

# Tema 4

## Balanceo de carga



José Manuel Soto Hidalgo  
Dpto. Arquitectura y Tecnología de Computadores  
Universidad de Granada

jmsoto@ugr.es

# Índice



- [ 1. Introducción ]
- 2. Funcionamiento básico de un servidor
- 3. Conceptos del balanceo de carga
- 4. Otras tecnologías
- 5. Estructura de la red
- 6. Algoritmos de balanceo de carga
  
- 7. Balanceo de carga global
- 8. Ejemplo 1
- 9. Ejemplo 2
- 10. Ejemplo 3
- 11. Futuro de las tecnologías de balanceo
- 12. Resumen y conclusiones

# Introducción

Inicialmente se usaban grandes mainframes como servidores

Caros de adquirir y mantener.



Hardware y software propietario.



Memoria y disco compartidos.

Las desventajas llevaron al desarrollo del balanceo de carga.

# Introducción



Varias máquinas trabajando en paralelo es mejor que una sola máquina muy grande (y cara).

Si una máquina del grupo se rompe, se puede sustituir fácilmente.



Las tareas se pueden repartir entre grupos de servidores.

Crear una granja web es más barato que adquirir un mainframe de la misma capacidad.



# Introducción

## Ejercicio T4.1:

*Buscar información sobre cuánto costaría en la actualidad un mainframe que tuviera las mismas prestaciones que una granja web con balanceo de carga y 10 servidores finales. Se deja a vuestra elección la capacidad de cómputo del mainframe. Por ejemplo, 10 RaspberryPi 4 frente a un mainframe*

*Comparar precio y potencia entre esa hipotética máquina y la granja web de unas prestaciones similares.*

# Introducción

El balanceo de carga distribuye el tráfico entre varios servidores.

Un dispositivo intercepta el tráfico, lo analiza y lo distribuye:



# Introducción

Existen diversos dispositivos hardware y software que pueden hacer funciones de balanceo:

- p.ej. los routers derivan los paquetes por diferentes caminos

Por otro lado, el balanceador realiza tareas adicionales:

- comprobar la disponibilidad de los servidores
- protege de diversos ataques
- derivar en función del tipo de tráfico

# Índice



1. Introducción
- 2. Funcionamiento básico de un servidor**
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
  
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Funcionamiento básico de un servidor

Proceso tras teclear una URL en el navegador:

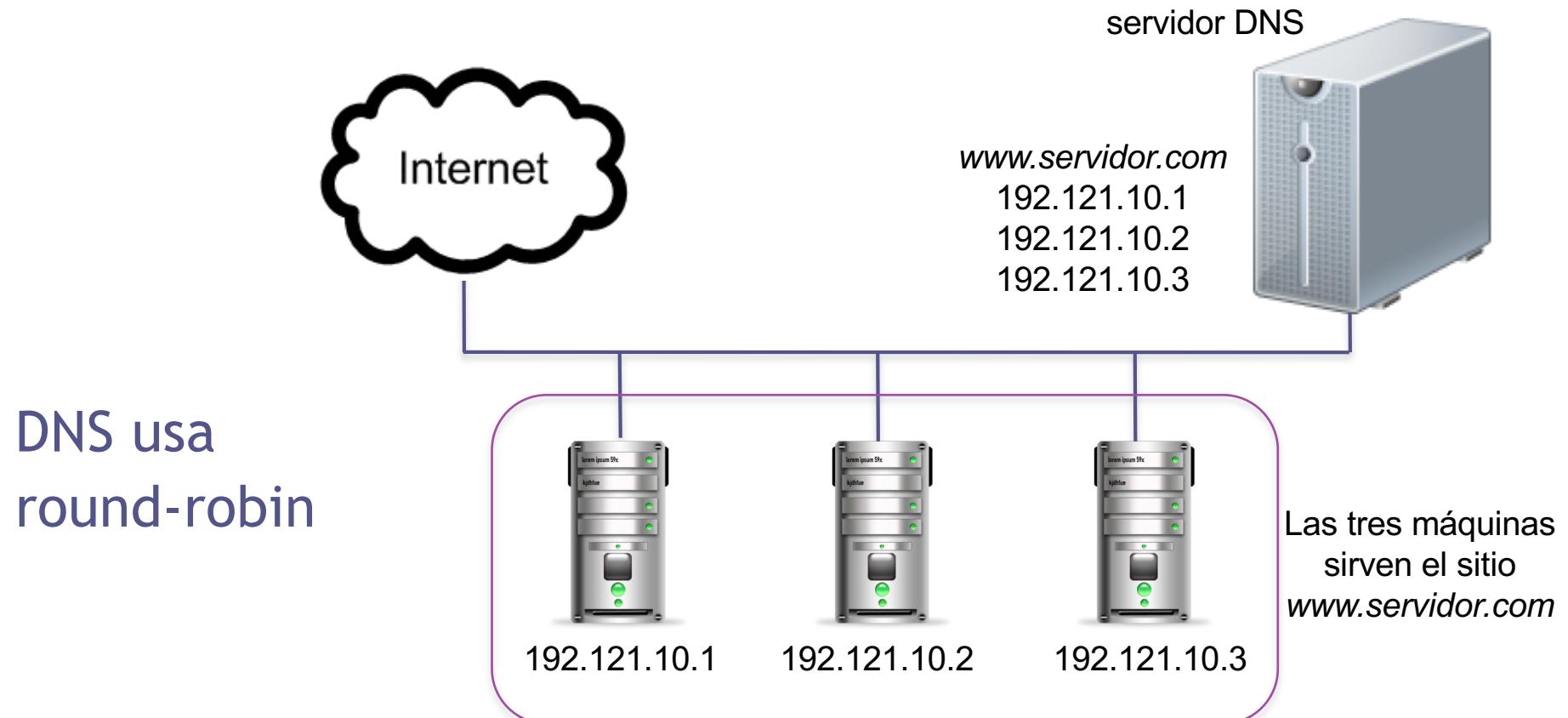
1. El navegador resuelve el nombre del sitio web (DNS).
2. Ya con la IP, el navegador contacta por TCP.
3. Justo a continuación, hace la petición HTTP.
4. El servidor devuelve la página HTML completa.
5. El navegador muestra la información.

¿Y si queremos que varias máquinas sirvan nuestro contenido web? (para disponer de más prestaciones)

# Funcionamiento básico de un servidor

## (1) Balanceo mediante DNS

Configurar el DNS asignando tres IP diferentes al mismo sitio  
(misma dirección)



# Funcionamiento básico de un servidor

## (1) Balanceo mediante DNS

Es una primera aproximación un poco rudimentaria.



Es muy sencilla de implementar.



DNS no tiene información sobre la disponibilidad de cada máquina servidora final.



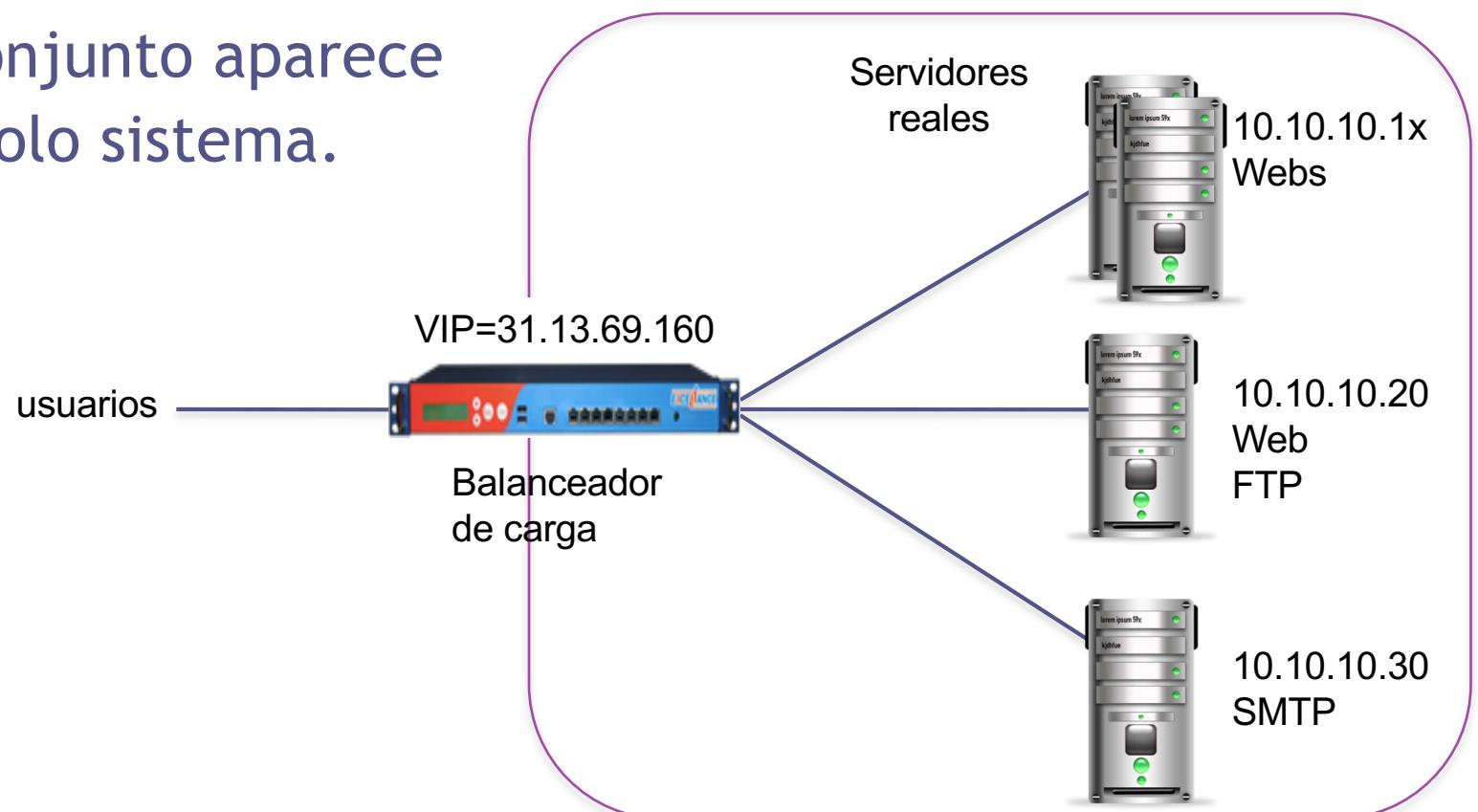
¡Los DNS se inventaron para resolver (traducir) nombres a IP!

# Funcionamiento básico de un servidor

## (2) Granja web con balanceo de carga

El balanceador se configura como punto de entrada para los usuarios.

Todo el conjunto aparece como un solo sistema.



## Funcionamiento básico de un servidor

### (2) Granja web con balanceo de carga

El usuario recibe del DNS la IP pública del balanceador (VIP), que se ocupa en distribuir el tráfico.

Ventajas:



- Mejora la escalabilidad
- Mejora la disponibilidad
- Facilita el mantenimiento (arreglar o añadir máquinas sobre la marcha)
- Mejora la seguridad (servidores finales están protegidos)

# Índice



1. Introducción
2. Funcionamiento básico de un servidor
- [ 3. Conceptos del balanceo de carga ]**
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Conceptos del balanceo de carga

Tener en cuenta algunos conceptos y términos:

- Modelo OSI
- Balanceador de carga
- VIP
- Servidor
- Grupo o cluster
- Niveles de acceso de usuario
- Redundancia
- Persistencia
- Disponibilidad de servicio
- Algoritmos de balanceo de carga
- Centro de datos
- NAT: network addess translation

# Conceptos del balanceo de carga

## Modelo OSI

Los concentradores y routers trabajan con IP y MAC.  
Reenvían tráfico a servidores.

No pueden reenviar tráfico concreto a un servidor y para una aplicación concreta (no tienen en cuenta puertos).

No pueden comprobar si una máquina está disponible.

Se basan en el modelo OSI:

# Conceptos del balanceo de carga

## Modelo OSI

Los protocolos IP, TCP, UDP, HTTP se mapean en las 7 capas.

SWAP

Capa OSI	Función	Unidades	Ejemplo	Relación con el balanceo de carga
Capa 1	Física	Bits (unos y ceros)	Cable, fibra SX	Cable usado para conectar los switches y hubs
Capa 2	Enlace	Tramas Ethernet	Switches, hubs	Dispositivos que gestionan el tráfico
Capa 3	Red	Direcciones IP	Routers	Dispositivos con características de router
Capa 4	Transporte	TCP, UDP, ICMP	Puerto TCP 80 para http, puerto UDP 161 para SNMP	Nivel típico del balanceo. Principalmente se trabaja con IP y puertos
Capas 5-7	Sesión, presentación y aplicación	URL, cookie, etc	<a href="http://www.servidor.dom">http://www.servidor.dom</a> cookie	Informaciones incluidas en las URL o en las cookies

# Conceptos del balanceo de carga

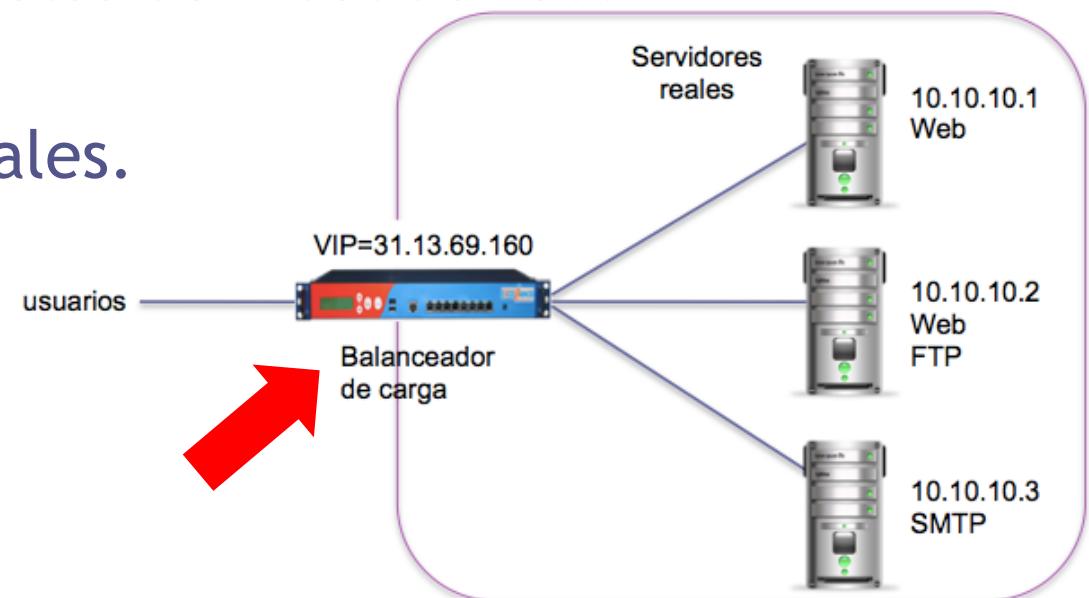
## Balanceador de carga

Es un puente entre la red y los servidores finales.

Puede manejar los protocolos de alto nivel.

Puede manejar los protocolos de nivel de red.

Protege los servidores finales.



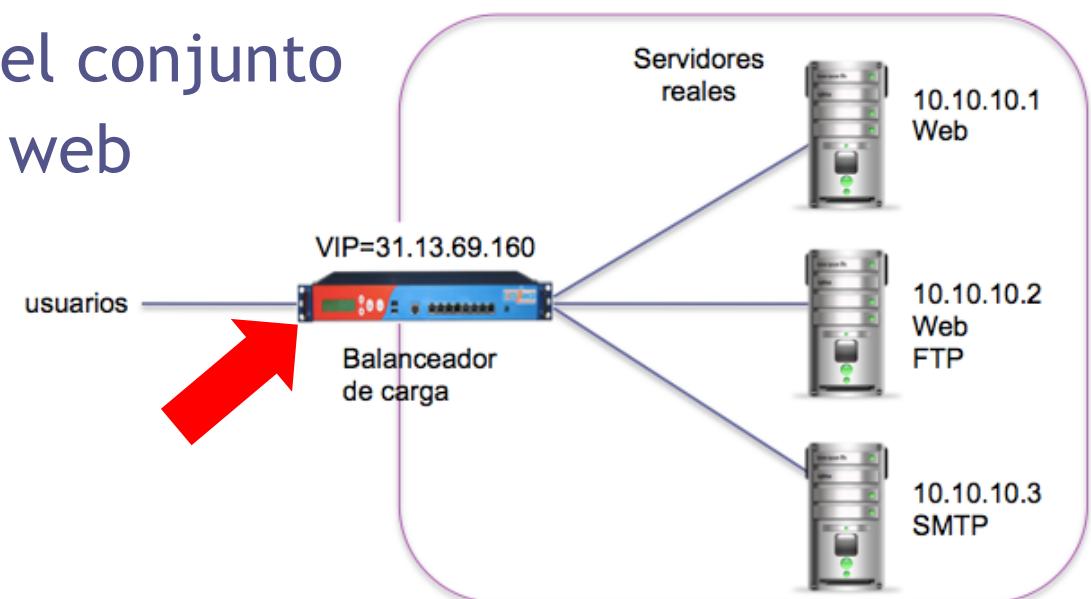
# Conceptos del balanceo de carga

## Virtual IP (VIP)

Dirección pública a la que acceden los usuarios cuando piden un servicio al sistema.

Suele ser la IP del balanceador de carga.

Es la forma de hacer que el conjunto de máquinas de la granja web aparezca como una sola.



