

# Autoscaling en Amazon Web Services EC2

Diciembre 2013

**str**sistemas

# Comenzamos

# hola \*

## #tallerSTRamazon

# Entorno del taller

Si alguien quiere reproducir el taller necesitará:

- una cuenta en Amazon Web Services (AWS en adelante) con tarjeta de crédito válida
- [esta presentación](#) y el vídeo que colgaremos en Youtube tras el taller
- ganas y un rato de dedicación

# Planificación

Si da tiempo la planificación es:

- conceptos básicos de Cloud Computing
- conceptos básicos de AWS EC2
- creación de instancias EC2
- configuración de autoscaling en EC2
- configuración de políticas de autoescalado

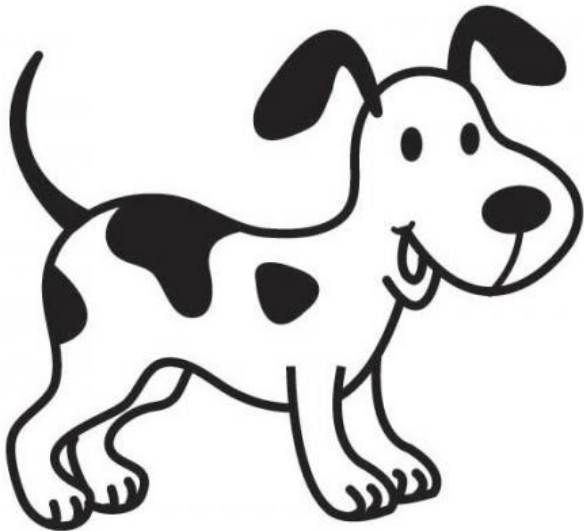
# Cloud Computing (I)

Cada uno cuenta la película a su manera. Nuestro punto de vista es:

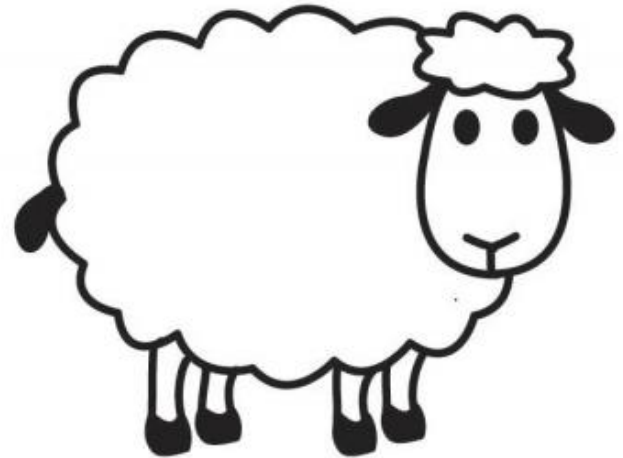
- plataforma escalable en tiempo casi real disponible como servicio (público o privado)
- dispone de algo más que máquinas virtuales (balanceadores, almacenamiento, etc.)
- **los servicios disponen de API**

# Cloud Computing (II)

Cambio de paradigma



VS



[Lectura recomendada](#)

# Cloud Públicas



# Cloud Privadas



Todas las marcas, productos y logos son propiedad de sus respectivas empresas u organismos enlazados



# Amazon Web Services (I)

- Rápido crecimiento y evolución
- Posiblemente la plataforma pública de cloud con más servicios y opciones en IAAS
- Soluciones para muchos tipos de problemas: web, big data, científico, etc.
- Utilizada como base de amazon.com
- Según Jeff Bezos será el negocio más grande de Amazon en los próximos años

# Amazon Web Services (II)

- Todo se basa en pagos por recursos utilizados
- Todos los servicios tienen limitaciones por cliente/cuenta/instancia
- Precio > hosting tradicional (coste alto!= caro)
- No siempre es lo mejor
- Performance variable, existe disputa de recursos

