PaaS OpenShift.

2.6.2.3. Pilas de aplicaciones de autoservicio y en demanda

[Red Hat Inc., 2013b.] Al permitir a los desarrolladores desplegar rápida y fácilmente las pilas de aplicaciones, la plataforma OpenShift puede aumentar la productividad y fomentar la innovación en

el diseño y la entrega de aplicaciones. Las nuevas ideas pueden ser prototipos rápidamente y proyectos de misión crítica pueden ser llevados al mercado más rápidamente.

2.6.2.4. Flujos de trabajo estandarizados para Desarrolladores

[Red Hat Inc., 2013b.] OpenShift como plataforma de aplicaciones cloud permite que la organización de desarrollo de aplicaciones pueda normalizar el flujo de trabajo del programador, y crear procesos repetibles para la entrega de aplicaciones para agilizar todo el proceso.

2.6.2.5. Políglota: Elección de los lenguajes de programación y los softwares de base

[Red Hat Inc., 2013b.] La capacidad de los desarrolladores para elegir entre Java, Ruby, Node.JS, Python, PHP y Perl en la plataforma OpenShift les permite elegir la herramienta adecuada para el trabajo, y hacer una elección diferente para cada proyecto, según sea necesario. Además de estos y otros lenguajes de código abierto, muchos de los marcos de código abierto populares se incluyen también dentro de OpenShift. Algunos ejemplos incluyen Rails, Django, EE6, Spring, Play, Sinatra y Zend.

2.6.2.6. Aplicaciones empresariales con Java EE6

[Red Hat Inc., 2013b.] Ser capaz de desplegar aplicaciones Java EE6 en JBoss EAP funcionando en OpenShift permite que muchos departamentos de TI que se han estandarizado con Java EE puedan migrar sus aplicaciones legadas a la nube sin necesidad de aplicar reingeniería al código o a la arquitectura.

2.6.2.7. Servicios de base de datos incorporados

[Red Hat Inc., 2013b.] Con las opciones de tecnología de bases de datos disponibles en OpenShift y la capacidad de tener estas instancias de bases de datos conectadas automáticamente a las pilas de aplicación según sea necesario, los desarrolladores y arquitectos de la empresa pueden elegir entre los almacenes de datos relacionales clásicos y los modernos NoSQL.

2.6.2.8. Sistema de cartucho Extensible para agregar servicios

[Red Hat Inc., 2013b.] Además de la función de idiomas y servicios, los desarrolladores pueden añadir otro idioma, base de datos o componentes de middleware que necesitan a través del sistema de cartuchos OpenShift personalizable. Esta extensibilidad Cartucho permite a los desarrolladores (y operaciones) extender las PaaS para soportar los estándares o requisitos específicos de la empresa.

2.6.2.9. Soporte para múltiples entornos (desarrollo / Pruebas / Producción)

[Red Hat Inc., 2013b.] Con la capacidad de la plataforma OpenShift para soportar múltiples ambientes del ciclo de vida de desarrollo de las aplicaciones (como Desarrollo, Calidad, Pre-Producción y Producción), las empresas pueden adoptar e implementar la plataforma PaaS (Platform As A Service) OpenShift sin necesidad de cambiar sus metodologías o procesos actuales.

2.6.2.10. Dependencia y Gestión de la Construcción

[Red Hat Inc., 2013b.] La plataforma OpenShift incluye Dependencia y gestión de la Construcción para muchos de los lenguajes de programación más populares, incluyendo Bundler para Ruby, NPM para Node.JS y Maven para Java. Estas herramientas automatizan el proceso de identificación de las dependencias en el código fuente, y la construcción de la aplicación completa. Estas características incrementan la productividad y reducen la posibilidad de error, volviéndose críticas en una plataforma de aplicaciones de nube como PaaS.

2.6.2.11. Integración Continua y Gestión de Versiones

[Red Hat Inc., 2013b.] La plataforma OpenShift incluye Jenkins para integración continua y gestión de lanzamiento. Jenkins puede realizar pruebas cuando se sube código al repositorio, organizar el proceso de construcción, y promover o cancelar automáticamente una versión de la aplicación dependiendo de los resultados de las pruebas. Esta gestión automatizada de los releases se convierte en una parte fundamental para simplificar el desarrollo de aplicaciones.

2.6.2.12. Gestión de versiones de código fuente

[Red Hat Inc., 2013b.] La plataforma OpenShift incluye el control distribuido de versiones Git y un sistema de gestión de código fuente. El protocolo Git asegurado con SSH es utilizado por los desarrolladores para comprobar el código en el repositorio Git seguro que reside dentro de su contenedor de aplicaciones con OpenShift. Git permite tanto la gestión rápida, segura y controlada de origen de la aplicación de control de versiones de código.

2.6.2.13. Inicio de sesión Remoto por SSH al contenedor de aplicaciones

[Red Hat Inc., 2013b.] La arquitectura única basada en SELinux de la plataforma OpenShift permite a los usuarios (desarrolladores u operadores) que inicien sesión remotamente en contenedores de aplicaciones individuales para las aplicaciones implementadas en las PaaS. El usuario registrado podrá ver solamente sus procesos, sistema de archivos y archivos de registro. Esto le da a los usuarios el acceso que necesitan para una mejor arquitectura y administración de sus aplicaciones.

2.6.2.14. Integración con IDE

[Red Hat Inc., 2013b.] Con integración incorporada con Eclipse de la plataforma de OpenShift, JBoss Developer Studio y Titanium Studio, muchos desarrolladores pueden permanecer completamente dentro del IDE con el cual se sientan cómodos a la hora de trabajar con OpenShift.

2.6.2.15. Línea de comandos enriquecida

[Red Hat Inc., 2013b.] Para desarrolladores que prefieren trabajar desde la línea de comandos, la plataforma OpenShift incluye un amplio conjunto de herramientas de línea de comandos que

proporcionan acceso completo a la interfaz del programador de las PaaS. Estas herramientas son fáciles de usar, así como secuencias de comandos para las interacciones automatizadas.

2.6.2.16. Desarrollo de aplicaciones móviles

[Red Hat Inc., 2013b.] A través de una asociación con Appcelerator, la plataforma OpenShift incluye una estrecha integración con Titanium Studio IDE móvil que permite el desarrollo de aplicaciones móviles en la nube con respaldo para Android o iOS que pueden ser ejecutados por aplicaciones de servidor que se ejecutan en OpenShift.

2.6.2.17. Redundancia de componentes del sistema para alta disponibilidad

[Red Hat Inc., 2013b.] La plataforma OpenShift posee una arquitectura con un plano de control sin estado (stateless Brokers), una infraestructura de mensajería, e infraestructura de alojamiento de aplicaciones (nodos). Cada pieza de la plataforma se puede configurar con redundancia múltiple para conmutación por error y equilibrio de carga escenarios para eliminar el impacto de fallo de hardware o infraestructura.

2.6.2.18. Aprovisionamiento automático de la pila de aplicaciones

[Red Hat Inc., 2013b.] Cuando un desarrollador utiliza la plataforma OpenShift de autoservicio para crear una aplicación, OpenShift creará automáticamente los engranajes necesarios, desplegará los runtimes del lenguaje (a través de cartridges), configurará las interfaces de red y la prestación de los ajustes del DNS y, finalmente, retornará al usuario las credenciales que necesitará para comenzar a subir el código para la aplicación. Este aprovisionamiento automático reemplaza lo que históricamente podría tomar días, semanas o incluso meses para el equipo de operaciones de TI para hacerlo manualmente. Esto libera al equipo de operaciones para centrarse en las necesidades críticas de los clientes en lugar de ocuparse en la configuración de servidores de forma repetida.

2.6.2.19. Escalabilidad automática de aplicación

[Red Hat Inc., 2013b.] La plataforma OpenShift permite elasticidad cloud, proporcionando escalabilidad de aplicación horizontal automática a medida que aumenta la carga de aplicaciones,

eliminando la necesidad de Operaciones para aumentar manualmente el número de instancias de aplicación.

2.6.2.20. Elección de la infraestructura de nube (No disponible en OpenShift Online)

[Red Hat Inc., 2013b.] OpenShift empresarial está diseñado para ser desplegado en la capa superior, y se ejecuta en Red Hat Enterprise Linux (RHEL). RHEL es necesario debido a la utilización de SELinux dentro de la plataforma OpenShift, y permite además brindar Servicios de Soporte Global de Red Hat proveer soporte a las PaaS y a los runtimes y bibliotecas incluidas. OpenShift empresarial no tiene requisitos específicos para la capa de infraestructura aparte de que debe ser capaz de proveer instancias de RHEL. Para ello, los clientes OpenShift empresarial tienen la opción de infraestructura. Los clientes de la plataforma como servicios de OpenShift empresarial cuentan con opciones de infraestructura, pudiendo implementar la solución mediante servidores físicos, mediante una plataforma de virtualización, una Infraestructura como servicio, o un proveedor de nube pública, siempre y cuando las instancias de RHEL puedan ser configuradas y accedidas, se pueden implementar. Esto otorga la libertad de implementar la solución PaaS de OpenShift empresarial en la forma que mejor se ajuste a sus opciones de infraestructura existentes.

2.6.3. Autoescalabilidad

[Red Hat Inc., 2013g] Por defecto, cuando se crea una aplicación OpenShift, es automáticamente escalada basándose en la cantidad de peticiones (del inglés “request”). OpenShift también permite escalar manualmente las aplicaciones ajustando los umbrales mínimos y máximos permitidos para escalar.

[Red Hat Inc., 2013c.] Cómo funciona la escalabilidad de OpenShift

El cartucho HAProxy se sienta entre la aplicación y el Internet público y las rutas de tráfico web a sus cartuchos web. Cuando aumenta el tráfico, HAProxy notifica a los servidores OpenShift que necesita capacidad adicional. OpenShift Chequea que tenga disponibilidad (de los tres engranajes libres) y luego crea otra copia de su cartucho web en ese nuevo equipo. El código en el repositorio git se copia en cada nuevo equipo, pero el directorio de datos comienza vacío. Cuando se inicia el nuevo cartucho de copia realizará los procedimientos necesarios y el proxy de alta disponibilidad (del inglés High availability Proxy, de acrónimo "HAProxy") comenzará a realizar el enrutamiento de las peticiones web al mismo.

El algoritmo para la escalabilidad hacia arriba y hacia abajo (expansión y decremento) se basa en el número de solicitudes simultáneas a su aplicación. OpenShift asigna 10 conexiones por equipo. Si el HAProxy ve que la aplicación está sosteniendo el 90% de su capacidad máxima, añade otra instancia. Si la demanda cae un 50% de su capacidad máxima durante varios minutos, HAProxy elimina dicha instancia.

Debido a que cada cartucho es "nada compartido", si desea compartir datos entre los cartuchos puede utilizar un cartucho de base de datos. Cada uno de los engranajes creados durante la escala tiene acceso a la base de datos y puede leer y escribir datos consistentemente.

OpenShift, en su plan de expansión y crecimiento, pretende añadir más capacidades como almacenamiento compartido, bases de datos a escala, y el almacenamiento en caché compartida.

La consola web OpenShift muestra cuántas instancias están actualmente siendo consumidas por la aplicación.

2.6.4. Blueprints / Imágenes para acelerar el aprovisionamiento

[Red Hat Inc., 2013e.] En OpenShift se crean aplicaciones y las aplicaciones se componen de equipos (o instancias). Por razones de simplicidad, es conveniente pensar en un equipo como un usuario con un directorio raíz y su propio contexto.

Las instancias son lo que ponemos dentro de su equipo para que sean útiles. Si desea utilizar PHP, usted necesitará un cartucho de PHP. Si desea utilizar Jboss, deberá hacer uso del cartucho Jboss.

[Red Hat Inc., 2013f.] Flujo de trabajo:

1. Las herramientas de cliente contactan con el agente a través de una API REST. El agente encuentra un nodo donde instalar la instancia. Se crea entonces en ese nodo un equipo vacío.
2. Una vez que se crea el equipo, nuestro cartucho se copia en ese equipo.
3. Una vez copiado el cartucho, cada archivo que aparece en `./metadata/locked_files.txt` se crea y se establece como propiedad del usuario.
4. El script de configuración se ejecuta.
5. Se establece como propietario a "root" a todos los archivos que aparecen en `./metadata/locked_files.txt`, para que el usuario no pueda modificarlos
6. Los Ganchos de conexión (del inglés "connection hooks") se ejecutan
7. El cartucho se inicia a través de Inicio de control

2.6.6. Soporte para Sistemas operativos Linux

[Red Hat Inc., 2013d.] OpenShift funciona únicamente con el sistema operativo Red Hat Enterprise Linux en su versión 6.3 (o superior) para las plataformas de hardware x86 o bien x64. A la fecha de creación de este documento no es factible correr OpenShift bajo otro sistema operativo.

2.6.7. Soporte para almacenamiento de datos

[Red Hat Inc., 2013h] Las aplicaciones online de OpenShift soportan diferentes motores de bases de datos. Entre ellos podemos mencionr a MySQL 5.1, PostgreSQL 8.4, MongoDB 2.2 y SQLite.

Tanto MySQL como PostgreSQL son motores de bases de datos relacionales tradicionales basadas en el lenguaje de consultas SQL.

[SQLite, 2013a.] SQLite, en cambio, es una base de datos que se implementa simplemente en una librería transaccional también basada en una base de datos SQL que no requiere configuración alguna.

[MongoDB, 2013a.] Además, MongoDB es una base de datos documental OpenSource, de tipo NoSQL, escrita en lenguaje C++ que está principalmente orientada al almacenamiento de documentos basados en formato JSON y técnicas de Map-Reduce.

2.6.8. Soporte para colas

[Red Hat Inc., 2013i.] Red Hat OpenShift ofrece soporte para colas de mensajería con el producto IronMQ.

[IronMQ. 2013a] IronMQ es un servicio de colas de mensajería fácil de usar y de alta disponibilidad. IronMQ está construido para distribuir aplicaciones en la nube con necesidades de mensajería críticos. Proporciona colas de mensajes en demanda con el transporte HTTPS, entrega FIFO, y persistencia de mensajes, con rendimiento optimizado en la nube.

Algunas de sus características principales son:

- Listo para usar: Sólo tiene que conectar IronMQ a los puntos finales (del inglés "endpoints") y ya está listo para crear colas y enviar y recibir mensajes. Perfecto para hacer mucho más cosas de forma asíncrona.

- alta disponibilidad: Se ejecuta sobre infraestructuras de nube y utiliza múltiples centros de datos de alta disponibilidad. Utiliza almacenes de datos fiables para la durabilidad y persistencia de sus mensajes.
- Rendimiento escalable / Alto: Escrito con la elasticidad y la escalabilidad en su núcleo. Construido con lenguajes de alto rendimiento diseñados para la concurrencia, se ejecuta en las nubes de potencia industrial.
- Múltiples enlaces de lenguaje: hace uso de un amplio conjunto de bibliotecas de idiomas IronMQ como Ruby, PHP, Python, Java, Go, y mucho más.
- Gateways Seguros / OAuth2: Utiliza HTTPS y SSL para proporcionar pasarelas (del inglés "gateways") seguras para la gestión de colas y mensajes. OAuth2 proporciona flexibilidad, escalabilidad y seguridad.