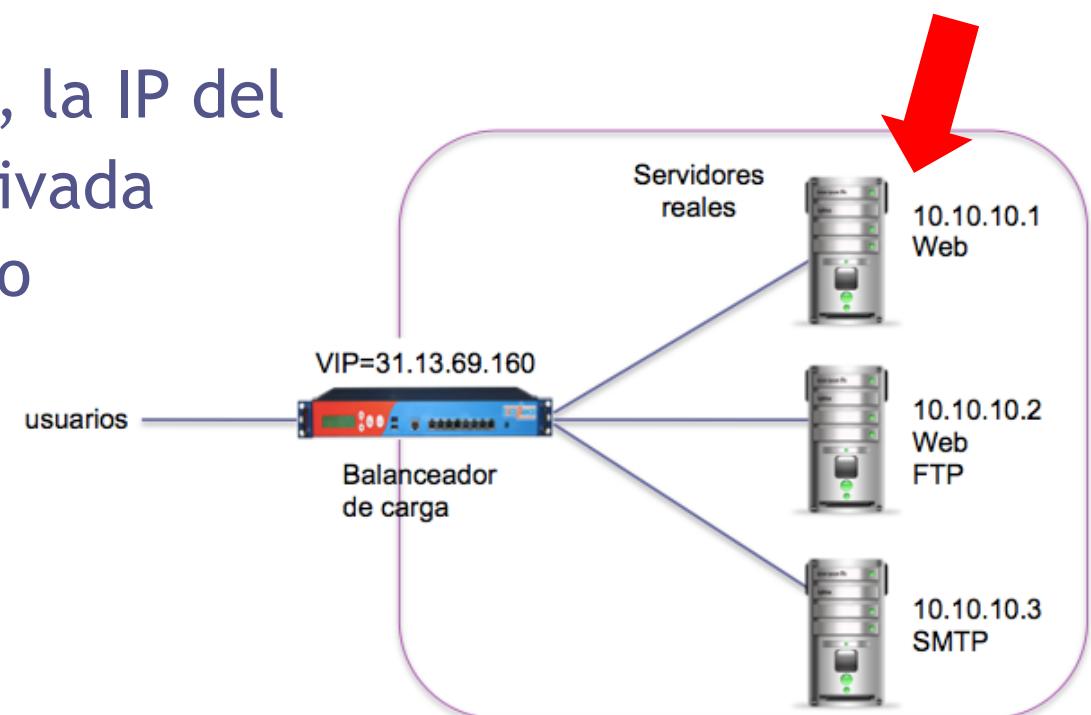
# Conceptos del balanceo de carga

## Servidor

Dispositivo que ejecuta un servicio.

En nuestro caso nos referiremos a servidores HTTP.

Según la topología de red, la IP del servidor será pública o privada (accesible a través de otro dispositivo de red que haga NAT).

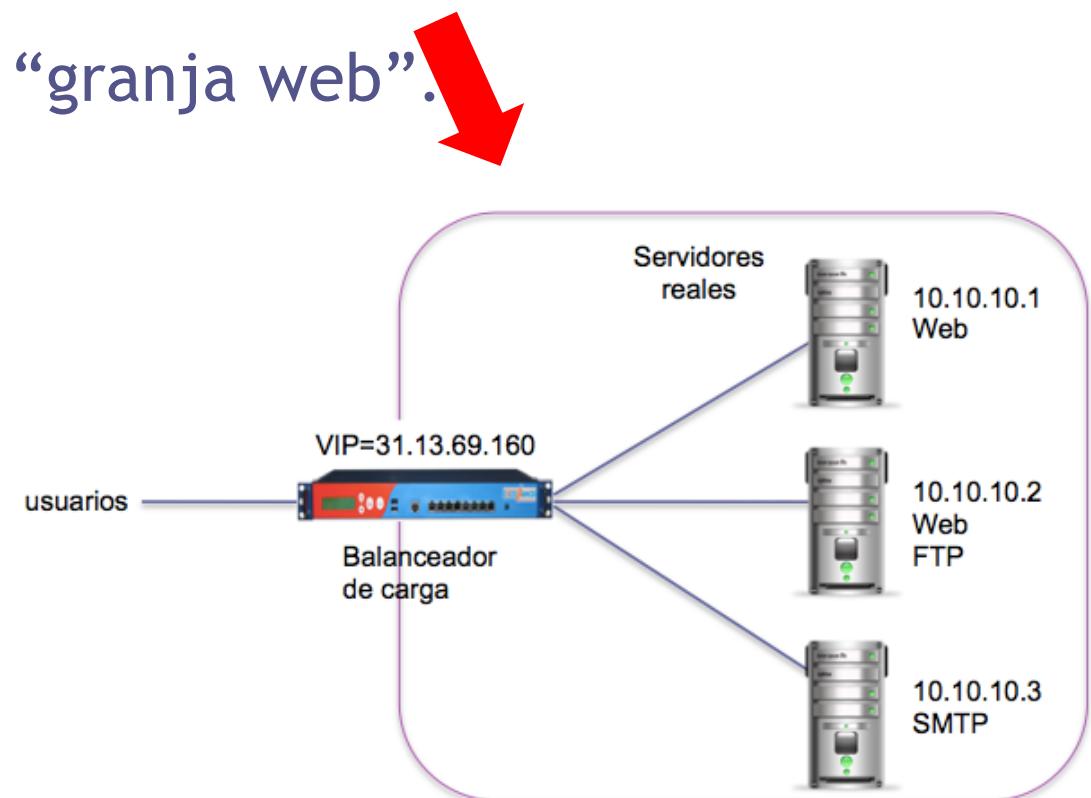


# Conceptos del balanceo de carga

## Grupo o cluster

Conjunto de servidores con un平衡ador al frente para repartir la carga.

Lo que estamos llamando “granja web”.



# Conceptos del balanceo de carga

## Niveles de acceso de usuario

Nos referimos a los permisos de control de los usuarios.

Ej: sólo lectura, superusuario

Entre las políticas de acceso y seguridad se suele definir un tipo de usuario “administrador-web”, con más privilegios que un usuario normal, pero menos que el “root”, y que se utiliza para administrar las configuraciones de los servidores que tienen que ver con la web.

# Conceptos del balanceo de carga

## Redundancia

Si un dispositivo falla, otro pasa a hacer su función.

Se trata de evitar la degradación del servicio.

Se suelen poner dos dispositivos idénticos que monitoricen el estado del otro:

- relación maestro-maestro
- relación maestro-esclavo

Protocolo VRRP (Virtual Router Redundancy Protocol).

# Conceptos del balanceo de carga

## Persistencia (sesiones de navegación)

Se refiere a mantener el tráfico de un usuario dirigido al servidor que comenzó a atenderlo.

El balanceador elige a qué servidor final dirige el tráfico, pero a partir de ahí, seguirá dirigiéndolo mientras dure la sesión de navegación.

Es importante en aplicaciones web que almacenan el estado del usuario mientras navega (ej. tienda web).

Otra opción es usar una memoria compartida-distribuida entre los servidores (memcache).

# Conceptos del balanceo de carga

## Comprobación de disponibilidad de servicio

El balanceador debe comprobar regularmente si un servidor está activo o caído para reenviarle más tráfico.

Se puede implementar con ICMP (ping) a los servidores finales.

También comprobando contenido: accede a un servicio web específico y se espera una respuesta concreta por parte del servidor.

# Conceptos del balanceo de carga

## Algoritmos de balanceo de carga

Hay varios algoritmos o estrategias de distribución del tráfico entre el grupo de servidores:

- por turnos, prioridad, ponderación, número de conexiones, etc.

El balanceador los implementa y los aplica según la configuración que hagamos.

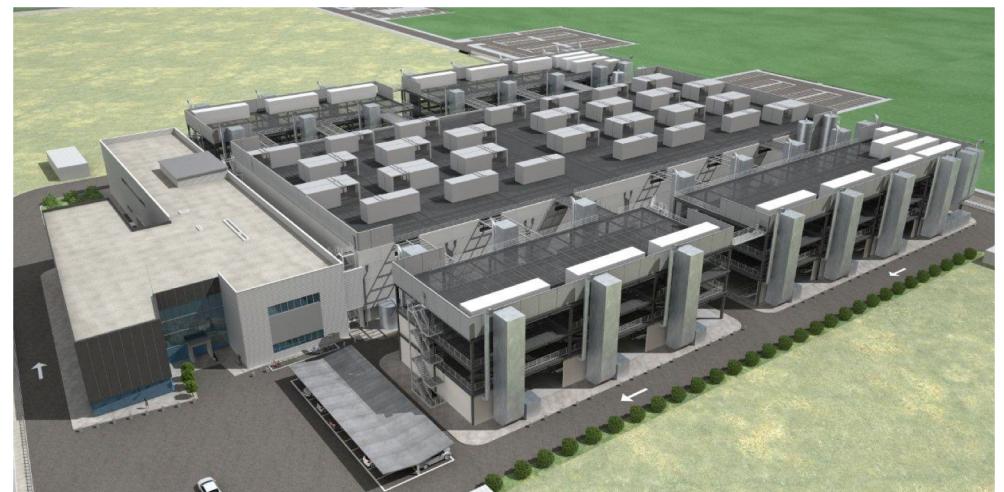
# Conceptos del balanceo de carga

## Centro de datos

Los equipos hardware están hospedados en un edificio con alta seguridad, refrigeración controlada, alimentación continua y sistemas antiincendios específicos.

Se ocupa de la seguridad, alimentación, refrigeración y conexión.

Ahorro en mantenimiento.



# Conceptos del balanceo de carga

## NAT: Network Address Translation

El balanceador de carga hace traducción de direcciones.

Toma los paquetes de entrada y cambia la IP de destino (la VIP por la IP privada de un servidor real final).

Cuando el servidor sirve contenido, el balanceador hace el cambio de nuevo (pone la VIP en lugar de la IP final).

Opción más eficiente =>

el servidor final ya genera los paquetes con la VIP como origen (el balanceador se evita un trabajo).

# Conceptos del balanceo de carga



el cliente hace una petición enviando un paquete:

$SRC=56.56.56.56\ DST=88.88.88.88$

Traducción =>  $SRC=88.88.88.88\ DST=172.16.0.5$

el servidor procesa la petición y devuelve la respuesta en un paquete:

$SRC=172.16.0.5\ DST=88.88.88.88$

Traducción =>  $SRC=88.88.88.88\ DST=56.56.56.56$

Opción más eficiente =>

el servidor final ya genera los paquetes con la VIP como origen (el balanceador se evita un trabajo).

# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Otras tecnologías similares

El balanceo de carga consiste en interceptar el tráfico para redirigirlo a varios servidores.

Hay otras tecnologías similares:

- Balanceo de carga en cortafuegos
- Balanceo de carga global
- Clustering

# Otras tecnologías similares

## Balanceo de carga en cortafuegos

Se trata de estructurar varios cortafuegos de forma balanceada.

Normalmente, un cortafuegos es una máquina más.

La CPU tiene una limitación: más de 80Mbps

En algunos casos puede no ser suficiente.

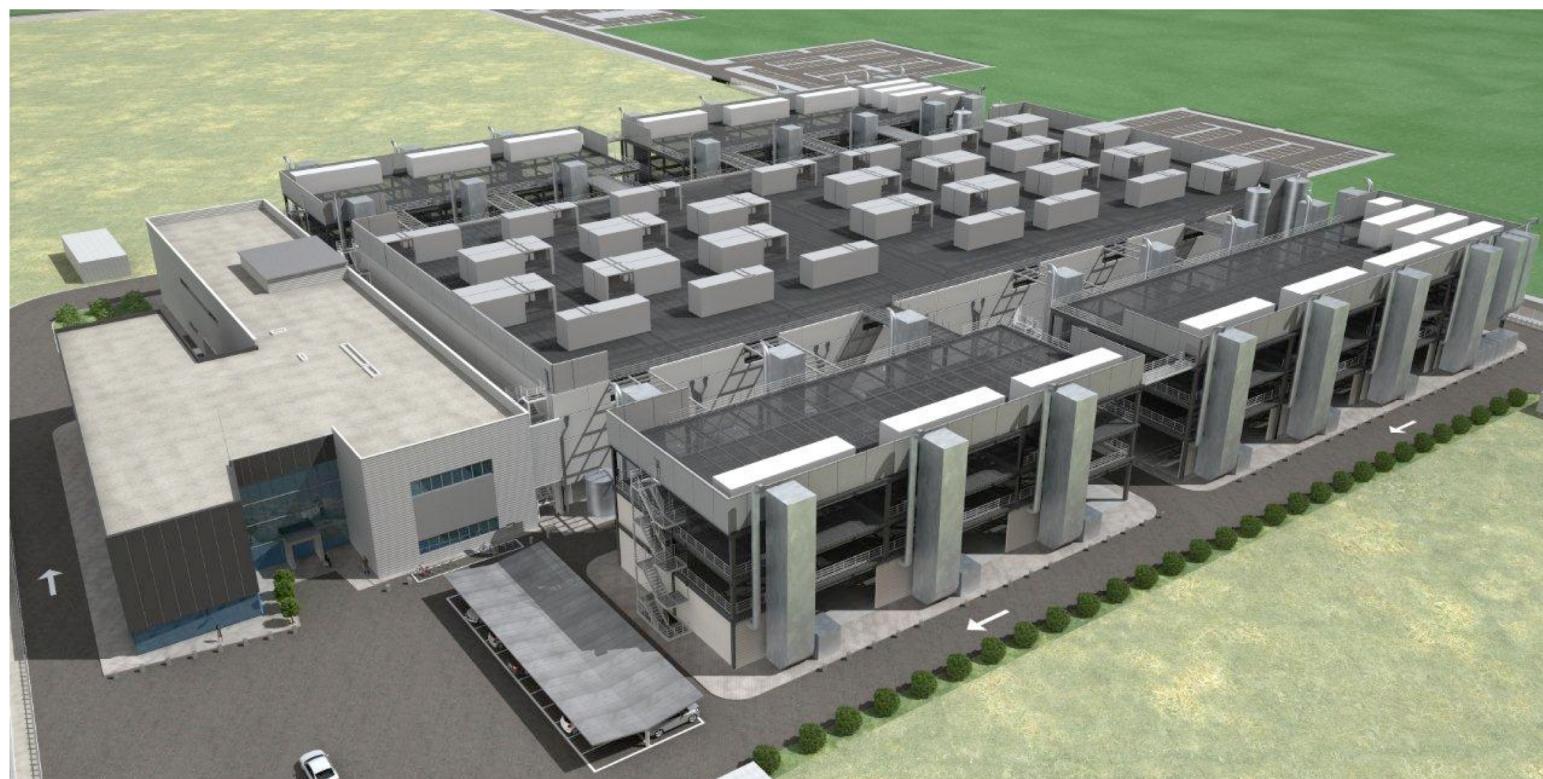


Configurar varias máquinas como cortafuegos balanceados.

# Otras tecnologías similares

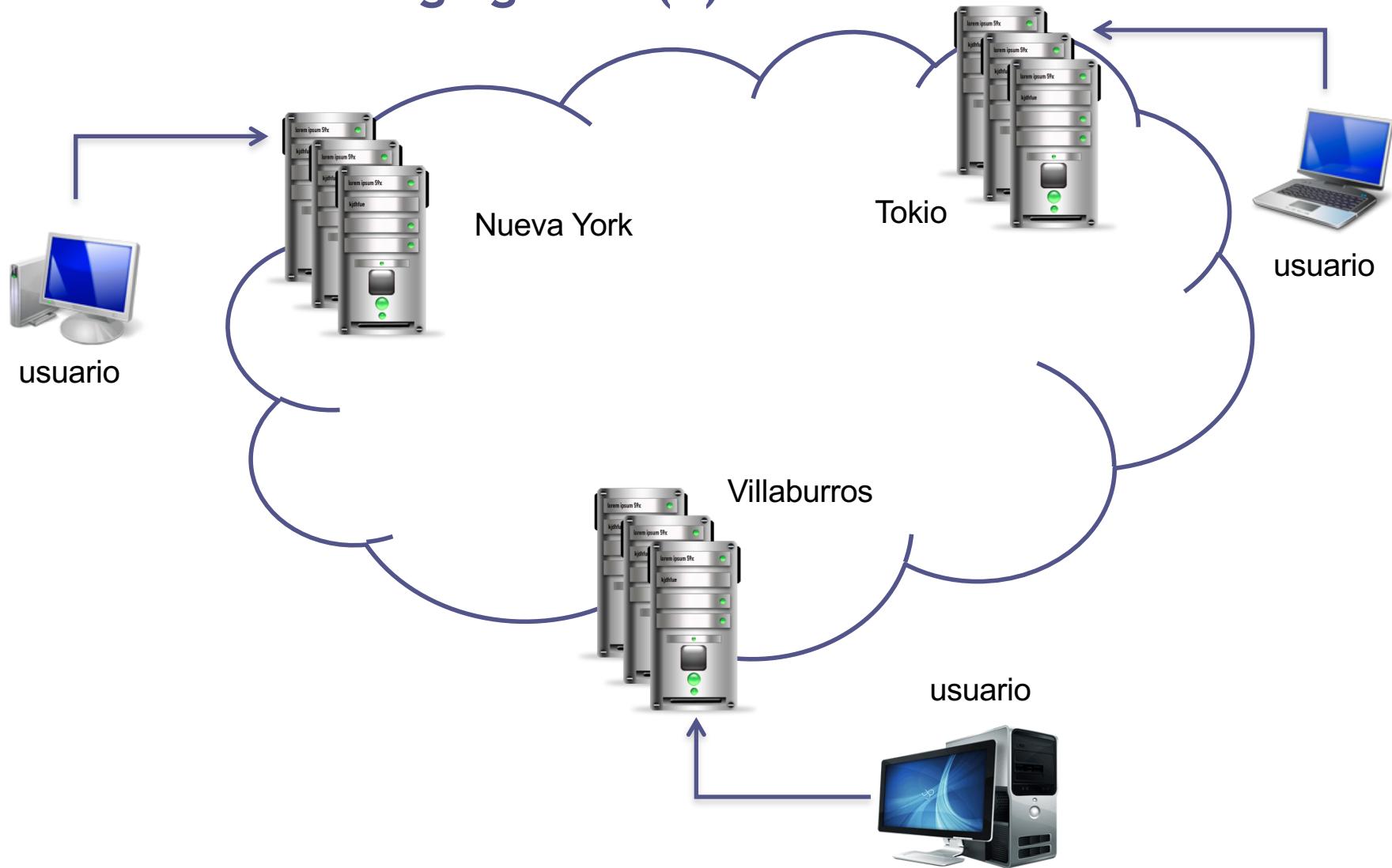
## Balanceo de carga global (I)

Realizar balanceo de carga, similar al estudiado, pero entre centros de datos.



# Otras tecnologías similares

## Balanceo de carga global (II)



# Otras tecnologías similares

## Balanceo de carga global (III)

Balanceo de carga entre centros de datos.

El tráfico de cierto usuario va desde su navegador cliente hasta el centro de datos más cercano geográficamente.

Una vez que las peticiones llegan a cierto centro de datos, se balancea también para distribuir entre el grupo de servidores.

# Otras tecnologías similares

## Balanceo de carga global (IV)

Se evita un retraso por el viaje de los paquetes a un centro situado a miles de kilómetros.