# Amazon Web Services (III)



**EC2**



**ROUTE 53**



**ELASTIC  
CACHE**



**RDS**



**CLOUDFRONT**



**GLACIER**

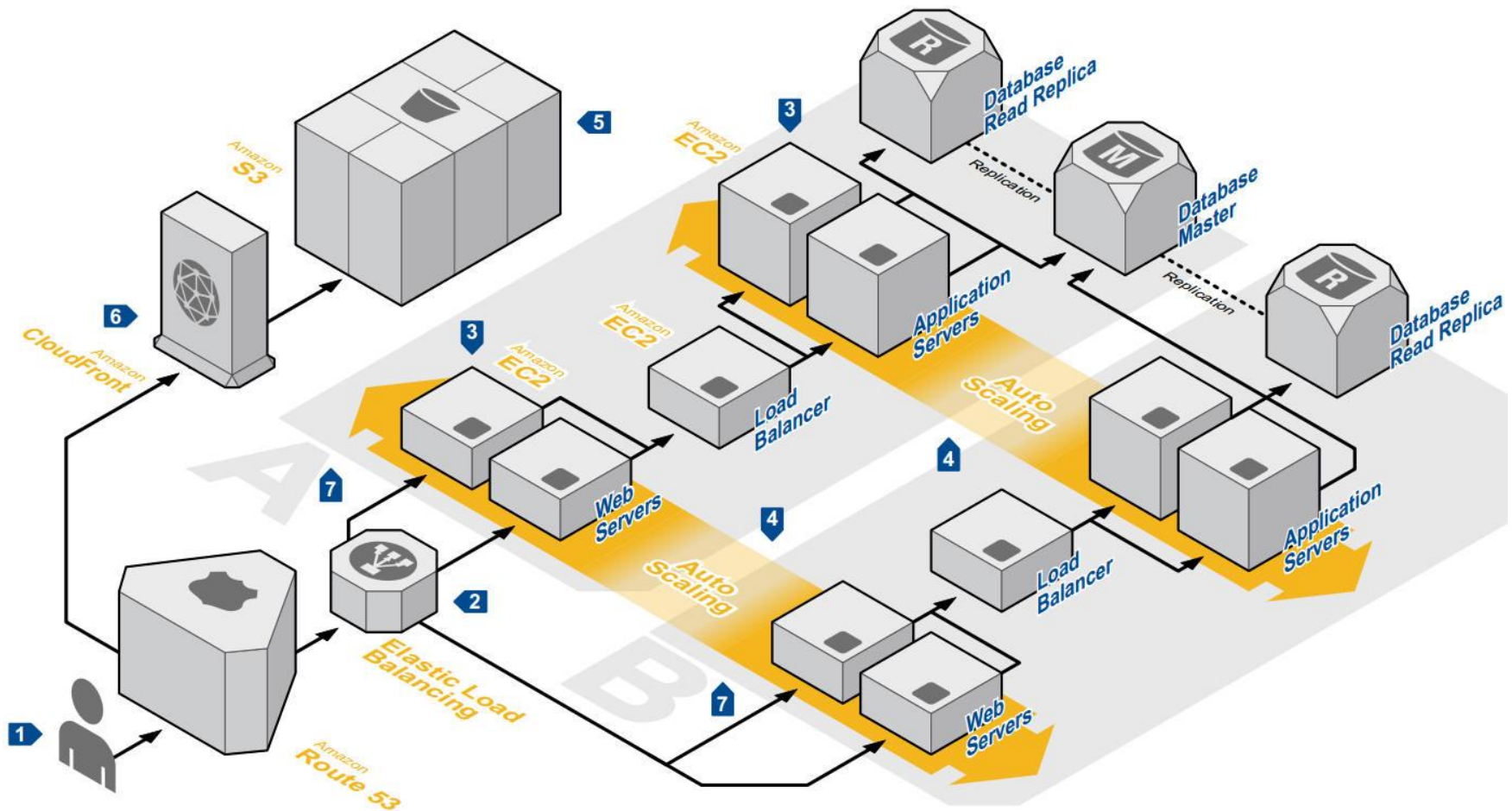


**S3**



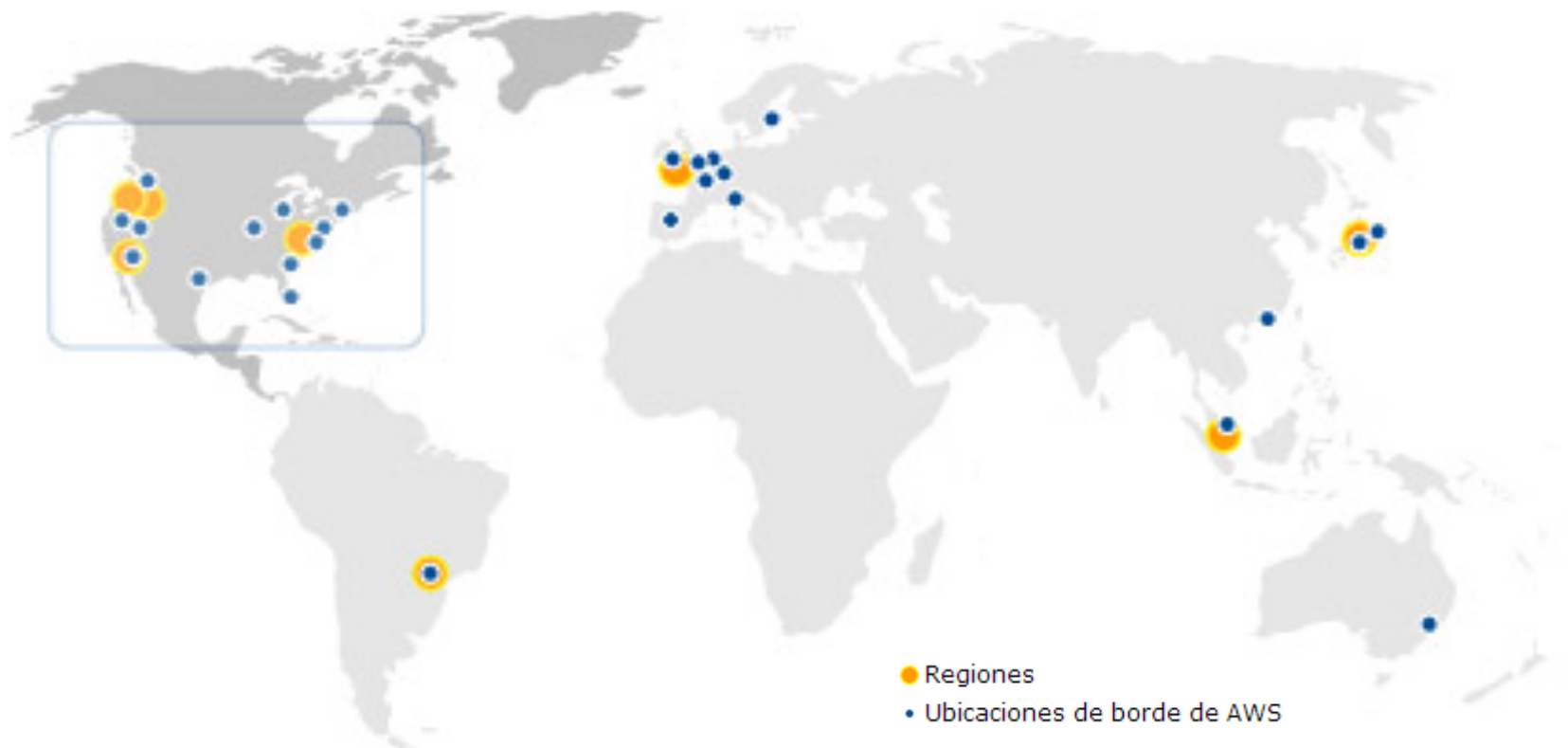
**CLOUDWATCH**

# Amazon Web Services (IV)



# Amazon Web Services (V)

## Regiones y Zonas de disponibilidad (AZ)



# Amazon Web Services (VI)



**LEY: LAS COSAS SE ROMPEN**

Imagen de <http://www.gen2phen.org/blogs/gt50/2009-05/too-much-data-causes-server-crash>

