

# Desarrollo de software en sistemas distribuidos

## Curso 2020



# Desarrollo de software en sistemas distribuidos

## Cloud Computing

# [Conceptos generales]

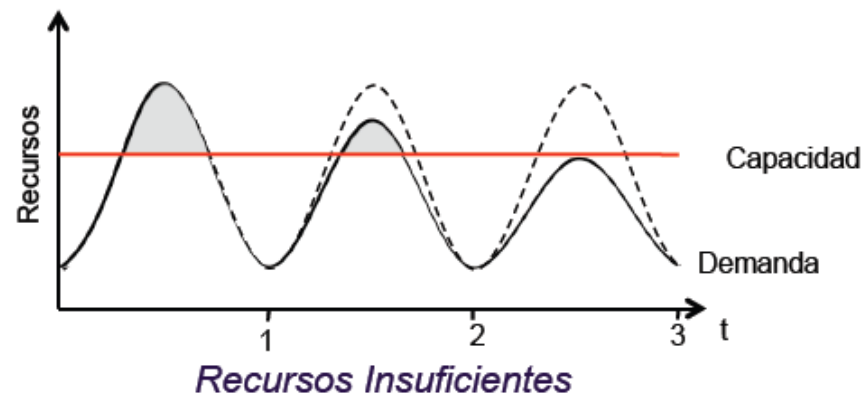
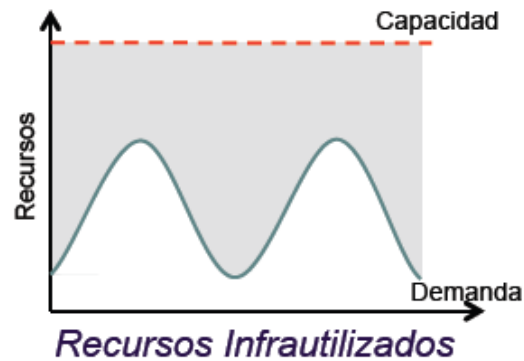
- La Computación Cloud es un paradigma que posibilita el acceso ubicuo bajo demanda a servicios TIC accesibles a través de Internet.
- El término cloud (nube) se refiere a la forma de representar la red (Internet) en los diagramas y es una abstracción de las complejidades de su infraestructura
- El desafío es como dimensionar la capacidad de los servidores para ajustarlos a la demanda (servicios estacionales, eventos masivos y puntuales, juegos on line)

# [Conceptos generales]



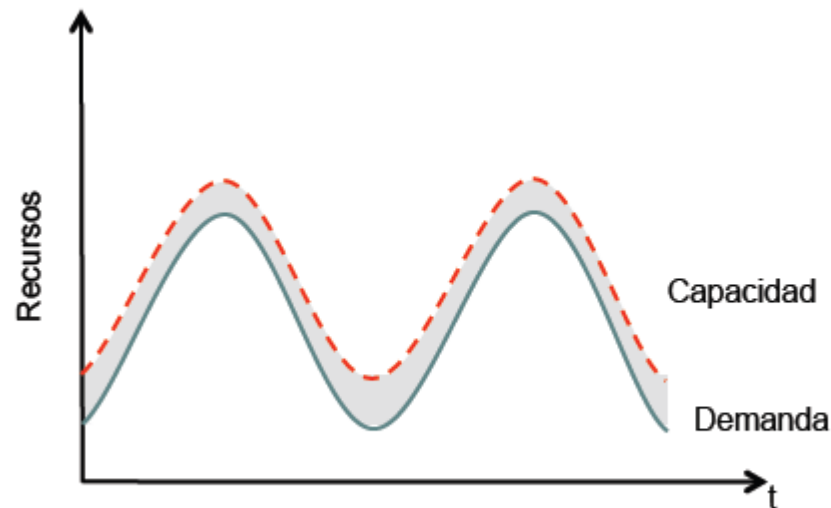
# Conceptos generales

- El desafío es cómo dimensionar los servidores para ajustarlos a la demanda.



# Conceptos generales

- La idea es contar con aprovisionamiento bajo demanda:
  - consume lo que necesites
  - paga por lo que consumes



