
Cloud as a service

PID_00247336

Màrius Montón Macián

Tiempo mínimo de dedicación recomendado: 1 hora



Universitat
Oberta
de Catalunya

Ninguna parte de esta publicación, incluido el diseño general y la cubierta, puede ser copiada, reproducida, almacenada o transmitida de ninguna forma, ni por ningún medio, sea éste eléctrico, químico, mecánico, óptico, grabación, fotocopia, o cualquier otro, sin la previa autorización escrita de los titulares del copyright.

Índice

Introducción	5
1. <i>Cloud computing</i> en la actualidad	6
2. <i>Infrastructure as a service (IaaS)</i>	7
2.1. Niveles de redundancia	8
2.2. Cambios en el software	8
3. <i>Platform as a service (PaaS)</i>	11
4. <i>Software as a service (SaaS)</i>	13
4.1. Arquitectura	14
4.2. Discusión	15
5. <i>Service-level agreement (SLA)</i>	16
Bibliografía	18

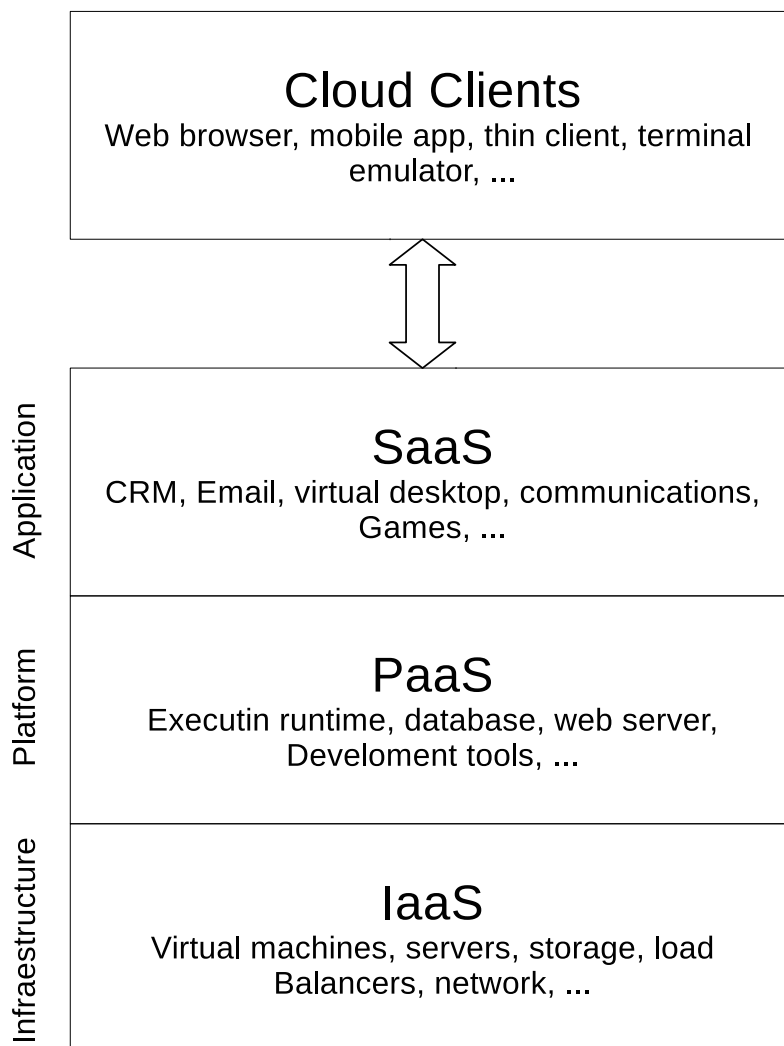
Introducción

En este contenido, se presentan los distintos modos de comercialización del *Cloud* actualmente disponibles.