# Amazon EC2

Servicio de computación de AWS

- Máquinas virtuales (instancias EC2)
- Imágenes EC2 (AMI)
- Almacenamiento efímero (Instance Storage)
- Almacenamiento permanente (EBS)
- Seguridad perimetral (Security Groups)
- Balanceadores (ELB)
- IPs dinámicas por defecto, estática a petición (Elastic IP)

# Amazon Machine Images

- Son la base para crear una instancia EC2, y contienen una imagen del sistema operativo y configuración de la instancia
- Hay AMIs públicas y privadas
- Existe un [marketplace](#) para comprar AMI con modelo de pago por uso
- Puedes crear las tuyas propias



# Instancias EC2 (I)

Máquinas virtuales basadas en virtualización Xen:

- Tiene [tipos](#) que definen el tamaño y performance de CPU, RAM, IO, etc. de la máquina virtual
- Una instancia se lanza desde una imagen “[Amazon Image Machine](#)” o AMI
- Puede arrancar desde almacenamiento del host, [instance storage](#), (almacenamiento efímero) o desde un disco duro de red llamado [Elastic Block Storage](#) (persistente) o EBS

# Instancias EC2 (II)

- Tiene IP dinámica
- Linux y Windows
- Una instancia está en una zona de disponibilidad y se puede comunicar con instancia de otras zonas de disponibilidad por red local
- Pertenece a un Security Group y le aplican por tanto sus reglas

# Elastic Block Storage

Almacenamiento persistente:

- almacenamiento de red
- funciona como un dispositivo de bloques al conectarse a una instancia EC2
- soporte para realizar snapshots almacenados en Amazon S3
- IOPS asegurados si lo necesitamos
- posibilidad de conectar varios y montar RAID por ejemplo

# Elastic Load Balancer

Es un balanceador de carga

- soporta HTTP, HTTPS, TCP, SSL
- terminación SSL
- se le añaden instancias EC2 a las que repartir tráfico
- probes configurables
- accesible desde internet o como balanceador interno si se usa en conjunto con AWS VPC

# Security Groups

Viene a ser como un firewall perimetral

- cada instancia pertenece a un único security group
- se pueden definir reglas entrantes y salientes por IP, subredes y grupos de seguridad propios o de terceros

# Autoscaling (I)

Permite escalar la capacidad de procesamiento:

- ahorro de costes
- aumenta la seguridad y disponibilidad del negocio
- hay diferentes tipos de escalado
  - por horarios (scheduled scaling)
  - dinámico (dynamic scaling)
  - manual
- no existe actualmente gestión desde la consola web