Redundancia y balanceo: si un centro falla (corte de luz, maremoto, etc) el tráfico se redirige automáticamente a otro centro disponible.

Se puede implementar con los DNS o con BGP (*border gateway protocol*).

# Otras tecnologías similares

## Clustering

Solución basada en la disponibilidad y escalabilidad.

Técnica orientada a disponer de un cluster de máquinas que aceptarán grandes tareas computacionales.

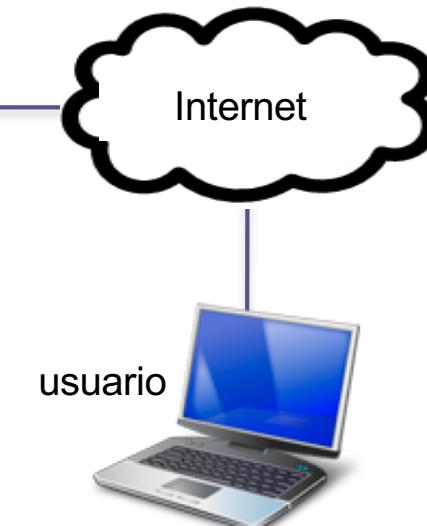
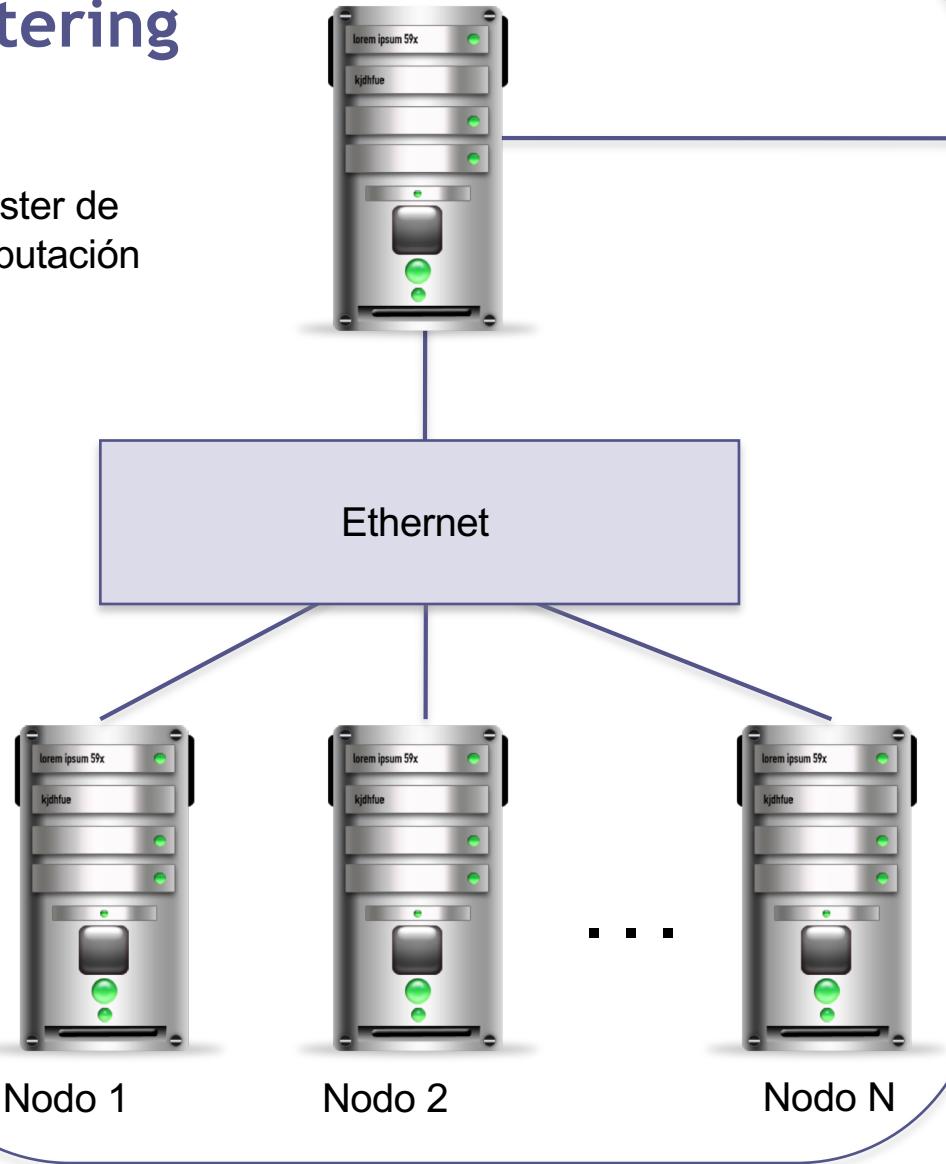
Ejecución paralela repartiendo el trabajo entre varias máquinas.

*Se accede por la máquina principal para lanzar trabajos (carga), y las máquinas del cluster trabajan juntas para dar el servicio (en este caso, de computación en lugar de servir peticiones)*

# Otras tecnologías similares

## Clustering

Cluster de computación



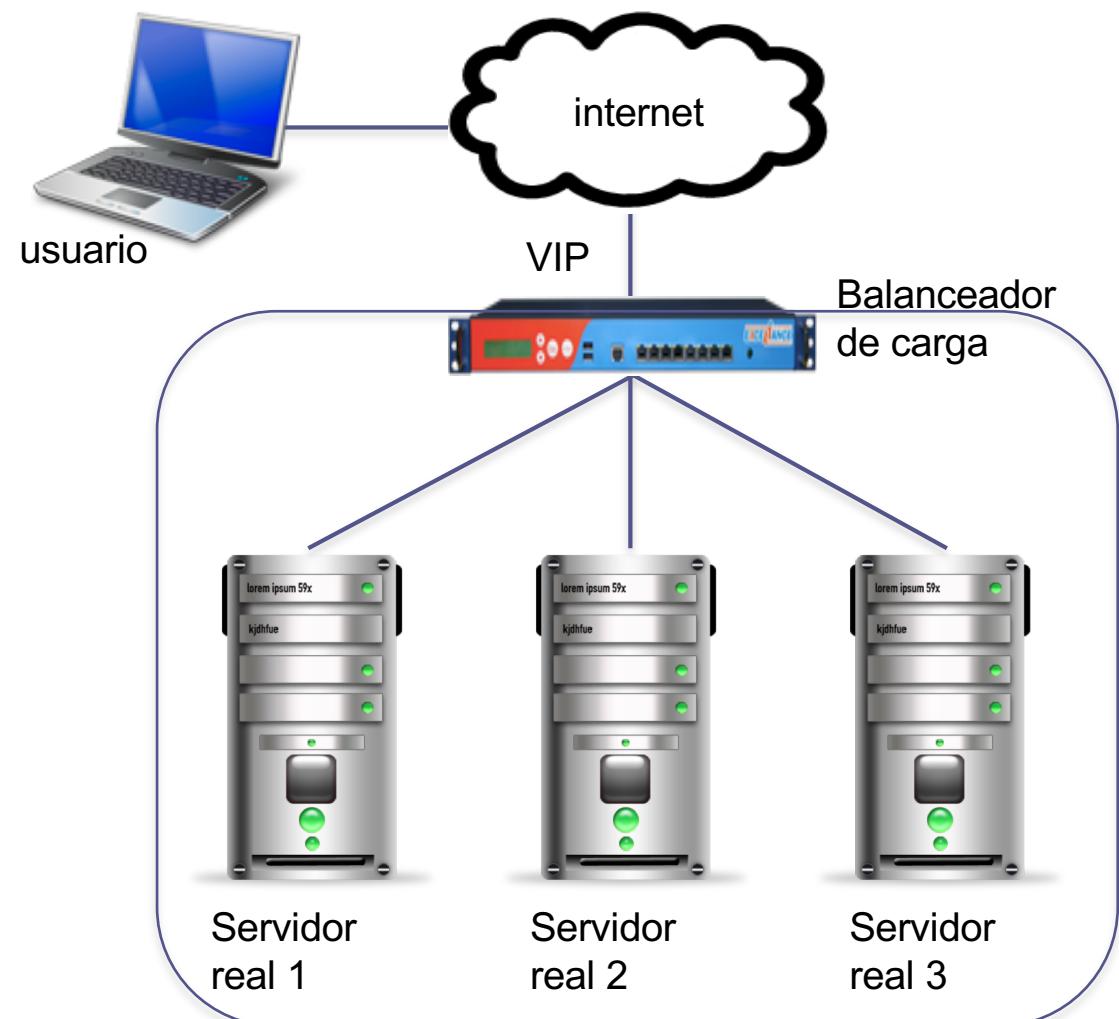
# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Estructura de la red

Una instalación basada en el uso de balanceador queda representada en el siguiente esquema:



# Estructura de la red

El tráfico generado por el usuario va a través del balanceador hasta la máquina servidora final.

La respuesta va hasta el cliente a través del balanceador.

La configuración se basará en (1) servidor con software específico, o bien (2) concentrador o switch con funciones de balanceo.

Cada opción tiene ventajas e inconvenientes...



# Estructura de la red

## (1) Configurar un servidor como balanceador

Existe software específico para configurar un PC como balanceador. Casi cualquier SO servirá.

Usa algoritmos de reparto de carga conocidos que podemos configurar.

# Estructura de la red

## (1) Configurar un servidor como balanceador

Opciones de software libre:

- HAProxy
- nginx
- Apache
- Pound

Opciones de software propietario:

- Local Director (cisco): [http://en.wikipedia.org/wiki/Cisco\\_LocalDirector](http://en.wikipedia.org/wiki/Cisco_LocalDirector)
- BIG-IP (F5): <http://www.f5.com/products/big-ip/>
- NLB (Microsoft): <http://support.microsoft.com/kb/323437>

# Estructura de la red

## (1) Configurar un servidor como balanceador

Más opciones de software libre:

<https://geekflare.com/open-source-load-balancer/>

The screenshot shows a web browser window with the URL <https://geekflare.com/open-source-load-balancer/> in the address bar. The main content area displays the title **10 Open Source Load Balancer for HA and Improved Performance**. Below the title is a green graphic featuring a white cloud icon with three blue squares underneath it, representing a load balancer architecture.

**10 Open Source Load Balancer for HA and Improved Performance**

- [Seesaw](#)
- [LoadMaster by KEMP](#)
- [HAProxy](#)
- [ZEVENET](#)
- [Neutrino](#)
- [Balance](#)
- [Pen](#)
- [Nginx](#)
- [Traefik](#)
- [Gobetween](#)

# Un ejemplo: Balanceo con gobetween

- Disponible para varias plataformas.
- Balanceo de carga, balanceo usando Docker/Swarm, ElasticSearch balancing.

<http://gobetween.io/downloads.html>

The screenshot shows a web browser window with the URL <http://gobetween.io/downloads.html> in the address bar. The page itself has a blue header with the gobetween logo and navigation links for Documentation and Downloads. Below the header, there are download links for Linux (x86/x64), Windows (x86/x64), Darwin (x86/x64), and a Docker Image (yyyar/gobetween). Each download link is accompanied by a small icon representing the platform.

Plataforma	Formato	Links
Linux	(tar.gz)	<a href="#">Linux (tar.gz)</a> x86   x64
Windows	(zip)	<a href="#">Windows (zip)</a> x86   x64
Darwin	(zip)	<a href="#">Darwin (zip)</a> x86   x64
Docker Image		<a href="#">Docker Image</a> yyyar/gobetween

# Un ejemplo: Balanceo con gobetween

The screenshot shows a web browser displaying the [gobetween documentation](http://gobetween.io/documentation.html#Static-balancing). The URL bar shows the full address. The page has a blue header with the gobetween logo and navigation links for 'Documentation', 'Introduction', 'Installation', 'Configuration', 'Protocols', and 'Balancing'. The main content area is titled 'Simple load balancing' and contains a diagram illustrating the flow from 'Internet' to a 'gobetween' proxy, which then routes traffic to three 'backend' boxes labeled 'backend-1', 'backend-3', and 'backend-3'.

Simple load balancing

```
graph LR; Internet((Internet)) --> gobetween[gobetween]; gobetween --> backend1[backend-1]; gobetween --> backend3_1[backend-3]; gobetween --> backend3_2[backend-3]
```

<http://gobetween.io/downloads.html>

# Balanceo con dispositivos hardware

## Ejercicio T4.2:

*Instala y configura en una máquina virtual el balanceador gobetween).*

*Compara con la dificultad de la instalación y configuración usando nginx o haproxy (práctica 3).*

# Estructura de la red

## (2) Usar un dispositivo balanceador específico:

Dispositivos tipo “caja negra” que incluyen hardware y software para el balanceo.

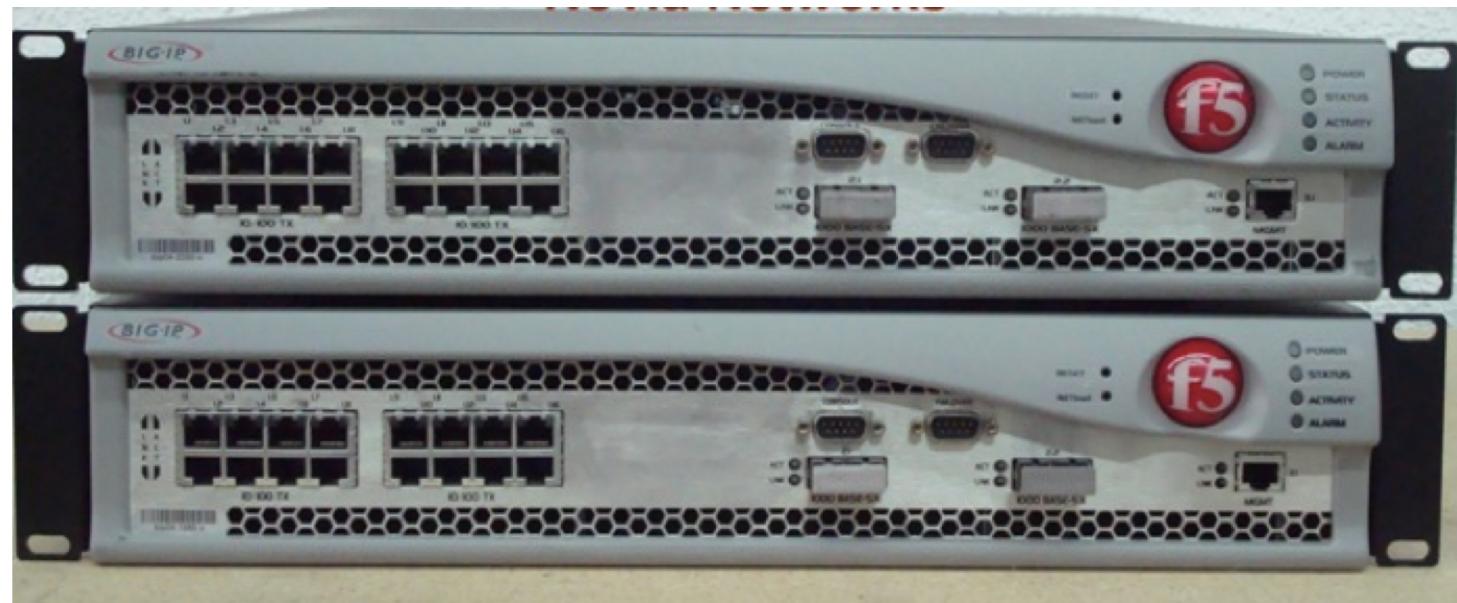
Un primer tipo usa **procesadores específicos (ASIC, application specific integrated circuit)** para realizar las tareas de modificación de paquetes.



# Estructura de la red

## (2) Usar un dispositivo balanceador específico:

Procesadores extremadamente rápidos haciendo esas modificaciones (aunque no pueden realizar otras tareas que sí pueden hacer las CPUs de propósito general).



# Estructura de la red

(2) Usar un dispositivo balanceador específico:



MODEL COMPARISON	MODEL 240	MODEL 340	MODEL 440	MODEL 640
<b>CAPACITY*</b>	<b>\$1,424</b>	<b>\$1,899</b>	<b>\$3,799</b>	<b>\$8,549</b>
Maximum Throughput	95 Mbps	950 Mbps	1 Gbps	4 Gbps
Real Server Support	10	35	50	250
SSL Offloading/Acceleration		500 TPS	4,000 TPS	15,000 TPS
<b>HARDWARE</b>				
Rackmount Chassis	1U Mini	1U Mini	1U Mini	1U Fullsize
Dimensions (in)	16.8 x 1.7 x 9	16.8 x 1.7 x 14	16.8 x 1.7 x 14	16.8 x 1.7 x 22.6
Dimensions (cm)	42.7 x 4.3 x 23	42.7 x 4.3 x 35.6	42.7 x 4.3 x 35.6	42.7 x 4.3 x 57.4
Weight (lbs/kg)	8 / 3.6	12 / 5.4	12 / 5.4	26 / 11.8
Ethernet	2 x 10/100	2 x Gigabit	2 x Gigabit	12 x Gigabit
AC Input Current (Amps)	1.0	1.2	1.4	1.8
ECC Memory				✓
<b>FEATURES</b>				
Layer 4 Load Balancing	✓	✓	✓	✓
Direct Server Return	✓	✓	✓	✓
Intrusion Prevention	✓	✓	✓	✓
High Availability		✓	✓	✓
VLAN		✓	✓	✓
Layer 7 Load Balancing		✓	✓	✓
SIP Call ID Persistence		✓	✓	✓
Cookie Persistence		✓	✓	✓
SSL Offloading		✓	✓	✓
Content Routing		✓	✓	✓
SNMP		✓	✓	✓
Programming Interface/API			✓	✓
Global Server Load Balancing			✓	✓
HTTP Compression			✓	✓
Content Caching			✓	✓
Link Bonding (LACP)				✓

# Estructura de la red

## (2) Usar un dispositivo balanceador específico:

Dispositivos que realmente son servidores con un SO comercial muy optimizado para realizar estas tareas.

Productos de Cisco Systems, Barracuda, Foundry Networks, Nortel Networks, F5 Networks y Radware.

Los balanceadores tipo caja-negra específicos resultan más rápidos que los basados en una máquina con software específico.