2.6.9. Alternativas de Hipervisor

[Red Hat Inc., 2013j] Si bien OpenShift ofrece tres alternativas para el uso de hipervisores. Estas son KVM (Kernel-Based Virtual Machine), Xen y Qemu.

Esta plataforma no ofrece soporte para virtualizadores de Microsoft como Hyper-V, lo cual implica limitaciones para la implementación de Sistemas operativos basados en Kernels Windows.

Las instalaciones típicas de OpenShift se realizan con el hipervisor KVM.

2.7. IBM SMARTCLOUD

En esta sección se ven en detalle los siguientes puntos de la plataforma IBM SmartCloud: Descripción (sección 2.7.1), características principales (sección 2.7.2), autoescalabilidad (sección 2.7.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.7.4), Soporte para Sistemas operativos Windows y Linux (sección 2.7.5), soporte para almacenamiento de datos (sección 2.7.6), soporte para colas (sección 2.7.7) y soporte para hipervisores (sección 2.7.8) y Datagrids para la plataforma (sección 2.7.9).

2.7.1. Descripción

[IBM Corp., 2011d] SmartCloud ofrece una gestión de cloud con el valor agregado que permite la elección y la automatización más allá del aprovisionamiento de máquinas virtuales.

IBM SmartCloud Enterprise+ es un entorno Cloud seguro, totalmente administrado y listo para producción, diseñado para garantizar una alta performance y disponibilidad.

SmartCloud Enterprise+ ofrece un control completo de "governance", administración y gestión, permitiendo definir acuerdos de nivel de servicio (SLA) para alinear las necesidades de negocio y los requisitos de uso.

Ofrece además múltiples opciones de seguridad y aislamiento, integrados en la infraestructura virtual y de red, manteniendo el cloud separado de otros entornos cloud.

2.7.2. Características Principales

- PureApplication System soporta múltiples virtualizadores. Es posible virtualizar, por ejemplo, con VMWare o Hyper-V.
- Los usuarios pueden autogestionar los recursos que necesitan (patterns de aplicación, requerimientos de disponibilidad, CPU, memoria, storage, etc) y el sistema realiza el deployment y el accounting automáticamente.
- Es posible definir políticas para las aplicaciones de manera que los recursos que esta insume sean elásticos, es decir, que crezcan o disminuyan de acuerdo con las variaciones de la demanda.
- Datagrid provisto por WebSphere eXtreme Scale, que se integra con PureApplication System. Ofrece un datagrid distribuido, manejando la persistencia y la transaccionalidad. Además, opcionalmente soporta la sincronización con una fuente de datos.
- WSX brinda una API simple y poderosa para manipular los datos del grid, permitiendo que sean accedidos por REST o directamente por conector nativo ADO.NET para las aplicaciones C# como SiSe 3G.
- Los servicios de PureApplication System son diferenciales conceptualmente con respecto a los servicios provistos por los otros clouds vendors mencionados, puesto que se implementa en un appliance. PureApplication System, que integra la funcionalidad de Cloud, de virtualización de aplicaciones y de datagrid, funciona todo en una misma "caja" (hardware) provista por IBM.

2.7.3. Autoescalabilidad

[IBM Corp., 2011c.] Obtener aplicaciones basadas en WebSphere rápidamente en la nube con el Servicio de Carga de Trabajo de aplicación de IBM SmartCloud: El Servicio de Carga de Trabajo de aplicación (SCAWS) de IBM SmartCloud elimina el tedio de la gestión de middleware y de infraestructura, lo que permite implementar más fácil, rápida y repetidamente aplicaciones basadas en WebSphere para la nube pública de IBM SmartCloud Enterprise.

[IBM Corp., 2011c.] Acelerar y simplificar la llegada a la nube con patrones: Como núcleo de la tecnología PaaS de IBM y construido en los años de experiencia de IBM, los patrones son plantillas que consisten en software y recursos de la máquina virtual. Los patrones simplifican radicalmente la configuración y gestión de los recursos de hardware y software, lo que permite a los equipos implementar rápidamente entornos de aplicaciones complejas para la nube con resultados validados, de alta calidad y consistentes.

[IBM Corp., 2011c.] Escale las Aplicaciones con escalabilidad automatizada basada en políticas: Los patrones tienen políticas para proporcionar automatización de gran alcance, incluyendo una política de enrutamiento para el balanceo de cargas, una política de registro, una política de JVM, y una política de escalabilidad basada en reglas automatizadas. Al establecer la política de ampliación, un patrón desplegado puede escalar de forma dinámica en función de las reglas (por ejemplo, sobre la base de tiempo de respuesta de solicitud).

[IBM Corp., 2013a.] Rápido aprovisionamiento y escalabilidad

Con el aprovisionamiento de IBM SmartCloud, las organizaciones de TI pueden implementar rápidamente el tipo específico de entorno cloud que necesitan sin importar el tamaño, para empezar a tomar ventaja del aumento de la flexibilidad y el rendimiento que ofrece el cloud computing. En cualquier momento, la nube se puede escalar rápidamente hacia arriba o hacia abajo según sea necesario, para ayudar a las organizaciones a satisfacer las necesidades cambiantes de TI sin interrumpir las operaciones.

Despliegue de cientos de máquinas virtuales en menos de cinco minutos: El aprovisionamiento de IBM SmartCloud permite a las organizaciones tomar ventaja inmediata de la mayor flexibilidad y rendimiento que ofrece el cloud computing. Al acelerar la entrega de los recursos informáticos, a su vez acelera la entrega de productos y servicios innovadores, lo que permite una respuesta más rápida a los retos y oportunidades del mercado.

El aprovisionamiento de IBM SmartCloud permite un rápido despliegue y la integración de las capacidades de la nube utilizando patrones de carga de trabajo. Los patrones pueden ser creados utilizando un patrón suministrado como una plantilla o bien se pueden crear desde cero. Una vez creado un patrón, se puede reutilizar una y otra vez para crear varias instancias idénticas en la nube. Se puede lograr una integración significativa con componentes de middleware y recursos de infraestructura para optimizar los componentes para un tipo particular de carga de trabajo de la aplicación. Capacidades dinámicas y elásticas se realizan plenamente. El sistema puede crear o eliminar recursos adicionales según sea requerido por la demanda de la aplicación.

2.7.4. Blueprints / Imágenes para acelerar el aprovisionamiento

[IBM Corp., 2013a.] Características destacadas:

- Reducir los costos y la complejidad al permitir solicitudes de autoservicio y automatización de las operaciones
- Obtener una rápida escalabilidad para satisfacer el crecimiento del negocio con un despliegue casi instantáneo de cientos de máquinas virtuales
- Evitar los proveedores de tecnología mediante la creación de cargas de trabajo en la nubes optimizadas para entornos heterogéneos
- Acelerar las implementaciones de aplicaciones utilizando los patrones de carga de trabajo repetibles
- Optimizar los entornos virtualizados con la gestión del ciclo de vida y análisis de imagen avanzada

[IBM Corp., 2013a.] Con herramientas como la Biblioteca Virtual de imagen, Construcción de Imagen y Composición (ICONO), y el Editor de diseños, El aprovisionamiento de IBM SmartCloud ayuda a las organizaciones a gestionar la expansión de imagen, velar por el cumplimiento, crear imágenes base, automatizar la instalación de software y crear patrones repetibles. Esta amplitud de capacidades ayuda a asegurar que las organizaciones no sólo puedan crear entornos de nube robustos, sino que también puedan controlar y gestionar estos entornos.

Implementaciones aceleradas con los patrones de carga de trabajo reutilizables: El aprovisionamiento de IBM SmartCloud ayuda a las organizaciones a desarrollar fácil y

rápídamente, y probar y desplegar aplicaciones de negocio, poniendo fin a la utilización de los procesos manuales, procesos complejos o aquellos que requieren mucho tiempo asociado a la creación de entornos de aplicación. Una vez que las solicitudes han completado sus tareas, los recursos retornan automáticamente al administrador de recursos compartidos. Esta solución también gestiona usuarios individuales y accesos grupales, dando a los administradores de TI el tipo correcto de los controles de acceso con niveles de eficiencia óptimos.

2.7.5. Soporte para Sistemas operativos Microsoft Windows y Linux

[IBM Corp., 2013a.] El aprovisionamiento de IBM SmartCloud ofrece una única plataforma de gestión a lo largo de las diferentes infraestructuras, ayudando a reducir la complejidad y el coste de las operaciones. Permite el diseño y despliegue de aplicaciones compuestas consistentes y repetibles en una nube de hardware virtualizado ejecutando cualquiera de los hipervisores soportados, incluyendo Linux en IBM System z®, máquinas virtuales basada en el kernel (KVM), Xen, Microsoft Hyper-V, IBM PowerVM® e IBM z/VM®. Integra cómputo, almacenamiento de red y distribución de aplicaciones para ayudar a permitir la integración organizacional. Además, ayuda a reducir los costos de licencias y hardware con su soporte para múltiples hipervisores y gestión de entornos mixtos.

2.7.6. Soporte para almacenamiento de datos

En esta sección se definen los 3 tipos de almacenamiento de datos provistos por IBM SmartCloud, organizados de la siguiente manera: IBM SONAS 1.3 (sección 2.7.6.1), IBM Storwize V7000 Unified (sección 2.7.6.2) e IBM XIV Storage Systems Gen3 (sección 2.7.6.3).

2.7.6.1. IBM SONAS 1.3

[IBM Corp. 2011e.] Características destacadas:

- 21 Petabytes de almacenamiento con rendimiento escalable y accesible a nivel global
- Proporciona escalabilidad extrema para ajustarse a la capacidad de crecimiento, satisfaciendo a las aplicaciones hambrientas de ancho de banda con rendimiento escalable.

- TCO hasta en un 40 por ciento inferior con la gestión del ciclo de vida y la migración automatizada en cinta
- Ofrece la opción de puerta de enlace (del inglés "gateway") con los para sistemas de discos IBM XIV e IBM Storwize V7000

2.7.6.2. IBM Storwize V7000 Unified

[IBM Corp., 2013b] Características destacadas:

- Sistema de almacenamiento virtualizado construido para complementar los entornos de servidores virtualizados
- Proporcionar hasta 200 por ciento de mejora en el rendimiento con la migración automática de las unidades de estado sólido de alto rendimiento (SSD) 1
- Habilita el almacenamiento de hasta cinco veces los datos primarios más activos en el mismo espacio físico en disco utilizando Real-time Compression 2 IBM
- Consolida el bloque y el almacenamiento de archivos para simplificar, obteniendo una mayor eficiencia y facilidad de gestión
- Habilita la disponibilidad casi continua de las aplicaciones a través de la migración dinámica
- Gestión de datos diseñada para que sea fácil de usar, con una interfaz gráfica de usuario y sistema de gestión con la capacidad "apuntar y hacer clic"
- Metro Mirror y Global Mirror para la replicación de datos síncrona o asíncrona entre sistemas para la eficiencia del backup
- SSD para aplicaciones que requieren alta velocidad y acceso rápido a los datos
- RAID 0, 1, 5, 6 y 10

2.7.6.3. IBM XIV Storage Systems Gen3

[IBM Corp., 2013c] IBM XIV es un sistema de almacenamiento en disco de alta gama que permite a miles de empresas a hacer frente al desafío creado por el crecimiento sorprendente de datos. IBM

XIV ofrece un alto rendimiento de punto de acceso y facilidad de uso para el cambio de juego. Por agilidad óptima en entornos de nube, IBM XIV ofrece un escalado simple, altos niveles de servicio para cargas de trabajo dinámicas, heterogéneas y una estrecha integración con hipervisores, así como también con la plataforma OpenStack.

[IBM Corp., 2013d] Características destacadas:

- Almacenamiento en disco virtualizado con hasta 3 terabytes de espacio.
- Un sistema revolucionario, probado, de alta gama de almacenamiento en disco diseñado para el crecimiento de los datos y una incomparable facilidad de uso
- Alto Rendimiento uniforme, logrado a través de paralelismo masivo y ajuste automático
- Almacenamiento virtualizado, fácil aprovisionamiento y agilidad extrema para la nube optimizada y entornos virtuales
- Opción de aumento de rendimiento adicional a través del manejo libre de almacenamiento en caché SSD
- Alta fiabilidad y disponibilidad a través de la redundancia completa, velocidad de reconstrucción sin precedentes
- TCO bajo permitido por el almacenamiento de alta densidad, la planificación simplificada, las características de la gratuidad y la gestión bajo-touch

2.7.7. Soporte para colas

[IBM Corp., 2013e] El bus de Integración de IBM (antes conocido como WebSphere Message Broker) es un bus de servicios empresariales (ESB) que proporciona conectividad y transformación de datos universal para la arquitectura orientada a servicios (SOA) y entornos no SOA. Ahora las empresas de cualquier tamaño pueden eliminar las conexiones punto a punto y procesamiento por lotes, independientemente de la plataforma, protocolo o formato de datos.

Con IBM Integration Bus es posible:

- Utilizar las capacidades robustas para hacer frente a los diferentes requisitos de integración para satisfacer las necesidades de cualquier tamaño de proyecto.
- Ayudar a toda su organización a tomar decisiones más inteligentes de negocios, proporcionando un rápido acceso, visibilidad y el control sobre los datos que fluyen a través de sus aplicaciones empresariales y sistemas.
- Conectar toda una serie de aplicaciones heterogéneas y servicios web, eliminando necesidades complejas de conectividad de punto a punto.
- Contar con el soporte de aplicaciones y servicios de Microsoft.
- Proporcionar una base de integración estandarizada, simplificada y flexible para ayudarle con mayor rapidez y facilidad a apoyar las necesidades del negocio y escalar con el crecimiento del negocio.

2.7.8. Alternativas de Hipervisor

[IBM Corp., 2013f.] IBM Systems Director es la columna vertebral de la plataforma de gestión para lograr una computación inteligente. Un componente integral de la cartera de Sistemas de IBM, IBM Systems Director permite la integración con Tivoli y plataformas de gestión de terceros, proviendo componentes para los servicios de gestión integrados. Con Systems Director puede:

- Automatizar las operaciones del centro de datos mediante la implementación de infraestructuras virtuales cloud-ready
- Unificar la gestión de los recursos físicos y virtuales, almacenamiento y redes, para los servidores IBM
- Simplificar la gestión de los sistemas optimizados
- Lograr una visión única de la utilización real de energía a través de su centro de datos
- Gestión del ciclo de vida simple de las cargas de trabajo con un uso intuitivo y reducción de la complejidad - IBM Systems Director ofrece una plataforma unificada integral de gestión de sistemas. Proporciona herramientas para el descubrimiento, el inventario, el estado, la

configuración, notificación de eventos de salud del sistema, monitoreo de los recursos, actualizaciones del sistema y la automatización de la gestión.