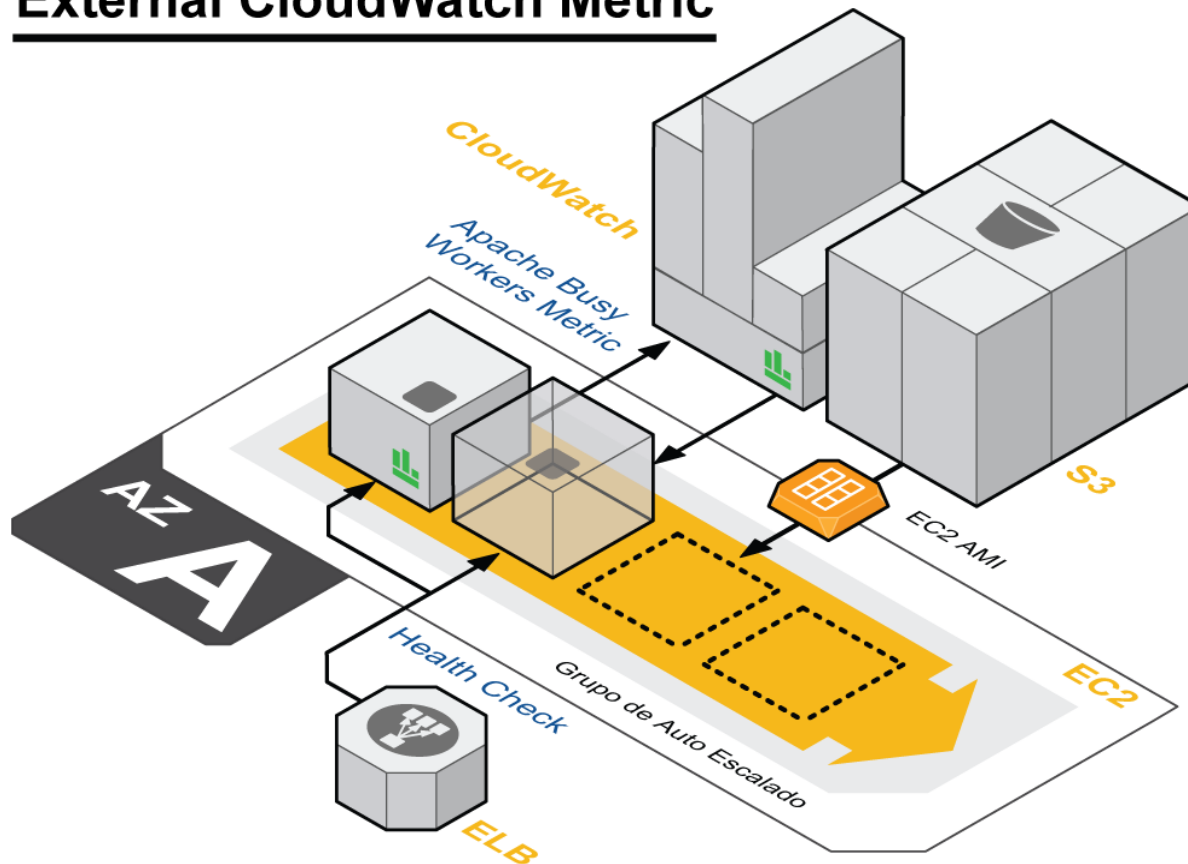
# Autoscaling (II)

Se define con

- configuración: AMI origen, security group, tipo de instancia, key, etc.
- grupo de autoescalado: zonas de disponibilidad en las que escalar, balanceador, algoritmos, etc.
- políticas autoscaling: como escala y en base a qué
- monitorización: métricas de cloudwatch para lanzar las políticas de autoscaling