1. Cloud computing en la actualidad

Actualmente, los servicios de *cloud computing* están agrupados en distintos niveles, según los servicios disponibles para los clientes. De este modo, habitualmente se habla de tres niveles (véase figura 1), empezando por el más bajo, donde el cliente obtiene acceso a las máquinas, discos y elementos de red (IaaS, véase el apartado 2); en el segundo nivel, el cliente tiene acceso a algún tipo de software que le permite configurar o desarrollar su propia aplicación (PaaS, véase el apartado 3); y en el último nivel, el cliente tiene acceso a un software ya desarrollado (SaaS, véase el apartado 4). Veremos con más detalle cada uno de estos niveles. Por último, se comentan los tipos de contratos de servicio entre proveedores y clientes (SLAs, véase el apartado 5)

Figura 1. Distintos servicios ofrecidos en el *Cloud*.



2. Infrastructure as a service (IaaS)

Uno de los servicios prestados por el *Cloud* es del *Infrastructure as a service* (IaaS), en el que los proveedores de servicios ofrecen una serie de recursos para que una empresa pueda montarse su propia infraestructura informática en la nube, a diferencia de servicios PaaS o SaaS que ofrecen soluciones ya instaladas y preparadas.

Cuando se habla de IaaS, pues, el servicio que contrata el cliente es la infraestructura necesaria para hacer funcionar un sistema informático, esto es, las máquinas, servidores, dispositivos de red (*routers*, *firewalls*, etc.), servidores de disco, etc., y el propio cliente (o alguna compañía auxiliar) configura, prepara e instala todo el software necesario para su aplicación (bases de datos, sistemas operativos, etc.).

Habitualmente, estos servicios tienen costes según la cantidad de recursos usados en cada momento, ya que por lo general se contrata un número básico de recursos y se incrementan de forma flexible según la demanda en cada momento.

Un ejemplo de IaaS son los servicios ofrecidos por la compañía Amazon bajo la marca *Amazon web services* (AWS). AWS es un conjunto de servicios de *Cloud*, que nos ofrecen servidores virtuales, bases de datos, redes de interconexión, almacenamiento en disco, etc. Estos servicios se ofrecen sin configurar ni estar listos para ser usados, con lo que son necesarios técnicos preparados y especializados para poder manejar y convertir todos los servicios contratados en una plataforma capaz de desempeñar la función precisa.

Este nivel sería el más cercano al modelo clásico de infraestructura informática, solo que en lugar de tener los servidores físicamente dentro de la compañía, se encuentran fuera y no hay propiedad de los mismos, sino una especie de alquiler.

Otra diferencia es la velocidad de adquisición de nueva infraestructura, ya que en el caso de un proveedor de IaaS, con tan solo pedir una nueva máquina o servicio el proveedor nos lo va a ofrecer en un periodo de tiempo muy breve (del orden de segundos o minutos). Esto contrasta con los tiempos (al menos días) de entrega de adquisición de material físico.

Además, cabe la posibilidad de tener automatizada la tarea de ajustar el tamaño y la capacidad de la infraestructura (capacidad de proceso, espacio de almacenamiento, etc.), según las necesidades de cada momento. Esto permite ajustar los costes de la infraestructura sin perder calidad en el servicio.

Amazon web services

Lo que empezó siendo una tienda por internet (Amazon.com) ha acabado siendo un proveedor de servicios *Cloud* muy importante. De hecho, actualmente la tienda Amazon.com funciona sobre AWS.

2.1. Niveles de redundancia

Debido a lo cruciales e importantes que son los datos almacenados y los servicios prestados por estas infraestructuras, el mantenimiento y cuidado de toda la infraestructura para evitar o minimizar el número y duración de los fallos es un aspecto crucial. Asimismo, garantizar la seguridad de los datos, tanto contra fallos catastróficos como contra accesos no autorizados, es la otra gran preocupación en cuanto a la operativa de una infraestructura informática.