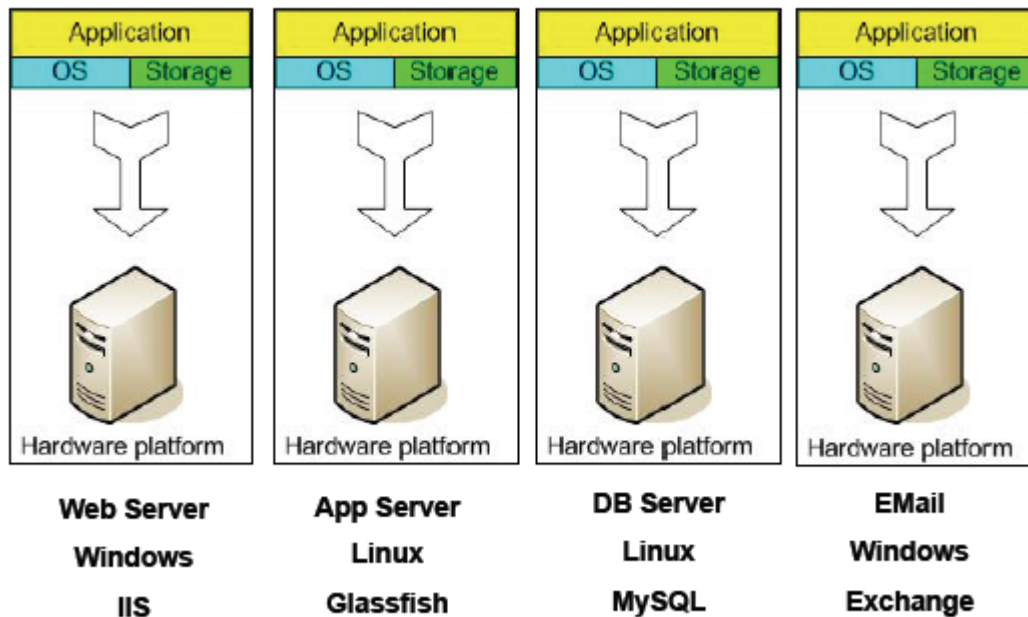
*Aprovisionamiento bajo demanda*

# [Datacenters]

- Esta evolución también ha supuesto un cambio profundo en la forma en que los ISPs gestionan y ofrecen sus recursos
- Los primeros proveedores cloud surgen de la evolución de las infraestructuras privadas de los principales ISPs
- Abren a los clientes las interfaces de gestión y añaden lo necesario para implementar el modelo pay-per-use
- Los clientes tienen así acceso a las mismas capacidades que antes se utilizaban sólo internamente

# Virtualización

## > Servidores tradicionales



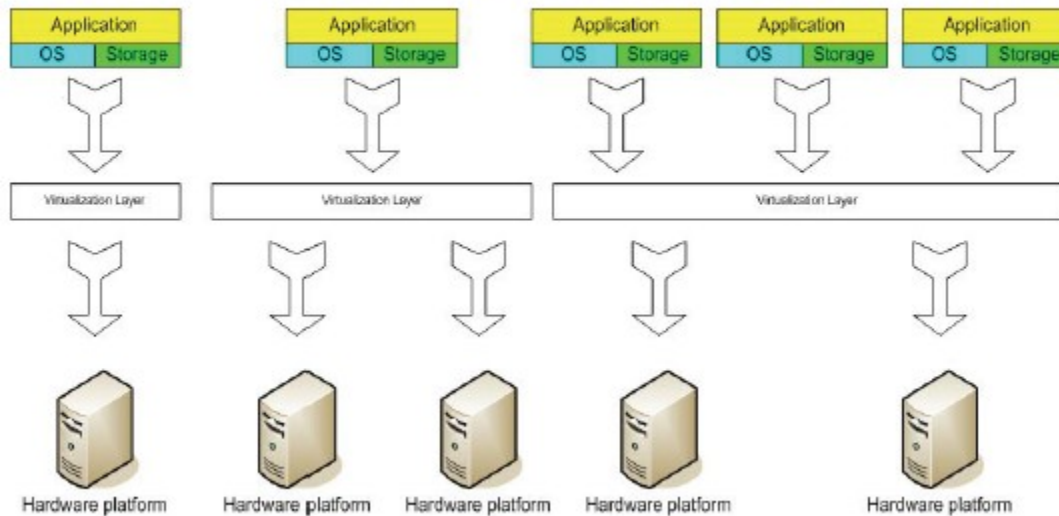
- A favor
  - Fácil de conceptualizar
  - El despliegue es bastante fácil
  - Puede ejecutar cualquier aplicación o servicio
- En contra
  - Es caro comprar y mantener el hardware
  - **No es muy escalable**
  - Es difícil replicarla
  - **Es difícil implementar la redundancia**
  - **Vulnerable a fallos hardware**
  - En situaciones de baja carga, el tiempo de CPU **se infrutiliza**



# Virtualización

## > Servidores virtuales

Una capa VMM (Virtual Machine Monitor)  
entre el SO alojado y el hardware



- A favor
  - **Resource pooling**
  - **Alta redundancia**
  - **Alta disponibilidad**
  - **Despliegue rápido** de nuevos servidores
  - **Reconfigurable “en caliente”** (los servicios siguen disponibles)
  - **Optimiza la utilización** de los recursos físicos
- En contra
  - Algo más difícil de conceptualizar
  - **Ligera pérdida de rendimiento** (hay que añadir el coste de la capa de abstracción)

# [Cloud Computing]

- No estamos realmente ante una nueva tecnología, sino un nuevo paradigma de uso que combina tecnologías existentes
  - SOAs
  - Grid Computing
  - Web Services
  - Virtualización

# [Definiciones]

## NIST (National Institute of Standards and Technology)

- “Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”
- Es la definición más referenciada y utilizada por el gobierno americano en sus relaciones con los proveedores.

# [Definiciones]

ENISA (European Network and Information Security Agency)

- “La computación en nube es un modelo de servicio bajo demanda para la prestación de TI, a menudo basado en la virtualización y en las tecnologías informáticas distribuidas. Las arquitecturas de computación en nube poseen:
  - (1) recursos con un alto grado de abstracción;
  - (2) escalabilidad y flexibilidad prácticamente instantáneas;
  - (3) prestación casi instantánea;
  - (4) recursos compartidos (hardware, base de datos, memoria, etc.);
  - (5) servicio bajo demanda que suele incluir un sistema de facturación de pago por uso;
  - (6) gestión programática (por ejemplo, mediante la API del WS)”