# Autoscaling (III)

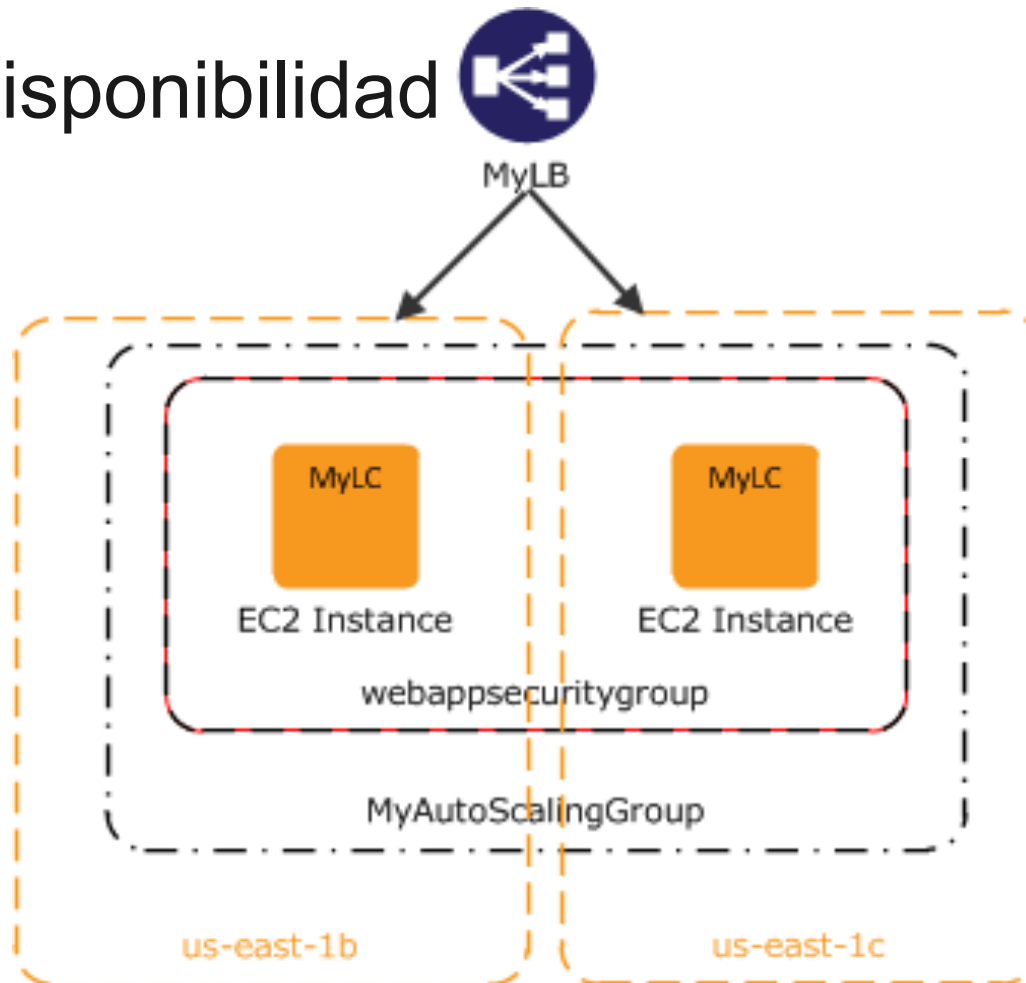
## AWS EC2 Auto Scaling: External CloudWatch Metric





