

Cloud Computing

Introducción amena
1er. Cuatrimestre 2023



Cloud Computing



Agenda

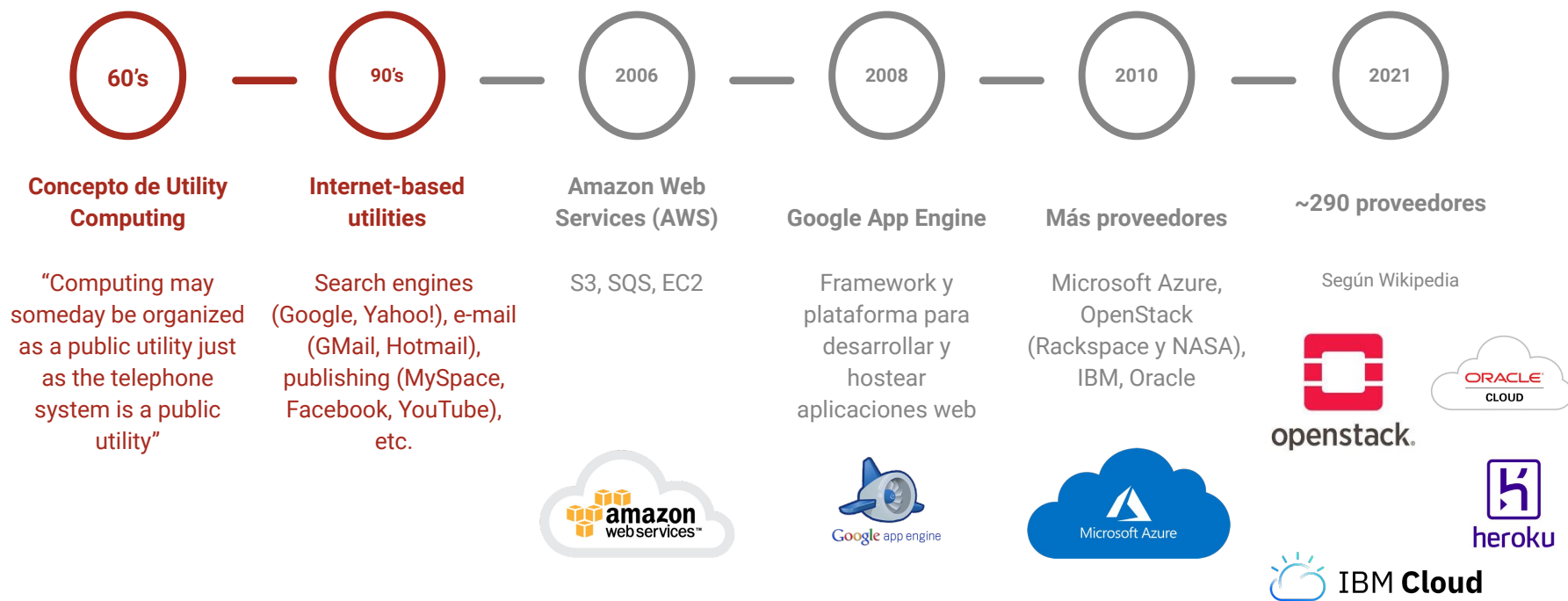
- Brevísima historia
- Definición
- Características
- Arquitecturas ejemplo
- Business Drivers
- Tecnologías
- Delivery Models
- Deployment Models
- Beneficios y Riesgos
- Hasta luego

¿De dónde viene? ¿Qué es?

¡Decime ya mismo!



Brevísima historia



Cloud Computing

Características esenciales

*Modelo para permitir el acceso **ubicuo, conveniente, y bajo demanda** a un **conjunto compartido de recursos computacionales configurables** (redes, servers, almacenamiento, aplicaciones, y servicios) a través de la red, que pueden ser **rápidamente provistos** y desplegados con un esfuerzo mínimo de administración o de interacción con el proveedor del servicio*