- Integración - Tivoli y plataformas de gestión de terceros proporcionan la base para la gestión integrada de los servicios de virtualización. Con una amplia gama de tareas disponibles en la plataforma de gestión y las herramientas automatizadas, Director asiste al personal de gestión de sistemas en el aumento de la productividad, lo que resulta en una mayor capacidad de respuesta y servicio.
- El valor de la clave de IBM Systems Director es su capacidad para trabajar en diferentes entornos de TI. Reduce drásticamente el número de herramientas de gestión e interfaces, simplificando la forma en que los administradores de TI realizan sus tareas, y la liberación de su tiempo para satisfacer las cambiantes necesidades del negocio.

[IBM Corp. 2013g] IBM Systems Director VMControl: gestión de infraestructuras físicas y virtuales lista para la nube. IBM Systems Director VMControl le ayuda a obtener más de la infraestructura de virtualización, con una gestión lista para la nube. La combinación de IBM Systems Director y VMControl le permite reducir el coste total de propiedad de su entorno virtualizado - servidores, almacenamiento y redes - al disminuir los costes de gestión, aumentando la utilización de activos, y la vinculación de rendimiento de la infraestructura de los objetivos de negocio. VMControl simplifica la gestión de los entornos virtuales a través de múltiples tecnologías de virtualización y plataformas de hardware. VMControl es una solución de gestión de virtualización multi-plataforma líder que se incluye con ediciones de IBM Systems Director, disponibles por separado o como un plug-in para IBM Systems Director. Esta diversidad de opciones de despliegue le permite seleccionar el nivel óptimo de funcionalidad de VMControl (Express, Standard o Enterprise) para su infraestructura virtualizada y para escalar sin problemas a medida que la misma evoluciona.

Características destacadas:

- Reúne la gestión física y virtual en una interfaz única para reducir la complejidad
- Ofrece una inigualable plataforma cruzada para la gestión de múltiples sistemas operativos, lo que ayuda a mejorar la prestación de servicios mediante la eliminación de silos aislados de virtualización en entornos heterogéneos

- Proporciona un "time-to-value" más rápido, y una mayor agilidad del negocio a través de la gestión de la virtualización simplificada que permite una utilización más eficaz de los recursos virtualizados
- Establece una precisión y consistencia repetibles gracias a la automatización
- Reduce los costos operativos y de infraestructura a través de una mayor eficiencia y utilización de los recursos

2.7.9. Soporte para Datagrids

[IBM Corp., 2013h] WebSphere eXtreme Scale proporciona almacenamiento esencial en caché de objetos distribuido para entornos de nube de próxima generación y escalabilidad elástica. El Software de IBM WebSphere eXtreme Scale proporciona un marco caché escalable de alto rendimiento y tecnología de redes. WebSphere eXtreme Scale proporciona una mayor calidad de servicio en entornos de computación de alto rendimiento a través de "caching elástico". El Caching elástico mejora el rendimiento y la rentabilidad de la inversión.

WebSphere eXtreme Scale es una herramienta esencial para la escalabilidad elástica y ofrece estos valiosos beneficios:

- Procesa grandes volúmenes de transacciones con extrema eficiencia y escalabilidad lineal.
- Construye rápidamente una rejilla elástica transparente, flexible y de alta disponibilidad que escala acorde a las necesidades de las aplicaciones, eliminando los límites de rendimiento de la base de datos.
- Proporciona alta disponibilidad y seguridad con copias redundantes de datos de la caché. Dispone de esquemas de autenticación que ayudan a garantizar la seguridad del sistema.
- Permite a los sistemas de back-end existentes soportar una cantidad significativamente mayor de aplicaciones, lo que reduce el coste total de propiedad (del inglés "Total cost of ownership", por sus siglas TCO).

2.8. VMWARE V CLOUD SUITE

En esta sección se ven en detalle los siguientes puntos de la plataforma VMware vCloud Suite: Descripción (sección 2.8.1), características principales (sección 2.8.2), autoescalabilidad (sección 2.7.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.8.4), Soporte para Sistemas operativos Windows y Linux (sección 2.8.5), soporte para almacenamiento de datos (sección 2.8.6), soporte para colas (sección 2.8.7) y soporte para hipervisores (sección 2.8.8).

2.8.1. Descripción

[VMware Inc., 2013d] La virtualización ha reducido los costos de IT dramáticamente, mejorando la eficiencia en gran medida. Ahora las unidades de negocio necesitan un acceso rápido a los recursos de IT para lograr mayor eficiencia en el "time to market" de los proyectos. VMware vCloud Suite es una solución de infraestructura integral y completa que simplifica las operaciones de IT ofreciendo mejores SLAs para las aplicaciones. Ayuda además a mejorar la agilidad, la eficiencia y a realizar una gestión inteligente de la administración de operaciones de computación en la nube.

2.8.1.1. ¿Qué es VMware vCloud Suite?

[VMware Inc., 2013e]. La virtualización de VMware ha ayudado a los clientes a reducir drásticamente los gastos de capital gracias a la consolidación de servidores. Ha mejorado los gastos de explotación mediante la automatización y ha minimizado la pérdida de ingresos porque reduce el tiempo de inactividad, planificado y no planificado. Sin embargo, las empresas actuales también necesitan reducir el tiempo de comercialización de sus productos y servicios. Las unidades de negocio exigen acceso rápido a los recursos de TI y a las aplicaciones. Se presenta a continuación (figura 2.3), la solución para la nube propuesta por VMware, y también un cuadro que define las responsabilidades de cada uno de los productos para la gestión e infraestructura de la nube (figura 2.4).



Figura 2.3. Solución completa e integrada en la nube [VMware Inc., 2013e].

2.8.2. Características Principales

En esta sección se presentan las principales características de la plataforma VMware VCloud Suite. Las siguientes son: Agilizar el acceso a los recursos y a las aplicaciones (sección 2.8.2.1), Simplifique y automatice la gestión de operaciones (sección 2.8.2.2), Mejores acuerdos de nivel de servicio para todas las aplicaciones (sección 2.8.2.3), Infraestructura de la nube (sección 2.8.2.4) y Operaciones en la nube (sección 2.8.2.5).

INFRAESTRUCTURA EN LA CLOUD Y GESTIÓN DE LA CLOUD		vCLOUD SUITE STANDARD	vCLOUD SUITE ADVANCED	vCLOUD SUITE ENTERPRISE
Gestión de la cloud	vCloud Automation Center • Aprovisionamiento y gestión de servicios de TI en entornos vSphere			•
	vFabric Application Director for Provisioning • Configuración e implementación automatizadas de aplicaciones de múltiples niveles en la cloud			•
	vCenter Operations Management • Supervisión de aplicaciones: SO, middleware, bases de datos • Configuración del cumplimiento normativo: cambio y configuración de máquinas virtuales y de hosts de vSphere • Optimización del rendimiento y la capacidad: analíticas, paneles y alertas • Reconocimiento de aplicaciones: detección y asignación de dependencias • Imputación de costes: informe y contabilidad de la utilización de recursos		vCOPS Advanced • • • •	vCOPS Enterprise • • • •
Infraestructura en la cloud	vCenter Site Recovery Manager • Planificación, pruebas y ejecución de la recuperación ante desastres automatizadas			SRM Enterprise •
	vCloud Networking and Security • Disponibilidad continua avanzada de red y firewall* • Redes y seguridad definidas por software e integración del ecosistema*	vCloud Net & Sec Standard •	vCloud Net & Sec Advanced • •	vCloud Net & Sec Advanced • •
	vCloud Director y vCloud Connector • Centros de datos virtualizados multicliente y capacidad de ampliación de la cloud pública	vCD y vCC Adv •	CD y vCC Adv •	CD y vCC Adv •
	vSphere • Infraestructura virtualizada con automatización basada en políticas	vSphere Enterprise Plus •	vSphere Enterprise Plus •	vSphere Enterprise Plus •

Figura 2.4. Responsabilidades de productos [VMware Inc., 2013e].

2.8.2.1. Agilice el acceso a los recursos y a las aplicaciones

[VMware Inc., 2013e] vCloud Suite permite a los técnicos de TI prestar servicios de TI controlados por políticas en régimen de autoservicio, así como aprovisionar máquinas virtuales y aplicaciones de múltiples niveles en solo unos minutos. Además, amplía la capacidad disponible para las unidades de negocio tanto de forma interna como externa. vCloud Suite incluye VMware vCloud Automation Center, que introduce un potente modelo de aprovisionamiento de servicios de TI gestionados en régimen de autoservicio a través de un portal seguro y fácil de usar controlado mediante políticas.

Las redes definidas por software permiten a las máquinas virtuales de una cloud de VMware utilizar los recursos en cualquier lugar del centro de datos, sin importar los límites de las redes. Además, con VMware vCloud Connector, el equipo de TI puede estar preparado para una demanda inesperada, recurriendo a clouds públicas de confianza, fuera de sus instalaciones, basadas en VMware cuando sea preciso.

2.8.2.2. Simplifique y automatice la gestión de operaciones

[VMware Inc., 2013e] A medida que el centro de datos se convierte en una cloud privada o híbrida, la gestión de operaciones cobra una importancia fundamental. Un entorno ágil proporciona escalabilidad elástica y permite un cambio constante de las cargas de trabajo. Para respaldar de manera proactiva a la empresa, el departamento de TI necesita planificación inteligente de la capacidad, solución automatizada de los problemas de rendimiento y gestión continua del cumplimiento normativo. vCloud Suite incluye las completas prestaciones de gestión de TI de vCenter Operations Management. La supervisión del rendimiento y las alertas proactivas garantizan que el departamento de TI pueda reaccionar a los desafíos de cumplimiento de los acuerdos de nivel de servicio en la cloud antes de que lleguen a afectar al negocio.

2.8.2.3. Mejores acuerdos de nivel de servicio para todas las aplicaciones

[VMware Inc., 2013e] Además de proporcionar agilidad a los propietarios de los negocios y de realizar una gestión óptima de los recursos del centro de datos, se espera que los equipos de TI aporten los acuerdos de nivel de servicio de rendimiento, seguridad y recuperación ante desastres que la empresa necesita.

2.8.2.4. Infraestructura de la nube

[VMware Inc., 2013f]

- **Aprovisionamiento e implementación automatizados.** Forme nuevas aplicaciones con componentes reutilizables e impleméntelas en minutos, no semanas.

- **Administración automatizada de operaciones.** Administre su nube en forma eficiente con herramientas desarrolladas especialmente para optimizar el rendimiento, asegurar la seguridad y rectificar problemas potenciales antes de que los usuarios se enteren siquiera.
- **Disponibilidad, recuperación ante desastres y cumplimiento normativo.** Suministre acuerdos exigentes de nivel de servicio, proteja sus datos y verifique el cumplimiento de políticas y regulaciones.
- **Visibilidad de los costos del departamento de TI.** Planifique la capacidad, optimice la asignación de recursos y desarrolle un modelo de cobro retroactivo de gastos completo para el departamento de TI con sensatez.
- **Capacidad completa de extensibilidad.** Personalice su entorno, integre soluciones de terceros e interopere con los servicios de computación en nube pública regidos por VMware.

2.8.2.5. Operaciones en la nube

[VMware Inc., 2013f]

- **Servicios según demanda.** Implemente un nuevo modelo de autoservicio para reducir los costos del departamento de TI y aumentar la agilidad.
- **Implementación y aprovisionamiento automatizados.** Revolucione el cumplimiento de los pedidos, el desarrollo de aplicaciones y los procesos de implementación para generar gran eficiencia.
- **Administración anticipativa de problemas e incidentes.** Aproveche la administración según políticas y la automatización para eliminar los procesos manuales propensos a errores y administrar los sistemas anticipándose a los problemas.
- **Administración de riesgo, el cumplimiento normativo y la seguridad.** Proteja su negocio asegurando estándares de nivel empresarial para la manera en que se protegen los sistemas en todo su entorno de nube.
- **Administración financiera del departamento de TI.** Adopte un nuevo paradigma financiero que le brinde transparencia y conecte los costos de los servicios de TI directamente con la demanda y el consumo.

2.8.3. Autoescalabilidad

[VMware Inc., 2013e] VMware vCenter Server proporciona una plataforma centralizada y extensible para la gestión de la infraestructura virtual. vCenter Server administra entornos de VMware vSphere, proporcionando a los administradores de TI un control simple y automatizado en el ambiente virtual para entregar la infraestructura con confianza.

Principales beneficios:

- Analizar y solucionar problemas rápidamente con visibilidad de la infraestructura virtual vSphere.
- Entregar la seguridad y disponibilidad de vSphere a través de funciones de gestión proactiva automatizadas, como el balanceo de carga automático y fuera de la caja (de la expresión en inglés "out-of-the-box") de automatización de flujos de trabajo.
- Ampliar las capacidades de virtualización con soluciones ambientales de terceros

2.8.4. Blueprints / Imágenes para acelerar el aprovisionamiento

[VMware Inc., 2013e] VMware vCenter Server proporciona una gestión centralizada de la infraestructura virtual vSphere. Los administradores de TI pueden asegurar la seguridad y disponibilidad, simplificar las tareas del día a día, y reducir la complejidad de la gestión de la infraestructura virtual.

¿Cómo funciona el vCenter Server?: El Control y visibilidad centralizada proporciona una gestión centralizada de hosts virtuales y máquinas virtuales desde una única consola. Esto proporciona a los administradores una mayor visibilidad en la configuración de los componentes críticos de una infraestructura virtual, todo desde un solo lugar. Con vCenter Server, los entornos virtuales son más fáciles de manejar: un solo administrador puede gestionar cientos de cargas de trabajo, más del doble de la productividad típica en la gestión de la infraestructura física.

2.8.4.1. Infraestructura Virtual entregada con confianza

Cumplir consistentemente con los Acuerdos de nivel de servicio de aplicación críticos para el negocio (del inglés "Service level agreement", y su acrónimo SLA), exige una gestión proactiva automatizada para maximizar las capacidades de vSphere. Entre las principales funcionalidades habilitadas por vCenter Server incluyen: VMware vSphere vMotion, VMware vSphere Distributed Resources Scheduler, VMware vSphere High Availability (HA) y VMware vSphere Fault Tolerance. VMware vCenter Orchestrator también proporciona a los administradores la capacidad de crear e implementar fácilmente los flujos de trabajo con las mejores prácticas. Con una gestión proactiva automatizada, vCenter Server permite que se cumplan los niveles de servicio mediante el aprovisionamiento de nuevos servicios de forma dinámica, equilibrando los recursos y la automatización de alta disponibilidad.

2.8.5. Soporte para Sistemas operativos Microsoft Windows y Linux

[VMware Inc., 2013e] El Gestor Multi-hipervisor de vCenter Server proporciona una gestión integrada y simplificada de hosts VMware e Hyper-V, brindando la posibilidad de implementar tanto sistemas operativos de la línea Windows como también aquellos basados en kernels Linux.