# Autoscaling (IV)

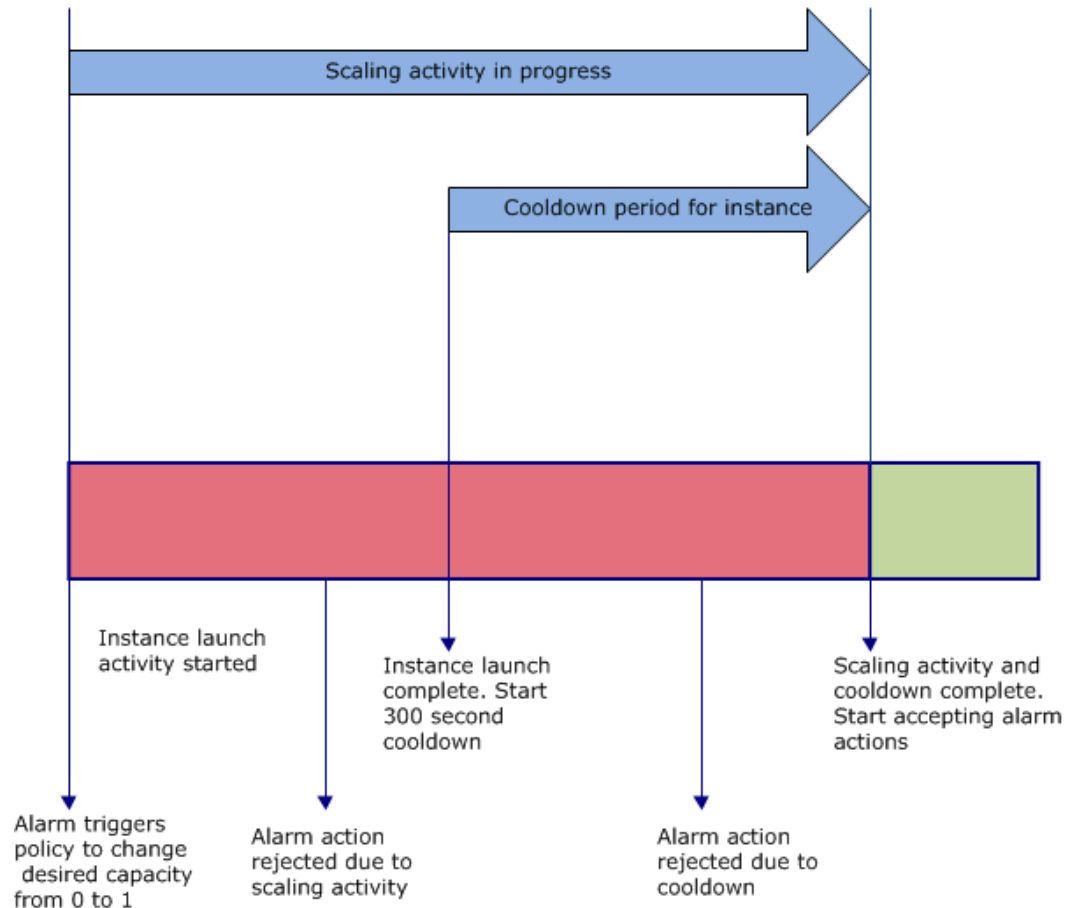
Zonas de disponibilidad



# Autoscaling (V)

## Cooldown

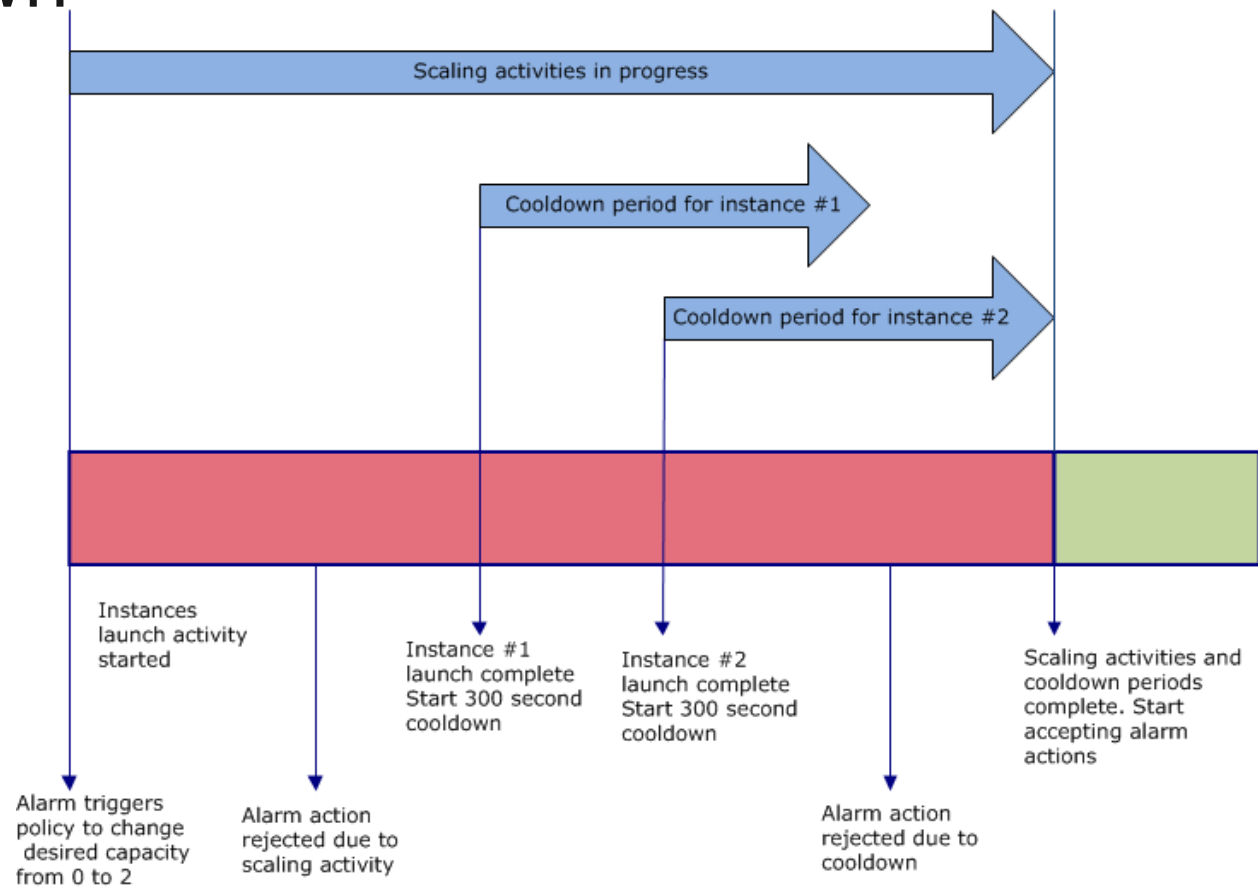
Scaling Activity and Cooldown Period for One Instance