1. On-demand self-service
2. Broad network access
3. Resource pooling
4. Rapid elasticity
5. Measured service

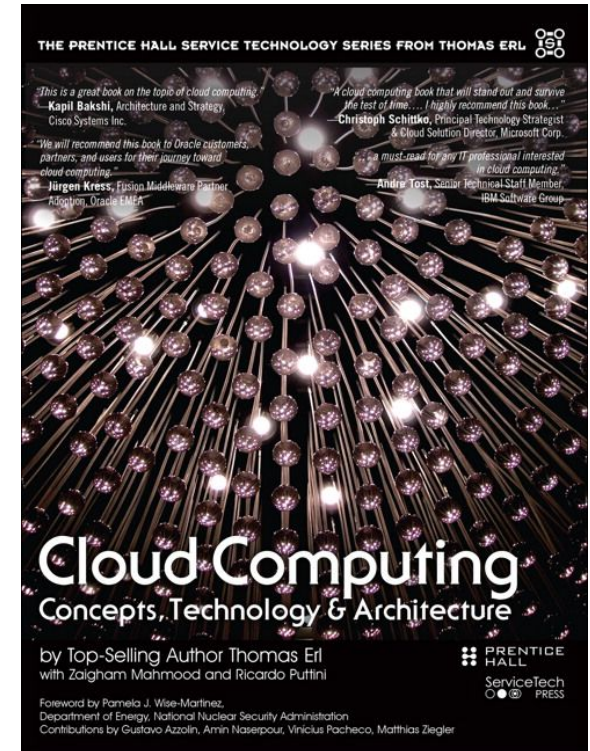
Service Models: SaaS, PaaS, IaaS

Deployment Models: Private, Community, Public, Hybrid

Cloud Computing

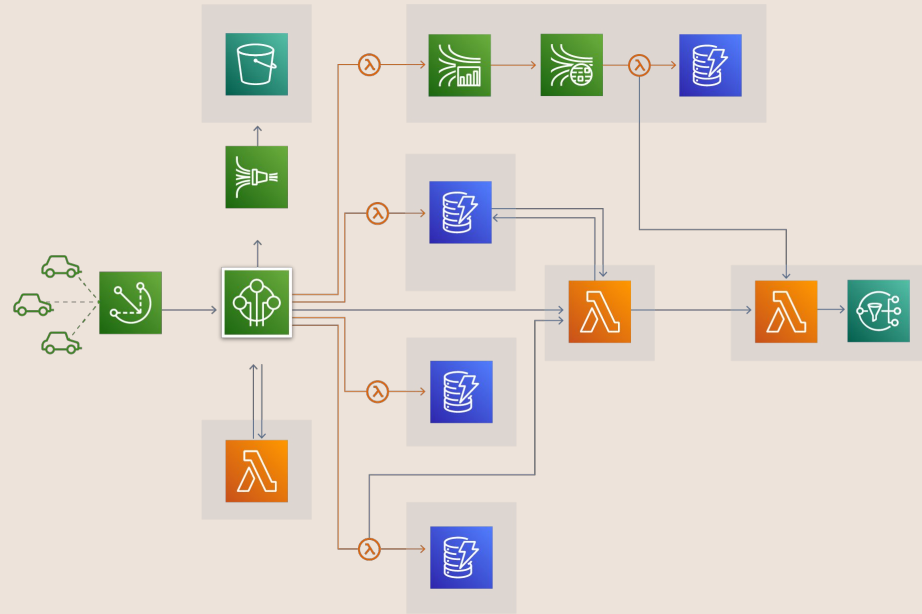
Definición complementaria

*Forma especializada de **computación distribuída**,
que introduce **modelos de utilización**
para la **provisión remota**
de **recursos escalables y medibles***