2.8.6. Soporte para almacenamiento de datos

[VMware Inc., 2010a] En el entorno de vCloud Director, el proveedor expone un conjunto de cómputo virtualizado, almacenamiento y recursos de red para ser consumidos por los usuarios en la nube. La puesta en común de los recursos virtuales se define y se gestiona a través de vCloud Director por el administrador de una manera que ofrece tanto elasticidad como escalabilidad. Desde la perspectiva de los usuarios (consumidores) los recursos de almacenamiento aparecen como un conjunto ilimitado de almacenamiento de rendimiento y costo uniforme. El objetivo de vCloud Director es el de proporcionar a los consumidores la capacidad de almacenamiento ilimitado. Un Proveedor VDC es un fondo de recursos de un clúster de servidores VMware ESX que acceden a un recurso de almacenamiento compartido. El proveedor VDC puede contener:

- 1) parte de un almacén de datos (compartida por otros CDA Provider)
- 2) un almacén de datos completo

3) múltiples almacenes de datos.

Como el almacenamiento se aprovisiona por Organización de VDCs, la agrupación de almacenamientos compartido para el proveedor de VDC se considera como una agrupación de almacenamiento sin distinción de características de almacenamiento, protocolos u otras características que lo diferencian de ser algo más que un espacio de direcciones de gran tamaño. En el caso de un Proveedor de VDC que se compone de más de un almacén de datos, se considera buena práctica que los almacenes de datos tengan capacidades de rendimiento iguales, así como protocolos y calidad de servicio. En caso contrario, el rendimiento del Proveedor de agrupación de almacenamiento VDC se verá afectado por el almacenamiento más lento. Algunos VDCs podrían terminar con el almacenamiento más rápido que otros. Con el fin de obtener los beneficios de los diferentes niveles o protocolos de almacenamiento, será necesario definir los Proveedores VDCs separados donde cada proveedor de VDC tendría almacenamiento de diferentes protocolos y diferentes calidades de servicio de almacenamiento. Por ejemplo, se podría suministrar un proveedor de VDC que se base en un almacén de datos de respaldo de discos de 15K RPM FC con un montón de caché en el disco para obtener el más alto rendimiento de disco, y un segundo proveedor de VDC que se basa en un almacén de datos de respaldo de unidades SATA y con poco caché en la matriz de un nivel inferior. Cabe señalar que cuando un proveedor de VDC tiene un almacén de datos que se comparte con otro Proveedor de VDC, uno puede encontrar que un Proveedor de VDC está causando impacto en el rendimiento en otro Proveedor de VDC. Por lo tanto, se considera buena práctica tener un proveedor de VDC que tenga un almacén de datos dedicado de forma que el aislamiento del almacenamiento reduzca las posibilidades de diferentes calidades de servicios de almacenamiento en un solo proveedor de VDC.

2.8.7. Soporte para colas

[VMware Inc., 2013h] El plug-in AMQP de VMware vCenter Orchestrator permite a las organizaciones activar automáticamente los flujos de trabajo basados en mensajes Advanced Message Protocol Queue Server (AMQP). AMQP es un protocolo de alta escalabilidad de publicación y suscripción de mensajes que se utiliza cada vez más en las arquitecturas de nube. Es el protocolo de mensajería por defecto para vCloud Director. Con el AMQP plug-in, las organizaciones pueden definir políticas que activan automáticamente los flujos de trabajo específicos en función de ciertos mensajes AMQP. Por ejemplo, como parte de una actividad de pre-aprovisionamiento de una vApp, vCloud Director Orchestrator puede detectar la petición de provisión y recuperar automáticamente la dirección IP de un sistema externo antes de permitir que

vCloud Director continúe con la actividad de aprovisionamiento. Cuando Orchestrator detecta que el aprovisionamiento de vApp se ha completado, Orchestrator puede actualizar las bases de datos de gestión de la configuración y otros sistemas de gestión de la información sobre la nueva instancia de vApp.

El AMQP plug-in proporciona la capacidad de controlar y publicar mensajes AMQP, así como para llevar a cabo tareas administrativas, como la configuración de corredores AMQP y gestión de colas. El plug-in AMQP de VMware soporta RabbitMQ así como otras implementaciones de AMQP.

2.8.8. Alternativas de Hipervisor

[VMware Inc., 2013e] El Gestor de Multi-hipervisor proporciona una gestión integrada y simplificada de hosts VMware e Hyper-V.

VMware vCenter Server proporciona la gestión centralizada de infraestructura virtual vSphere. Los administradores de TI pueden asegurar así la seguridad y disponibilidad, simplificar las tareas del día a día, y reducir la complejidad de la gestión de la infraestructura virtual.

[VMware Inc., 2013g] ¿Qué es VMware vCenter Multi-Hypervisor Manager 1.1?

VMware vCenter Multi-Hypervisor Manager es un componente que activa el soporte para los hipervisores heterogéneos en VMware vCenter Server. Ofrece los siguientes beneficios a su entorno virtual:

- Una plataforma integrada para la gestión de VMware y los hipervisores de terceros desde una única interfaz.
- Alternativas de hipervisores para las diferentes unidades de negocio de la organización para satisfacer sus necesidades específicas.
- Múltiples proveedores de hipervisores soportados.
- Cuando se agrega un host de terceros a vCenter Server, todas las máquinas virtuales que existen en el host se detectan automáticamente y se agregan al inventario de los ejércitos de terceros.

[VMware Inc., 2013g] Características destacadas:

vCenter Multi-Hypervisor Manager 1.1 introduce el siguiente conjunto de capacidades básicas de gestión sobre otros fabricantes:

- Gestión de hosts de terceros incluyendo agregar, quitar, conectar, desconectar y ver la configuración del host.
- Posibilidad de migrar máquinas virtuales desde hosts de terceros para ESX o ESXi.
- Capacidad para la provisión de máquinas virtuales en hosts de terceros.
- Capacidad para editar la configuración de la máquina virtual.
- mecanismo de autorización integrado del servidor vCenter en ESX / ESXi e inventarios de los host de terceros para obtener privilegios, roles y usuarios.
- Detección automática de las máquinas virtuales de terceros preexistentes
- Capacidad para realizar operaciones sobre hosts y máquinas virtuales.
- Posibilidad de conectar y desconectar DVD, CD-ROM, unidades de disco y las imágenes de disco para instalar sistemas operativos.

2.9. OPENSTACK

En esta sección se ven en detalle los siguientes puntos de la plataforma OpenStack: Descripción (sección 2.9.1), características principales (sección 2.9.2), autoescalabilidad (sección 2.9.3), Blueprints / Imágenes para acelerar el aprovisionamiento (sección 2.9.4), Soporte para Sistemas operativos Windows (sección 2.9.5), Soporte para Sistemas operativos Linux (sección 2.9.6), soporte para almacenamiento de datos (sección 2.9.7), soporte para colas (sección 2.9.8) y soporte para hipervisores (sección 2.9.9).

2.9.1. Descripción

[OpenStack, 2013o] OpenStack es un conjunto de proyectos de software de código abierto que las empresas / proveedores de servicios pueden usar para configurar y ejecutar su nube de computación

e infraestructura de almacenamiento. Rackspace y la NASA son los contribuyentes iniciales clave para la pila. Rackspace contribuyó con su plataforma "Archivos en la Nube" (código) para alimentar la parte de almacenamiento de objetos de OpenStack, mientras que la NASA aportó su plataforma "Nebulosa" (código) para alimentar la parte Compute. El Consorcio OpenStack ha logrado tener más de 100 miembros, incluyendo Canonical, Dell, Citrix, etc en menos de un año. OpenStack hace que sus servicios se encuentren disponibles por medio de una API compatible con Amazon EC2/S3. Por lo tanto, las herramientas cliente escritas para AWS se pueden utilizar con OpenStack.

2.9.2. Características Principales

En esta sección se describen las 3 familias de servicios principales de OpenStack: Infraestructura de cómputo, llamada Nova (sección 2.9.2.1), Infraestructura de Almacenamiento, llamada Swift (sección 2.9.2.2) y Servicios de imagen, llamados Glance (sección 2.9.2.3)

2.9.2.1. Infraestructura de Cómputo (Nova)

[OpenStack, 2013a] Nova es el controlador de fábrica para Cómputo para la nube OpenStack. Todas las actividades necesarias para apoyar el ciclo de vida de las instancias dentro de la nube OpenStack se manejan por Nova. Esto hace que Nova sea una plataforma de gestión que administra los recursos de cómputo, redes, autorización, y las necesidades de escalabilidad de la nube OpenStack. Sin embargo, Nova no proporciona ninguna capacidad de virtualización por sí mismo, sino que utiliza las API de libvirt para interactuar con los hipervisores compatibles. Nova expone todas sus capacidades a través de una API de servicios web que sea compatible con la API de EC2 de Amazon Web Services.

[OpenStack, 2013b] Funciones y características:

- Gestión del ciclo de vida de Instancias
- Gestión de recursos informáticos
- Redes y Autorización
- API basada en REST
- Comunicación consistente, eventualmente asincrónica

- Agnóstico de Hypervisor: soporte para Xen, XenServer / XCP, KVM, UML, VMware vSphere y Hyper-V

2.9.2.2. Infraestructura de Almacenamiento (Swift)

[OpenStack, 2013c] Swift proporciona un almacén de objetos virtuales eventualmente consistentes distribuida para OpenStack. Es análogo a Amazon Web Services - Simple Storage Service (S3). Swift es capaz de almacenar miles de millones de objetos distribuidos en diferentes nodos. Swift ha incorporado redundancia y tolerancia a fallos de gestión y es capaz de transmitir y guardar multimedia. Es sumamente escalable tanto en términos de tamaño (varios petabytes) y de capacidad (Número de objetos).

[OpenStack, 2013d] Funciones y características:

- Almacenamiento de gran cantidad de objetos
- Almacenamiento de objetos de tamaño grande
- Redundancia de datos
- Capacidad de Archivo - Trabajo con conjuntos de datos grandes
- Contenedor de datos de máquinas virtuales y aplicaciones cloud
- Capacidad de transmisión de multimedia
- Almacenamiento seguro de los objetos
- Copia de seguridad y archivado
- Escalabilidad Extrema

2.9.2.3. Servicios de Imagen (Glance)

[OpenStack, 2013e] Servicios de imágenes de OpenStack es un sistema de búsqueda y recuperación de imágenes de máquinas virtuales. Puede ser configurado para utilizar uno cualquiera de los siguientes backends de almacenamiento:

- Sistema de archivos local (por defecto)
- Almacenar objetos OpenStack para almacenar imágenes
- Almacenamiento directo en S3 (Simple storage service de Amazon EC2)
- Almacenamiento S3 con almacén de objetos como intermediario para el acceso a S3.
- HTTP (de sólo lectura)

[OpenStack, 2013f] Funciones y características:

- Proporciona servicios de imagenes

2.9.3. Autoescalabilidad

[OpenStack, 2013n] Heat es un servicio para orquestar múltiples aplicaciones compuestas en la nube utilizando el formato de la plantilla CloudFormation AWS, tanto a través de una API REST OpenStack-nativa como un API de consultas CloudFormation-compatible.

Heat proporciona una implementación CloudFormation AWS para OpenStack que orquesta una plantilla CloudFormation AWS que describe una aplicación de nube ejecutando las llamadas apropiadas a la API OpenStack, para generar la ejecución de aplicaciones de nube.

El software integra otros componentes básicos de OpenStack en un sistema de plantillas de un archivo. Las plantillas permiten la creación de la mayoría de los tipos de recursos OpenStack (tales como instancias, IPs flotantes, volúmenes, grupos de seguridad, usuarios, etc), así como también algunas funciones avanzadas tales como una alta disponibilidad de instancias, autoescalabilidad de instancias, y pilas anidadas. Al proporcionar una integración tan estrecha con otros proyectos núcleo de OpenStack, todos los proyectos núcleo de OpenStack podrían recibir un mayor número de usuarios.

2.9.4. Blueprints / Imágenes para acelerar el aprovisionamiento

En esta sección se presentan las herramientas disponibles en esta plataforma para la creación y automatización de máquinas virtuales. Estas son: Oz (sección 2.9.3.1), vmbuilder (sección 2.9.3.2), BoxGrinder (sección 2.9.3.3), VeeWee (sección 2.9.3.4), ImageFactory (sección 2.9.3.5).

2.9.3.1. Oz

[OpenStack, 2013g] Oz es una herramienta de línea de comandos que automatiza el proceso de creación de un archivo de imagen de máquina virtual. Oz es una aplicación Python que interactúa con KVM para pasar por el proceso de instalación de una máquina virtual. Se utiliza un conjunto predefinido de arranque rápido (sistemas basados en RedHat) y ficheros de preconfiguración (sistemas basados en Debian) para los sistemas operativos que soporta, y también permite crear imágenes de Microsoft Windows.

2.9.3.2. vmbuilder

[OpenStack, 2013g] vmbuilder (Generador de máquinas virtuales) es una herramienta de línea de comandos que se puede utilizar para crear imágenes de máquinas virtuales para diferentes hipervisores. La versión de vmbuilder que viene con Ubuntu sólo puede crear máquinas virtuales de Ubuntu. La versión de vmbuilder que viene con Debian pueden crear máquinas virtuales Ubuntu y Debian.

2.9.3.3. BoxGrinder

[OpenStack, 2013g] BoxGrinder es otra herramienta para la creación de imágenes de máquinas virtuales. BoxGrinder puede crear imagenes de máquinas virtuales Fedora, Red Hat Enterprise Linux, CentOS. BoxGrinder sólo está soportado en Fedora.

2.9.3.4. VeeWee

[OpenStack, 2013g] VeeWee se utiliza a menudo para construir cajas de Vagrant, pero también puede ser usado para construir imágenes KVM.

2.9.3.5. Imagefactory

[OpenStack, 2013g] Imagefactory es una herramienta nueva diseñada para automatizar la construcción, y convertir y subir imágenes a diferentes proveedores de la nube. Utiliza Oz como su back-end e incluye soporte para las nubes basadas en OpenStack.

2.9.5. Soporte para Sistemas operativos Microsoft Windows

[OpenStack, 2013h.] Es posible utilizar Hyper-V como un nodo de cálculo dentro de una implementación de OpenStack. El servicio de cómputo nova se ejecuta como un servicio de 32 bits directamente en la plataforma de Windows con la función Hyper-V habilitada. Los componentes de Python necesarios, así como el servicio de Cómputo Nova se instalan directamente en la plataforma Windows. Los Servicios de Cluster Server de Windows no son necesarios para la funcionalidad de la infraestructura OpenStack. El uso de la plataforma Windows Server 2012 se recomienda para obtener mejores resultados. Las plataformas Windows siguientes han sido probadas como nodos de cómputo:

- Windows Server 2008R2: Tanto Server y Server Core con el rol Hyper-V activados (Nada Compartido La migración en vivo no es compatible con 2008r2)
- Windows Server 2012: Server y Core (con la función Hyper-V habilita) e Hyper-V Server

2.9.6. Soporte para Sistemas operativos Linux

[OpenStack, 2013i.] ¿Qué es RDO?