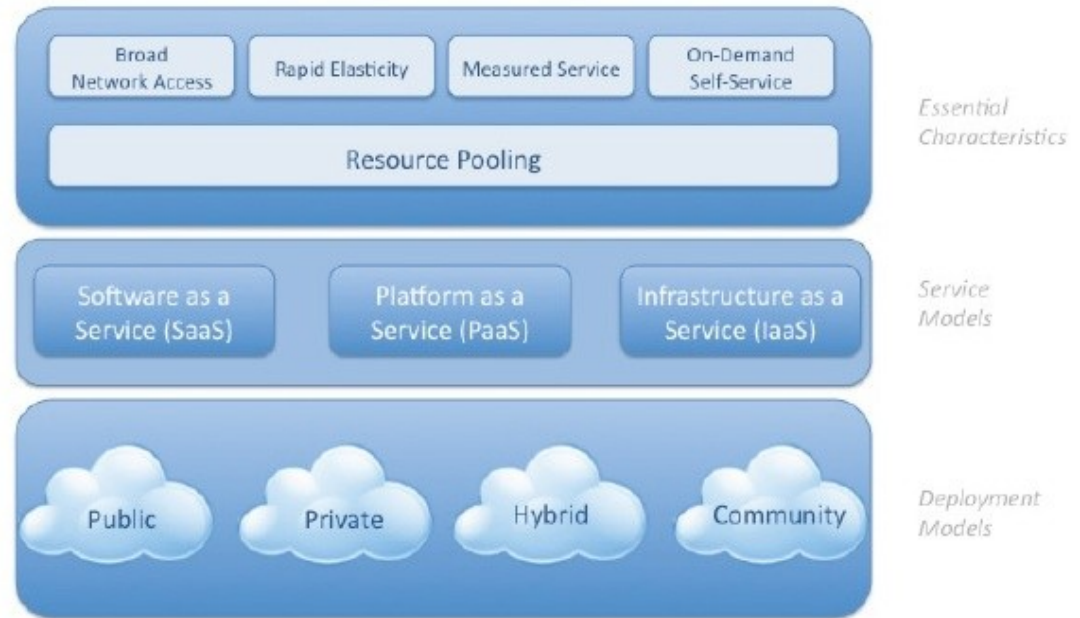
# [Modelo 5-3-4]