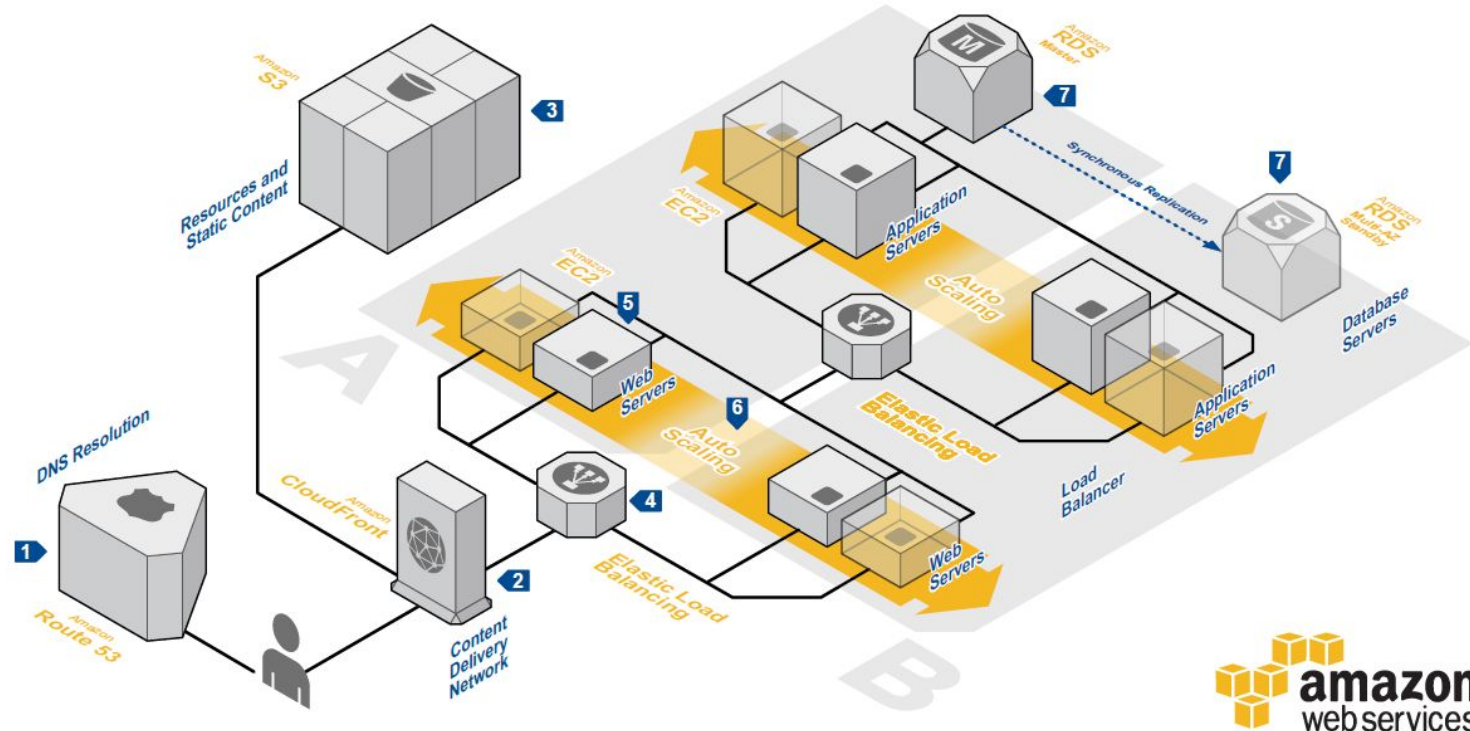


Y la Arquitectura...

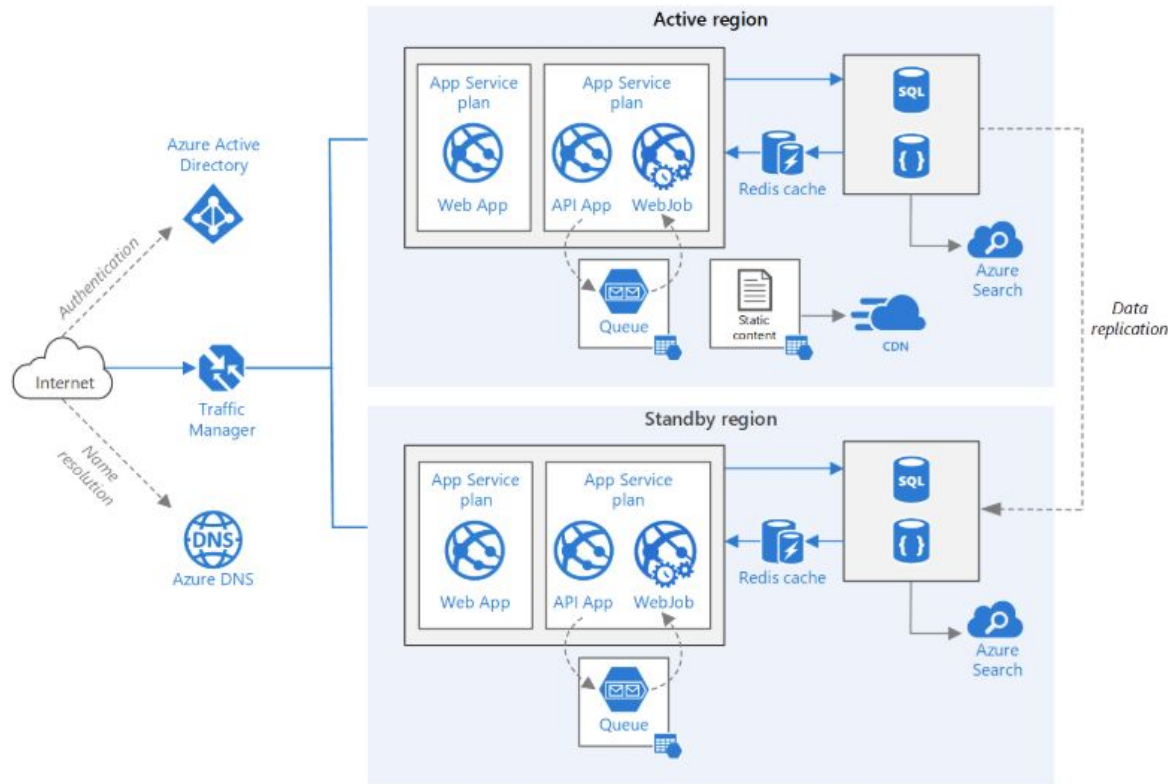
¿más o menos qué onda?



Arquitectura ejemplo: Web App en AWS



Arquitectura ejemplo: Web App Multi-Region en Azure



Características



6 Características – 3 por aquí...

- **On-Demand usage**

- Consumidor accede unilateralmente
- Self-provision
- El uso puede automatizarse

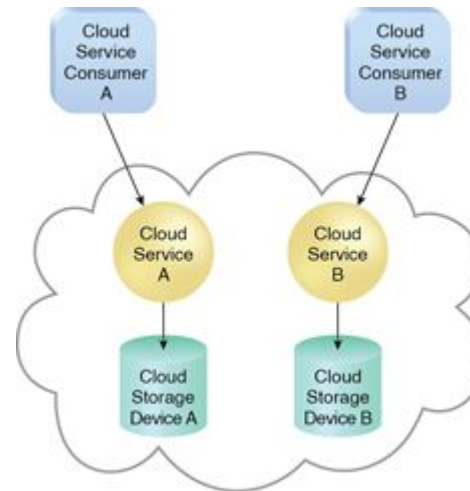
- **Ubiquitous Access**

- Ampliamente accesible
- Dispositivos, protocolos, interfaces, seguridad

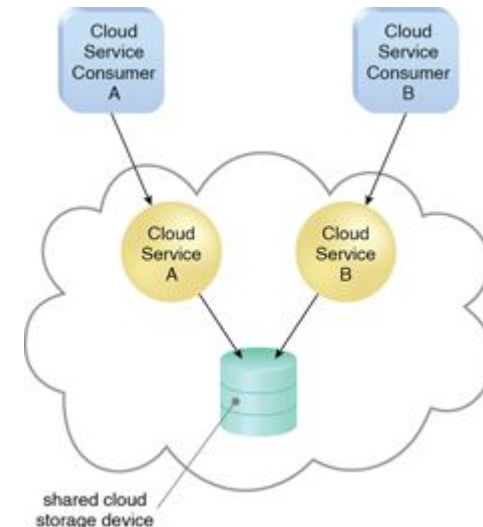
- **Multitenancy**

- Una instancia del software sirve a múltiples consumidores
- Cada consumidor está aislado de los otros
- Resource pooling