RDO es una disposición de distribución libre, apoyada por la comunidad de OpenStack, que se ejecuta en Red Hat Enterprise Linux, Fedora y sus derivados. Además de proporcionar un conjunto de paquetes de software, RDO permite además a los usuarios de la plataforma de computación en la nube en los sistemas operativos Red Hat Linux obtener ayuda y comparar notas sobre la ejecución de OpenStack.

El proyecto OpenStack se beneficia de un amplio grupo de proveedores y distribuidores, pero ninguno cuenta con la experiencia en producción de Red Hat, la experiencia técnica y el compromiso con la forma de código abierto de la producción de software. Algunas de las más grandes nubes de producción en el mundo se ejecutan y son apoyadas por Red Hat, y los ingenieros de Red Hat contribuyen a todas las capas de la plataforma OpenStack. Desde el núcleo de Linux y los componentes del hipervisor KVM hasta los componentes de nivel superior del proyecto OpenStack, Red Hat se encuentra cerca de la parte superior de la lista en términos de número de desarrolladores y de contribuciones.

2.9.7. Soporte para almacenamiento de datos

[OpenStack, 2013j.] Además de la tecnología de almacenamiento de clase empresarial tradicional, muchas organizaciones ahora tienen una variedad de necesidades de almacenamiento con requisitos de rendimiento y precios variables. OpenStack tiene soporte para almacenamiento de objetos y Block Storage, con muchas opciones de implementación para cada uno dependiendo del caso de uso. El almacenamiento de objetos es ideal para el almacenamiento eficaz con escalabilidad horizontal. Proporciona una plataforma de almacenamiento accesible por medio de una API completamente distribuida que se puede integrar directamente en las aplicaciones o utilizar para copia de seguridad, archivo y conservación de los datos. Block Storage permite mejorar el rendimiento y la integración con las plataformas de almacenamiento empresarial, como NetApp, Nexenta y SolidFire.

OpenStack ofrece, almacenamiento de objetos escalable y redundante utilizando clusters de servidores estandarizados capaces de almacenar petabytes de datos

El almacenamiento de objetos no es un sistema de archivos tradicional, sino más bien un sistema de almacenamiento distribuido de datos estáticos, como imágenes de máquina virtual, almacenamiento de fotos, almacenamiento de correo electrónico, copias de seguridad y archivos. Al no tener "cerebro" central, el punto principal de control proporciona una mayor escalabilidad, redundancia y durabilidad.

Los objetos y los archivos se escriben en múltiples unidades de disco distribuidos en el centro de datos (del inglés "Data Center"), con la responsabilidad del software de OpenStack de asegurar la replicación y la integridad de los datos en el clúster.

Las agrupaciones de almacenamiento (del inglés "Storage Clusters") escalan horizontalmente simplemente añadiendo nuevos servidores. En caso de que un servidor fallara, OpenStack replica el contenido a otros nodos activos en nuevas ubicaciones del clúster. Debido a que OpenStack utiliza la lógica del software para asegurar la replicación y distribución a través de diferentes dispositivos, se pueden utilizar, en lugar de los equipos más caros, los discos duros y servidores de datos de materias primas baratas.

Bloque Capacidades de almacenamiento:

OpenStack ofrece dispositivos de almacenamiento persistentes a nivel de bloque para su uso con instancias de proceso OpenStack.

El sistema de almacenamiento de bloques gestiona la creación, montaje y desmontaje de los dispositivos de bloque en los servidores. Los volúmenes de almacenamiento de bloques se integran

plenamente en OpenStack Compute y su tablero de control permite a los usuarios en la nube gestionar sus propias necesidades de almacenamiento.

Además, utiliza el almacenamiento del servidor Linux simple, que tiene soporte de almacenamiento unificado para numerosas plataformas de almacenamiento, incluyendo Ceph, NetApp, Nexenta, SolidFire y Zadara.

Bloquear el almacenamiento es adecuado para escenarios sensibles al rendimiento, como el almacenamiento de bases de datos, sistemas de archivos expandibles, o la prestación de un servidor con acceso al almacenamiento a nivel de bloque en bruto.

La gestión de Snapshots proporciona una funcionalidad de gran alcance para hacer copias de seguridad de los datos almacenados en volúmenes de almacenamiento en bloque. Los Snapshots se pueden restaurar y utilizar para crear un nuevo volumen de almacenamiento en bloque.

2.9.8. Soporte para colas

[OpenStack, 2013k.] Message Queue (Rabbit MQ Server)

OpenStack se comunica entre sí utilizando la cola de mensajes a través de AMQP (Protocolo avanzado de colas de mensajería, del inglés "Advanced Message Queue Protocol). Nova utiliza llamadas asincrónicas para la solicitud de respuesta, con una devolución de llamada que se desencadena una vez que se recibe una respuesta. Dado que se utiliza la comunicación asincrónica, ninguna de las acciones del usuario se bloquea por mucho tiempo en un estado de espera. Esto es efectivo ya que muchas de las acciones previstas por la API de llamadas tales como el lanzamiento de una instancia o añadir una imagen demandan mucho tiempo.

[OpenStack, 2013l.] ¿Qué puede hacer RabbitMQ?

Las funciones de Mensajería permiten conectar y ampliar las aplicaciones de software. Las aplicaciones pueden conectarse entre sí, como componentes de una aplicación más grande, o a los dispositivos de usuario y datos. La mensajería es asíncrona, desacoplando las aplicaciones mediante la separación del envío y recepción de datos.

La entrega de datos, las operaciones no bloqueantes, notificaciones push, publicación/suscripción, procesamiento asincrónico y colas de trabajo son patrones que forman parte de la mensajería.

RabbitMQ es un broker de mensajería que opera como intermediario. Proporciona a las aplicaciones una plataforma común para enviar y recibir mensajes, de manera que los mensajes permanezcan en un lugar seguro hasta que sean recibidos.

2.9.9. Alternativas de Hipervisor

[OpenStack, 2013m.] El módulo de cómputo de OpenStack soporta varios hipervisores. La mayoría de las instalaciones sólo utilizan un único hipervisor, sin embargo, es posible utilizar el ComputeFilter e ImagePropertiesFilter para permitir la programación de diferentes hipervisores dentro de la misma instalación.

- KVM - Máquina Virtual basada en el Kernel. Los formatos de disco virtual que soporta son heredados de QEMU, ya que utiliza un programa de QEMU modificado para poner en marcha la máquina virtual. Los formatos soportados incluyen imágenes en bruto (del inglés "raw images"), la qcow2 y formatos de VMware.
- LXC - Linux Containers (a través de libvirt), se utiliza para ejecutar máquinas virtuales basadas en Linux.
- QEMU - Emulador rápido, por lo general sólo se utiliza para fines de desarrollo.
- UML - User Mode Linux, por lo general sólo se utiliza para fines de desarrollo.
- VMware vSphere 4.1 Update 1 y versiones posteriores, ejecuta Linux y Windows basados en imágenes VMware a través de una conexión con un servidor vCenter o directamente con un servidor ESXi.
- Xen - XenServer, Plataforma de Nube Xen (XCP), utilizado para ejecutar máquinas virtuales de Windows o Linux. Es necesario instalar el servicio nova-compute en una máquina virtual para-virtualizado.
- PowerVM - virtualización de servidores con IBM PowerVM, utilizado para ejecutar AIX, IBM i y Linux en la tecnología IBM POWER.
- Hyper-V - virtualización de servidores con Hyper-V de Microsoft, utilizado para ejecutar indows, Linux, y las máquinas virtuales FreeBSD. Ejecuta nova-cálculo de forma nativa en la plataforma de virtualización de Windows.
- Bare Metal - No es un hipervisor en el sentido tradicional, este controlador dispone de hardware físico a través de controladores configurables (por ejemplo PXE para el despliegue de imágenes, e IPMI para la administración de energía).

3. ESTUDIO COMPARADO DE LAS ARQUITECTURAS RELEVADAS

En este capítulo se realiza un estudio comparativo identificando las características de los principales proveedores de servicios de plataformas Cloud Computing que deben ser consideradas en línea con el objetivo de este trabajo. La realización de la comparación de características se manifestará por medio de un cuadro de doble entrada, en el cual se presentará en una de sus dimensiones a los principales proveedores de servicios de Cloud Computing, y en la otra, a las características más relevantes que estas plataformas ofrecen.

El estudio antedicho se organiza en este capítulo de la siguiente manera: descripción del problema (sección 3.1), características consideradas para el estudio comparativo (sección 3.2), matriz comparativa (sección 3.3), y por último se realiza un estudio de la tabla comparativa (sección 3.3).

3.1. DESCRIPCIÓN DEL PROBLEMA

A la fecha de creación de este trabajo, el mercado de tecnologías de la información cuenta con una amplia oferta de plataformas de servicios de Cloud Computing comercializados por múltiples proveedores. Muchas de las plataformas de estos proveedores proponen servicios análogos que pueden ser explotados con diversos lenguajes de programación y plataformas de desarrollo. Es por ello que se considera necesario realizar una comparación, como producto resultante de este trabajo de investigación, acerca de los servicios y características ofrecidos por los principales proveedores de las tecnologías antedichas, en pos de colaborar al esclarecimiento sobre qué plataforma puede ser conveniente para cada caso, identificando además las fortalezas y debilidades más relevantes de cada una de ellas, así como también las carencias o prestaciones faltantes por parte de cada proveedor de soluciones.

3.2. CARACTERÍSTICAS CONSIDERADAS

En la instancia de diseño del cuadro comparativo se seleccionaron un conjunto de características que por diversas razones fueron consideradas relevantes. A continuación se define cada una de ellas.

3.1.1. Definición de las características evaluadas

En la tabla 3.1 se presentan las características analizadas, y sus descripciones correspondientes para cada una de las plataformas analizadas en el capítulo 2.

Carácterística	Descripción
Escalabilidad automática (auto-scaling)	Brinda la posibilidad de incrementar o reducir de manera automática, utilizando un monitor provisto por la plataforma, la cantidad de recursos asignado a un sistema o aplicación.
Blueprints / Imágenes para acelerar el aprovisionamiento	Las imágenes o blueprints son máquinas virtuales que ya disponen de un sistema operativo y de los aplicativos o marcos de trabajo (frameworks) instalados y preconfigurados, para que sea más rápido comenzar a trabajar en la plataforma, permitiendo al usuario final focalizarse en la construcción o despliegue de sus aplicaciones. Un ejemplo popular de blueprint es llamado “LAMP”, imagen de máquina virtual conformada por Linux Apache MySQL y PHP.
Soporta Sistema operativo Windows	Esta característica permite evaluar la capacidad de implementar sistemas o aplicaciones de usuarios finales que operen bajo Sistemas Operativos Windows, y en caso afirmativo, también definir cuáles de sus versiones son soportadas.
Soporta Sistema operativo Linux	Esta característica permite evaluar la capacidad de implementar sistemas o aplicaciones de usuarios finales que operen bajo Sistemas Operativos Linux, y en caso afirmativo, también definir cuáles de sus versiones son soportadas.
Soporte para lenguajes	Esta característica permite definir cuáles son los lenguajes soportados por las distintas plataformas en análisis
Soporte para almacenamiento de datos	Esta característica define cuáles son los medios físicos que ofrecen las plataformas analizadas para la persistencia de datos.

Tabla 3.1a. Características consideradas para el análisis de las plataformas en la nube.

Característica	Descripción
Soporte para Colas (queues)	Esta característica define cuáles son los soportes para colas brindados por las diferentes plataformas. Una cola es una estructura de datos, caracterizada por ser una secuencia de elementos en la que la operación de inserción push se realiza por un extremo y la operación de extracción pop por el otro. También se le llama estructura FIFO (del inglés First In First Out), debido a que el primer elemento en entrar será también el primero en salir.
Servidor Web	Esta característica permite evaluar cuáles son las opciones de servidores web (del inglés web server) ofrecidas por cada proveedor. Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales, y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente.
Alternativas de hipervisor	Esta característica permite evaluar cuáles son los hipervisores disponibles ofrecidos por cada plataforma. Un hipervisor (del inglés hypervisor) o monitor de máquina virtual (virtual machine monitor) es una plataforma que permite aplicar diversas técnicas de control de virtualización para utilizar al mismo tiempo diferentes sistemas operativos en una misma computadora.

Tabla 3.1b. Características consideradas para el análisis de las plataformas en la nube.

Característica	Descripción
Cache In-Memory distribuido / DataGrid	<p>Los caches distribuidos o datagrids son frecuentemente implementados por tablas de hash distribuidas. Las tablas de hash distribuidas (en inglés, Distributed Hash Tables, DHT) son una clase de sistemas distribuidos descentralizados que proveen un servicio de búsqueda similar al de las tablas de hash, donde pares (clave, valor) son almacenados en el DHT, y cualquier nodo participante puede recuperar de forma eficiente el valor asociado con una clave dada. Esta clase de productos ofrecen el beneficio de mejorar los tiempos de respuesta para la búsqueda de datos, con respecto a los mecanismos de persistencia tradicionales, tales como base de datos relacionales (BDR), puesto que para acceder a un set de datos alojado en una BDR generalmente se debe establecer una comunicación TCP y luego acceder al dato realizando una lectura de disco, lo cual es menos eficiente que los caches distribuidos, a los cuales generalmente se accede por protocolo TCP y luego se accede al dato almacenado en memoria RAM (Random Access memory).</p>
Soporte para tecnologías BigData	<p>Las tecnologías Big Data (del idioma inglés grandes datos) hace referencia a los sistemas que manipulan grandes conjuntos de datos (o data sets). Las dificultades más habituales en estos casos se centran en la captura, el almacenado, búsqueda, compartición, análisis, y visualización. La tendencia a manipular ingentes cantidades de datos se debe en muchos casos a la necesidad de incluir los datos del análisis en un gran conjunto de datos relacionado, tal es el ejemplo de los análisis de negocio, los datos de enfermedades infecciosas, la lucha contra el crimen organizado, etc.</p>

Tabla 3.1c. Características consideradas para el análisis de las plataformas en la nube.