El modelo de Cloud del NIST  
es conocido popularmente  
como 5-3-4

5 Características  
esenciales

3 Modelos de servicio

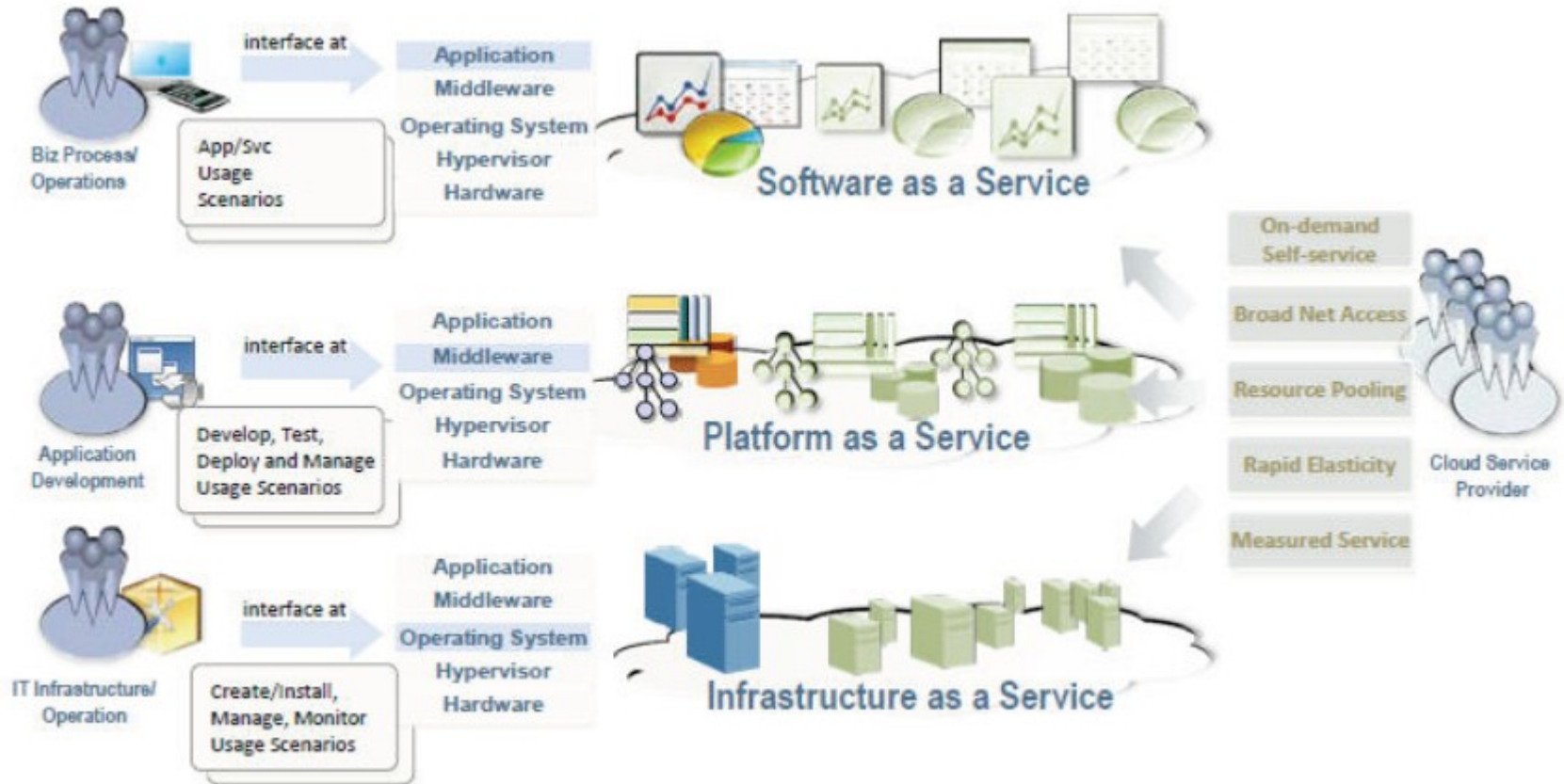
4 Modelos de despliegue



# [ 5 Características esenciales ]

1. **On-demand self-service:** un consumidor puede aprovisionar unilateral y automáticamente recursos de computación según necesite
2. **Broad network access:** las capacidades están accesibles en la red a través de mecanismos estándar que permiten el acceso desde plataformas cliente heterogéneas
3. **Resource pooling:** los recursos de computación de los proveedores son agrupados para servir a múltiples clientes usando un modelo multi-tenant (inquilino)
4. **Rapid elasticity:** las capacidades son aprovisionadas y liberadas rápida y elásticamente (y en algunos casos automáticamente)
5. **Measured Service:** los sistemas cloud controlan y optimizan automáticamente el uso de los recursos proporcionando alguna capacidad de medición (habitualmente pay-peruse o charge-per-use)

# 3 Modelos de servicio



# IaaS

1. El consumidor aprovisiona recursos de computación (p.e. CPU, almacenamiento, red, ...) en los que ejecuta su software (incluidas aplicaciones y sistemas operativos).
2. El consumidor no controla la infraestructura Cloud subyacente pero sí los SO, el almacenamiento, las aplicaciones desplegadas y a veces la red (firewalls).





# IaaS: Características

Las características comunes de un servicio IaaS son:

- **Provisión de recursos bajo demanda**
- **Elasticidad dinámica de los recursos**
- **Uso eficiente de los recursos: virtualización, arquitectura multi-tenant**
- **Modelo de pago por uso**
- **Gestión más eficiente de los datacenters: beneficios de la economía de escala, estandarización, reducción de costes, eficiencia energética**

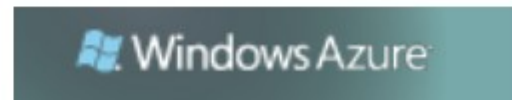
# IaaS: funciones de gestión

Algunas de las funciones de gestión típicas para el usuario son:

- Gestión dinámica de recursos (p.e. inicio/parada de VMs, creación/eliminación de volúmenes, ... )
- Funciones de automatización (p.e. balance automático de carga, reglas de autoescalado horizontal, ... )
- Monitorización de recursos (cpu, memoria, ancho de banda, ... )
- Gestión de consumos y costes

# [PaaS]

1. El consumidor despliega aplicaciones tanto propias como adquiridas, desarrolladas usando entornos de programación soportados por el proveedor, en la infraestructura cloud de este
2. El consumidor no controla la infraestructura Cloud subyacente pero sí las aplicaciones desplegadas y posiblemente el entorno de despliegue



# [PaaS: tipos de ofertas]

- Desarrollo de extensiones SaaS. Proveedores de SaaS proveen APIs para construir por sobre ellos (SalesForce, Facebook)
- Plataformas propietarias: algunos solo despliegan otros soportan todo el ciclo de vida del software: Amazon Elastic BeansTalk (despliegue de aplicaciones en Amazon EC2), Microsoft Azure (plataforma .NET), Google AppEngine (APIs propias)
- Open Platform: Ofertas que se basan en tecnologías abiertas para evitar el vendor lock-in. Son arquitecturas extensibles en las que se puede escoger el entorno de desarrollo, el lenguaje, la bd, el servidor de apps. Ej: Ejemplos: CloudFoundry (VMWare), OpenShift (RedHat), CumuLogic, CloudBees

# [PaaS: qué provee?

- Gestión del ciclo de vida de las aplicaciones
- Herramientas de desarrollo colaborativo
- Herramientas para el desarrollo de Uis
- Integración con WS (SOAP, REST): mashups, composición de servicios
- Integración con BD SQL y NoSQL
- Mecanismos de extensibilidad (plugins) para configurar nuevos frameworks
- Independencia del proveedor IaaS
- APIs (SOAP, REST) de gestión
- Mecanismos de monitorización y medida del rendimiento

# SaaS

1. El consumidor utiliza las aplicaciones del proveedor que son ejecutadas en una infraestructura cloud.
2. El consumidor no controla ni la infraestructura cloud subyacente ni las capacidades de la aplicación (pero posiblemente si puede controlar la configuración personal)



# Beneficios y desventajas SaaS

## BENEFICIOS

- No hay costos de licencias
- La Gestión de TI se simplifica
- Las actualizaciones son automáticas y constantes
- Ubicuidad (accesible desde cualquier dispositivo)
- Integración a través de APIs

## DESVENTAJAS

- Dependencia de la red
- Adaptabilidad
- Aspectos legales y de seguridad
- Dependencia del vendedor
- Costos: para pymes es ventajoso.