Los proveedores de IaaS suelen clasificarse según la seguridad y redundancia de sus equipos. Los parámetros que mide esta clasificación permiten tener un buen indicador de la redundancia y la disponibilidad de la instalación. Así, un *data center* de nivel 1 tendrá solo una conexión a internet, un único proveedor de energía eléctrica y ninguna redundancia en la refrigeración del equipo; mientras que uno de calidad 4 tendrá redundancia de dos o más en todos los puntos anteriores (entre otros muchos).

Hay varias de estas clasificaciones. Las más usuales son la propuesta por ANSI (ANSI/TIA-942-A [ANSI (2012)]) y la propuesta por el Uptime Institute [Uptime Institute (2012)]. En los dos casos, los niveles (o *tiers*) empiezan en el 1, más básico, y llegan al 4, que es el que proporciona más redundancia y seguridad.

La redundancia especificada es la de los equipos de alimentación eléctrica y sistemas de alimentación ininterrumpida (UPS), equipos de refrigeración, componentes de las redes internas, cableado, suministro de electricidad (de varias compañías), equipo anti-inundaciones, etc. (véase tabla 1).

Tabla 1. Resumen de la redundancia requerida para cada nivel.

Nivel o Tier	Grado de redundancia	Tiempo fuera de línea anual	Cantidad 9 s
1	No hay	28,8 horas	99,671 %
2	Redundancia simple, con UPS y única fuente de energía	22,0 horas	99,741 %
3	Redundancia en todos los sistemas, varias fuentes de energía	1,6 horas	99,982 %
4	Múltiple redundancia en todos los sistemas, puede funcionar hasta 4 días sin alimentación externa.	0,4 horas	99,995 %

Elaboración propia a partir de [ANSI(2012)] y [ADC Telecommunications, Inc.(2006)]

Según el nivel de redundancia del *data center*, los precios de sus servicios varían considerablemente.

2.2. Cambios en el software

Si bien hasta ahora hemos hablado del IaaS como un cambio en el uso del hardware en los sistemas informáticos, el otro gran cambio que ha ido de la mano ha sido el de los sistemas de gestión de datos de software. Junto con el cambio de uso de los siste-

Standard ANSI/TIA-942-A

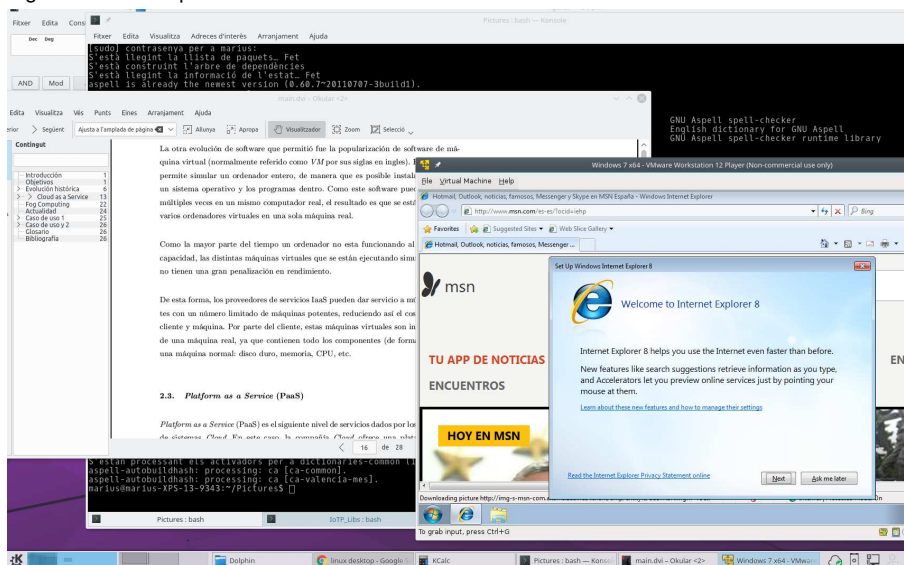
Estas especificaciones llegan, incluso, a cómo deben construirse los edificios, puertas de entrada, control de acceso, etc.

mas informáticos, cuando paulatinamente los costes de almacenamiento y procesado fueron disminuyéndose, aparecieron nuevas estrategias para la gestión de datos.