Single-tenant



Multitenant



6 Características – ...y otras 3 por acá

- **Elasticity**

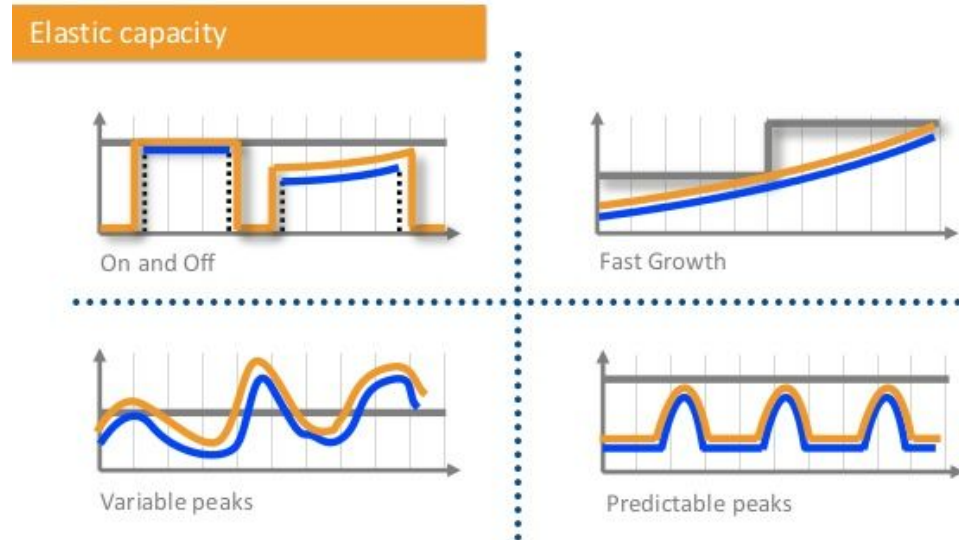
- Escalar recursos automáticamente
- Trigger: condiciones en runtime ó predeterminadas

- **Measured Usage**

- Mantener registro del uso de los recursos
- Pay-as-you-go. Sólo por los recursos utilizados
- Monitoreo de recursos para reportes

- **Resiliency**

- Distribución de recursos redundantes a lo largo de distintas ubicaciones físicas



¿Para qué?



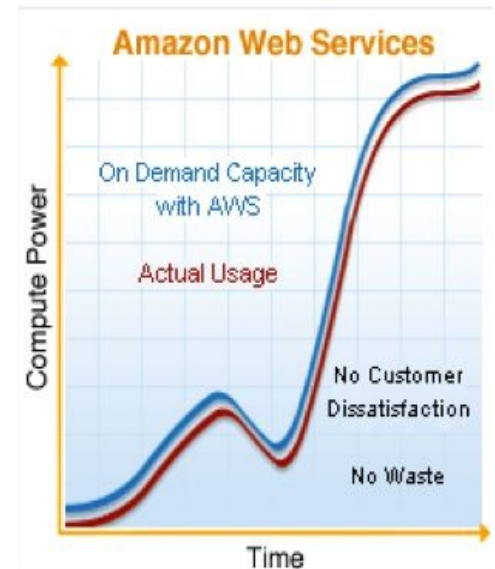
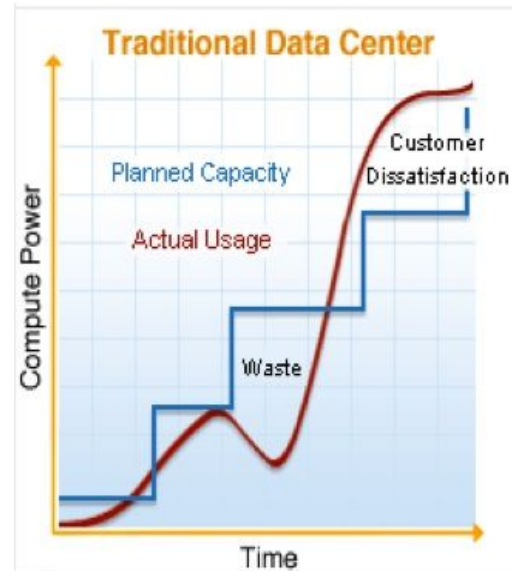
Planificación de la capacidad

¿Cuándo agregar capacidad?