# Beneficios SaaS vs on-premise

## Análisis del ROI de implantar un ERP con horizonte 3 años

Costes		On-premise	SaaS
Software ERP		\$1.000.000	(mensual) \$10.000
Hardware		\$300.000	\$0
Instalación/consultoría		\$500.000	\$0
Soporte interno		\$0	(mensual) \$10.000
Operación (anual)		\$600.000	\$0
Total		\$3.600.000	\$780.000
año 1		año 2	año 3
On-premise	\$2.400.000	\$600.000	\$600.000
SaaS	\$240.000	\$240.000	\$300.000

Fuente: <http://www.forbes.com/sites/microsoft/2011/03/21/the-clear-roi-of-saas/>



# Diferencia entre SaaS y ASP

Muchas similitudes

Aplicaciones bajo demanda, acceso ubicuo, pago por uso, el cliente no gestiona ni controla la infraestructura subyacente, configuraciones personalizadas, reducción de costes de instalación y operación

La diferencia principal está en la infraestructura del proveedor (single-tenant vs multi-tenant)

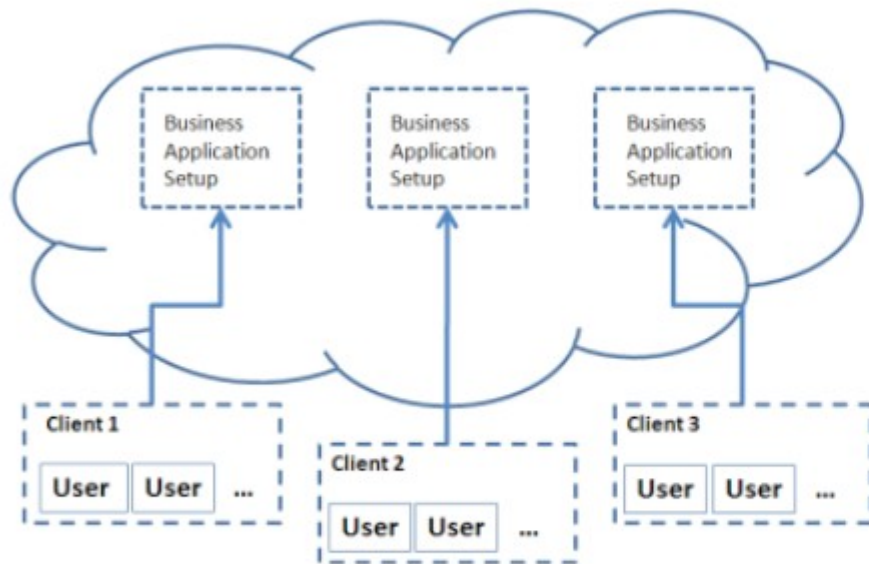
La importancia del aaS en SaaS

Interfaces WS abiertas: REST, SOAP, JSON, ...

Facilidad de integración, mashup, orquestación, ...

Desde el punto de vista del usuario es difícil percibir la diferencia

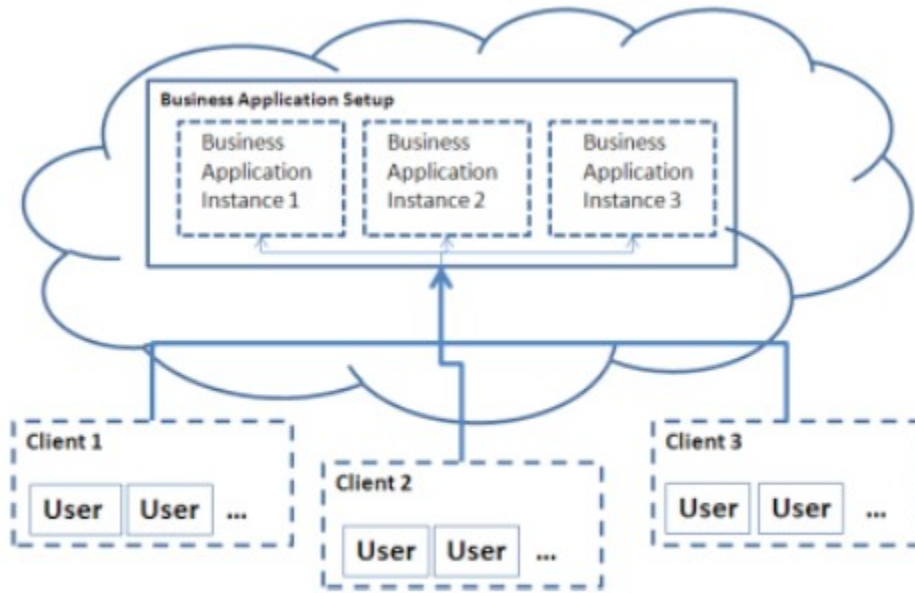
# [ASP: single-tenant]



**ASP: single-tenant**

- Cada cliente tiene su propia instalación personalizada que no se comparte y puede incluir la configuración del software, BD, servidores, etc.
- Los entornos ASP son más heterogeneos para dar soporte a las diferencias entre clientes y a los “casos especiales”
- El mantenimiento es, por tanto, más costoso
- La escalabilidad es limitada y difícil de implementar.