3.3. MATRIZ COMPARATIVA

Plataforma → Característica ↓	Amazon EC2	Microsoft Windows Azure	Google App Engine	Red Hat OpenShift	IBM SmartCloud	VMWare VCloud Suite	OpenStack
Escalabilidad automática (auto scaling)	Sí, a través de Amazon CloudWatch	Autoscaling application block y Windows Azure Fabric Controller.	BigTable y GFS	OpenShift HA Proxy	IBM SmartCloud Application Workload Service	VCloud Director	OpenStack Heat
Blueprints / Imágenes para acelerar el aprovisionamiento	Sí (AMI) – Imagen de máquina Amazon	Sí, provistas en una galería, y también imágenes propias guardadas	No	Sí (Single and Multitier VM Applications)	Sí	Sí, imágenes propias guardadas de máquinas virtuales VMWare	Sí, imágenes creadas por OpenStack y también compartidas por usuarios de la plataforma.
Soporta Sistema operativo Windows	<ul style="list-style-type: none"> Windows Server® 2003 R2 Windows Server 2008 Windows Server 2008 R2 Windows Server 2012 	<ul style="list-style-type: none"> Windows Server 2012 Datacenter Windows Server 2008 R2 SP1 	No	No	<ul style="list-style-type: none"> Sí Microsoft Windows Server 2003 Microsoft Windows Server 2008 	Sí, todas las distribuciones virtualizables	Sí, Windows Server 2008 R2
Soporta Sistema operativo Linux	<ul style="list-style-type: none"> Sí: SUSE Linux Enterprise Server Red Hat Enterprise Linux 	<ul style="list-style-type: none"> openSUSE 12.3 SUSE Linux Enterprise Server 11 Service Pack 2 Ubuntu Server 12.04 LTS Ubuntu Server 12.10 Ubuntu Server 13.04 OpenLogic CentOS 6.3 Ubuntu Server 12.10 DAILY 	Sí, pero las aplicaciones corren en un sandbox y Google provee acceso limitado al sistema operativo, el cual no puede ser alterado.	Sí, Red Hat Linux Enterprise	Sí Red Hat Enterprise Linux SUSE Linux Enterprise Server	Sí, todas las distribuciones virtualizables	<ul style="list-style-type: none"> Debian GNU/Linux wheezy Fedora / Red Hat Enterprise Linux / CentOS / Scientific Linux openSUSE / SLES11 SP2 Ubuntu 12.04 LTS (Precise Pangolin)
Soporte para lenguajes	<ul style="list-style-type: none"> C++ C# Java Perl Python Ruby 	<ul style="list-style-type: none"> .Net Java Node.js Python 	<ul style="list-style-type: none"> Python Java Go(experimental) 	<ul style="list-style-type: none"> Java Ruby node.js Python PHP Perl 	<ul style="list-style-type: none"> Java PHP 	<ul style="list-style-type: none"> Java C# C++ 	APIs Para: <ul style="list-style-type: none"> PHP Python Java C#.NET Ruby
Soporte para almacenamiento de datos	<ul style="list-style-type: none"> Amazon SSS Amazon Relational DB Service Amazon SimpleDB SQL Server® Express SQL Web SQL Server Standard 	<ul style="list-style-type: none"> SQL Relacional Almacenes de tablas NoSQL Blob no estructurado 	<ul style="list-style-type: none"> Base de datos no relacional "BigTable". No soporta bases de datos relacionales. 	<ul style="list-style-type: none"> MySQL PostgreSQL Mongo DB SQLite 	<ul style="list-style-type: none"> DB2 Oracle MS SQL MySQL Informix Sybase 	<ul style="list-style-type: none"> Oracle SQL Server VMware vFabric Postgres Múltiples distribuciones de Hadoop 	<ul style="list-style-type: none"> Object Storage (Swift) Block Storage (Cinder) MySQL hosts DB for Nova, Glance, Cinder, and Keystone
Soporte para Colas	Amazon Simple Queue Service	Windows Azure Service Bus, Colas FIFO con protocolos Rest, AMQP, WS	App Engine Task Queue	IronMQ	WebSphere Message Broker V8.0	RabbitMQ Protocolos AMQP, MQTT and STOMP	Rabbit MQ Server, AMQP

Tabla 3.2a. Matriz Comparativa

Plataforma → Característica ↓	Amazon EC2	Microsoft Windows Azure	Google App Engine	Red Hat OpenShift	IBM SmartCloud	VMWare VCloud Suite	OpenStack
Servidor Web	<ul style="list-style-type: none"> • Apache • IIS • Otros 	IIS V7.5	Jetty Web Server	Apache	WebSphere Application Server V7.0 and V8.0	<ul style="list-style-type: none"> • Apache • IIS • Otros 	Ofrece IaaS, no PaaS
Alternativas de hipervisores	XEN y LXC (Linux Containers)	Windows Azure Hypervisor (customized Hyper-V)	XEN/KVM	<ul style="list-style-type: none"> • KVM (Kernel-based VM) • Xen • QEMU 	<ul style="list-style-type: none"> • VMWare • Hyper-V • Otros 	VMWare	<ul style="list-style-type: none"> • XenServer/XCP • KVM • QEMU • LXC • ESXi/V • Hyper-V • Baremetal • PowerVM
Cache In-Memory distribuido / DataGrid	Open: VMWare Gemfire, Oracle Coherence, Gigaspaces XAP, Hazelcast, etc.	Windows Azure Caching / Memcached	Memcached	Infinispan	WebSphere eXtreme Scale	GemFire	Ofrece IaaS, no PaaS

Tabla 3.2b. Matriz Comparativa

3.4. ESTUDIO DE LA TABLA COMPARATIVA

A continuación se presenta un estudio comparativo de las características para las plataformas analizadas. Las secciones se organizan de la siguiente manera: Escalabilidad automática (sección 3.4.1), BluePrints/Imágenes para acelerar el aprovisionamiento (sección 3.4.2), Soporte para lenguajes (sección 3.4.3), Soporte para almacenamiento de datos (sección 3.4.4), Soporte para Colas y Servidores Web (sección 3.4.5), Alternativas de Hipervisores (sección 3.4.6), Cache In-Memory distribuido / Datagrid (sección 3.4.7)

3.4.1 Escalabilidad automática (AutoScaling)

Con respecto a la escalabilidad automática, algunos proveedores de servicios ofrecen capacidades más evolucionadas para definir los criterios de escalabilidad. Con esto nos referimos tanto a la escalabilidad que incrementa los recursos disponibles (el término popularmente utilizado en inglés es “scale up”), como a la escalabilidad que decrementa los recursos disponibles para las aplicaciones (el término popularmente utilizado en inglés “scale down”). Ejemplos de estos proveedores más evolucionados son las plataformas Amazon EC2 Cloudwatch, Windows Azure Autoscaling application Block y VMware VCloud Director, que otorgan la posibilidad de definir los criterios y o umbrales que determinarán las reglas de escalabilidad que dispararán los mecanismos necesarios para incrementar o decrementar la cantidad de recursos de hardware virtualizados que

serán asignados a las aplicaciones, de manera que estas puedan cumplir ajustándose dinámicamente a la demanda de sus clientes. En contraposición, otras plataformas como Google Apps ofrecen escalabilidad automática no controlada por el arquitecto de aplicación Cloud. En este caso, el arquitecto de aplicación cloud no podrá definir los criterios de escalabilidad para sus aplicaciones sino que este criterio se verá regido por los principios ya definidos por Google Apps, el cual cuenta con un modelo propio de escalabilidad que no puede ser ajustado u optimizado de manera alguna por el usuario de la plataforma, en función de sus necesidades particulares.

Con respecto a la autoescalabilidad de la plataforma OpenShift, provista por HA Proxy, sólo es posible definir una cantidad mínima y máxima de Cartidges (básicamente instancias) que podrán ser utilizadas en la aplicación durante los procesos de escalabilidad (tanto hacia arriba como hacia abajo) en función de la demanda de recursos de la aplicación, sin embargo, el arquitecto de una aplicación que se ejecuta en la plataforma OpenShift no cuenta con la posibilidad de definir cuáles serán los umbrales que dispararán los mecanismos de escalabilidad automática.

En el caso de IBM SmartCloud, que implementa sus mecanismos de autoescalabilidad con Smart Cloud Application Workload Scale (SCAWS), ofrece la posibilidad de utilizar plantillas predefinidas provistas por un sitio oficial de IBM que facilitan la configuración de escenarios de escalabilidad para diferentes arquetipos de aplicaciones, como por ejemplo “IBM Mobile Application Platform Pattern Type”, que optimiza las características de escalabilidad automática a los criterios típicos operacionales de aplicaciones Móviles. La plataforma de IBM también brinda cierta flexibilidad para definir reglas de escalabilidad.

Con respecto a la plataforma OpenStack, su propuesta de solución para la escalabilidad automática está implementada en “Heat”, el cual permite definir plantillas con criterios más avanzados, que permiten definir concretamente los umbrales cuantificados en niveles porcentuales (por ejemplo de uso de RAM) que serán considerados como criterios primarios para aplicar la escalabilidad automática. Heat ofrece además la posibilidad de eliminar automáticamente las instancias que se generaron al incrementarse la carga, cuando esta carga de trabajo (workload) se encuentre por debajo del umbral definido.

3.4.2 BluePrints/Imágenes para acelerar el aprovisionamiento

Amazon con su plataforma EC2 tiene la mayor oferta de imágenes para acelerar el aprovisionamiento de los proveedores analizados, contando con casi 2000 plantillas de máquinas virtuales con diferentes configuraciones. Otro proveedor que brinda buenas soluciones en este

aspecto es Microsoft Azure, que otorga la posibilidad de acelerar el aprovisionamiento de máquinas virtuales con diferentes sistemas operativos y configuraciones listadas en una galería, brindando además la posibilidad de crear las propias imágenes de máquinas virtuales personalizadas para que se adecúen perfectamente a las necesidades de sus clientes. Aunque Microsoft Azure virtualiza sus entornos con su hipervisor Hiper-V, otorga igualmente la posibilidad de convertir máquinas virtuales de VMware de manera que se puedan subir y utilizar en la plataforma Windows Azure, facilitando considerablemente la migración de aplicaciones existentes a su plataforma.

Google App Engine, en cambio, no brinda la posibilidad de acelerar el aprovisionamiento de entornos puesto que su plataforma es Infraestructura como servicio (IaaS); esto implica que App Engine no otorga la posibilidad de crear máquinas virtuales propias ni tampoco utilizar otras existentes. Existe un nuevo servicio de Google llamado Google App Compute que otorga la posibilidad de crear máquinas virtuales basadas en el sistema operativo Linux y tiende a otorgar servicios de plataforma como servicio (PaaS), sin embargo, esta plataforma aún se está gestando y no ha alcanzado un grado de madurez siquiera comparable con las plataformas que en este estudio se tratan y analizan.

La plataforma abierta de Red Hat, OpenShift, permite gestionar y acelerar el aprovisionamiento de máquinas virtuales por medio de su producto RHC (Red Hat Client), y el uso de lenguajes de scripting, mayoritariamente en lenguaje Bash, bajo el sistema operativo Red Hat Linux.

VMware brinda una flexibilidad muy grande al permitir acelerar el aprovisionamiento de entornos por medio de VCloud Director y sus imágenes de máquinas virtuales basadas en el virtualizador de VMware. Bajo estos lineamientos un usuario de la plataforma VCloud, basada en CloudFoundry, puede personalizar sus propias máquinas virtuales para que las mismas se ajusten completamente a sus necesidades, instalando los sistemas operativos que requiera (como por ejemplo Microsoft Windows Server 2008, Red Hat Linux Enterprise, Ubuntu, etc) y también el software de base necesario, tales como plataformas de desarrollo Java, .net, o el paquete clásico conocido por su acrónimo “LAMP” (Linux, Apache, MySQL y PHP).

IBM SmartCloud brinda también las herramientas para que sus usuarios puedan acelerar el aprovisionamiento de entornos en su Plataforma como servicios, otorgando 10 tamaños de instancias diferentes para sus máquinas virtuales de manera que se puedan ajustar a los requerimientos de sus aplicaciones. Asimismo, brinda 3 modelos de licenciamiento: máquinas virtuales pre-configuradas con la modalidad de pago basado en el uso, acceder a las imágenes utilizando licencias propias ya adquiridas, o subir programas de software de IBM bajo la modalidad “traer software y licencias propias (del inglés ”Bring Your Own Software and License”).

Por último, OpenStack brinda un modelo de aprovisionamiento acelerado otorgando la posibilidad de contar con más de 1100 máquinas virtuales con diferentes configuraciones (sistemas operativos y plataformas de desarrollo). Algunas de estas imágenes fueron creadas por el equipo de OpenStack y el resto publicadas por los propios usuarios de la plataforma, que decidieron compartirlas para enriquecer la comunidad.

3.4.3 Soporte para lenguajes

El soporte brindando para diferentes lenguajes de programación es crucial al momento de seleccionar una plataforma, puesto que es un factor limitante significativo en cuanto a las posibilidades que un proveedor puede ofrecer, y que sus clientes pueden explotar.

En este aspecto, Amazon EC2 ofrece un amplio abanico que cubre los principales lenguajes y plataformas de desarrollo del mercado actual. Este abanico se deriva del soporte y compatibilidad de la plataforma con gran cantidad de versiones de sistemas operativos cubriendo desde aplicaciones .net escritas en C#, aplicaciones Java multiplataforma, aplicaciones C++, aplicaciones Ruby, y también lenguajes interpretados como Perl y Python.

Microsoft Windows Azure, por su parte, brinda soporte para los lenguajes .Net (C#, Vb.net, J#, Asp.net, etc), Java (tanto con máquinas virtuales con sistema operativo Microsoft Windows 2012, como también con Sistemas operativos basados en Kernel Linux, tales como Ubuntu u OpenSUSE), Node.js para ejecutar código javascript del lado del servidor (por su expresión en inglés “server side”), y también Python. Podrían soportarse además otros lenguajes y plataformas utilizando imágenes de máquinas virtuales propias, por ejemplo las VMware, convertidas a su equivalente Windows Azure definida como disco duro virtual (por su acrónimo en inglés “VHD”, de Virtual Hard Disc).

Las opciones que ofrece Google AppEngine para soporte de lenguajes se encuentran limitadas exclusivamente a Python y java, con la opción adicional de Go, que aún se encuentra en fase experimental.

OpenShift mejora las opciones de lenguajes soportados por App Engine, otorgando la posibilidad de implementar aplicaciones desarrolladas en Java, Ruby, node.js, python, PHP y Perl; sin embargo, dadas las restricciones de sistema operativo que se derivan de que se trata de una plataforma abierta, y al no soportar sistemas operativos basados en Windows, no es posible implementar en OpenShift aplicaciones Win32 u otras basadas en .Net Framework, con lenguajes como C#, J#, Vb.net, Asp.net, etc.

Este hecho excluye a un sector del mercado que elige las plataformas Microsoft como opción para desarrollar sus aplicaciones.

IBM SmartCloud limita las posibilidades de soporte nativo en su plataforma para los lenguajes Java y PHP, aunque visto que es posible hacer uso de imágenes de máquinas virtuales soportadas por múltiples hipervisores, sería también factible implementar aplicaciones desarrolladas en otros lenguajes tales como aplicaciones .net (C#, J#, Asp.net, Vb.net, etc), aplicaciones PHP, Python y otros lenguajes, incrementando su potencial de lenguajes para múltiples plataformas, de manera que permite cubrir un segmento más amplio del mercado de aplicaciones.