1. **Lead strategy:** anticiparse a la demanda
2. **Lag strategy:** cuando el recurso alcanza su máximo uso

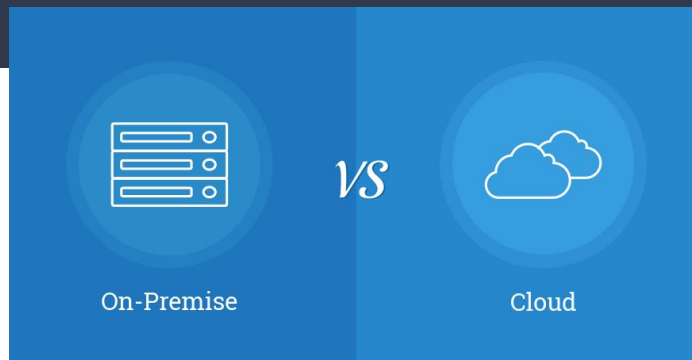
Over-provisioning vs. under-provisioning

3. **Match strategy:** pequeños incrementos / decrementos acompañando a la demanda



Elasticity favorece
Match Strategy

Reducción de Costos



1) **Adquisición nueva infraestructura**

2) **Costo de su mantenimiento**

- a) Personal
- b) Upgrades/patches → testing
- c) Electricidad
- d) Refrigeración
- e) Seguridad sobre recursos físicos
- f) Licencias y personal que las administre

Recursos on-demand, pay-as-you-go, corto plazo
(\$/h CPU)

Liberar recursos inmediatamente

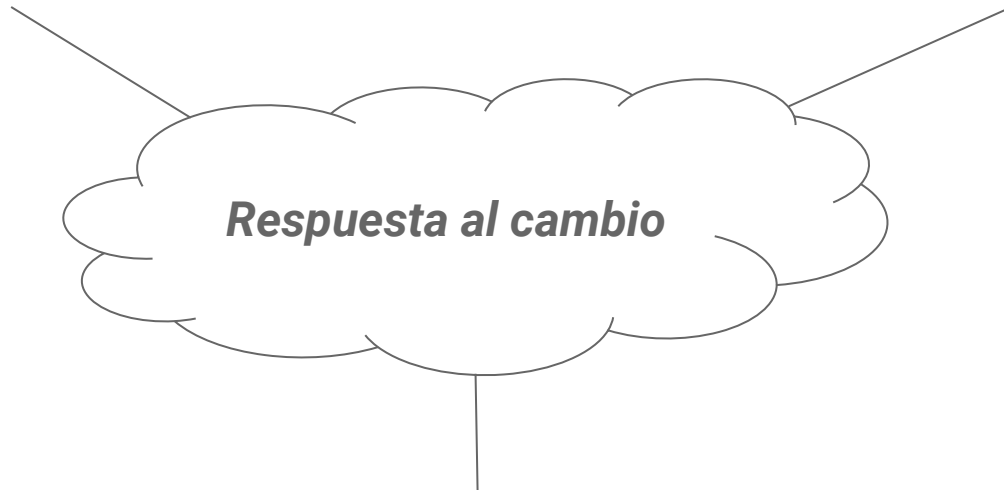
No se necesitan grandes inversiones up-front

Recursos ofrecidos como multi-tenant

Agilidad organizacional

Escalar más allá de
lo previsto

Responder a fluctuaciones
en el uso de los servicios



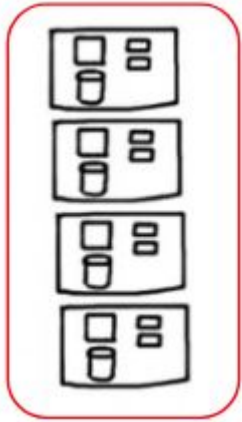
Evitar grandes
inversiones up-front

Ajá... ¿Y cómo es posible construir esas nubes?

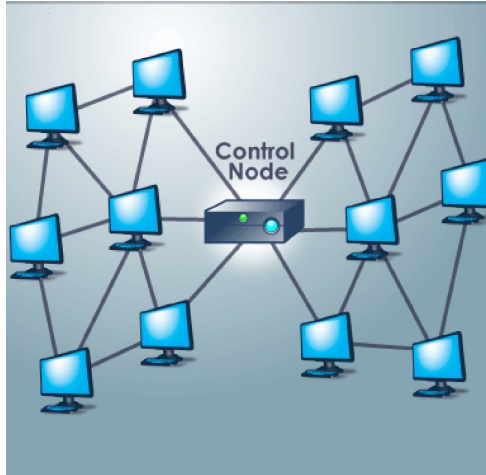


Tecnologías que inspiraron la nube

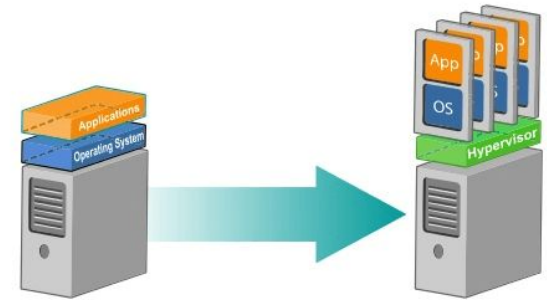
Clustering



Grid Computing



Virtualización



Tecnologías que posibilitan la nube

- Redes interconectadas
- ISPs
- Mejora de calidad en conectividad
- TCP/IP

Internet

Acceso ubicuo

- Web browsers
- HTTP, URI, HTML, XML, js

Web

Administración remota
Implementación de servicios cloud

- Recursos físicos y virtuales
- Agrupados en estrecha proximidad
- Commodity hardware
- Alta redundancia
- Operación remota
- Ubicaciones custom-designed