Una de las evoluciones más importantes que permitieron la popularización del *Cloud* fue la del software de máquina virtual (normalmente referido como VM (*virtual machine*) por sus siglas en inglés). Este software permite simular un ordenador entero, de manera que es posible instalar y ejecutar un sistema operativo y los programas dentro. Dado que este software puede ejecutarse múltiples veces en un mismo ordenador real, el resultado es que se están simulando varios ordenadores virtuales en una sola máquina real, y estas múltiples copias pueden funcionar de manera independiente una de la otra (véase figura 2).

Puesto que la mayor parte del tiempo un ordenador no funciona al 100 % de su capacidad, las distintas máquinas virtuales que se están ejecutando de manera simultánea no tienen una gran penalización en rendimiento.

Figura 2. Una máquina virtual corriendo Windows 7 en un ordenador real con Kubuntu



Elaboración propia

De esta forma, los proveedores de servicios IaaS pueden dar servicio a múltiples clientes con un número limitado de máquinas potentes, de modo que reducen el coste para cada cliente y máquina. Por parte del cliente, estas máquinas virtuales son indistinguibles de una máquina real, ya que contienen todo los componentes (de forma virtual) de una máquina normal: disco duro, memoria, CPU, etc.

Otras características interesantes de las máquinas virtuales son la migración de VM entre distintas máquinas y la elasticidad. La migración permite mover una VM de una máquina real a otra, bien para equilibrar la carga de trabajo de las máquinas reales o para cambiarla físicamente de emplazamiento. De este modo, un proveedor que detecte que una VM está sobrecargando una máquina real, debido a su cómputo extensivo, puede decidir mover esa VM a otra máquina que esté más descargada. Esta operación puede hacerse sin tener que desconectar ni apagar la VM en ningún momento,

por lo que es totalmente transparente para el cliente. La elasticidad es la característica que permite modificar los parámetros de la VM. Así, si una VM necesita más memoria RAM de la que originalmente se creó, es posible añadirle más memoria. De esta forma, se puede ajustar cada una de las VM a las necesidades de computación particulares de cada una de ellas, y reducir así los costes. Además, esto permite que si en un futuro las necesidades aumentan por el crecimiento de los requisitos de la compañía o de los clientes, se puedan ajustar de nuevo las VM responsables, de forma sencilla.

La irrupción de las VM y el abaratamiento de los costes, como ya se ha mencionado, permitieron adoptar estrategias de mejora de la computación y el tratamiento de datos que hasta el momento no se pudieron implementar por sus elevados costes.

Así, por ejemplo, se pasó de bases de datos centralizadas en un solo ordenador, de las que se tomaban copias de seguridad periódicamente, a bases de datos distribuidas en múltiples máquinas interconectadas, en las que cada máquina puede ser, dado el caso, una copia de seguridad de todos los datos. De esta forma, se gana en redundancia y en protección ante fallos, ya que el fallo de una máquina no tiene por qué poner en riesgo ni los datos ni la operativa habitual del sistema.

Así, la arquitectura, esto es, el diseño de qué servicios se usan, cómo se interconectan y cómo se balancea la carga, tuvo una revolución con la aparición y la normalización de estas tecnologías.

Por ejemplo, se pasó de un modelo de servidor monolítico, en el que el servidor de páginas web, el software de gestión de usuarios y el gestor de bases de datos están en una sola máquina real, a un modelo o arquitectura en el que cada uno de los servicios está alojado en una máquina virtual, todas ellas interconectadas. De esta manera, si por cualquier motivo es necesario ampliar el servicio para permitir, por ejemplo, más usuarios o muchos más datos, cabrá la posibilidad de mejorar la VM que sirve la base de datos o la VM que manejan los usuarios.