# Balanceadores hardware específicos

## Ejercicio T4.3:

*Buscar información sobre precio y características de balanceadores comerciales (hardware) específicos.*

*Compara las prestaciones que ofrecen unos y otros.*

# Balanceo con dispositivos hardware

Los balanceadores hardware están basados en un núcleo Linux/Unix que ejecuta los algoritmos estudiados.

Fabricantes:

- Cisco Systems: <http://www.cisco.com/web/ES/index.html>
- Foundry Networks: <http://www.brocade.com/index.page>
- Nortel Networks: <http://www.nortel-us.com/>
- F5 Networks: <http://www.f5.com/>
- Radware: <http://www.radware.com/>

# Balanceo con dispositivos hardware

La funcionalidad descrita hasta ahora está soportada por los balanceadores hardware.

Algunas funciones no se implementan en la mayoría del software de balanceo.

Los balanceadores hardware son más eficientes que los sistemas basados en un ordenador con software específico.

# Balanceo con dispositivos hardware

La configuración y control de su funcionalidad se puede hacer con interfaz de ventanas o por línea de comandos.

<https://www.zevenet.com/zen-load-balancer/>



The screenshot displays the ZEN Load Balancer interface. On the left, there's a large ZEVENET logo and the text "Zen Load Balancer". The main area shows a dashboard with two sections: "Manage::Farms::" and "Settings::Interfaces". The "Manage::Farms::" section includes fields for "Farm's name", "Backend response time", "Frequency to check results", and "Timeout request from client". The "Settings::Interfaces" section lists three interfaces (eth0, eth0.1, eth1) with their respective IP addresses, netmasks, and gateway settings. Below these are sections for "Default gateway" and "Actions". At the bottom, there's a modal window titled "Edit real IP servers configuration" containing a table with two rows for servers 0 and 1, showing columns for Server, Address, Port, Timeout, Weight, and Actions.

Server	Address	Port	Timeout	Weight	Actions
0	192.168.0.100	80	-	5	
1	192.168.0.101	80	-	3	

Cancel

# Balanceo con dispositivos hardware



Zen Load Balancer



- [Products](#) ▾
- [Services](#) ▾
- [Alliances](#) ▾

[ZVA64 EE 3110 Virtual Appliance](#)

[ZNA64 EE 3300 HW Appliance](#)

## ZNA64 Enterprise Edition 3300 Appliance



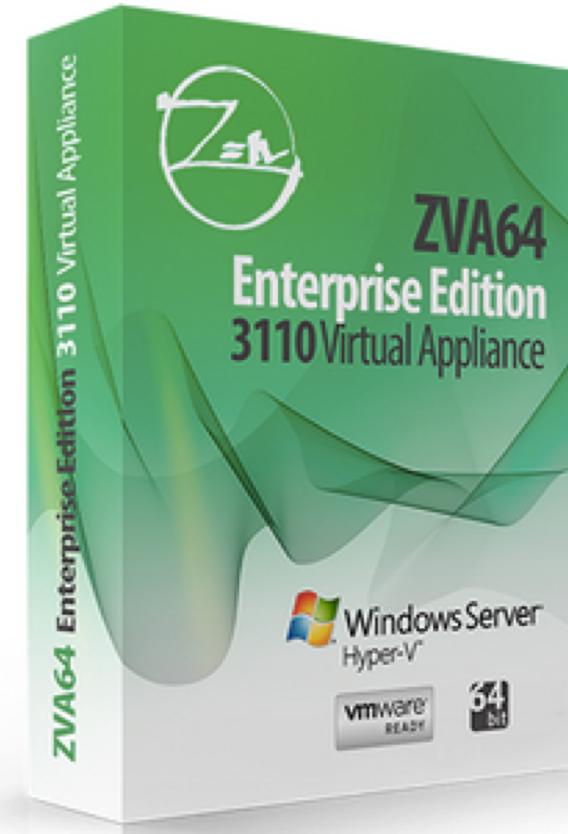
- ✓ Intel Multi-core New CPU's Generation.
- ✓ Powered by Zen Load Balancer Enterprise Edition.
- ✓ 64 bits support.
- ✓ Light weight distribution.
- ✓ Ready for fast deployment without installation.
- ✓ Full warranty.



# Balanceo con dispositivos hardware



Zen Load Balancer



Products ▾ Services ▾ Alliances ▾

ZVA64 EE 3110 Virtual Appliance

ZNA64 EE 3300 HW Appliance

## ZVA64 Enterprise Edition 3110 Virtual Appliance



- ✓ VMware ESXi 5.5/5.1/5.0 host platform support.
- ✓ Hyper-V 2012/2012 R2 host platform support.
- ✓ KVM 3.2.0 host platform support
- ✓ Zen Load Balancer Enterprise Edition.



# Balanceo con dispositivos hardware

The screenshot shows a browser window with the URL <https://www.zevenet.com/zen-load-balancer>. The page header includes the Zevenet logo, navigation links for PRODUCTS, SERVICES, PARTNERS, and ABOUT US, a green 'TRY ZEVENET' button, and a shopping cart icon. A banner at the top reads "Zen Load Balancer is now ZEVENET | Load Balancing made easy". Below this, a sub-section for "Community Edition" features a network graph background, the GitHub logo, and the SF logo. Two buttons are present: "DOWNLOAD SOURCE CODE" and "DOWNLOAD ISO IMAGE". The "DOWNLOAD ISO IMAGE" button is highlighted with a red border.

<https://www.zevenet.com/products/community/#repository>

# Balanceo con dispositivos hardware

## Ejercicio T4.4:

*Instala y configura en una máquina virtual el balanceador ZEVENET (ZenLoadBalancer).*

*Compara con la dificultad de la instalación y configuración usando nginx o haproxy (práctica 3).*

# Balanceo con dispositivos hardware

## Funcionalidad (I):

- 1) Los balanceadores no solo **organizan** la red y **reparten** tráfico entre los servidores de la granja, sino que **monitorizan** la disponibilidad y fallos en aplicaciones.
- 2) Algunos pueden balancear la carga de dispositivos como cortafuegos o concentradores.
- 3) Los balanceadores hardware tienen varios niveles de **redundancia**: configurar varios en paralelo, componentes internos redundantes (cpu, ram, alimentación).

# Balanceo con dispositivos hardware

## Funcionalidad (II):

- 4) Podemos hacer configuraciones por parejas, tipo activo/pasivo o activo/activo. Si uno cae, el otro asume su carga.
- 5) Los balanceadores hardware mantienen y aseguran la **sesión de navegación** de los usuarios. Así se hace compatible el uso de cookies.
- 6) Si un grupo de servidores o un balanceador falla, el suplente puede copiar la información de sesión a otro grupo de servidores para mantener la navegación.

# Balanceo con dispositivos hardware

## Funcionalidad (III):

- 7) Los balanceadores permiten la entrada/salida de servidores de forma automática. Los monitoriza, y si uno falla, deja de enviarle tráfico, hasta que vuelva.
- 8) Hacen **traducción NAT**, y analizan las cabeceras TCP/IP para derivar tráfico al servidor más adecuado, en función del servicio solicitado.
- 9) Se pueden configurar para **evitar algunos ataques** del tipo TCP SYN, DoS, ping of death, IP spoofing, etc.

# Balanceo con dispositivos hardware

## Funcionalidad (IV):

10) Los balanceadores hardware permiten servir el contenido estático directamente: **caching service**.

Así quitamos trabajo a los servidores finales.

Analizan la petición HTTP y lo sirven de la forma más eficiente:

- o bien enviándolo a un grupo de servidores que servirán contenido estático
- o bien determinan el contenido estático más demandado y lo sirven directamente, *cacheándolo* en su memoria RAM.



# Índice

1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
- [ 6. Algoritmos de balanceo de carga ]**
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Algoritmos de balanceo de carga

Los dispositivos y el software de balanceo ofrecen métodos y opciones para repartir carga.

Algunos métodos son estáticos, pero otros tienen en cuenta el estado de las máquinas servidoras.

- Si las máquinas van a ser similares y vamos a servir contenido estático => algoritmos estáticos
- Si las máquinas son heterogéneas y vamos a servir contenido dinámico => algoritmos basados en ponderación.



Probar varios métodos para decidir.

# Algoritmos de balanceo de carga

Los algoritmos más comunes son:

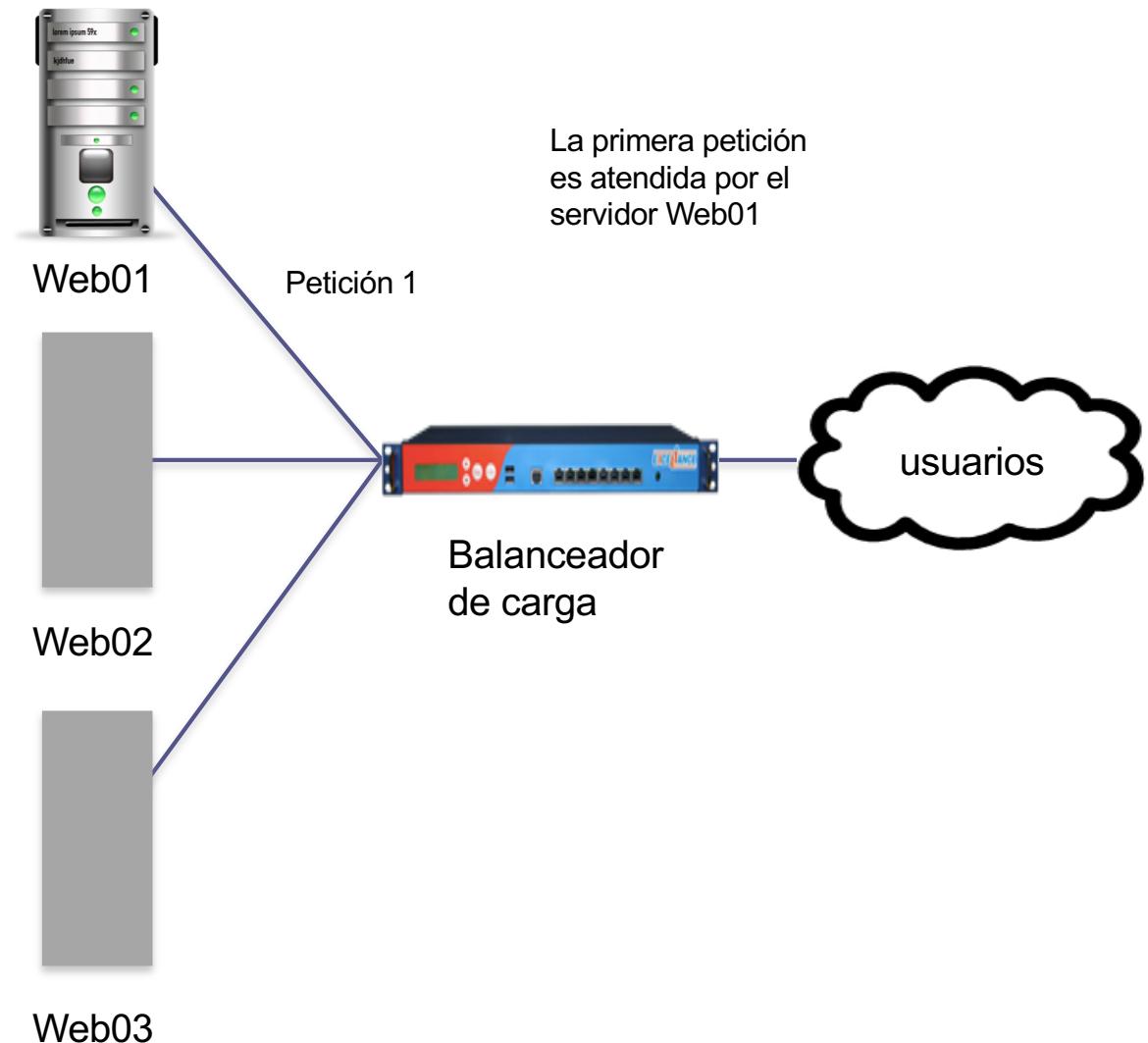
1. balanceo basado en turnos (round-robin)
2. balanceo basado en el menor número de conexiones
3. balanceo basado en ponderación
4. balanceo basado en prioridad
5. balanceo basado en tiempo de respuesta
6. combinación de los algoritmos de tiempo de respuesta y menor número de conexiones

# Algoritmos de balanceo de carga

## 1. Algoritmo de balanceo basado en turnos (round-robin)

Imaginemos que están en una fila.

El primero sirve la petición y pasa al final.

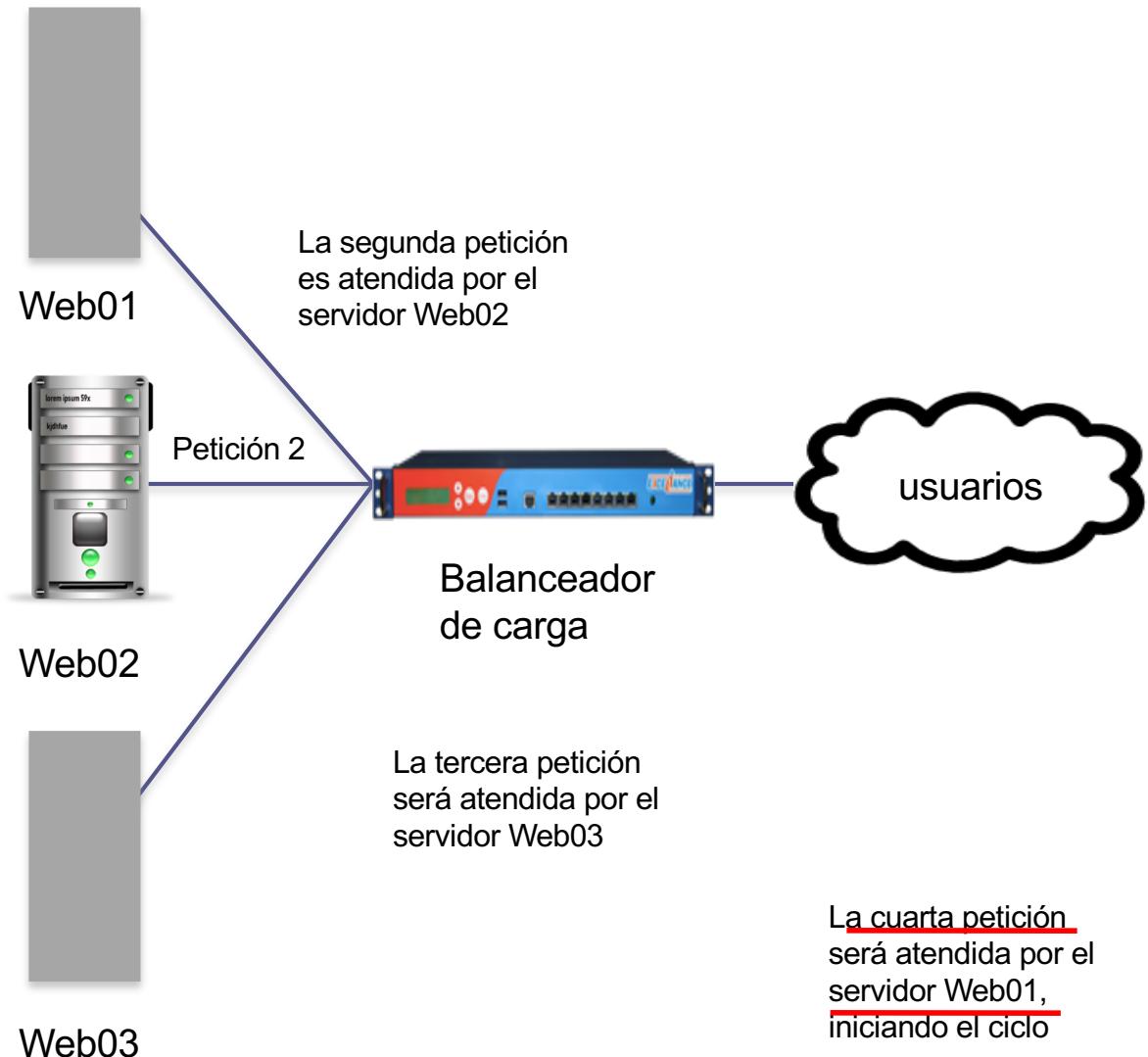


# Algoritmos de balanceo de carga

## 1. Algoritmo de balanceo basado en turnos (round-robin)

El segundo en servir  
es el servidor 2.

etc...



# Algoritmos de balanceo de carga

## 1. Algoritmo de balanceo basado en turnos (round-robin)

Este algoritmo supone a todas las máquinas con la misma potencia.