Data Centers

Power sharing
Compartir recursos
Accesibilidad del personal
Alta Disponibilidad

- Múltiples tenants acceden a la misma aplicación simultáneamente
- La aplicación asegura que un tenant sólo pueda acceder a sus datos
- Tenants pueden customizar la aplicación

Multitenancy

Compartir recursos físicos y virtuales

- Web Services (SOAP)
- HTTP/Web APIs
- REST APIs

Services

Exponer/Encapsular funcionalidad
Comunicación entre máquinas

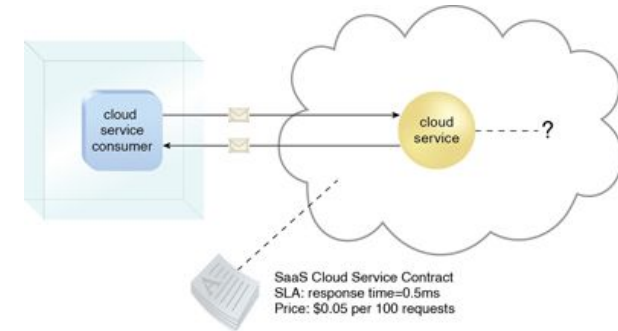
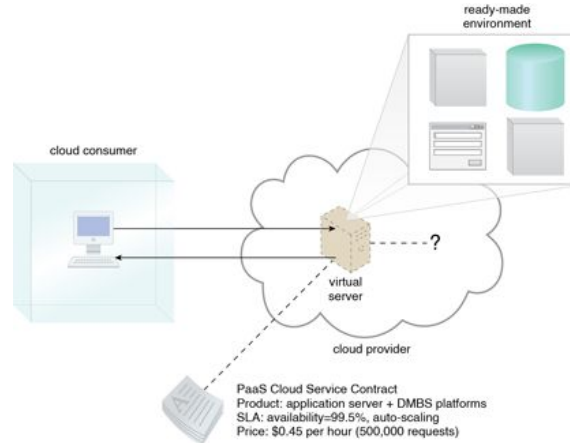
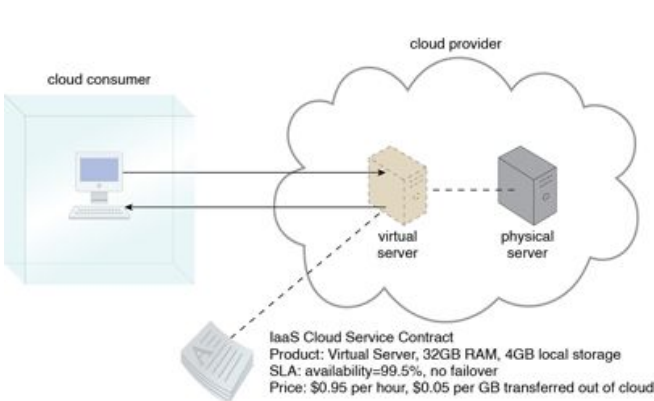
¿Y de qué maneras se puede usar?