3. Platform as a service (PaaS)

Platform as a service (PaaS) es el siguiente nivel de servicios dados por los proveedores de sistemas *Cloud*. En este caso, la compañía *Cloud* ofrece una plataforma y un entorno que permiten desarrollar aplicaciones y servicios de forma sencilla y unificada (mismas herramientas y entorno para el mismo proveedor PaaS). Según el grado de complejidad de la aplicación, el servicio PaaS puede consistir en un conjunto de funcionalidades preconfiguradas que el usuario elige según la aplicación que quiera.

Así, las soluciones PaaS pueden ser tan complicadas que necesiten a usuarios expertos, o tan sencillas que puedan usarse solo con el ratón y haciendo clic en las opciones deseadas. En cualquier caso, toda la infraestructura necesaria está oculta al usuario final, ya que es la compañía de PaaS la que la gestiona directamente.

Un ejemplo sencillo de qué es un servicio PaaS es el software de gestión de blogs WordPress. Este software permite de forma muy sencilla, desde la misma web y usando solo el ratón, configurar el comportamiento del blog, su aspecto visual, y añadir o quitar funcionalidades por medio de *plug-ins* (un *plug-in* es un pequeño componente de software que añade alguna característica nueva a un programa ya existente).

Si usamos la plataforma web que ofrecen sus creadores (<http://wordpress.com/>), tenemos a nuestra disposición una instancia del software (un blog listo para usar solo para nosotros) y podemos cambiar todo lo comentado, sin tener que preocuparnos de la infraestructura necesaria (servidores, bases de datos, almacenamiento en discos, etc.).

Si nuestros conocimientos lo permiten, podemos escribir *plug-ins* para añadir alguna funcionalidad necesaria que no esté disponible. Lo mismo podemos decir para los temas, es decir, el aspecto visual del blog.

Otro ejemplo es el de la plataforma de Google *Google app engine*, que ofrece múltiples lenguajes de programación para el desarrollo de aplicaciones basadas en web. Esta plataforma también ofrece almacenamiento en base de datos. En este caso, el servicio es gratuito hasta cierto límite de uso, con la posibilidad de añadir recursos de forma sencilla y rápida y obtener soporte de personal especializado de la compañía.

Así pues, las empresas que ofrecen PaaS tienen multitud de servicios preparados para ser usados y configurados de forma rápida y sencilla. Esto permite, por ejemplo, preparar todo un entorno para alojar una nueva aplicación sin tener que entrar en todos los detalles de la infraestructura usada, simplemente se elige qué tipo de servidor web o base de datos se requiere y el sistema lo prepara todo en cuestión de minutos o segundos. Esta preparación «oculta» implicará, la mayoría de las veces, la creación de

Wordpress, éxito de código libre

Actualmente, Wordpress está presente en unos 69 millones de páginas, con usos distintos de los de un blog, como tienda en línea o carrusel de fotografías.

una o varias VM, su configuración y la interconexión entre ellas si fuese necesario. Una vez terminado el proceso, el cliente tiene acceso al nuevo sistema listo para ser usado.

En este tipo de servicios, la infraestructura subyacente queda escondida, de manera que es el gestor del PaaS el que debe gestionar correctamente los recursos para dar cabida a todas las necesidades de sus clientes.

Volviendo al ejemplo, no sabemos (ni falta que hace) qué cantidad de máquinas reales o virtuales están soportando actualmente el servicio Wordpress. Es la compañía que ofrece la plataforma la que se encarga de dimensionar su infraestructura *cloud* para que los usuarios del servicio de blogs no tengan problemas ni fallos en el uso diario.

4. *Software as a service (SaaS)*

La evolución en la infraestructura informática permitió la aparición de nuevos servicios, hasta llegar al *software as a service (SaaS)*.

Se entiende por SaaS todo aquel software que funciona en la nube y que de forma sencilla y flexible puede ser usado por distintos usuarios, simultáneamente y por separado.

Por tanto, en este momento aparece otro actor en la ecuación: por un lado, encontramos los proveedores de servicios *cloud* IaaS y PaaS, es decir, proveedores de la infraestructura necesaria del sistema en diferentes niveles. Por otro lado, encontramos los proveedores de software que han adaptado su software para funcionar en la nube.