Es adecuado

si todos tienen potencia similar

o bien

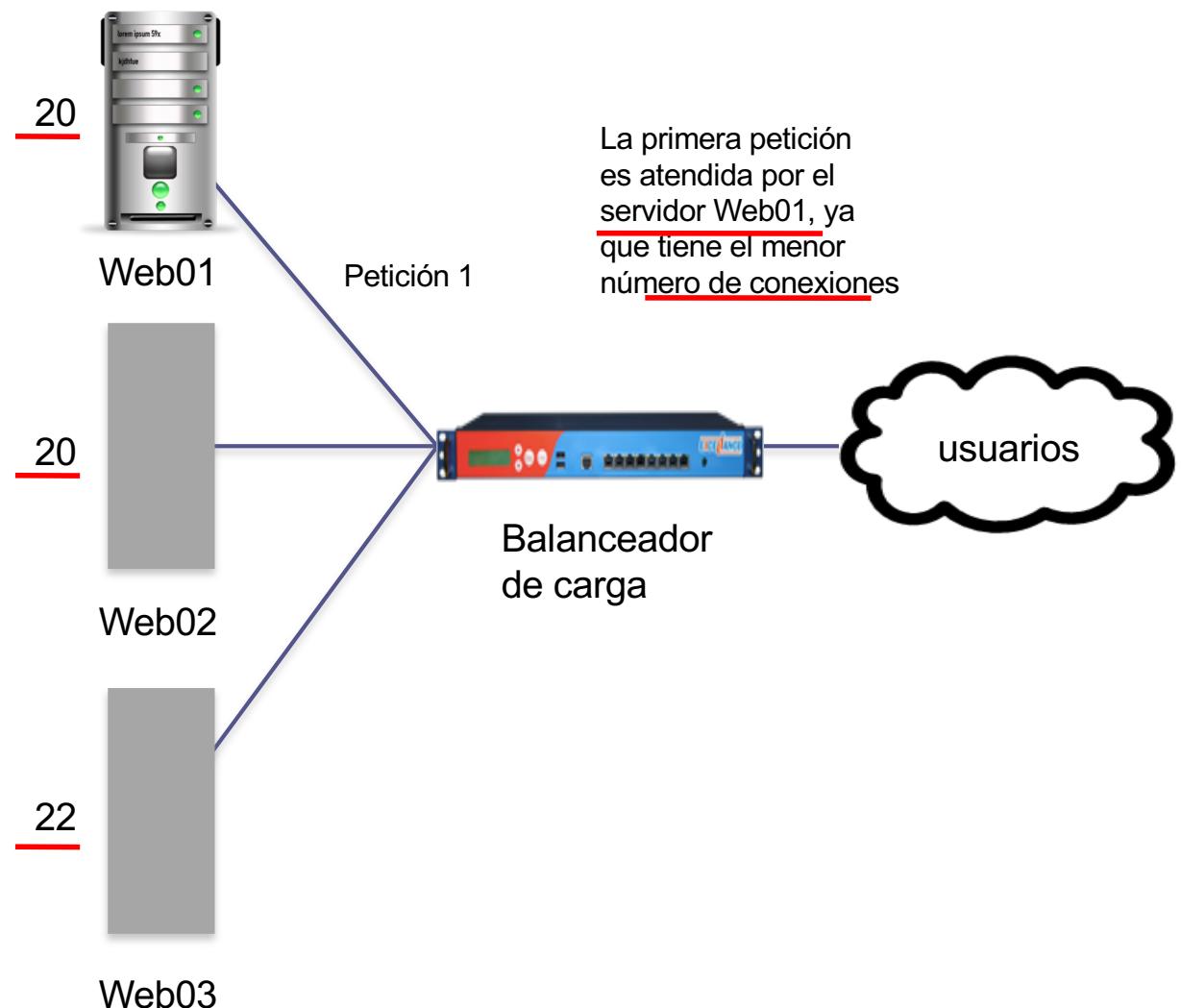
si vamos a servir aplicaciones o servicios sencillos.

# Algoritmos de balanceo de carga

## 2. Algoritmo de balanceo basado en el menor número de conexiones

El balanceador lleva la cuenta del número de conexiones a cada servidor.

Así repartiremos el trabajo según la utilización de cada una.

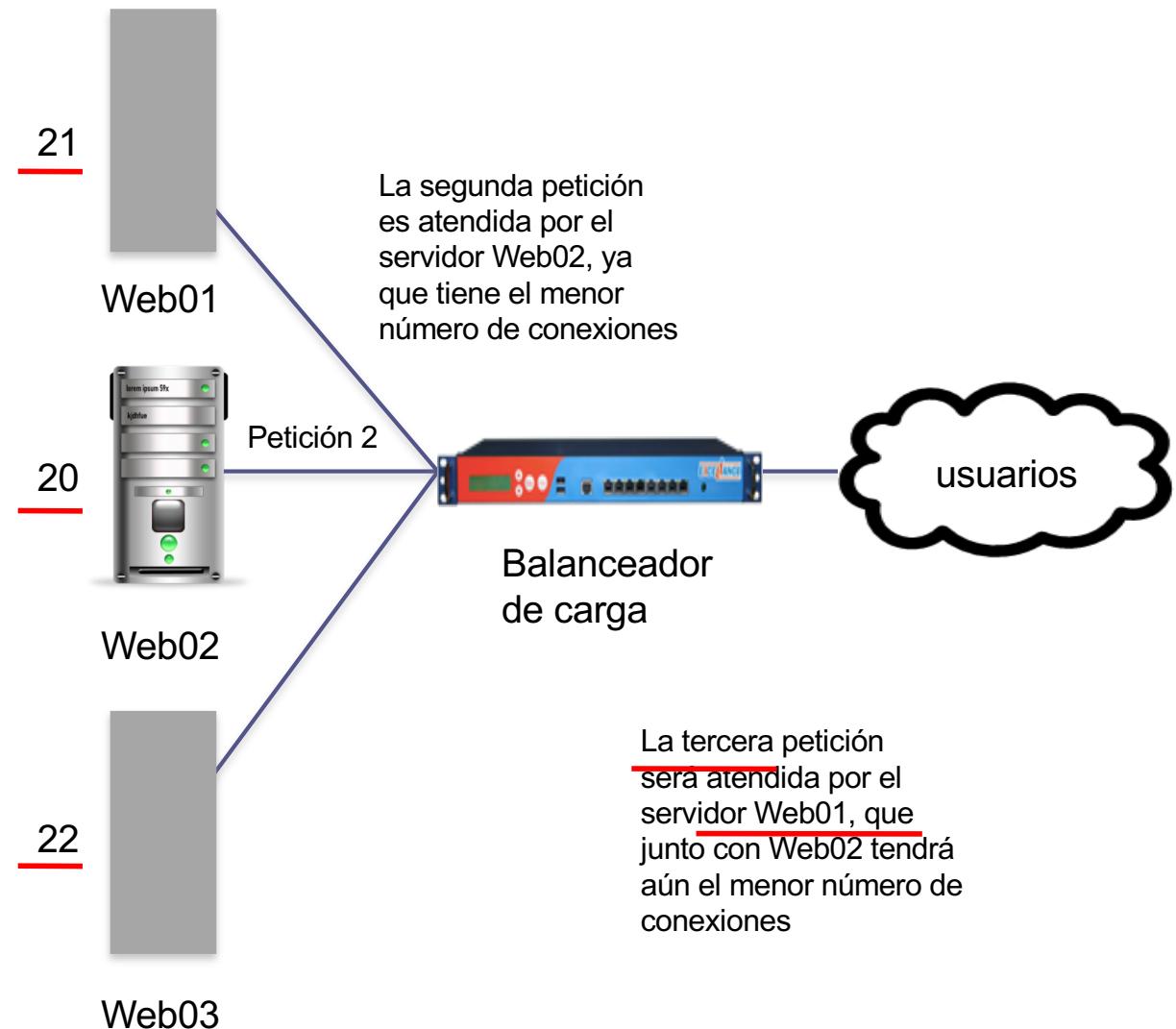


# Algoritmos de balanceo de carga

## 2. Algoritmo de balanceo basado en el menor número de conexiones

Mientras que los servidores 1 y 2 no igualen al 3 en núm. conexiones, seguirán recibiendo.

Sólo entonces se le pasa trabajo al 3.



# Algoritmos de balanceo de carga

## 2. Algoritmo de balanceo basado en el menor número de conexiones

Con este algoritmo se consigue una distribución del trabajo muy adecuada entre máquinas similares.

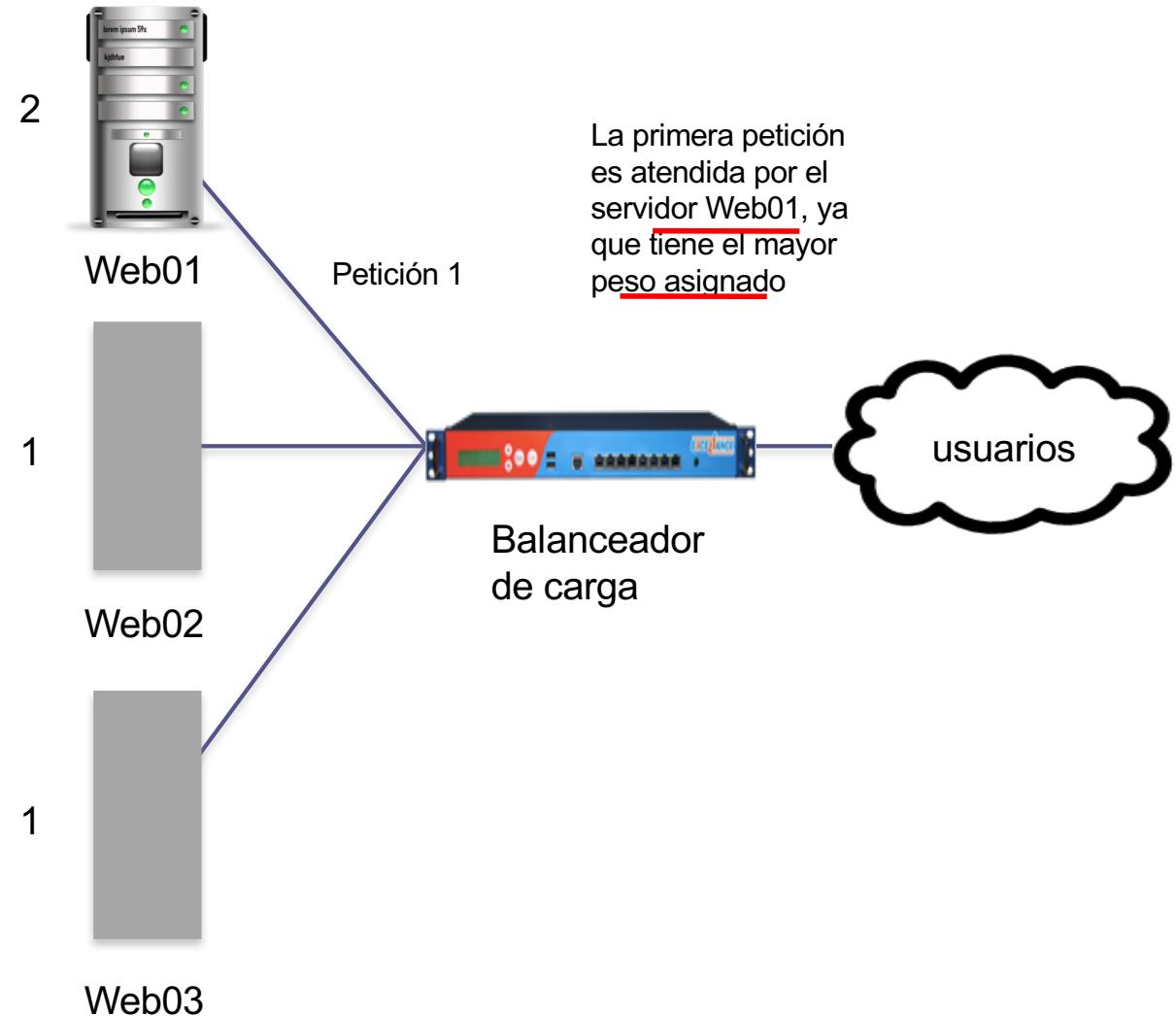
Se evita la sobrecarga de las que puedan tener más trabajo.

# Algoritmos de balanceo de carga

## 3. Algoritmo de balanceo basado en ponderación

Podemos asignar un peso a cada máquina.

Así controlamos el % de conexiones que le pasamos a cada una.

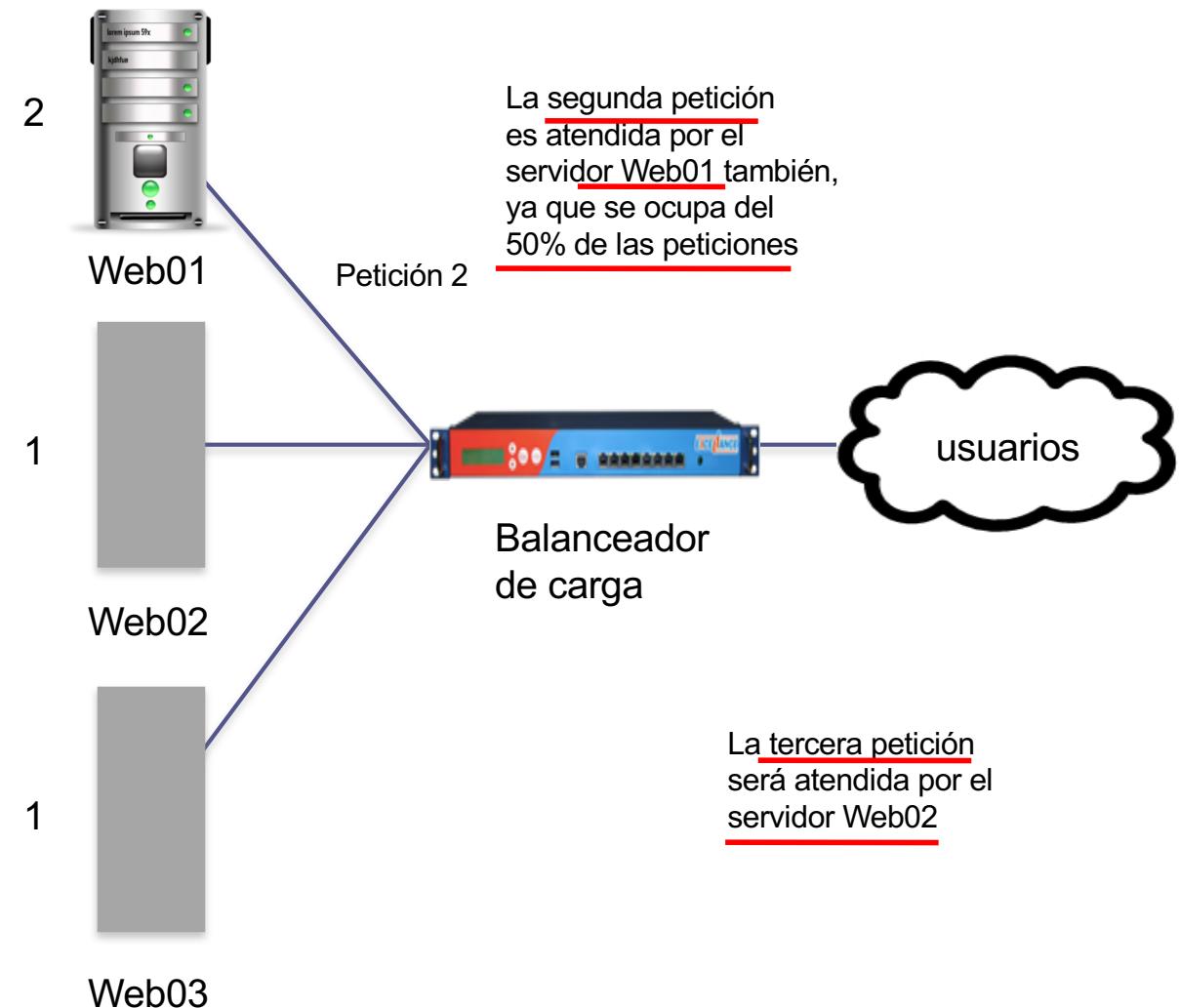


# Algoritmos de balanceo de carga

## 3. Algoritmo de balanceo basado en ponderación

Si en una granja de 4 les damos 25% => todas trabajan igual.

Si damos 50%, 20%, 15%, 15% => la primera se ocupa de la mitad del trabajo y las otras hacen menos.



# Algoritmos de balanceo de carga

## 3. Algoritmo de balanceo basado en ponderación

Probar este algoritmo **cuando los dos anteriores no den buenos resultados.**

Puede resultar adecuado

- cuando el número de conexiones no es un buen indicador
- el trabajo por turnos hace que se pierda tiempo en las máquinas más lentas

# Algoritmos de balanceo de carga

## 4. Algoritmo de balanceo basado en prioridad

Similar a la ponderación pero con grupos.

Cada grupo tiene una prioridad y máximo de conexiones que puede atender.

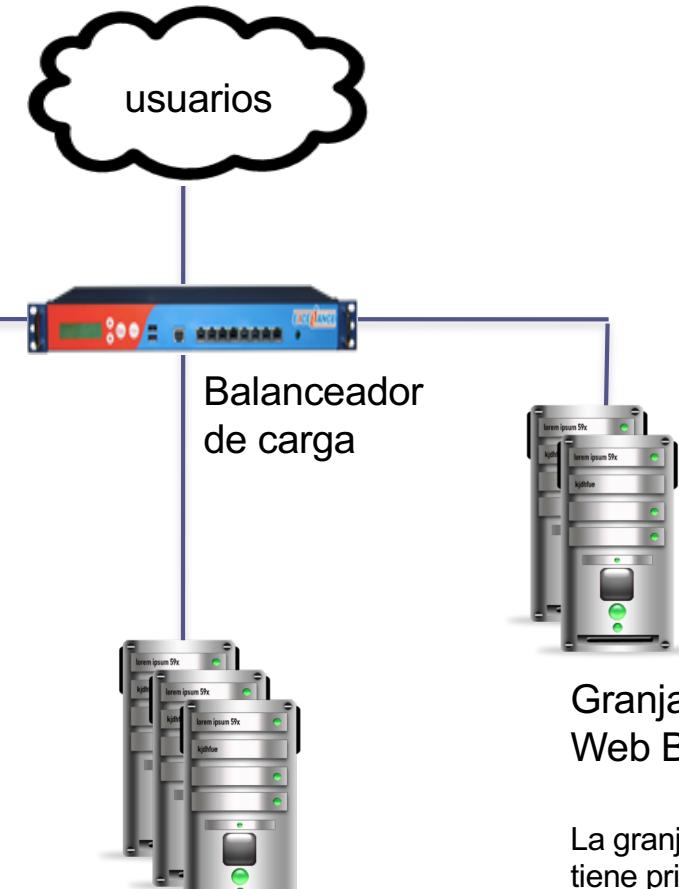
Hay reparto por turnos dentro del mismo grupo.

La granja web A tiene prioridad 1 y puede gestionar 1000 conexiones



Granja Web A

La granja web C tiene prioridad 2 y no actúa a menos que la A pase de las 1000 conexiones o B pase de las 500



Granja Web C

La granja web B tiene prioridad 1 y puede gestionar 500 conexiones

# Algoritmos de balanceo de carga

## 4. Algoritmo de balanceo basado en prioridad

Dentro del mismo grupo se hace reparto por turnos.

Si un grupo recibe más peticiones de las que tiene establecidas entonces el siguiente grupo con más prioridad pasa a recibir.

# Algoritmos de balanceo de carga

## 5. Algoritmo de balanceo basado en tiempo de respuesta

Método dinámico presente en todos los sistemas de balanceo de carga.

Si una petición tarda más en una máquina concreta, es que está más cargada que el resto.

Los平衡adores pueden ir calculando esos tiempos para decidir a qué máquina le pasa la siguiente petición.

Algunos además pueden calcular y tener en cuenta el uso de CPU y memoria de cada máquina.