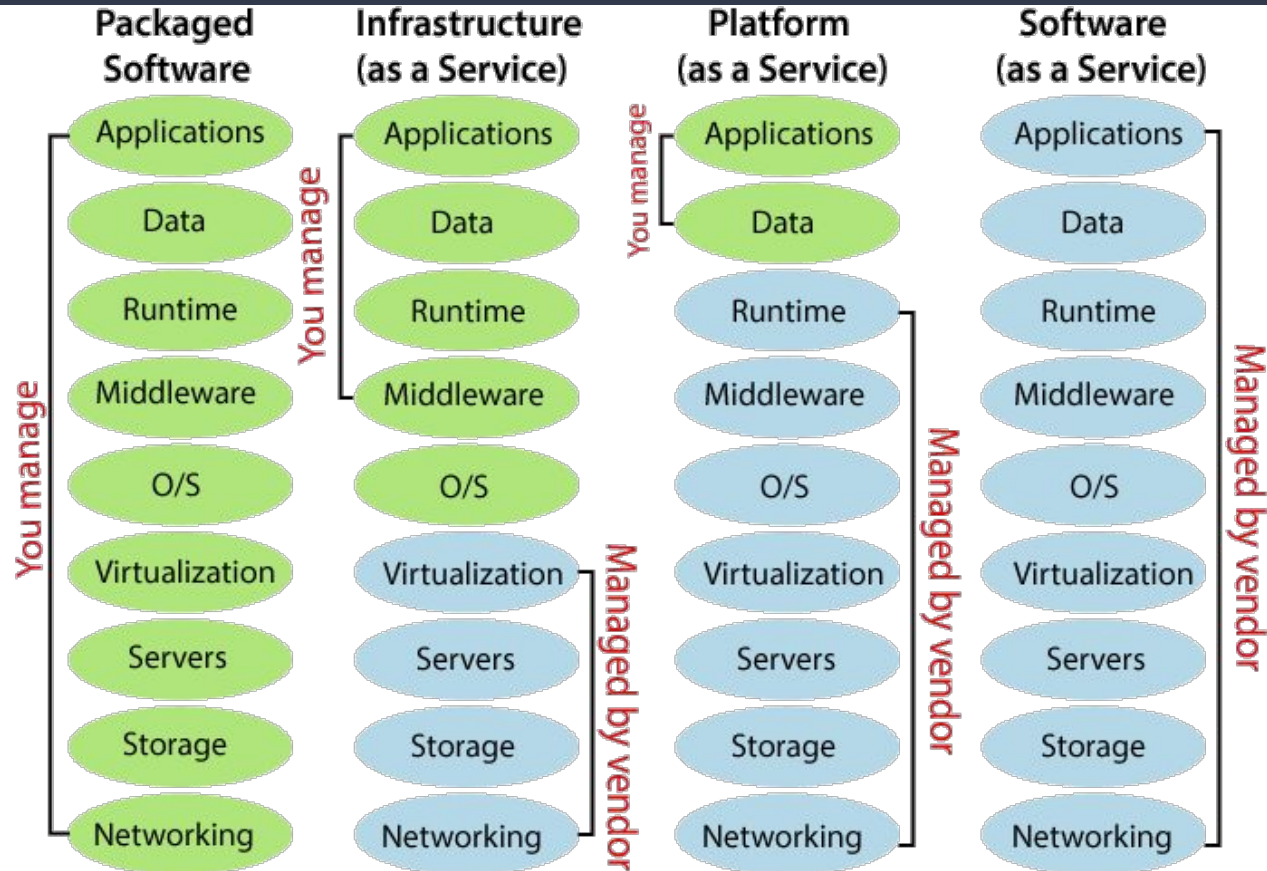
Delivery Models

Combinación de recursos específica y pre-empaquetada por el proveedor cloud

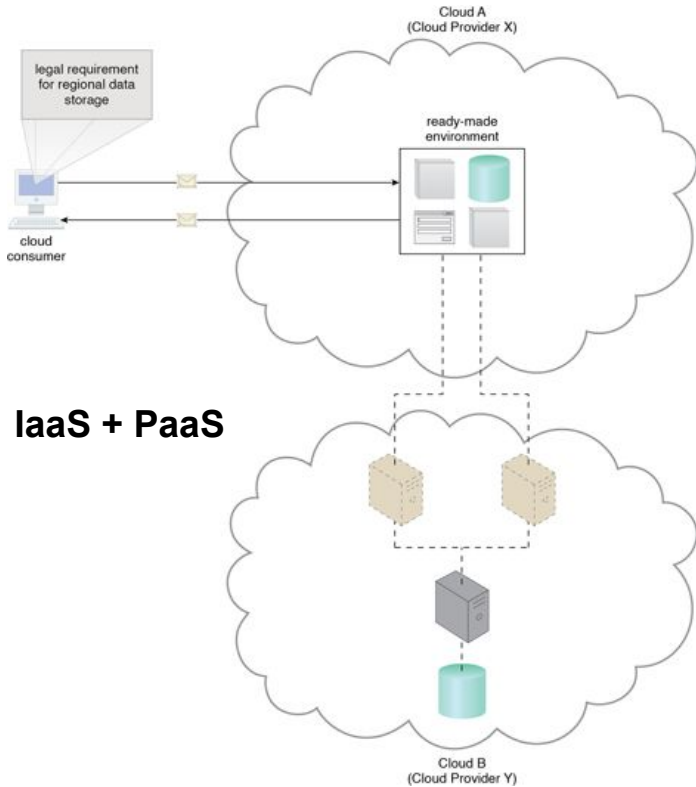
IaaS	PaaS	SaaS
Infraestructura (hardware, network, OSs, etc.)	Ambiente predefinido listo para usar	Software expuesto como servicio
Alto nivel de control	Control limitado	Control sólo sobre configuración de uso
AWS, Azure, Google Compute Engine, Rackspace, OpenStack, Apache CloudStack	Azure (.NET), Google App Engine (Java/Python)	Google Apps, Salesforce