Un buen ejemplo es el servicio de correo de Google GMail o Microsoft Outlook. Este servicio funciona en servidores propios de Google, de los cuales el usuario no tiene ninguna información. Estos servidores y máquinas proporcionan un servicio (en este caso, consulta y envío de correo electrónico) a múltiples usuarios a la vez y por separado: es decir, un solo servidor servirá a miles de usuarios a la vez, pero los usuarios solo ven su cuenta de correo. Al estar estos servidores conectados a internet, para los usuarios es posible consultar su correo desde cualquier punto del planeta con conexión a internet.

Hay que notar que los usuarios no tienen ningún control sobre qué está haciendo funcionar el servicio, ni las posibles mejoras o versiones, ni se deben preocupar de que el servicio esté funcionando sin problemas, o de hacer copias de seguridad de sus correos, etc. Todos estos servicios asociados están proporcionados por el proveedor del servicio principal (en este caso, Google proporcionando GMail).

A diferencia de las licencias de software usuales, que tienen un precio inicial alto y, opcionalmente, cargos por actualizaciones, los proveedores de software mediante SaaS acostumbran a usar un modelo distinto: la política de precios para SaaS suele ser de un coste inicial muy bajo o inexistente y un precio fijo mensual o anual. De esta forma, los clientes no deben afrontar un cargo alto para empezar a usar el software, sino que pagan según ciertos parámetros preestablecidos (número de usuarios utilizando la aplicación, cantidad de datos, etc.).

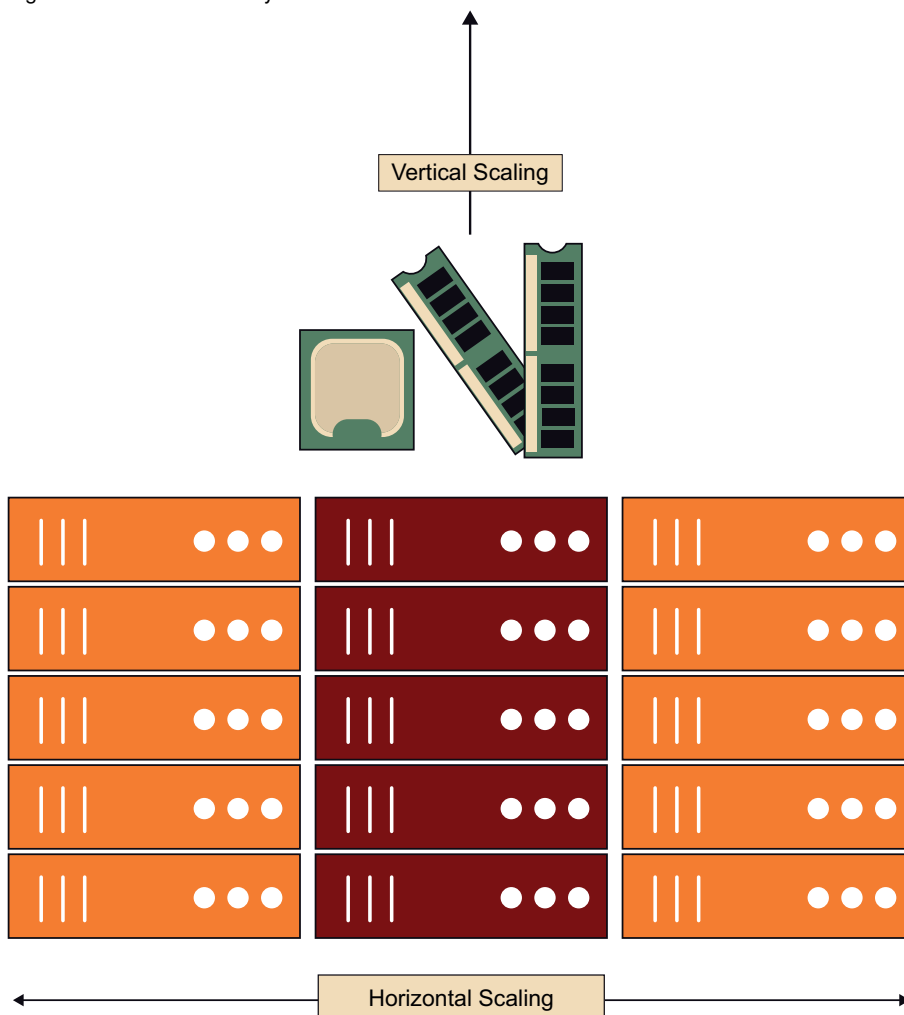
Debido a los bajos costes que supone añadir usuarios a sistemas *Cloud*, muchas compañías ofrecen suscripciones tipo *freemium*. En este modelo, se ofrece de forma gratuita una versión limitada del software, con la posibilidad de añadir ciertas mejoras o características bajo pago. Otras aplicaciones SaaS pueden ofrecerse de forma total-