# Algoritmos de balanceo de carga

## Ejercicio T4.5 (opcional):

*Implementar un pequeño servicio web en los servidores finales que devuelva el % CPU y % RAM que en un instante tiene en uso dicho servidor.*

*Lo debe devolver como una cadena de texto plano que representa ambos porcentajes, p.ej:*

**“CPU 45% RAM 76%”**

# Algoritmos de balanceo de carga

## 6. Combinación de los algoritmos de tiempo de respuesta y menor número de conexiones

Algunos dispositivos平衡adores pueden usar combinaciones de métodos.

Por ejemplo:

tener en cuenta el tiempo de respuesta y el menor número de conexiones para elegir la máquina a la que enviar la siguiente petición.

# Índice



1. Introducción
  2. Funcionamiento básico de un servidor
  3. Conceptos del balanceo de carga
  4. Otras tecnologías
  5. Estructura de la red
  6. Algoritmos de balanceo de carga
7. Balanceo de carga global
  8. Ejemplo 1
  9. Ejemplo 2
  10. Ejemplo 3
  11. Futuro de las tecnologías de balanceo
  12. Resumen y conclusiones

# Balanceo de carga global

Balanceo de carga global (GSLB, *global server load balancing*).

GSLB parte de la idea del balanceo basado en DNS.

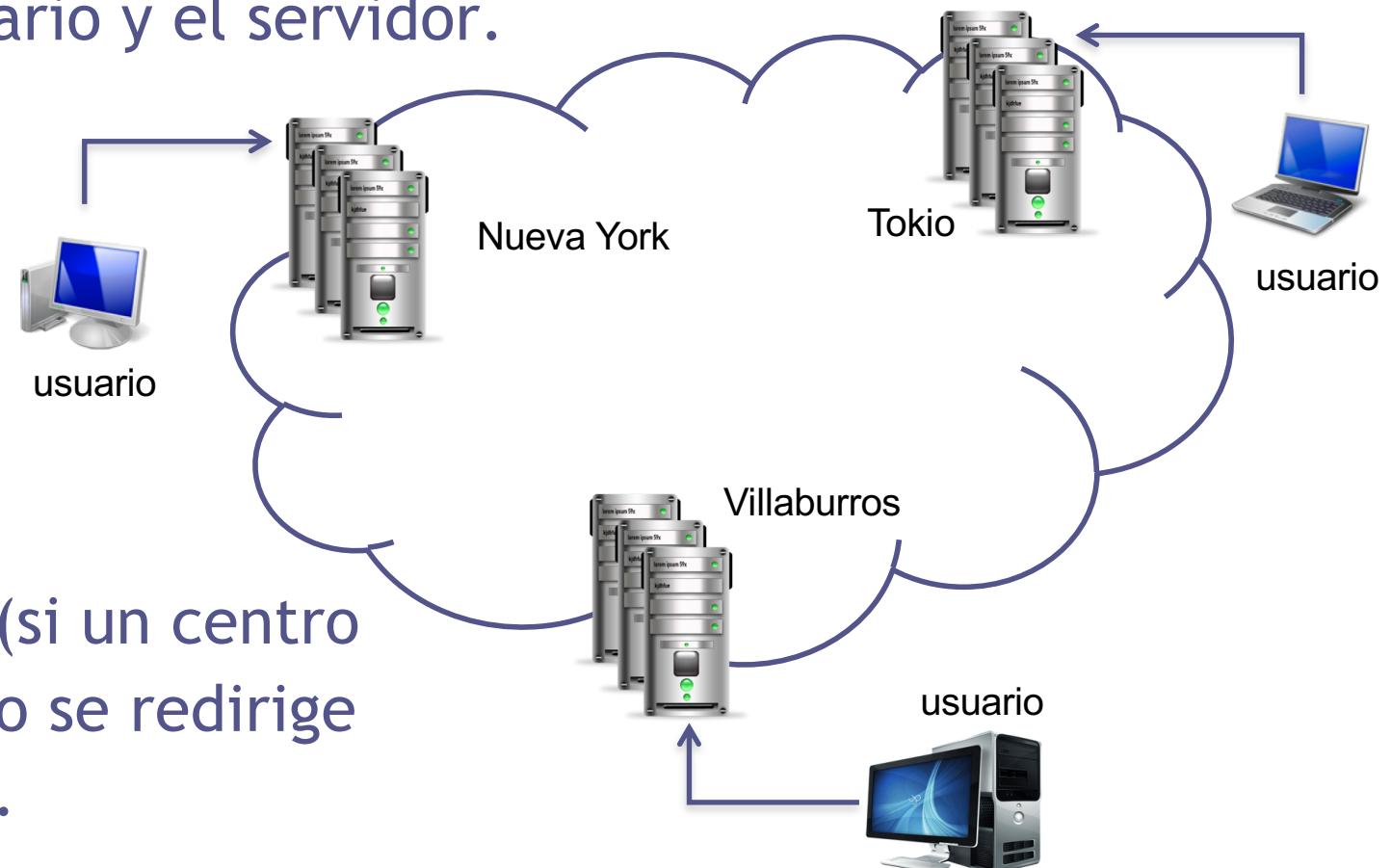
Mejorar el sistema con los conceptos de “alta disponibilidad” y “tiempo de respuesta”.

Queremos evitar una caída total del sistema por un problema en el centro de datos (corte de luz, red, o desastre natural).

# Balanceo de carga global

Distribuir la carga entre varios centros.

Evitar retrasos en las comunicaciones por las distancias entre el usuario y el servidor.



Redundancia (si un centro falla, el tráfico se redirige a otro centro).

# Balanceo de carga global

Posibles implementaciones:

- Uso del DNS
- Redirección HTTP
- GSLB basado en DNS
- GSLB usando protocolos de enrutamiento



# Balanceo de carga global

## Primera aproximación: uso del DNS

DNS puede usarse para hacer balanceo de carga (turnos).

El mismo funcionamiento podemos aprovecharlo para configurar a nivel de DNS las IP de los平衡adores de varios centros (granjas web).

Así repartiremos la carga entre los centros.



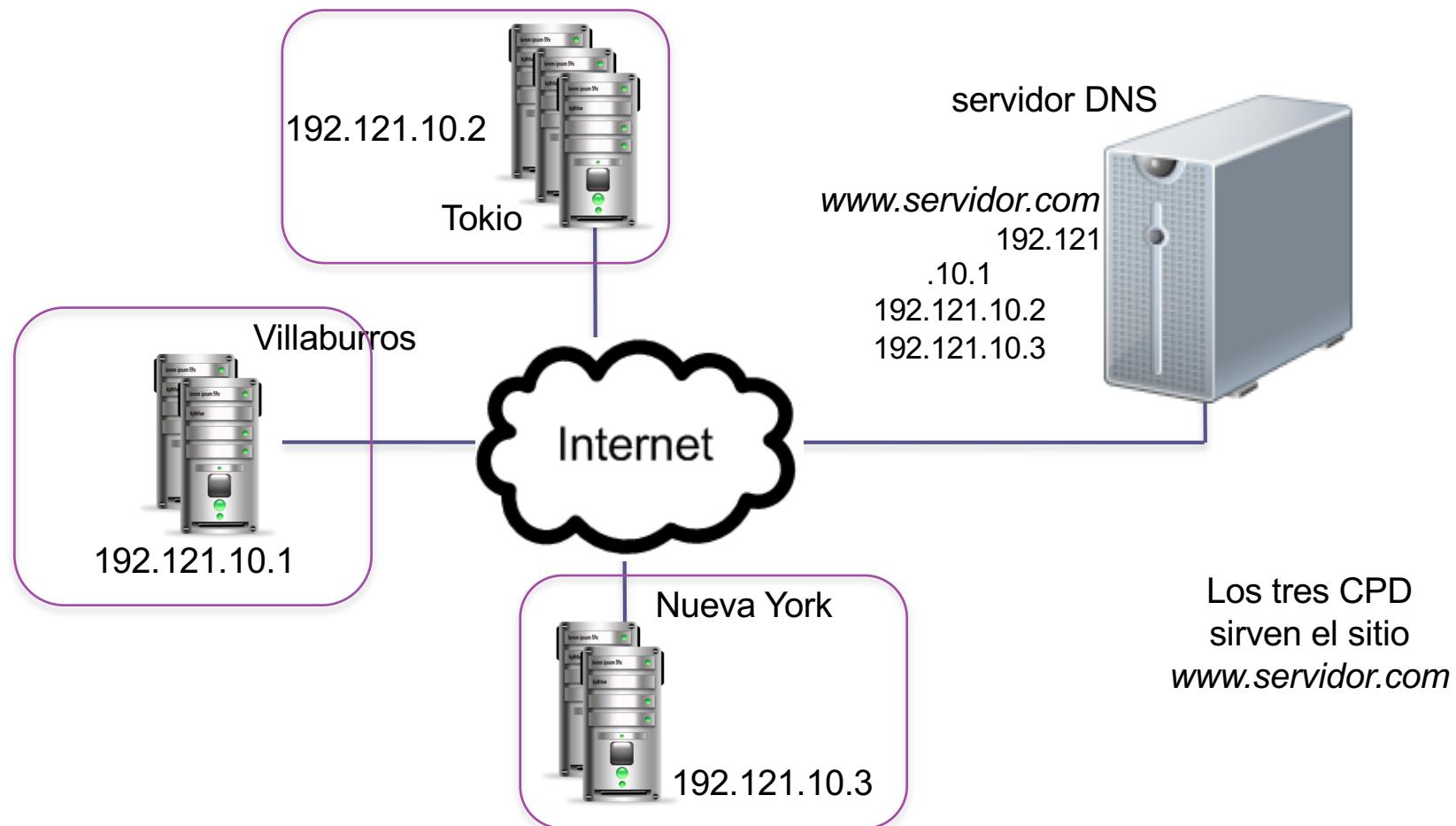
Los DNS no pueden saber si uno de los centros está caído ni reencaminar el tráfico al centro más cercano.



# Balanceo de carga global

## Primera aproximación: uso del DNS

DNS puede usarse para hacer balanceo de carga (turnos).



# Balanceo de carga global

Primera aproximación: uso del DNS

Así repartiremos la carga entre los centros.



Los DNS no pueden saber si uno de los centros está caído ni reencaminar el tráfico al centro más cercano.

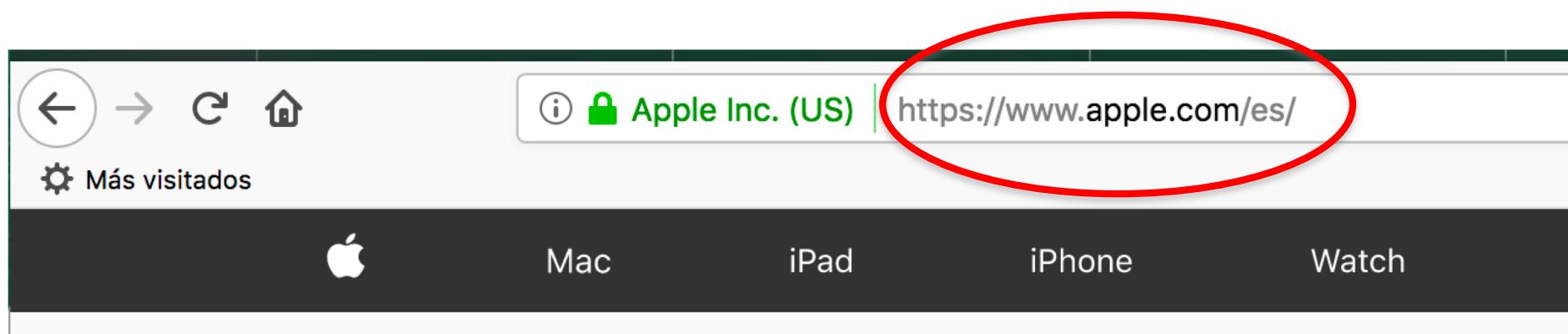


# Balanceo de carga global

## Redirección HTTP

El protocolo HTTP tiene un mecanismo para hacer redirección a otra URL.

Así, si un usuario está en España y accede a la IP de una web, el servidor puede ver la IP del cliente, determinar dónde está exactamente, y redirigirlo a una parte de la web en español o a un servidor en España.



# Balanceo de carga global

## Redirección HTTP

Esta técnica no necesita hacer cambios en los DNS. Sólo programación web muy sencilla.



Necesita dos accesos (latencia). Cuello de botella en la IP



# Balanceo de carga global

## Redirección HTTP (con HTML, JS o PHP)

<http://es.kioskea.net/faq/537-php-redireccionar-a-otra-pagina-web>  
[http://en.wikipedia.org/wiki/Meta\\_refresh](http://en.wikipedia.org/wiki/Meta_refresh)

```
<?php
    $idioma0 = explode(";", $_SERVER['HTTP_ACCEPT_LANGUAGE']);
    $idioma1 = explode(",", $idioma0['0']);
    $idioma2 = explode("-", $idioma1['0']);
    $id = $idioma2['0'];
    switch ($id)
    {
        case 'en':
            header("Location: /en/");
            break;
        case 'es':
            header("Location: /es/");
            break;
    }
?>
```

```
<html>
<head>
<meta http-equiv="Refresh"
      content="5;url=http://servidor.com">
</head>
<body>
</body>
</html>
```

```
<?php header ("Location: http://servidor.com"); ?>
```

```
<body>
<script type="text/javascript">
    window.location="http://servidor.com";
</script>
</body>
```

# Balanceo de carga global

Redirección básica con HTML:

[http://en.wikipedia.org/wiki/Meta\\_refresh](http://en.wikipedia.org/wiki/Meta_refresh)

```
<html>
<head>

<meta http-equiv="Refresh"
      content="5;url=http://servidor.com">

</head>
<body>
</body>
</html>
```

# Balanceo de carga global

## Redirección con PHP (I):

<http://es.kioskea.net/faq/537-php-redireccionar-a-otra-pagina-web>  
<http://stackoverflow.com/questions/6038236/http-accept-language>

```
<?php

$idioma0 = explode(";", $_SERVER['HTTP_ACCEPT_LANGUAGE']);
$idioma1 = explode(", ", $idioma0['0']);
$idioma2 = explode("-", $idioma1['0']);
$id = $idioma2['0'];

switch ($id) {
    case 'en':
        header("Location: /en/");
        break;
    case 'es':
        header("Location: /es/");
        break;
}

?>
```

# Balanceo de carga global

## Redirección con PHP (II):

<http://www.thefutureoftheweb.com/blog/use-accept-language-header>

### Cabecera “Accept-Language”:

en-ca,en;q=0.8,en-us;q=0.6,de-de;q=0.4,de;q=0.2

```
<?php
$langs = array();
if (isset($_SERVER['HTTP_ACCEPT_LANGUAGE'])) {
    preg_match_all('/(([a-z]{1,8}(-[a-z]{1,8})?)\s*(;\s*q\s*=\s*(1|0\.[0-9]+))?)?/i',
        $_SERVER['HTTP_ACCEPT_LANGUAGE'], $lang_parse);

    if (count($lang_parse[1])) {
        $langs = array_combine($lang_parse[1], $lang_parse[4]);

        foreach ($langs as $lang => $val) {
            if ($val === '') $langs[$lang] = 1;
        }
        arsort($langs, SORT_NUMERIC);
    }
}
?>
```

# Balanceo de carga global

## Redirección con JavaScript (I):

<http://blog.joason.com/2013/08/detectar-idioma-del-navegador-y.html>

```
<script>

if (navigator.appName == 'Netscape')
    var lenguage = navigator.language;
else
    var lenguage = navigator.browserLanguage;

if (lenguage.indexOf('en') > -1){
    window.location = 'http://www.google.com';
}else if (lenguage.indexOf('es') > -1){
    window.location = 'http://www.google.es';
}else if (lenguage.indexOf('fr') > -1){
    window.location = 'http://www.google.fr';
}else if (lenguage.indexOf('pt') > -1){
    window.location = 'http://www.google.pt';
}else{
    window.location = 'http://www.ugr.es';
}
</script>
```

# Balanceo de carga global

## Redirección con JavaScript (II):

<http://www.forosdelweb.com/f13/redireccion-segun-idioma-del-navegador-981111/>

```
<script>
var langcodes=new Array("es", "en", "default")
var langredirects=new Array("http://www.google.es", "http://www.google.com", "http://www.ugr.es")

var languageinfo = navigator.language ? navigator.language : navigator.userLanguage
var gotodefault=1

function redirectpage(dest){
    if (window.location.replace)
        window.location.replace(dest);
    else
        window.location=dest;
}

for (i=0;i<langcodes.length-1;i++) {
    if (languageinfo.substr(0,2)==langcodes[i]) {
        redirectpage(langredirects[i]);
        gotodefault=0;
        break;
    }
}
if (gotodefault)
    redirectpage(langredirects[langcodes.length-1]);
</script>
```

# Balanceo de carga global

Ejercicio T4.7 (opcional):

*Probar las diferentes maneras de redirección HTTP.*

*¿Cuál es adecuada y cuál no lo es para hacer balanceo de carga global? ¿Por qué?*

# Balanceo de carga global

## GSLB basado en DNS (I)

Balanceo de carga a nivel del DNS para ayudar a seleccionar la IP del mejor sitio.

DNS consiste en un conjunto jerárquico de servidores DNS. Cada dominio o subdominio tiene una o más zonas de autoridad (authoritative DNS).