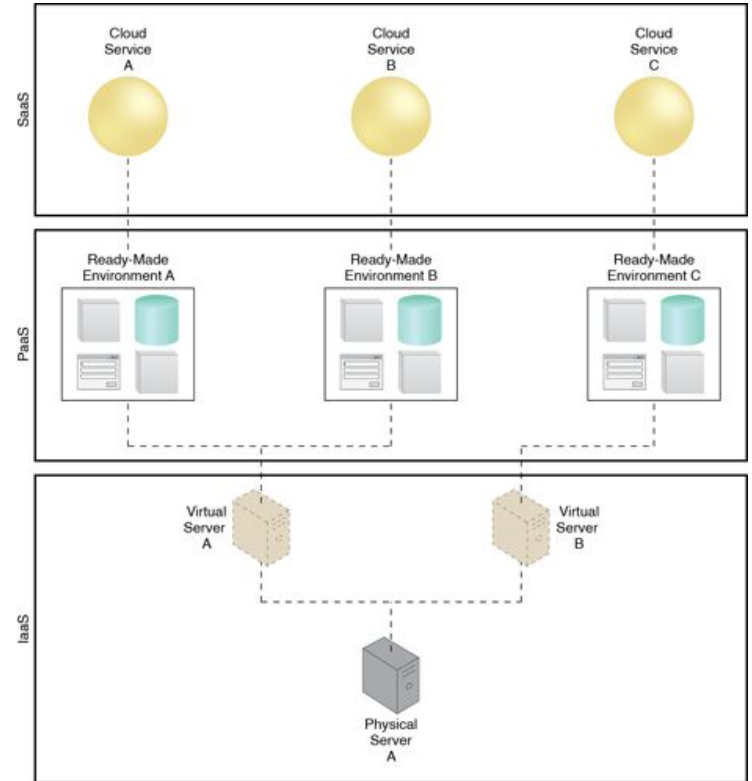
Delivery Models



Delivery Models: se pueden combinar



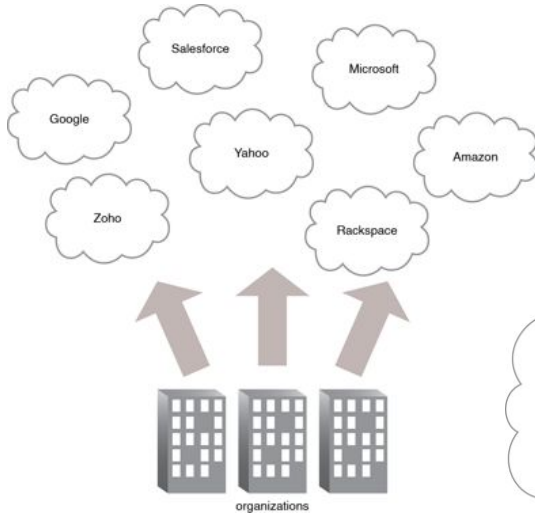
IaaS
+
PaaS
+
SaaS



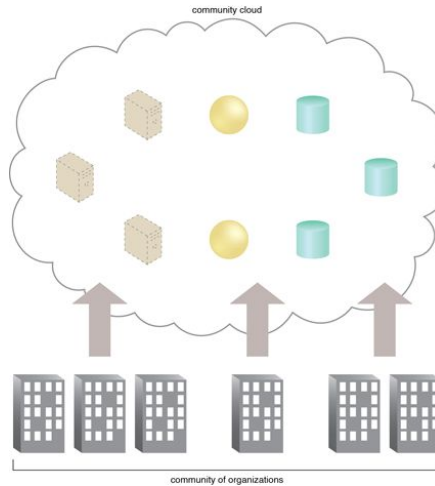
Deployment Models

Tipo específico de ambiente Cloud, distinguible por quién lo posee, su tamaño y el tipo de acceso

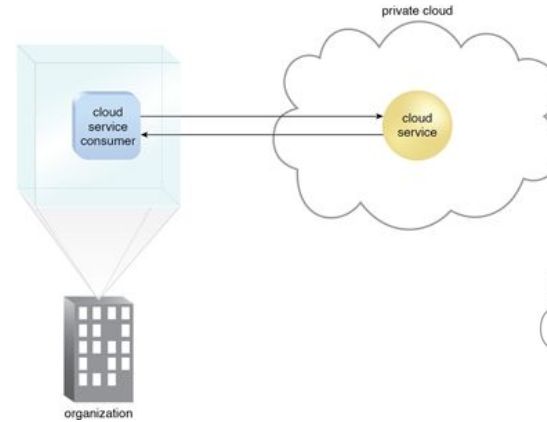
Public Cloud



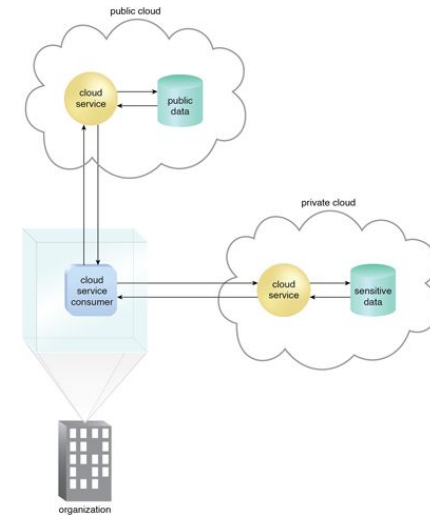
Community Cloud



Private Cloud



Hybrid Cloud



Bueno, sí, ya quiero que
esto termine, Guille...



Tradeoffs – Beneficios y Riesgos

- + Reducción de inversiones up-front
- + Costos proporcionales (consumo medible)
- + Percepción de infinitos recursos
- + Agregar/quitar recursos fácilmente
- + Abstracción de la infraestructura
- + Scalability
- + Elasticity
- + Availability

- Security
 - La seguridad de los datos es responsabilidad conjunta del consumidor y del proveedor cloud
 - El proveedor cloud tiene acceso privilegiado a los datos del consumidor
- Menor control que on-premise
 - Uso de SLAs
- Portabilidad baja entre proveedores cloud
- Cuestiones legales
 - Ubicación geográfica de los datos
 - Revelación de datos por leyes de los países donde están los data centers



¿Consultas?

Feedback

<http://bit.ly/arq-soft-feedback-clases>

Guillermo Rugilo

grugilo@fi.uba.ar

¡Gracias!