# [ SaaS: multi-tenant ]



**SaaS: multi-tenant**

- Todos los clientes comparten la misma infraestructura (servidores, BD, aplicaciones, ...) al tiempo que se preserva la separación entre ellos
- Los entornos SaaS son más homogéneos
- El mantenimiento es, por tanto, más barato y la escalabilidad más fácil de implementar.
- Tiene la contrapartida de ser menos flexible para adaptarse a las peculiaridades de los clientes

# [Variantes aaS]

- **HaaS: Hardware-aaS**
- **BPaaS: Bussines Process-aaS**
- **IDaaS: Identity-aaS**
- **EaaS: Entertainment-aaS**
- **BDaaS: Bases de datos-aaS**

# [ 4 Modelos de despliegue ]

---

1. Cloud público
2. Cloud privado
3. Cloud híbrido
4. Cloud comunitario

# [ 4 Modelos de despliegue ]

## 1. Cloud público

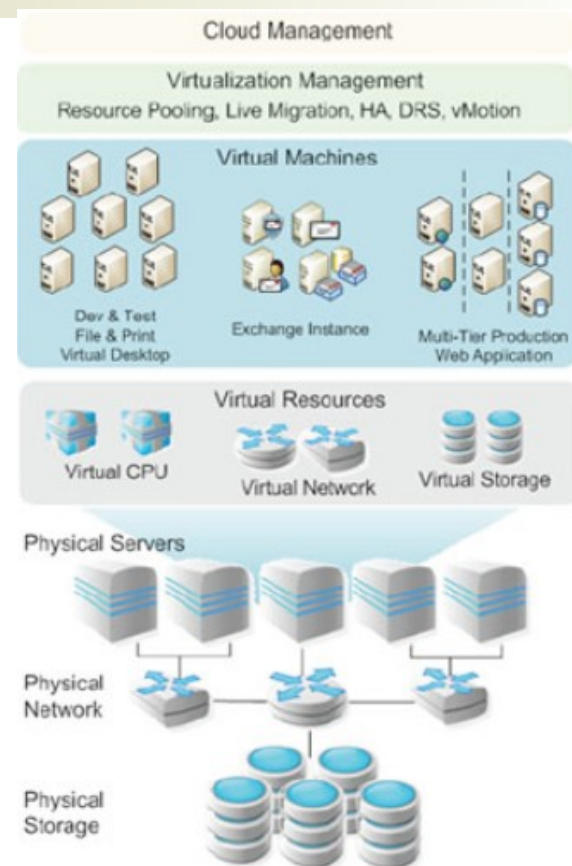
- Infraestructura para uso del público o empresas en general
- Propiedad del proveedor (p.e. empresa, universidad, organismo estatal, ...)
- Estos clouds tienen una interface para la gestión interna de la infraestructura virtualizada y exponen otra externa para la gestión de los recursos de los clientes



# 4 Modelos de despliegue

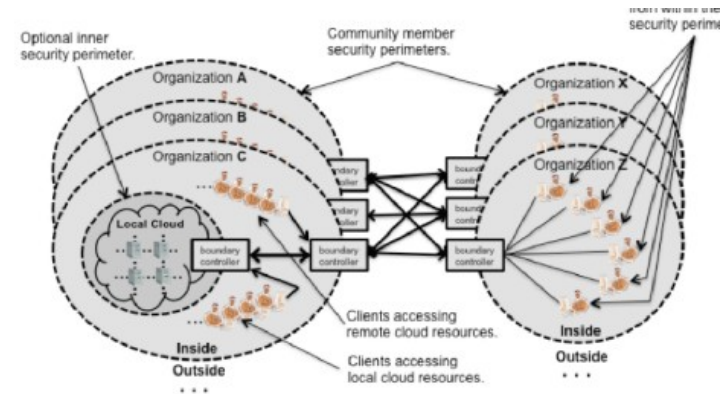
## 2. Cloud privado

- Para uso exclusivo de una organización con múltiples clientes (p.e. departamentos de una empresa)
- Puede ser de su propiedad (on-premise), alquilarlo (off-premise) o una combinación de ambas
- Permite una gestión flexible y ágil de los recursos de la organización
- Los principales proveedores públicos de IaaS comenzaron creando clouds privados para mejorar la gestión de sus datacenters



# [ 4 Modelos de despliegue ]

3. Cloud comunitario: infraestructura para uso de una comunidad de organizaciones que comparten intereses
  - Cada componente conserva su autonomía
  - La relación entre componentes se hace mediante tecnologías (propietarias o estandarizadas) que permitan la portabilidad de datos y aplicaciones
  - Puede ser propiedad de una o varias de las organizaciones, ser alquilado o una combinación de ambas