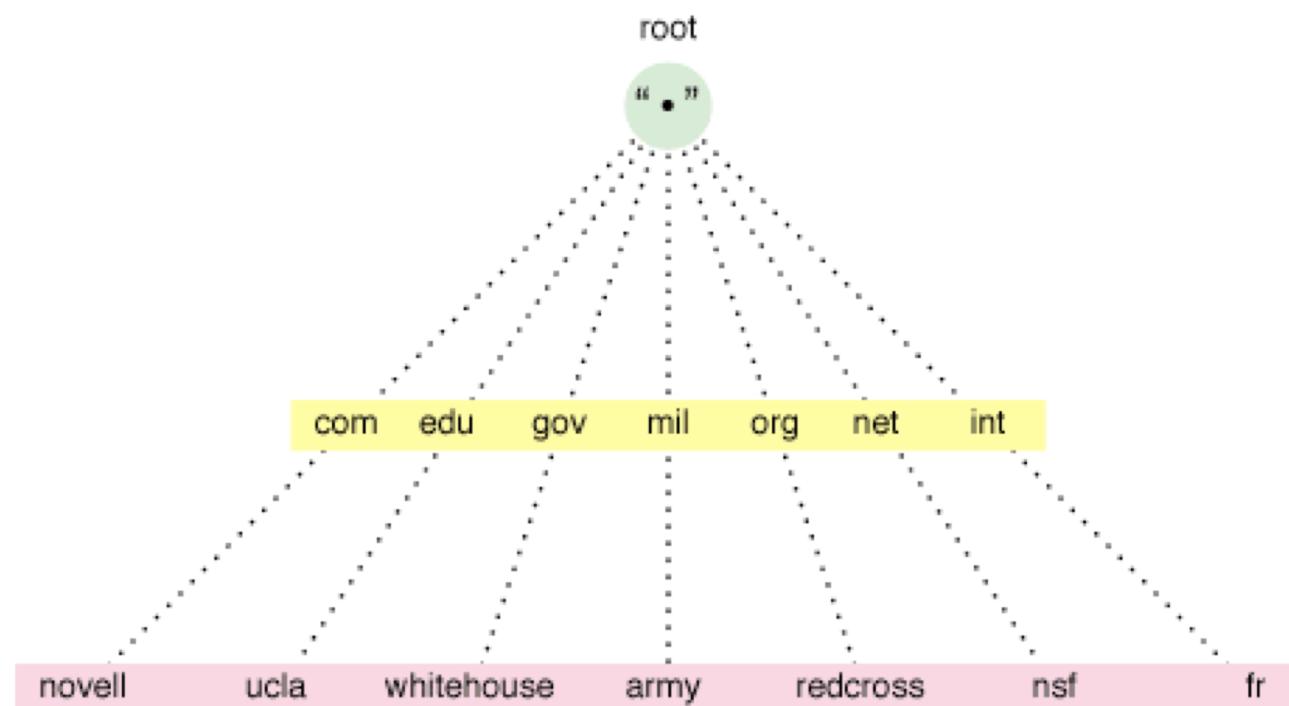
Publican la información acerca del dominio y los nombres de servicios de cualquier dominio incluido.

Al inicio de esa jerarquía se encuentran los servidores raíz.

# Balanceo de carga global

## GSLB basado en DNS (I)

DNS consiste en un conjunto jerárquico de servidores DNS. Cada dominio o subdominio tiene una o más zonas de autoridad (authoritative DNS).



# Balanceo de carga global

## GSLB basado en DNS (II)

Una opción es poner un **balanceador** como servidor DNS a **nivel de zona de autoridad**.

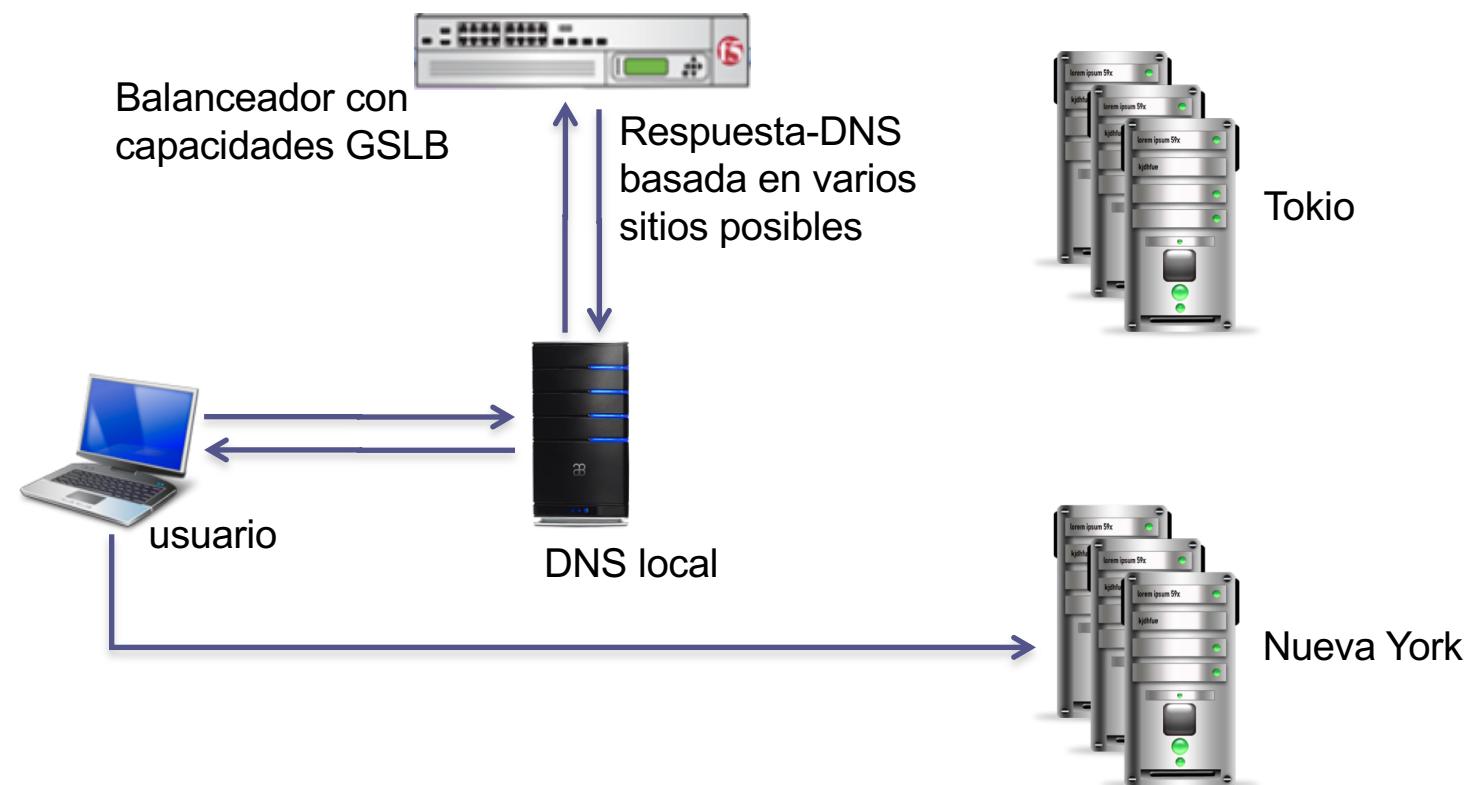
La VIP del balanceador será a la que se enviarán las peticiones al servicio DNS.

Forma en que casi todos los productos GSLB funcionan actualmente.

# Balanceo de carga global

## GSLB basado en DNS (III)

El dispositivo GSLB actúa al nivel *authoritative DNS*.



# Balanceo de carga global

## GSLB basado en DNS (IV)

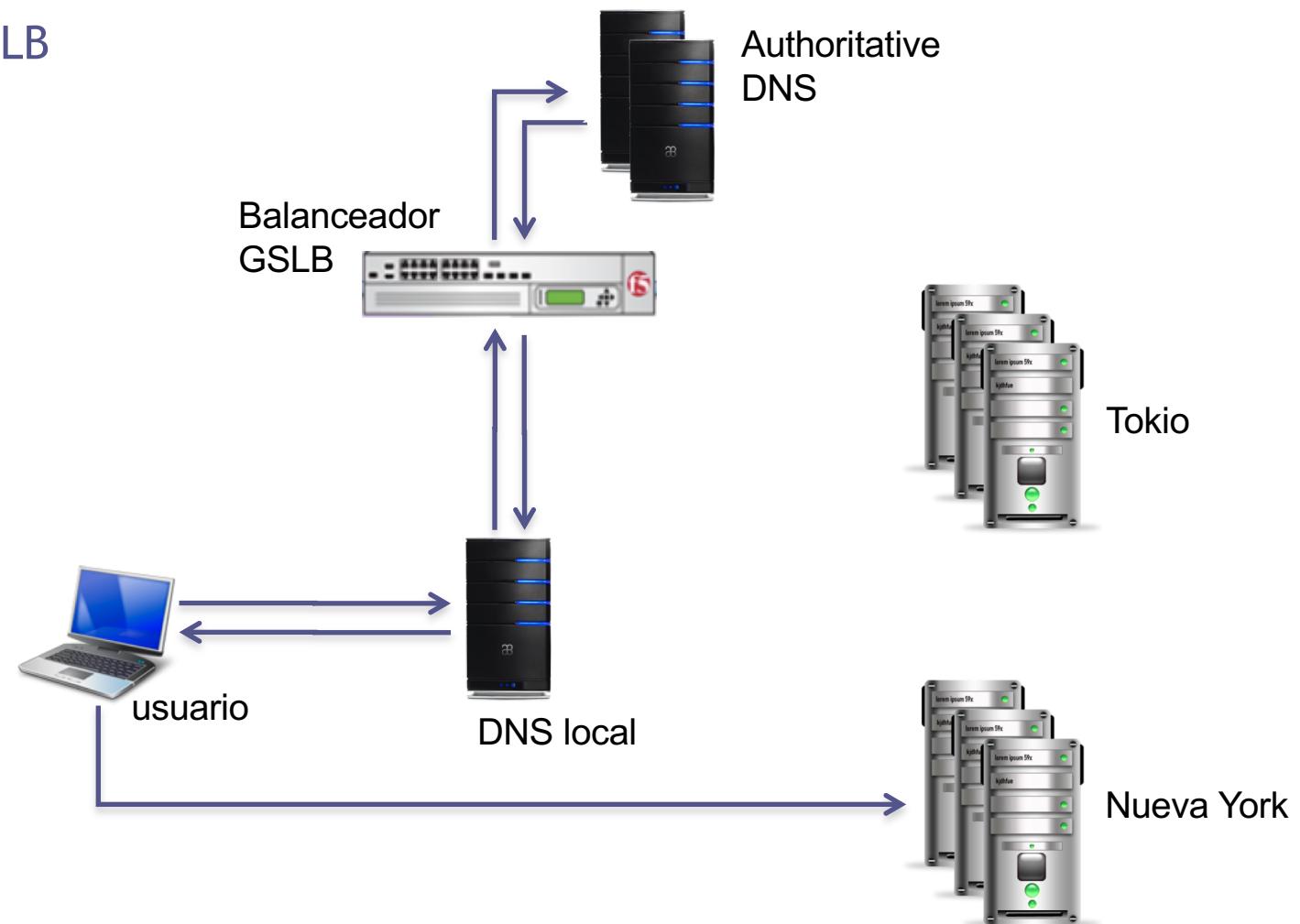
Otra tecnología con la que podemos hacer la integración es configurar un balanceador como proxy.

La idea es poner un **balanceador** delante del "authoritative DNS", de forma que algunas peticiones se pasarán al DNS directamente (y reenvía las respuestas también directamente).

# Balanceo de carga global

## GSLB basado en DNS (V)

El balanceador GSLB  
deja pasar las  
respuestas para  
los sitios que no  
requieren GSLB  
y hace tareas  
de DNS para  
los sitios que  
sí lo requieren



# Balanceo de carga global

## GSLB basado en DNS (VI)

El balanceador puede recoger la respuesta y si hay más de un sitio disponible, la modifica para ofrecer el mejor sitio al cliente.

El balanceador debe hacer ese trabajo extra para modificar la respuesta del DNS para aquellos nombres de dominio que requieren GSLB.

El balanceador no tendrá implementada toda la funcionalidad de un DNS, sino solo la necesaria para mejorar la respuesta del DNS.

# Balanceo de carga global

## GSLB basado en DNS (VII)

Queda determinar el mejor sitio a proponer como respuesta, ya que es algo crítico y que debe decidir el balanceador GSLB a partir sólo de la información intercambiada con el DNS local.

El **GSLB** está monitorizando continuamente el estado de disponibilidad de los diferentes sitios. Así, reenviará tráfico sólo a los sitios activos.

Esta tarea la puede realizar enviando una petición HTTP a cierta URL predefinida, para comprobar el código de retorno devuelto.

# Balanceo de carga global

## GSLB basado en DNS (VIII)

Junto con esa petición HTTP, el balanceador GSLB puede **medir el tiempo de respuesta** que experimenta él.

Cada sitio podrá aceptar un número de conexiones concurrentes diferente. Esta capacidad de aceptar tráfico puede afectar al tiempo de respuesta, aunque no es determinante.

Conviene que el **GSLB conozca la capacidad teórica** de conexiones concurrentes de cada sitio para reenviar tráfico hacia ese sitio. En principio esto mejorará la navegación de los usuarios, pero no asegura nada.

# Balanceo de carga global

## GSLB basado en DNS (IX)

Otra forma de determinar el mejor sitio al que reenviar el tráfico de cierto usuario es **usar información geográfica**.

El GSLB no conoce la IP del usuario, pero sí conoce la del DNS local al usuario (suele ser un DNS cercano al mismo).

La información geográfica **no es el mejor criterio** ya que un sitio más cercano puede estar colapsado por las conexiones.

# Balanceo de carga global

## GSLB basado en DNS (X)

<http://www.ripe.net> ofrece información sobre los bloques de IPs asignadas a la región que incluye Europa y África.

De la misma forma existen webs para obtener información sobre las regiones Asia-Pacífico y América.

# Balanceo de carga global

<https://www.ripe.net/manage-ips-and-asns/dns/reverse-dns>

The screenshot shows a web-based interface for managing IP and ASNs. At the top, it displays the RIPE NCC logo and the text "RIPEstat Reverse DNS (150.214.205.46)". Below this is a yellow button labeled "Reload this widget by ". A message indicates "1 reverse delegation object related to this prefix found in the RIPE Database". A "Show more fields" link is visible above a table. The table lists the following delegation details:

domain 214.150.in-addr.arpa	
descr	Centro Informatico Cientifico de Andalucia
descr	Av. de Reina Mercedes SN
descr	Sevilla, 41012
nserver	dns1.cica.es
nserver	dns2.cica.es
nserver	sun.rediris.es
nserver	chico.rediris.es
nserver	ns.ripe.net

A note at the bottom states "Last updated 6 years ago". The footer of the page says "Showing results for 150.214.205.46 as of 2019-04-05 10:33:00 UTC".

# Balanceo de carga global

**Ejercicio T4.8 (opcional):**

*Buscar información sobre los bloques de IP para los distintos países o continentes.*

*Implementar en JavaScript o PHP la detección de la zona desde donde se conecta un usuario.*

# Balanceo de carga global

## GSLB basado en DNS (productos)

El 3DNS de F5 Networks implementa estas ideas:

[http://support.f5.com/content/kb/en-us/archived\\_products/3-dns/manuals/product/3dns4\\_5ref/\\_jcr\\_content/pdfAttach/doc.pdf](http://support.f5.com/content/kb/en-us/archived_products/3-dns/manuals/product/3dns4_5ref/_jcr_content/pdfAttach/doc.pdf)



Otros productos:

[http://www.tractionet.com/index.php?main\\_page=index&cPath=73](http://www.tractionet.com/index.php?main_page=index&cPath=73)

# Balanceo de carga global

## GSLB basado en DNS (desventajas)

GSLB es complicado de configurar y poner en funcionamiento, ya que la tecnología DNS no se desarrolló para hacer balanceo de carga.

Además, el funcionamiento de las **cachés de los navegadores y sistemas operativos** hace que ciertas IP se guarden durante horas e incluso días, por lo que si las condiciones de cierto sitio cambian, el usuario seguirá enviando tráfico a dicho sitio.

**Solución:** limpiar la caché del navegador.

<http://www.tenereillo.com/GSLBPageOfShame.htm>

# Balanceo de carga global

## GSLB usando protocolos de enrutamiento (I)

Independiente de la tecnología DNS.

Utilizar a nivel de proveedor de servicio.

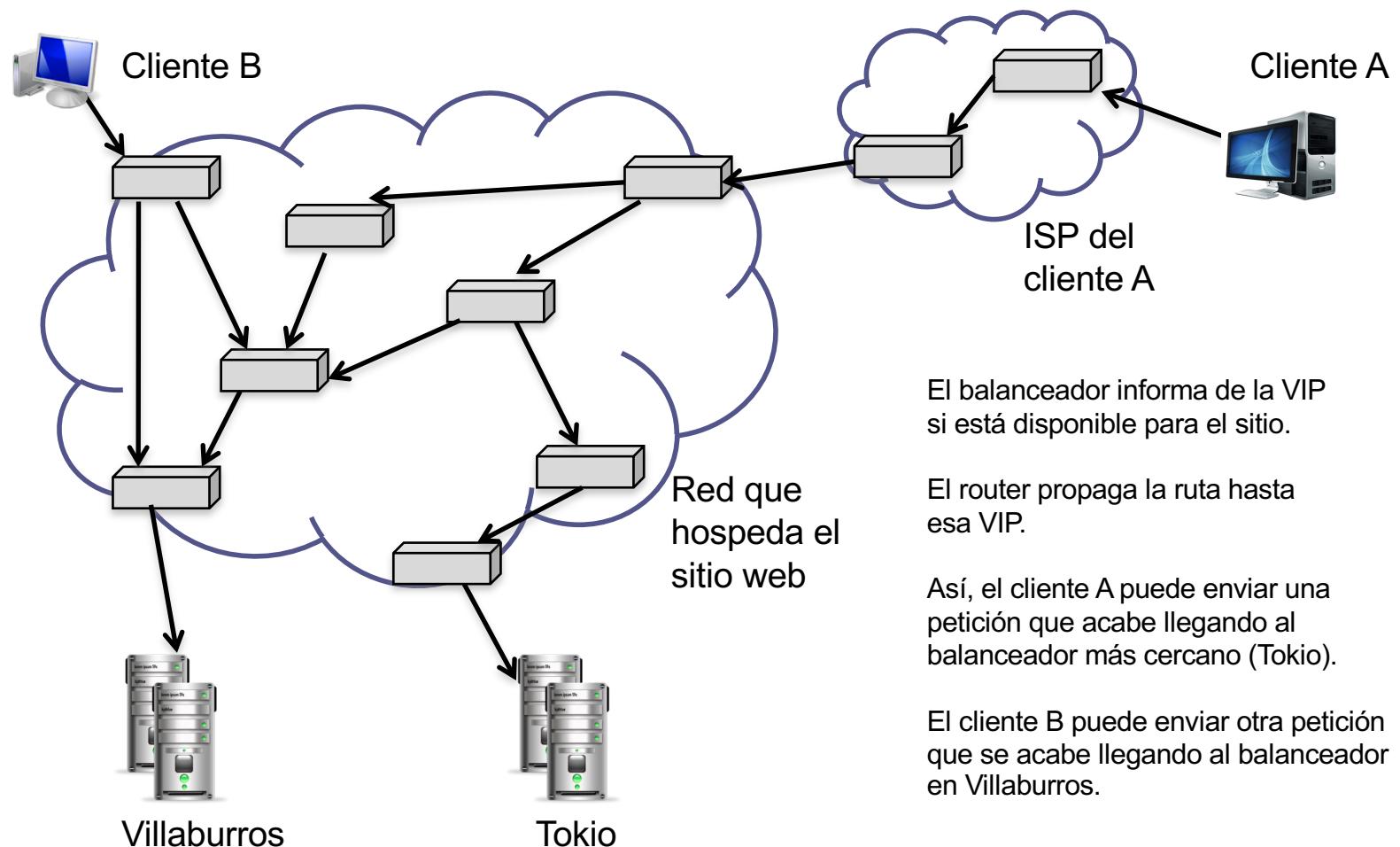
La idea es tener la dirección del sitio web hospedada en diferentes DNS.