La plataforma de VMware brinda soporte nativo para los lenguajes Java, C# y C++, maximizando la integración de estas aplicaciones con la Suite de productos que su plataforma ofrece, por ejemplo para DataGrids distribuidos en memoria con el producto GemFire e implementaciones de Big Data, que pueden interfacear con APIs nativas de estos lenguajes para mayor performance. Asimismo, dadas las capacidades flexibles de virtualización que VMware ofrece, sería también posible implementar aplicaciones desarrolladas en otros lenguajes (por ejemplo PHP o Ruby), e inclusive hacer uso de sus productos de DataGrids mediante interfaces basadas en protocolos interoperables como por ejemplo REST sobre HTTP.

OpenStack, por su parte, también ofrece soporte con APIs nativas para gran diversidad de lenguajes de programación. Algunos de los lenguajes que pueden gozar de los beneficios de esta plataforma son PHP, Python, Java, C#, Ruby. Esta oferta es sumamente atractiva y abarcadora, puesto que cubre los lenguajes más populares y utilizados por las aplicaciones tanto de escritorio como aquellas basadas en plataformas web e interpretadas, maximizando las posibilidades de atraer nuevos clientes.

3.4.4 Soporte para almacenamiento de datos

Los servicios de la plataforma Amazon EC2 se destacan por sus alternativas de almacenamiento de datos, puesto que cuenta con varias opciones disponibles que pueden ser aprovechadas de manera independiente por sus clientes en función de las necesidades puntuales que cada aplicación tenga que cubrir. Algunas de ellas son:

Amazon Simple Storage Service, que proporciona una interfaz de servicios web (generalmente basadas en los protocolos REST o SOAP sobre HTTP) que puede utilizarse para almacenar y recuperar prácticamente cualquier cantidad de datos desde cualquier parte de la Web. Hace uso de la misma infraestructura (económica, escalable, y segura) que utiliza Amazon para tener en

funcionamiento su propia red internacional de sitios web. Este servicio tiene como fin maximizar las ventajas del escalado y trasladar estas ventajas a los desarrolladores. Otra opción de almacenamiento de datos ofrecida por Amazon EC2 consiste en Amazon Relational DB Service, que ofrece servicios de bases de datos relacionales, las cuales son altamente compatibles con la amplia mayoría de las aplicaciones ya existentes y con las técnicas de persistencia de datos más populares del mercado (maximizando los recursos humanos disponibles con conocimientos de estas técnicas y bases de datos basadas en esta clase de tecnología). Amazon EC2 también ofrece el servicio Amazon SimpleDB, que es un almacén de datos no relacionales de alta disponibilidad y flexible que no requiere trabajo de administración de bases de datos por parte de los clientes de su plataforma. Los desarrolladores simplemente almacenan elementos de datos y los consultan mediante solicitudes de servicios Web (en general utilizando APIs basadas en el protocolo REST o SOAP); Amazon SimpleDB se encarga del resto.

Además, Amazon ofrece también soporte para múltiples versiones de SQL Server, que otorgan primordialmente la posibilidad de integrar aplicaciones que persistan sus datos utilizando las tecnologías de Microsoft SQL para cumplir su propósito. Algunas aplicaciones que suelen hacer uso más frecuente de los motores de base de datos SQL Server, son aquellas aplicaciones desarrolladas en tecnologías .Net y PHP.

Windows Azure en este aspecto ofrece 3 tipos de almacenamiento de datos. Uno para dar soporte a SQL Relacional, que permite que las aplicaciones ya desarrolladas se puedan adaptar y migrar fácilmente a la nube sin necesidad de modificar sus capas de acceso a datos (por los conectores y consultas), y tampoco modificar el modelo de datos de la base de datos ya disponible.

Para las aplicaciones nuevas o aquellas que deseen aplicar técnicas de reingeniería para aprovechar los beneficios de las nuevas tecnologías tales como los productos NoSQL, Azure ofrece almacenes de tablas NoSQL permitiendo el almacenamiento de grandes cantidades de datos no estructurados, que se pueden escalar automáticamente para satisfacer un rendimiento y volumen masivos de hasta 100 terabytes, accesibles prácticamente desde cualquier lugar a través de REST y las API administradas. La última opción que ofrece Microsoft Windows Azure para almacenamiento consiste en Blobs no estructurados, que otorga la posibilidad de almacenar grandes cantidades de texto no estructurado o datos binarios tales como vídeo, audio e imágenes.

Google App Engine por su parte sugiere una única alternativa para dar solución a la persistencia de datos, consistente en una base de datos no relacional conocida como “Big Table”. Si bien Google fue uno de los proveedores pioneros en esta tecnología, su mercado competitivo ha avanzado a pasos agigantados y todos sus proveedores competidores de servicios cloud ofrecen actualmente muchas más alternativas para dar solución a la persistencia de datos. En consecuencia, Google App

Engine no soporta bases de datos relacionales, lo cual dificulta y obstaculiza la migración de aplicaciones existentes tradicionales a su plataforma.

Para la plataforma OpenShift, por estar basada y pensada para aplicaciones que corren sobre sistemas operativos Linux (más particularmente sobre distribuciones de Red Hat Linux), no ofrece la posibilidad de persistir datos en bases de datos SQL Server de Microsoft, pero sí ofrece otras alternativas de persistencia de datos relacionales basadas en los populares motores MySQL, PostgreSQL y SQLite. Además, OpenShift ofrece también la posibilidad de persistencia de datos en un motor de base de datos NoSQL muy popular del mercado (MongoDB) que es una base de datos Open Source de tipo documental con documentos de estilo JSON y esquemas dinámicos.

La oferta de IBM SmartCloud para el almacenamiento de datos es amplia. Otorga la posibilidad de utilizar almacenes de bases de datos relacionales, como el motor IBM DB2, Oracle, Microsoft SQL Server, Informix y Sybase. Con respecto a los productos NoSQL, SmartCloud implementa almacenes de datos basados en productos muy populares como por ejemplo Hadoop.

En lo que al almacenamiento de datos se refiere, VMware ha optado la estrategia de apuntar a los motores de persistencia tradicionales que mayor segmento del mercado actual ocupan, ofreciendo soporte nativo para bases de datos relacionales tales como Oracle, Microsoft SQL Server, y PostgreSQL. Además, es posible persistir datos con otros productos que integran la suite de VMware vFabric, tales como los almacenes de datos propietarios de su datagrid GemFire. En adición, para la línea de tecnologías de BigData, VMware brinda soporte para múltiples distribuciones de Hadoop que permiten obtener los beneficios de NoSQL.

Por último, OpenStack brinda diferentes posibilidades de almacenamiento de datos, tales como Object Storage (persistencia de objetos implementada por el producto Swift), Block Storage (persistencia de bloques implementada con el producto Cinder) y también brinda soporte para distribuciones de bases de datos relacionales MySQL, tales como Nova, Glance, Cinder y Keystone.

3.4.5 Soporte para Colas y Servidores Web

En líneas generales, todos los proveedores de servicios cloud ofrecen un único producto para implementar técnicas de Colas.

En el caso de la plataforma Amazon EC2, cuenta con un producto propietario cuyo nombre comercial es Amazon Simple Queue Service.

Google App Engine también ofrece un producto propietario comercializado como App Engine Task Queue.

OpenShift implementa soluciones para colas con IronMQ que es un producto de colas pensado para aplicaciones que corren en la nube, el cual basa sus comunicaciones en los protocolos HTTP/Rest, brindando además soporte para JSON.

IBM SmartCloud ofrece un producto de colas propietario con el cual ya contaba en su suite WebSphere, y que es comercializado como WebSphere Message Broker.

VMware y OpenStack, en cambio, proponen como alternativa para dar solución a las colas el producto Open Source popularizado bajo el nombre de RabbitMQ, que se basa en el protocolo estándar AMQP, que provee además APIs para Java y .Net.

Algo similar ocurre con la estrategia elegida por los proveedores para dar solución a las necesidades de servidores Web: la mayoría de los proveedores ofrecen una única alternativa para publicar aplicaciones web, tales son el caso de Google App Engine, con Jetty Web Server, Red Hat OpenShift con Apache Server, o IBM SmartCloud con WebSphere Application Server.

Otros casos como Amazon EC2 y Microsoft Windows Azure ofrecen al menos dos alternativas para dar soporte a las aplicaciones web, y esto se deriva de que estas plataformas soportan múltiples lenguajes, algunos de los cuales que no pueden compatibilizar sus ejecuciones en los mismos servidores Web, como por ejemplo aplicaciones Web de Microsoft (Asp.net) que requieren el servidor web Internet Information Server, y aplicaciones web Java, que requieren Servidores de tipo Apache/Tomcat.

3.4.6 Alternativas de Hipervisores

Esta característica es crucial y determinante para el modelo de negocio ofrecido por los proveedores de servicios cloud, puesto que en función de las alternativas de virtualización que estos ofrecen, se deriva la facilidad de portabilidad de máquinas virtuales que contienen las aplicaciones ya existentes en los datacenters (on premise) de sus potenciales clientes a sus entornos Cloud. En muchos casos, aplicar reingeniería para migrar las aplicaciones o instalarlas y adaptarlas en nuevas plataformas puede demandar mucho tiempo y resultar costoso en extremo. De allí se desprende la relevancia de esta característica.

Amazon EC2, al igual que Google App Engine y OpenShift, utiliza hipervisores basados en XEN y LXC (Linux Containers)

IBM SmartCloud, en cambio, ofrece muy buenas capacidades de virtualización, soportando múltiples hipervisores que van desde VMware, Hyper-V hasta otros basados en XEN.

Windows Azure, por su parte, trabaja con Windows Azure hipervisor, que se trata de una versión de Hyper-V (el conocido y tradicional hipervisor de Microsoft) ajustada y optimizada para la

virtualización en la nube. Adicionalmente, Microsoft permite (como se mencionó con anterioridad) la posibilidad de migrar máquinas virtuales de VMware a formatos aceptados por este virtualizador, de manera que brinda también una alternativa de compatibilidad con esta tecnología.

VMware VCloud permite trabajar con hipervisores ESX, ESXi y también con el hipervisor de Microsoft Hyper-V.

Por último, OpenStack ofrece el más amplio abanico para dar soluciones a la virtualización, soportando hipervisores XEN, Hyper-V, KVM, QEMU, Contenedores Linux (LNC) y muchos otros.

3.4.7 Cache In-Memory distribuido / Datagrid

Los cache In-Memory distribuidos y los datagrids juegan un rol significativo a la hora de optimizar la performance de las aplicaciones (y más aún cuando se trata de aplicaciones que correrán en la nube, donde la escalabilidad y la performance son un punto central), puesto que sustituyen los mecanismos de persistencia y búsqueda de datos tradicionales que suelen estar basados en hardware de bajo costo, pero también de baja performance, como son los casos de los discos rígidos (storage) magnéticos. Estos mecanismos, al estar implementados sobre memorias de acceso aleatorio (del inglés Random Access memory –RAM-) son en general inclusive más veloces que las unidades de estado sólido para obtener y persistir datos.

En el caso de Amazon EC2, ofrece gran flexibilidad otorgando una alternativa abierta para la implementación de productos de Cache In-Memory y datagrids, pudiendo mencionarse el soporte de productos tales como GemFire, Oracle Coherence, Gigaspaces XAP, Hazelcast, etc.

En el caso de Windows Azure, ofrece principalmente dos alternativas para dar solución al acceso rápido a los datos: Memcached, el cual se trata de un producto Cache OpenSource de tipo Clave valor (del inglés Key-value) muy popular para caching distribuido, con gran adherencia en el mercado (algunos de los clientes que usan este producto son Facebook, Twitter, Wikipedia, YouTube, WordPress, etc.). Este producto tiene sus propios protocolos (inicialmente trabajaba únicamente con un protocolo de tipo texto y las últimas versiones del producto implementan un nuevo protocolo de tipo binario, optimizando así el desempeño del mismo. Para hacer uso de este producto, las aplicaciones deben implementar técnicas de caching de datos en sus capas de acceso a datos, es decir que su implementación y uso en Microsoft Azure no resulta transparente para las aplicaciones que requieran optimizar su desempeño por este camino.

La segunda opción que ofrece Azure para caching distribuido de datos es Windows Azure Caching, que se trata de un producto propietario de Microsoft y que además es compatible con el protocolo de Memcached, de manera que las aplicaciones que ya implementaban mecanismos de cache basados en el protocolo de memcached en sus capas de acceso a datos puedan comenzar a hacer uso de este producto minimizando el costo de adaptación de tecnología.

Google App Engine también brinda la posibilidad de utilizar Memcached para optimizar la performance de las aplicaciones que corren en esta plataforma.

Con respecto a OpenShift, Red Hat cuenta con su propio producto de cache distribuido conocido por el nombre comercial Infinispan, que ya existía antes de que se iniciara la era cloud, y formaba parte de la suite de productos de JBoss. Este cache es también de tipo Key-value y brinda soporte transaccional, con el adicional de soporte para NoSQL.

En el caso de IBM SmartCloud, su estrategia de cache distribuida está basada en un producto propietario comercializado como WebSphere eXtreme Scale, que se puede utilizar tanto en clouds privados como públicos, obteniendo una gran mejora de performance para las aplicaciones.

VMware para dar solución a la necesidad de contar con un producto de grid de datos distribuidos, y no perder competencia de mercado con los otros proveedores, puesto que no contaba con productos de este tipo, adquirió GemFire y lo integró a su Suite de VFabric. GemFire es un potente cache que permite distribuir la carga y procesamiento de datos en múltiples nodos, en propósitos de optimizar el rendimiento, permitiendo transaccionar de manera asíncrona con su propia base de datos, o con cualquier otra base de datos (por ejemplo SQL Server, Oracle, MySQL, etc.). Además, GemFire tiene la particularidad que puede trabajar inclusive con nodos que pueden encontrarse distribuidos en diferentes datacenters. Este mecanismo lo hace especialmente atractivo brindando modelos de alta flexibilidad y performance para centros de recuperación del desastre (del inglés “Disaster recovery” datacenters).