mente gratuita, ya que el modelo de negocio no está en el pago de los usuarios sino en otros modelos, ya sea el de publicidad o venta de estadísticas de los usuarios y sus actos.

4.1. Arquitectura

La gran mayoría de las aplicaciones SaaS se basan en una arquitectura *multitenant* ('multiusuario', en español). En este modelo, una sola aplicación funcionando en una sola configuración (máquina, disco, etc.) es usada por varios usuarios simultáneamente y por separado. Para soportar gran cantidad de usuarios, la aplicación se instala en varias máquinas que funcionan de forma separada. De esta manera, se reparte la carga de las distintas máquinas según la carga que soportan en cada momento.

Figura 3. Escalado vertical y horizontal



Un escalado vertical es aquel en el que se mejoran las prestaciones de la infraestructura (mejor CPU, más memoria, más almacenamiento) para poder añadir más usuarios o mejorar el rendimiento. En el escalado horizontal, se replican servicios añadiendo más máquinas a la infraestructura (véase figura 3).

4.2. Discusión

El modelo SaaS de distribución de software tiene una serie de ventajas e inconvenientes que vamos a tratar a continuación:

Ventajas:

- El cliente o la compañía no necesitan personal cualificado para mantenimiento de servidores y demás infraestructura de servidores. En nuestro ejemplo, no tenemos que dedicarnos a mantener servidores, maquinaria especializada, etc.
- La responsabilidad de operación recae en la compañía de servicio, no en la propia. En el ejemplo, no tenemos que estar permanentemente pendientes de que el servicio funcione de manera correcta.
- Manejo de copias de seguridad y otras tareas de integridad de datos como seguridad de acceso, seguridad física, etc. De nuevo, no debemos preocuparnos por la seguridad de nuestros datos o correos.

Y como todo, es posible detectar un conjunto de inconvenientes:

- Pérdida del control sobre los datos. Al estar en una tercera empresa, se puede perder el control de los datos generados. En nuestro caso, quizá no podríamos recopilar todos nuestros correos para llevárnoslos a otro servicio. Las normativas europeas de protección de datos exigen que los datos estén siempre en territorio europeo; esto puede provocar problemas con algunos proveedores en otros países.
- No hay acceso al software como tal. De este modo, en caso de querer añadir o modificar alguna funcionalidad, quizá no resulte posible. En nuestro caso, nosotros no podemos modificar el funcionamiento de GMail o añadir funcionalidades que se ajusten a nuestras necesidades.
- Capturado por un solo proveedor. Esto puede ser un inconveniente si el proveedor deja de ofrecer el servicio o sus tarifas dejan de ser competitivas. En el ejemplo, si GMail dejase de ser gratuito y quisiéramos abandonarlo por otro servicio, podríamos padecer varias molestias (copia de todos los correos, cambio de dirección de correo, etc.).