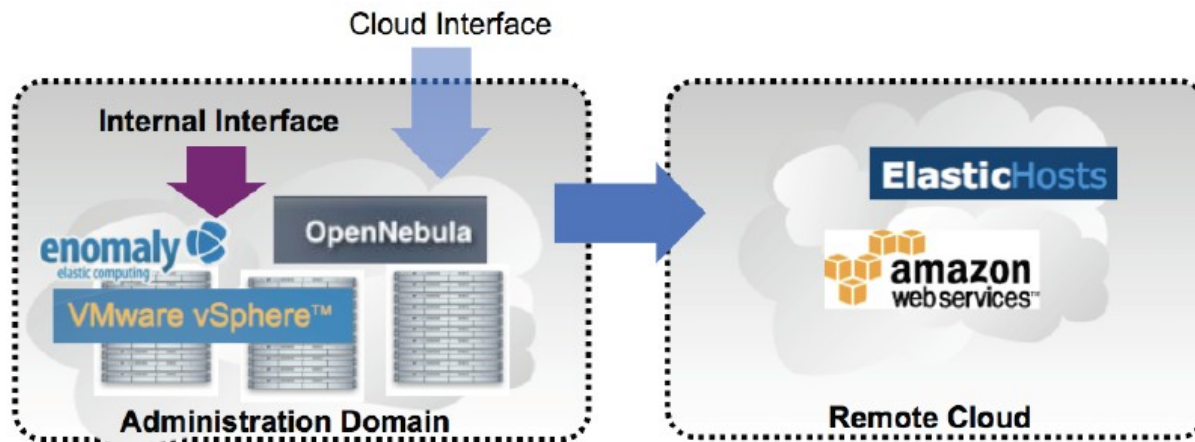




# 4 Modelos de despliegue

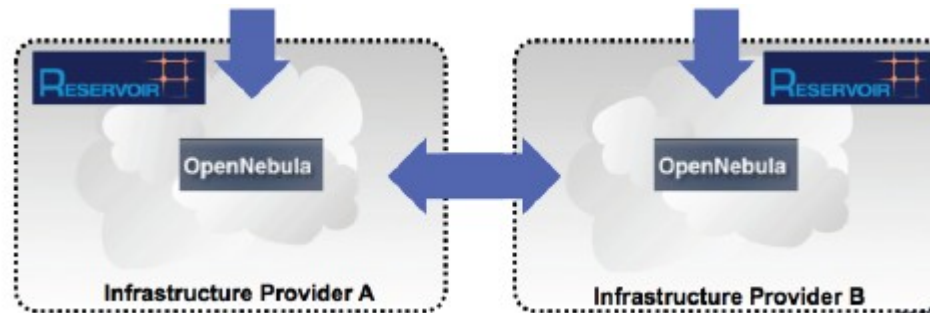
## 4. Cloud híbrido

- Combinación de cloud privado y público
- Permite gestionar los picos de carga obteniendo recursos del cloud público
- El uso de un cloud público es totalmente transparente para los usuarios del cloud privado

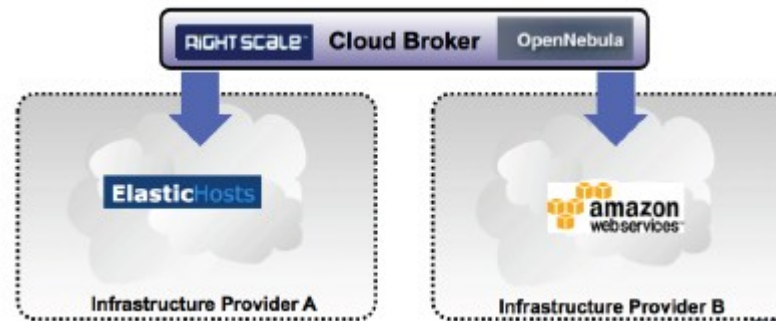


# Otros modelos de despliegue

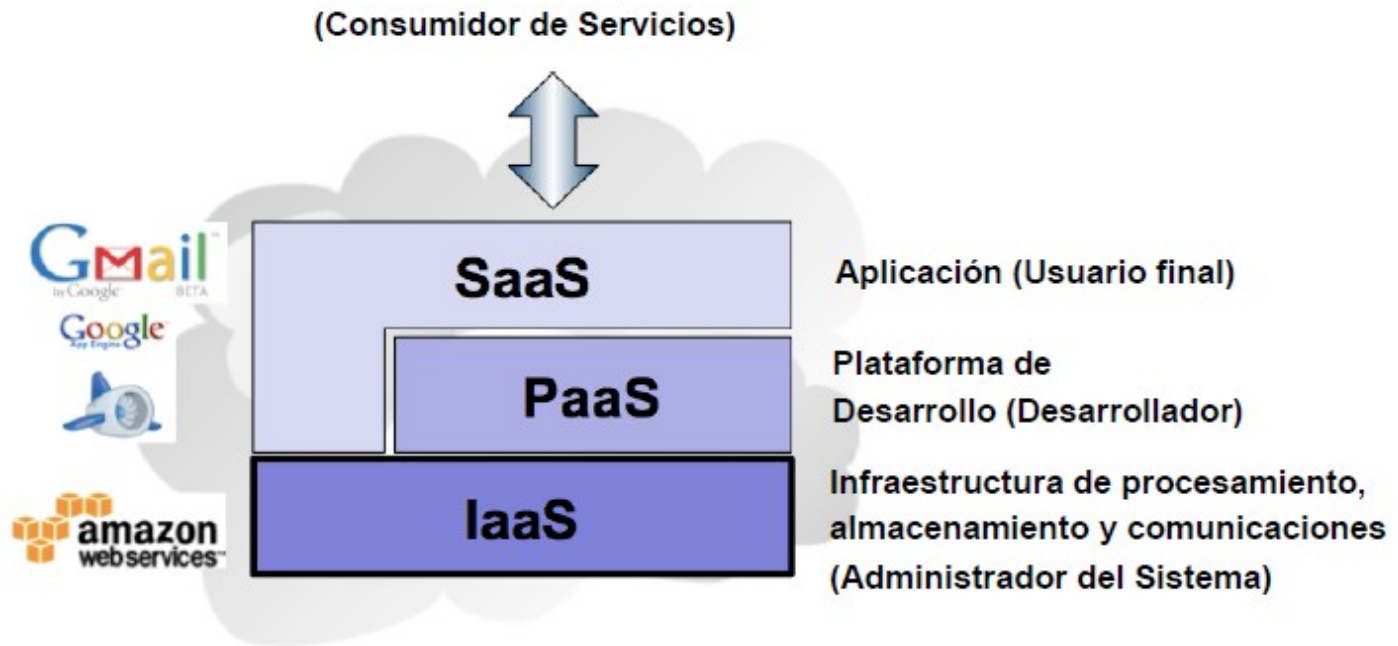
Federación de clouds (requiere estándares)