4. REFERENCIAS

- Amazon.com Inc. 2013a. Amazon Elastic Compute Cloud (Amazon EC2). <http://aws.amazon.com/es/ec2/#functionality>. Página vigente al 12/05/2013
- Amazon.com Inc. 2013b. Auto Scaling. <http://aws.amazon.com/es/autoscaling/>. Página vigente al 12/05/2013
- Amazon.com Inc. 2013c. Amazon Simple Storage Service (Amazon S3). <http://aws.amazon.com/es/s3/>. Página vigente al 20/05/2013
- Amazon.com Inc. 2013d. Amazon Relational DB Service (Amazon RDS). <http://aws.amazon.com/es/rds/>. Página vigente al 20/05/2013
- Amazon.com Inc. 2013e. Amazon SimpleDB (Beta). <http://aws.amazon.com/es/simpledb/>. Página vigente al 20/05/2013
- Amazon.com Inc. 2013f. Amazon Amazon Simple Queue Service (Amazon SQS). <http://aws.amazon.com/es/sqs/>. Página vigente al 20/05/2013
- Amazon.com Inc. 2013g. Overview of Security Processes. The Hipervisor. <http://aws.amazon.com/articles/1697>. Página vigente al 21/05/2013
- Amazon.com Inc. 2013h. Amazon Cloud Watch. <http://aws.amazon.com/es/cloudwatch/>. Página vigente al 21/05/2013
- Amazon.com Inc. 2013i. Amazon Machine Images (AMIs). <https://aws.amazon.com/amis/>. Página vigente al 21/05/2013
- Amazon.com Inc. 2013j. Amazon EC2 con Microsoft Windows Server y SQL Server. <http://aws.amazon.com/es/windows/>. Página vigente al 21/05/2013
- Amazon.com Inc. 2013k. AMI de Amazon Linux. <http://aws.amazon.com/es/amazon-linux-ami/>. Página vigente al 21/05/2013
- Carter, S. 2007. The New Language of Business. SOA & Web 2.0. Editorial IBM Press, Pearson plc. ISBN-10: 0-13-195654-X

- Dasari A., Snitzer B, 2010. IBM on the Cloud: What do you want to know? Answering some of the frequently asked questions about
- Douglas, B. 2003. Web Services and Service-Oriented Architecture: The Savvy Manager's Guide. Editorial Morgan Kaufmann. ISBN-10: 1-55-860906-7
- Erl, T. 2004. Service-Oriented Architecture. A Field Guide to Integrating XML and Web Services. Editorial Techne group (Prentice Hall). ISBN-10: 0-13-142898-5
- Erl, T. 2005. Service-Oriented Architecture: Concepts, Technology, and Design. Editorial Prentice Hall PTR. ISBN-10: 0-13-185858-0
- GigaSpaces Inc. GigaSpaces XAP – Your Key to Application Performance and Low Latency through our In-Memory Data Grid Technology. <http://www.gigaspaces.com/datagrid>. Página vigente al 12/05/2013
- Google Inc. 2012a. Why App Engine. <https://developers.google.com/appengine/whyappengine>. Página vigente al 12/05/2013
- Google Inc. 2012b. Google App Engine – the platform for your next great idea. <https://cloud.google.com/files/GoogleAppEngine.pdf>. Página vigente al 22/05/2013.
- Google Inc. 2012c. What Is Google App Engine?. <https://developers.google.com/appengine/docs/whatisgoogleappengine>. Página vigente al 12/05/2013
- Google Inc. 2012d. Using the Datastore. <https://developers.google.com/appengine/docs/java/gettingstarted/usingdatastore>. Página vigente al 22/05/2013
- Google Inc. Google App Engine 2013f. Tareas programadas y Colas de tareas. <https://developers.google.com/appengine/docs/whatisgoogleappengine?hl=es>. Página vigente al 22/05/2013
- Hasan J., Duran M. 2006. Expert Service-Oriented Architecture in C# 2005, second edition. Editorial Apress. ISBN 1-59059-701-X
- Hansen, M. 2007. SOA Using Java™ Web Services. Editorial Prentice Hall. ISBN-10: 0-13-04496-8-7

- IBM Corp. 2011a. IBM SmartCloud Enterprise. <http://www.ibm.com/developerworks/cloud/library/cl-cloudfaq.html>. Página vigente al 12/05/2013
- IBM Corp. 2011b. IBM SmartCloud Application Services (Beta). http://www-05.ibm.com/it/cloud/assets/IBMSmartCloud_Application_Services_FAQ.pdf. Página vigente al 12/05/2013
- IBM Corp. 2011c. Deploy an app to the cloud at lunch hour and still have time to eat. <http://www.ibm.com/cloud-computing/us/en/paas.html>. Página vigente al 12/05/2013
- IBM Corp. 2011d. IBM SmartCloud Enterprise. Enterprise class cloud managed infrastructure. <http://www-935.ibm.com/services/us/en/managed-cloud-hosting/>. Página vigente al 15/06/2013
- IBM Corp. 2011e. IBM Scale Out Network Attached Storage. <http://www-03.ibm.com/systems/storage/network/sonas/>. Página vigente al 15/06/2013.
- IBM Corp. 2013a. IBM SmartCloud Provisioning. Increase business agility and optimize virtualization with the workload-optimized cloud. <http://public.dhe.ibm.com/common/ssi/ecm/en/ibd03003usen/IBD03003USEN.PDF>. Página vigente al 15/06/2013
- IBM Corp. 2013b. IBM Storwize V7000 and Storwize V7000 Unified Disk Systems. http://www03.ibm.com/systems/storage/disk/storwize_v7000/. Página vigente al 16/06/2013.
- IBM Corp. 2013c. IBM XIV Storage System. <http://www-03.ibm.com/systems/storage/disk/xiv/>. Página vigente al 16/06/2013.
- IBM Corp. 2013d. IBM XIV Storage System. <http://www-03.ibm.com/systems/storage/disk/xiv/overview.html>. Página vigente al 16/06/2013.
- IBM Corp. 2013e. IBM Integration Bus. <http://www-03.ibm.com/software/products/us/en/integration-bus/>. Página vigente al 16/06/2013.
- IBM Corp. 2013f. IBM Systems Director. <http://www-03.ibm.com/systems/software/director/index.html>. Página vigente al 16/06/2013.

- IBM Corp. 2013g. IBM Systems Director. <http://www-03.ibm.com/systems/software/director/vmcontrol/index.html>. Página vigente al 16/06/2013.
- IBM Corp. 2013h. WebSphere eXtreme Scale. <http://www-03.ibm.com/software/products/us/en/websphere-extreme-scale/>. Página vigente al 16/06/2013.
- IronMQ. 2013a. IronMQ, The Message Queue for the Cloud. <http://www.iron.io/mq>. Página vigente al 16/06/2013.
- Josuttis, N. 2007. SOA in Practice, The Art of Distributed System Design. Editorial O'Reilly. ISBN-10: 0-596-52955-4
- Krishnan, S. 2010. Programming Windows Azure. Editorial O'Reilly Media. ISBN-10: 0-59-680197-1
- Mattern, T., Woods, D. 2006. Enterprise SOA: Designing IT for Business Innovation. Editorial O'Reilly. ISBN-10: 0-596-10238-0
- Microsoft Corp. 2013a, Windows Azure. Qué es Windows Azure. <http://www.windowsazure.com/es-es/home/features/what-is-windows-azure/>. Página vigente al 12/05/2013
- Microsoft Corp. 2013b, Windows Azure. How to Use the Autoscaling Application Block. <http://www.windowsazure.com/en-us/develop/net/how-to-guides/autoscaling/>. Página vigente al 12/05/2013
- Microsoft Corp. 2013c, Windows Azure. Mensajería. <http://www.windowsazure.com/es-es/home/features/messaging/>. Página vigente al 12/05/2013
- Microsoft Corp. 2013d, Windows Azure. Máquinas Virtuales. <http://www.windowsazure.com/es-es/home/features/virtual-machines/>. Página vigente al 21/05/2013
- Microsoft Corp. 2013e, Windows Azure. Base de datos SQL. <http://www.windowsazure.com/es-es/pricing/details/sql-database/>. Página vigente al 21/05/2013
- Microsoft Corp. 2013f, Windows Azure. Administración de datos. <http://www.windowsazure.com/es-es/home/features/data-management/>. Página vigente al 21/05/2013

- Microsoft Corp. 2013g, Windows Azure. What are Service Bus Queues. <http://www.windowsazure.com/en-us/develop/net/how-to-guides/service-bus-queues/>. Página vigente al 21/05/2013
- Microsoft Corp. 2013h, Windows Azure. HDInsight. <http://www.windowsazure.com/en-us/manage/services/hdinsight/>. Página vigente al 21/05/2013
- MongoDB 2013a. Mongo DB. <http://www.mongodb.org/>. Página vigente al 18/06/2013
- OpenStack 2013a, Infraestructura de Cómputo (Nova). http://docs.openstack.org/diablo/openstack-compute/starter/content/Open_Stack_Compute_Infrastructure_Nova_-d1e124.html. Página vigente al 18/06/2013
- OpenStack 2013b, Infraestructura de Cómputo (Nova). Funciones y características. http://docs.openstack.org/diablo/openstack-compute/starter/content/Functions_and_Features-d1e132.html. Página vigente al 18/06/2013
- OpenStack 2013c, Infraestructura de Almacenamiento (Swift). http://docs.openstack.org/diablo/openstack-compute/starter/content/OpenStack_Storage_Infrastructure_Swift_-d1e271.html. Página vigente al 18/06/2013
- OpenStack 2013d, Infraestructura de Almacenamiento (Swift). Funciones y características. http://docs.openstack.org/diablo/openstack-compute/starter/content/Functions_and_Features-d1e279.html. Página vigente al 18/06/2013
- OpenStack 2013e, Servicios de Imagen (Glance). http://docs.openstack.org/diablo/openstack-compute/starter/content/OpenStack_Imaging_Service_Glance_-d1e329.html. Página vigente al 18/06/2013
- OpenStack 2013f, Servicios de Imagen (Glance). Funciones y características. http://docs.openstack.org/diablo/openstack-compute/starter/content/Functions_and_Features_Glance_-d1e352.html. Página vigente al 18/06/2013
- OpenStack 2013g, Tool support for image creation. http://docs.openstack.org/trunk/openstack-image/content/ch_creating_images_automatically.html. Página vigente al 18/06/2013

- OpenStack 2013h. Hyper-V Virtualization Platform. <http://docs.openstack.org/trunk/openstack-compute/admin/content/hyper-v-virtualization-platform.html>. Página vigente al 18/06/2013.
- OpenStack 2013i. What is RDO?. http://openstack.redhat.com/Frequently_Asked_Questions#What_is_RDO.3F. Página vigente al 18/06/2013.
- OpenStack 2013j. OpenStack Storage. <http://www.openstack.org/software/openstack-storage/>. Página vigente al 18/06/2013.
- OpenStack 2013k. Message Queue (Rabbit MQ Server). http://docs.openstack.org/essex/openstack-compute/starter/content/Message_Queue_Rabbit_MQ_Server_-d1e223.html. Página vigente al 18/06/2013.
- OpenStack 2013l. Message Queue (Rabbit MQ Server). Features. <http://www.rabbitmq.com/features.html>. Página vigente al 18/06/2013.
- OpenStack 2013m. Selecting a Hypervisor. <http://docs.openstack.org/trunk/openstack-compute/admin/content/selecting-a-hypervisor.html>. Página vigente al 18/06/2013.
- OpenStack 2013n. Selecting a Hypervisor. <https://wiki.openstack.org/wiki/Heat>. Página vigente al 18/06/2013.
- OpenStack 2013o. OpenStack. <http://docs.openstack.org/trunk/openstack-compute/starter/content/OpenStack-d1e94.html>. Página vigente al 18/06/2013.
- Pulier, E., Taylor, H. 2006. Understanding Enterprise SOA. Editorial Manning. ISBN 1-932394-59-1
- Red Hat Inc. 2013a. Open Shift. Open Shift All Versions User Guide. https://access.redhat.com/site/documentation/en-US/OpenShift/2.0/pdf/User_Guide/OpenShift-2.0-User_Guide-en-US.pdf. Página vigente al 25/05/2013
- Red Hat Inc. 2013b. Open Shift. Enterprise Features and Benefits. <https://www.openshift.com/enterprise-paas/features-and-benefits>. Página vigente al 25/05/2013
- Red Hat Inc. 2013c. Open Shift. Scale Your Applications on the web. <https://www.openshift.com/developers/scaling>. Página vigente al 25/05/2013

- Red Hat Inc. 2013d. Open Shift. How to Install OpenShift Enterprise PaaS. <https://www.openshift.com/blogs/installing-enterprise-paas-part-1>. Página vigente al 29/05/2013
- Red Hat Inc. 2013e. Open Shift. New OpenShift Cartridge Format Part1. <https://www.openshift.com/blogs/new-openshift-cartridge-format-part-1>. Página vigente al 29/05/2013
- Red Hat Inc. 2013f. Open Shift. New OpenShift Cartridge Format Part2. <https://www.openshift.com/blogs/new-openshift-cartridge-format-part-2>. Página vigente al 29/05/2013
- Red Hat Inc. 2013g. Open Shift. OpenShift Online User guide. https://access.redhat.com/site/documentation/en-US/OpenShift/2.0/html-single/User_Guide/index.html. Página vigente al 14/06/2013
- Red Hat Inc. 2013h, Managing Databases in the Cloud. <https://www.openshift.com/blogs/manipulate-your-paas-database>. Página vigente al 18/06/2013
- Red Hat Inc. 2013i, IronMQ on OpenShift. <https://www.openshift.com/quickstarts/ironmq-on-openshift>. Página vigente al 18/06/2013
- Red Hat Inc. 2013j. Build Your Own Paas from the OpenShift Origin LiveCD using liveinst. <https://www.openshift.com/wiki/build-your-own-paas-from-the-openshift-origin-livecd-using-liveinst>. Página vigente al 18/06/2013
- SQLite 2013a. SQLite. <http://www.sqlite.org/>. Página vigente al 18/06/2013.
- VMware Inc. 2010a. Storage Considerations for VMware vCloud Director. Pooling of Storage Resources. http://www.vmware.com/files/pdf/techpaper/VMW_10Q3_WP_vCloud_Director_Storage.pdf. Página vigente al 16/06/2013.
- VMware Inc. 2013a. VMware vFabric RabbitMQ™. MESSAGING THAT JUST WORKS. <http://www.vmware.com/products/application-platform/vfabric-rabbitmq.html>. Página vigente al 12/05/2013
- VMware Inc. 2013b. vFabric Team. Choosing Your Messaging Protocol: AMQP, MQTT, or STOMP. <http://blogs.vmware.com/vfabric/2013/02/choosing-your-messaging-protocol-amqp-mqtt-or-stomp.html>. Página vigente al 12/05/2013

- VMware Inc. 2013c. VMware Data Director. <https://www.vmware.com/products/applicationplatform/vfabric-data-director/features.html>. Página vigente al 12/05/2013
- VMware Inc. 2013d. VMware vCloud Suite. <http://www.vmware.com/es/products/datacenter-virtualization/vcloud-suite/how-it-works.html>. Página vigente al 16/06/2013
- VMware Inc. 2013e. VMware vCloud Suite. <http://www.vmware.com/files/es/pdf/products/vCloud/VMware-vCloud-Suite-Datasheet.pdf>. Página vigente al 16/06/2013
- VMware Inc. 2013f. Soluciones para computación en nube. <http://www.vmware.com/mx/cloud-computing.html>. Página vigente al 16/06/2013
- VMware Inc. 2013g. VMware vCenter Multi-Hypervisor Manager 1.1 Release Notes. <http://www.vmware.com/support/mhm/doc/vcenter-multi-hypervisor-manager-11-release-notes.html#capabilities>. Página vigente al 16/06/2013
- VMware Inc. 2013h. Introduction to the VMware vCenter Orchestrator AMQP Plug-In. <http://www.vmware.com/support/orchestrator/doc/amqp-plugin-101-release-notes.html>. Página vigente al 16/06/2013.