Incluso con estos inconvenientes, los ahorros en costes actuales hacen que la mayoría de las compañías estén adoptando esta forma de distribución de software en lugar de las tradicionales, con servidores y desarrollo de software propios o personalizados.

5. Service-level agreement (SLA)

Los proveedores de servicios *Cloud*, ya sean de IaaS, PaaS o SaaS, normalmente ofrecen un contrato con el tipo de servicio y la calidad que aseguran en sus servicios prestados. En inglés, se conoce como *service-level agreement* (SLA). Este contrato define las reglas entre el proveedor del servicio *Cloud* y el cliente, y fija el nivel de calidad para este servicio. Puede ocurrir que el mismo proveedor *Cloud* ofrezca distintos niveles o grados de SLA, según diferentes tarifas de precios, y que deje a elección del cliente qué grado de compromiso requiere del servicio.

En plataformas *Cloud* es habitual tratar los siguientes términos (por otra parte, comunes a proyectos informáticos en general):

- ABA (*abandonment rate* o razón de abandono): porcentaje de avisos abandonados mientras esperaban recibir atención personalizada.
- ASA (*average speed to answer* o tiempo medio de atención): tiempo medio empleado por la compañía proveedora del servicio para que responda al aviso.
- TSF (*time service factor* o factor del tiempo de servicio): porcentaje de avisos respondidos en un plazo de tiempo determinado, por ejemplo, el 90 % en 10 minutos.
- FCR (*first call resolution* o resolución en el primer aviso): porcentaje de avisos recibidos que pudieron ser resueltos sin necesidad de un segundo aviso.
- TAT (*turn around time* o tiempo de respuesta): tiempo utilizado para completar una tarea determinada.
- MTTR (*mean time to recover* o tiempo medio de recuperación): tiempo medio de recuperación tras un fallo del servicio.
- MTBF (*mean time between failures* o tiempo medio entre fallos: tiempo medio entre un fallo y otro del sistema.
- *Uptime* o tiempo de funcionamiento: tiempo en el que el servicio está disponible y funcionando perfectamente. Normalmente, se mide en % de tiempo.
- Cambios: respuesta del proveedor a la necesidad de introducir algún cambio en el software (actualizar versiones, resolución de *bugs*, etc.).

Para poder cumplir con estos compromisos, las compañías de servicio (o las empresas especializadas en monitorización de servicios) suelen monitorizar y controlar todo el sistema *Industria 4.0* (o cualquier otro sistema informático) con un servicio denominado 24/7, es decir, vigilado por personal especializado las 24 horas del día y los 7 días de la semana, para así poder responder rápidamente a las incidencias que puedan surgir en cualquier momento.

Es evidente que restricciones más severas (mayor tiempo de *uptime*, menor tiempo de respuesta a incidencias, etc.) harán que el coste del servicio se incremente, y pueda llegar a costar miles de euros al año para casos de sistemas críticos.

Es evidente, pero hay que recalcarlo, que el eslabón más débil de la cadena de SLA, es decir, el SLA más laxo que tengamos de todos nuestros proveedores, es el que va a marcar el SLA global de toda nuestra infraestructura informática.

Caída de AWS de febrero del 2017

El 28 de febrero del 2017, AWS de Amazon tuvo un fallo que provocó que estuviera inoperativo cerca de 5 horas. Este fallo en el servicio provocó que múltiples servicios de internet no pudieran funcionar o lo hicieran con muchos problemas durante ese tiempo. La compañía dio como explicación que un fallo humano fue el que provocó el incidente.

Bibliografía

ADC Telecommunications, Inc. (2006). «TIA-942:Data Center Standards Overview». Report técnico 102264AE, ADC Telecommunications, Inc.

ANSI (2012). «TELECOMMUNICATIONS INFRASTRUCTURE STANDARD FOR DATA CENTERS». Report técnico ANSI-TIA-942-A, American National Standards Institute.

Uptime Institute (2012). «Uptime Institute». URL: <https://uptimeinstitute.com/>.