Cloud brokering (uso conjunto de varios proveedores)



# [ En resumen .... ]



# [ En resumen .... ]

Con IaaS contratamos: computación, almacenamiento y red

	Amazon	Microsoft Azure	Google
Computación	EC2 Auto scaling ELB	Virtual Machines Traffic Manager	Compute Engine
Almacenamiento	S3 EBS DynamoDB Glacier RDS	Storage  Backup SQL Database	Cloud Storage  Cloud SQL
Red	Elastic IP VPC Route 53 Direct Connect	Virtual Network	

# Por último ... estándares en el Cloud

Hay numerosas iniciativas promoviendo estándares y su adopción: NIST, CSA (Cloud Security Alliance), Open Data Center Alliance, OCI (Open Cloud Initiative), Open Cloud Consortium, ...

Se han identificado varias áreas en las que es necesario intensificar los esfuerzos de estandarización:

- Portabilidad de VMs e imágenes
- Acceso y migración de datos
- Seguridad y gestión de la identidad
- Orquestación de procesos
- Monitorización y administración
- SLAs

# Estándares IaaS

**OVF (Open Virtualization Format):** Estándar para el empaquetamiento y la distribución de VMs. Define un formato abierto, extensible, eficiente e independiente de la plataforma. Propuesta apoyada por los principales actores en virtualización (VMware, Citrix, Microsoft, ...)

**CIMI (Cloud Infrastructure Management Interface)**

- Interfaz para la gestión de infraestructuras IaaS
- Alternativa al estándar de facto: Amazon EC2
- “Contendiente” del estándar OCCI (aparecido previamente)
- Su enfoque es hacia la gestión del ciclo de vida de los recursos proporcionados por los proveedores IaaS a sus clientes
- Define un modelo para los recursos (computación, almacenamiento, comunicaciones) que puede ser mapeado a la oferta de cualquier proveedor de infraestructura Cloud
- Aún no hay muchas implementaciones: Apache DeltaCloud, OW2 Sirocco

# Estándares IaaS (2)

## OCCI (Open Cloud Computing Interface)

- Comenzó siendo una API REST para la gestión de infraestructura IaaS. Alternativa al estándar de facto: Amazon EC2
- Propuesta apoyada por muchos proveedores IaaS y proyectos y plataformas
- Abiertos (Rackspace, Oracle, GoGrid, Cisco, Flexiscale, ElasticHosts, SLA@SOI, RESERVOIR, OpenStack, OpenNebula, ...)
- Numerosas implementaciones: OpenNebula, OpenStack, Eucaliptus, ...

## CDMI (Cloud Data Management Interface)

- Interfaz para el manejo de diferentes tipos de almacenamiento en el cloud. Soporta: contenedores y objetos, colas, tablas. Permite asociar metadatos a contenedores y objetos
- Apoyado por muchas empresas importantes: DELL, IBM, Oracle, Intel, Fujitsu, Microsoft, VMware, ...
- Adoptado en el 2012 como estándar ISO/IEC 17826:2012

# [Estándares PaaS]

## CAMP (Cloud Application Management for Platforms).

- Interfaz estándar para el acceso a los servicios de proveedores PaaS independiente de la plataforma, framework y lenguaje de desarrollo utilizados
- Define los artefactos y APIs necesarios para crear, ejecutar, administrar, monitorizar y “patchear” aplicaciones en el Cloud

## TOSCA (Topology and Orchestration Specification for Cloud Applications)

- Estándar para la descripción de aplicaciones y servicios IT complejos a ser ejecutados sobre infraestructuras Cloud
- Su enfoque es hacia la portabilidad de aplicaciones y servicios con independencia del proveedor Cloud utilizado
- Define un metamodelo declarativo de Service Templates, en los que se describe: La topología (o estructura) del servicio, Los planes u operaciones que permiten iniciarlo, terminarlo y gestionar su ciclo de vida



# [Estándares PaaS (2)]

## Beneficios

- **Despliegue de aplicaciones en cualquier proveedor manteniendo los requisitos de la aplicación: seguridad, compatibilidad, etc.**
- **Facilita la migración de aplicaciones existentes hacia entornos Cloud**
- **Abre la posibilidad a aplicaciones multiproveedor dinámicas**
- **Creación de un mercado de servicios IT**

# Tendencias

- **Service Mesh.** Istio es un proyecto proveniente de la alianza de Google, IBM y Lyft; considerado como muy estable y que fue lanzada en julio de 2018.
- **Serverless.** El proveedor de la capa de computación nos permite ejecutar durante un periodo de tiempo determinado porciones de código denominadas "funciones" sin necesidad de hacernos cargo de la infraestructura subyacente
- **Contenedores.** La tendencia podría extenderse a servicios que provean versiones hospedadas de las principales plataformas contenedoras como Docker Enterprise, Red Hat OpenShift u otros.
- **Nube híbrida.** Los entornos de Cloud híbrido han emergido con gran fuerza, ya que son muchas las empresas las que no quieren enfocarse exclusivamente en la nube pública.
- **IoT.** Se espera que tome mayor fuerza ya que ayuda a mejorar todos los procesos y aporta cada día una cantidad mayor de recursos.
- **El análisis masivo de datos (Big Data)** es el servicio cloud que despierta un mayor interés entre las empresas