Cuando un DNS responda al usuario con una IP para cierto dominio, esta aproximación dirigirá el tráfico al mejor sitio posible.

Balanceadores de carga en sitios diferentes (A y B), todos configurados con la misma VIP.

# Balanceo de carga global

## GSLB usando protocolos de enrutamiento (II)



# Balanceo de carga global

## GSLB usando protocolos de enrutamiento (III)

Para los routers sólo hay **caminos** para los paquetes.

A nivel de router, que existan dos sitios con la misma VIP significa que **hay dos caminos** para llegar al mismo sitio.

Cuando un usuario A teclea una URL para ir al sitio web, el DNS le facilita la IP (en este caso, la VIP).

El navegador cliente comienza la conexión enviando la petición a la VIP.

# Balanceo de carga global

## GSLB usando protocolos de enrutamiento (IV)

Esta petición se propagará por la red y llegará al router A, que mirará en su tabla de enrutamiento para decidir por qué ruta lo envía.

Encontrará dos "caminos" hasta la VIP, y en función de algún algoritmo de camino mínimo, decide hacia dónde enviar este tráfico.

Como ejemplo de protocolo de enrutamiento cabe destacar OSPF (Open Shortest Path First).

# Balanceo de carga global

## GSLB usando protocolos de enrutamiento (V)



### Possible problema:

Cuando un usuario navega, abre varias conexiones TCP. Los routers recibirán los paquetes generados y los reenviarán hacia cualquiera de los balanceadores de los sitios, en función del estado de la red.

Sin embargo para que la comunicación funcione, todos los paquetes deben ir al mismo balanceador.

Si en un momento, algún **router decide cambiar la ruta** hacia otro de los balanceadores, se perderá la integridad de la comunicación (las aplicaciones web necesitan mantener la persistencia en las comunicaciones).

# Balanceo de carga global

## Resumen

GSLB presenta ventajas: alta disponibilidad.



Es muy complejo de implementar y comprender.



Requiere de conocimientos de los DNS.

Si la aplicación web usa BD, éstas deben estar sincronizadas entre los diversos sitios. Muy complejo.



Para que GSLB funcione es necesario un entorno de red muy controlado, con alta coordinación entre los operadores.

# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

## Ejemplo: recorrido de un paquete a través de un balanceador

Veamos cómo funcionaría el tráfico de red a través de un sistema web con un balanceador de carga.

Supongamos 4 máquinas servidoras:

- m1 y m2 sirven HTTP
- m3 sirve FTP
- m4 sirve SMTP
- b hace de balanceador

Así se desacoplan las aplicaciones o servicios de las máquinas, dando alta flexibilidad.

## Ejemplo: recorrido de un paquete a través de un balanceador

Un cliente establece una conexión TCP, envía la petición HTTP, recibe la respuesta y cierra la conexión TCP.

b recibe en el inicio de la conexión un paquete que lleva como origen la IP del cliente y como destino la VIP del sistema web. También indica el puerto del servicio.

b comprueba la disponibilidad de m1 y m2.

b selecciona a una de las dos máquinas y cambia en el paquete la VIP por la IP privada de la máquina a la que enviará ese tráfico de ese cliente.

## Ejemplo: recorrido de un paquete a través de un balanceador

La máquina servidora envía datos como respuesta.

Los paquetes pasan por el balanceador, que tiene que cambiar la IP privada de la máquina servidora por la VIP antes de enviar el paquete hacia el cliente (que espera recibir los paquetes con origen la VIP).

# Índice

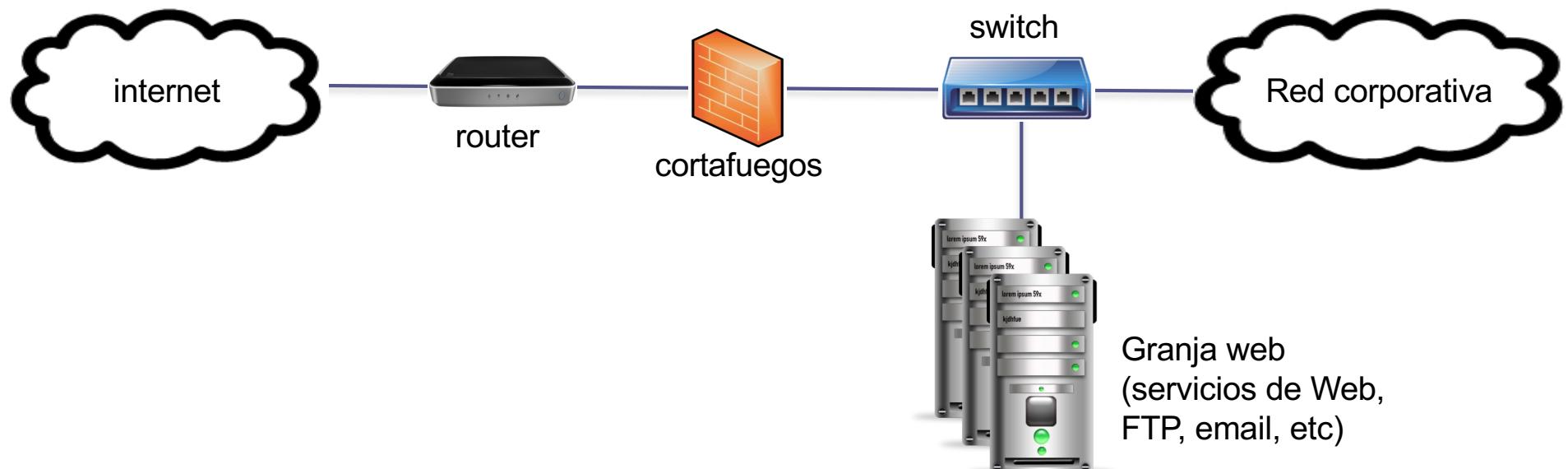


1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

# Ejemplo: diseño de la red de una empresa

Esquema, a alto nivel, de la red de una empresa con unas necesidades especiales en cuanto a seguridad, escalabilidad y alta disponibilidad del sitio web.

Partiremos del siguiente diseño de red:

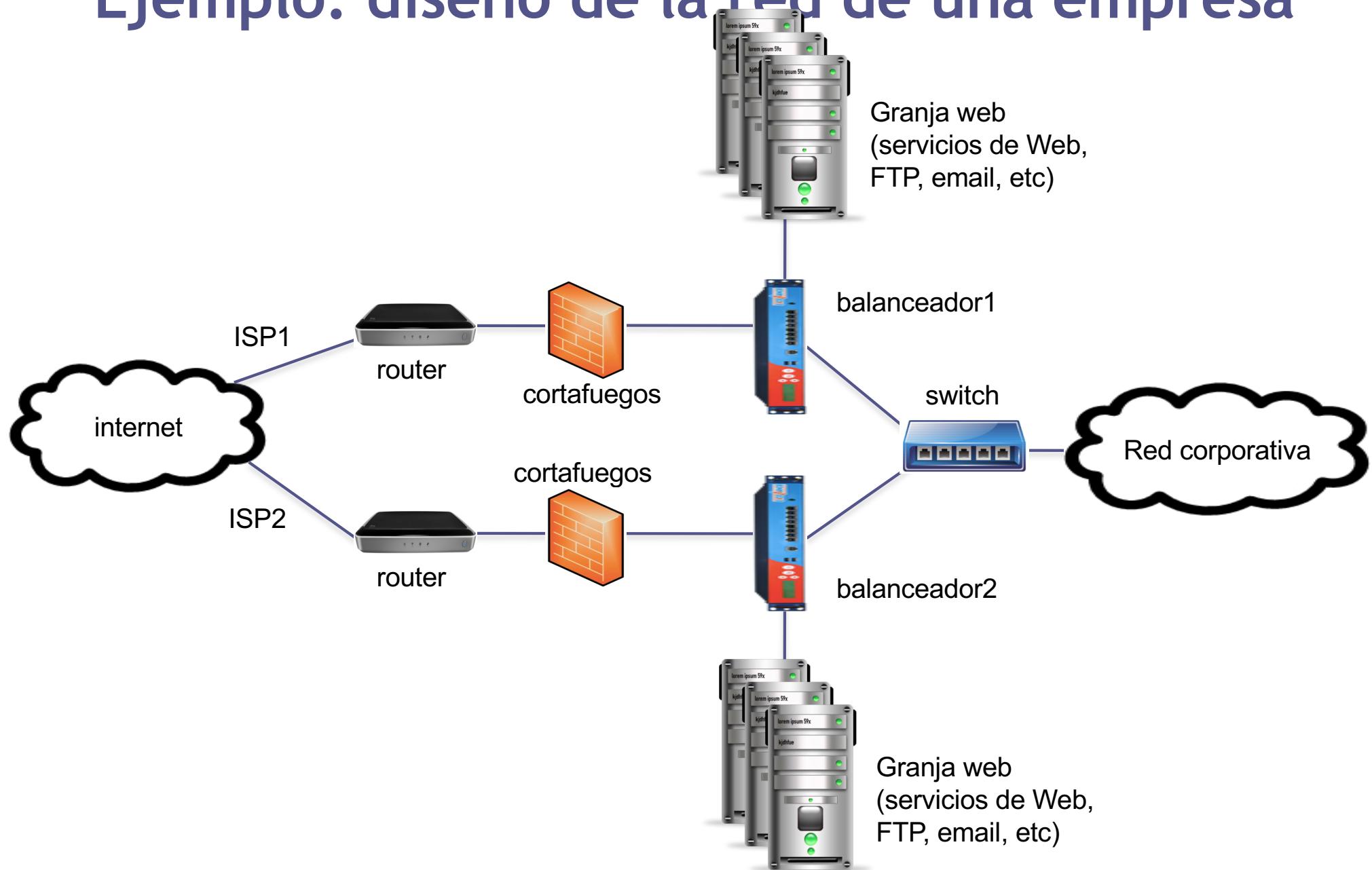


## Ejemplo: diseño de la red de una empresa

Modificar ese diseño inicial para mejorar la disponibilidad, escalabilidad y usabilidad:

- añadiremos dos routers así como dos conexiones replicadas a sendos proveedores de Internet, replicando cortafuegos
- podemos definir una zona segura de la red y mover los servidores web y FTP a esa zona DMZ
- para realizar balanceo de carga, mejorando así la escalabilidad, disponibilidad y manejabilidad
- replicar la instalación y configuración de red en varias localizaciones (países) para llevar a cabo balanceo global

# Ejemplo: diseño de la red de una empresa



# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
- [10. Ejemplo 3]**
11. Futuro de las tecnologías de balanceo
12. Resumen y conclusiones

## Ejemplo: redes de distribución de contenidos

Comentar la función de los balanceadores de carga en las redes de distribución de contenidos.

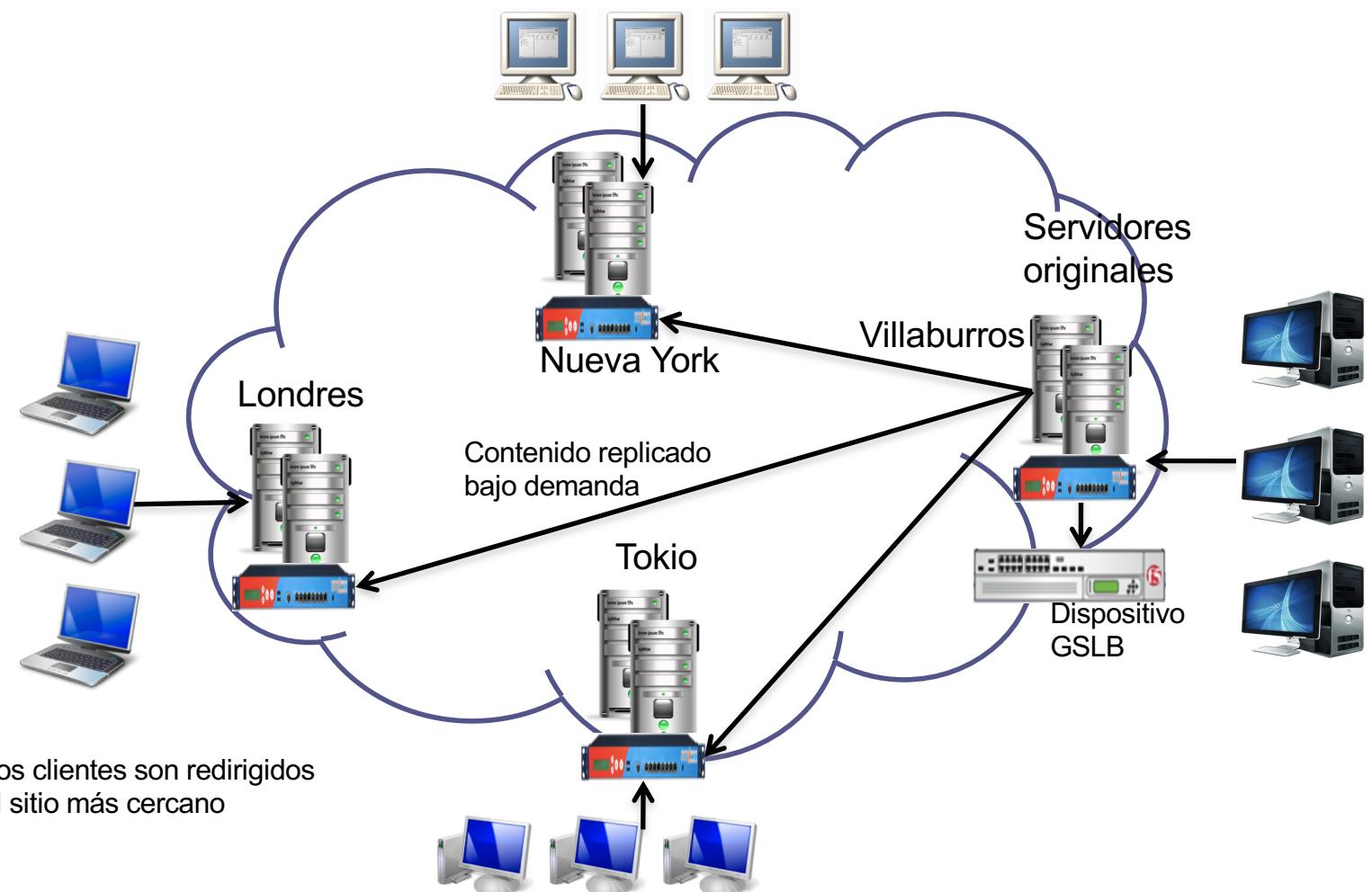
Los proveedores de contenido deben ofrecer el mejor tiempo de respuesta posible a los usuarios finales.

Localizar los centros de datos cerca de los usuarios.

Las empresas despliegan réplicas (caches) en diferentes centros de datos para servir el contenido estático.

# Ejemplo: redes de distribución de contenidos

Una posibilidad es usar GSLB:



## Ejemplo: redes de distribución de contenidos

Hay empresas que consiguen reducir los costes de este tipo de instalaciones usando los centros de datos existentes (especie de alquiler).

Todo el coste de la instalación y mantenimiento de la red en varios centros de datos por todo el mundo se lo ahorra la empresa.

Mediante GSLB se mejora la disponibilidad y el usuario experimenta menores tiempos de respuesta.

## Comentar: uso de CDN

Table 2-1. *CDN service providers used by top sites*

Web site	CDN
<a href="http://www.amazon.com">http://www.amazon.com</a>	Akamai
<a href="http://www.aol.com">http://www.aol.com</a>	Akamai
<a href="http://www.cnn.com">http://www.cnn.com</a>	
<a href="http://www.ebay.com">http://www.ebay.com</a>	Akamai, MirrorImage
<a href="http://www.google.com">http://www.google.com</a>	
<a href="http://www.msn.com">http://www.msn.com</a>	SAVVIS
<a href="http://www.myspace.com">http://www.myspace.com</a>	Akamai, Limelight
<a href="http://www.wikipedia.org">http://www.wikipedia.org</a>	
<a href="http://www.yahoo.com">http://www.yahoo.com</a>	Akamai
<a href="http://www.youtube.com">http://www.youtube.com</a>	

<http://stevesouders.com/hpws/ex-cdn.php>

<http://stevesouders.com/hpws/ex-nocdn.php>

# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
- [11. Futuro de las tecnologías de balanceo]**
12. Resumen y conclusiones

# Futuro de las tecnologías de balanceo

Reducir precios, incrementar prestaciones y funcionalidades.

Evolución de los balanceadores: cubrir las necesidades de servidores de ficheros, base de datos y otras aplicaciones.

Mejora de las prestaciones y reducción de la latencia.

Tareas relativas a la seguridad.

Servir directamente contenido estático (espacio en RAM)

El balanceador podrá integrar hardware y software para acelerar el procesamiento de datos SSL.

# Índice



1. Introducción
2. Funcionamiento básico de un servidor
3. Conceptos del balanceo de carga
4. Otras tecnologías
5. Estructura de la red
6. Algoritmos de balanceo de carga
7. Balanceo de carga global
8. Ejemplo 1
9. Ejemplo 2
10. Ejemplo 3
11. Futuro de las tecnologías de balanceo
- [12. Resumen y conclusiones]**

# Resumen y conclusiones

El balanceo de carga ha permitido ofrecer más servicios a un número creciente de usuarios.

El balanceador reparte el tráfico web entre varios servidores, y realiza comprobaciones para asegurar la disponibilidad.

Diversos algoritmos de balanceo de carga para repartir el tráfico entre los servidores.

Aporta diversos beneficios: escalabilidad, disponibilidad, mantenimiento, seguridad, calidad de servicio.