# Autoscaling (VI)

## Cooldown

Scaling Activities and Cooldown Periods for Multiple Instances



# Autoscaling (VII)

Autoescalado dinámico:

- debe hacerse en varias zonas de disponibilidad
- servicios de Internet necesitamos un ELB
- evitar flapeos, coger buenas métricas
- una instancia EC2 no es inmediata
- herramientas configuración y deploy
- en entornos web tener en cuenta las sesiones
- hacer números y pruebas ¿qué es mejor 6 instancias small 10 horas + 2 small 14 horas o 6 medium 10 horas + 2 medium 14 horas?

# Autoscaling (VIII)

Veámoslo en la práctica:

1. Configurar security group
2. Instancia EC2 manual y pruebas
3. Crear ELB y pruebas
4. Crear configuración de autoscaling
5. Crear grupo de autoscaling
6. Escalado manual
7. Escalado dinámico

# Comandos (I)

Variables de entorno

- AWS\_ACCESS\_KEY
- AWS\_SECRET\_KEY
- AWS\_CREDENTIAL\_FILE

Fichero de credenciales

```
AWSAccessKeyId=<id_aws>  
AWSSecretKey=<key_aws>
```

# Comandos (II)

## Comandos (configuración de lanzamiento)

- as-create-launch-config
- as-describe-launch-configs
- as-delete-launch-config

# Comandos (III)

## Comandos (configuración de lanzamiento)

- as-create-auto-scaling-group
- as-update-auto-scaling-group
- as-describe-auto-scaling-groups
- as-delete-auto-scaling-group
- as-describe-auto-scaling-instances
- as-set-instance-health
- as-describe-scaling-activities



# Comandos (IV)

## Comandos

- as-put-scaling-policy
- as-describe-policies
- as-delete-policy
- as-put-scheduled-update-group-action
- as-describe-scheduled-actions
- as-delete-scheduled-action
- mon-put-metric-alarm
- mon-describe-alarms
- mon-delete-alarms

# GRACIAS

 [www.strsistemas.com](http://www.strsistemas.com)

 [@STRSistemas](https://twitter.com/STRSistemas)

 [info@strsistemas.com](mailto:info@strsistemas.com)

 902027609



