

El objetivo de esta sesión de problemas es observar el funcionamiento y evaluar las ventajas de mecanismos para organizar el servicio web con.

- un único servidor,
- varios servidores (la consulta de dns devuelve varias direcciones)
- varias réplicas (mirrors)
- una red de distribución de contenidos

Como ejemplos de sitios modelo para cada caso hemos seleccionado uno para cada caso aunque pueden utilizarse muchos otros.

un único servidor	www.upc.es, www.apache.org
varios servidores	www.ibm.com, www.microsoft.com (+cdn)
varias réplicas	httpd.apache.org http://www.apache.org/mirrors/
red de distribución de contenidos	www.mtv.com

Las pruebas se pueden hacer tanto en Linux como en Windows, tal como os resulte más cómodo.

### **Prueba 1. Número de servidores: Probar nslookup**

Consulta de DNS: Prueba el comando nslookup y averigua cuántas direcciones IP parecen corresponder a cada página web. (Si tu PC no dispone del comando, puedes usar este: <http://www.infobear.com/nslookup.shtml>)

Consulta de DNS extendida:

- Prueba un recorrido detallado por el DNS usando el nslookup extendido en <http://www.squish.net/dnscheck/>
- Busca los registros de DNS tipo A (dirección IP) asociada a cada página web.

**Resultado:** ¿Cuántas direcciones salen en cada caso? ¿En los sitios web con CDN qué ocurre con el DNS?

### **Prueba 2. Ubicación y caminos: Probar visualroute**

El programa "Visualroute" permite averiguar y ver en un mapa la ubicación de máquinas en Internet. Puede encontrarse en <http://www.visualroute.com>, pero hemos dejado una copia en el web de aad para facilitar la descarga (Linux, Windows). Si se escribe el nombre de un servidor web, visualroute lo resuelve en la(s) direcciones IP y averigua el camino y estima la ubicación de los servidores (seleccionar "modo avanzado". Aunque la instalación de este programa te la podrías ahorrar ...

También es posible usarlo desde un navegador, sin necesidad de instalación previa. Entonces se pueden hacer las pruebas desde diversos lugares visitando: <http://www.visualware.com/visualroute/livedemo.html>

Para los sitios web propuestos, se puede ver la ubicación de cada servidor (por ejemplo con mtv.com puede verse cómo cambia la ubicación según el lugar desde el que se hagan las medidas).

**Resultado 1:** ¿En qué sitios web cambia la ubicación del servidor según el origen de la consulta (ubicación del cliente)?

La fundación Apache tiene un sistema para proponer la réplica o "mirror" más cercana. Está en la página web: <http://www.apache.org/dyn/closer.cgi>  
Mira con Visualroute la ubicación de los primeros sitios que proponen, el estado del camino por la red y su tiempo de respuesta.

**Resultado 2:** ¿ Es buena la elección que propone Apache ?

### **Prueba 3. Rendimiento en el tiempo: Probar JMeter**

JMeter es un programa de la fundación Apache que permite observar el comportamiento de un servidor web y medir su rendimiento. Puede generar una demanda de intensidad variable para probar cómo se porta un servidor sometido a diferentes niveles de carga. Entre otras cosas genera un análisis gráfico de rendimiento y guarda la historia de transferencias durante una prueba.

En nuestro caso, se usará JMeter para medir el tiempo de respuesta con una carga o demanda reducida y ver una gráfica de cómo varía el tiempo de respuesta y qué responde el servidor.

JMeter está en <http://jakarta.apache.org/jmeter/index.html>

Para usar JMeter hay que generar un "Plan de pruebas" siguiendo las instrucciones que indica en:

<http://jakarta.apache.org/jmeter/usermanual/build-web-test-plan.html>

**Resultado:** ¿Qué diferencia observas entre sitios próximos, lejanos, con CDN, con mirror de apache (sobre todo los que están en horas de mucha carga)?

### **Prueba 4. Rendimiento de cada servidor: Probar Apache Benchmark (ab)**

Apache Benchmark o (ab) es una utilidad de la distribución de Apache (./support) que incluye el servidor web Apache. Utiliza las librerías de apache para probar el rendimiento de un servidor web. La forma de uso puede encontrarse en: <http://httpd.apache.org/docs-2.0/programs/ab.html>

El programa puede generar resultados en un fichero con las opciones:

-g filename    Output collected data to gnuplot format file.  
-e filename    Output CSV file with percentages served

La opción -g genera un fichero que contiene: dirección IP contactada, fecha inicio conexión, los tiempos de respuesta medidos (conexión, transferencia, espera)

La opción -e genera una estadística de los tiempos de respuesta del servidor como la que aparece en la figura 1.

```
ab -n 6 -c 2 -v 1 -k -g log.txt -e distrib.txt http://www.upc.es/
```

Hemos modificado el programa para que al generar logs.txt incluya la dirección IP con que "ab" se ha conectado. Además, si se ejecuta ab varias veces con el mismo fichero en la opción -g los resultados se añaden. Esto permite visualizar fácilmente los datos en un programa como gnuplot o ms-excel.

**Resultado:** Hacer un script para medir el rendimiento de las réplicas de apache.  
(Opcional) Hacer un programa que tome el mejor en cada momento: hacer cdf con gnuplot o excel, ....  
(Opcional y difícil) Descubrir servidores de Akamai y hacer lo mismo tal como se describe en [2].

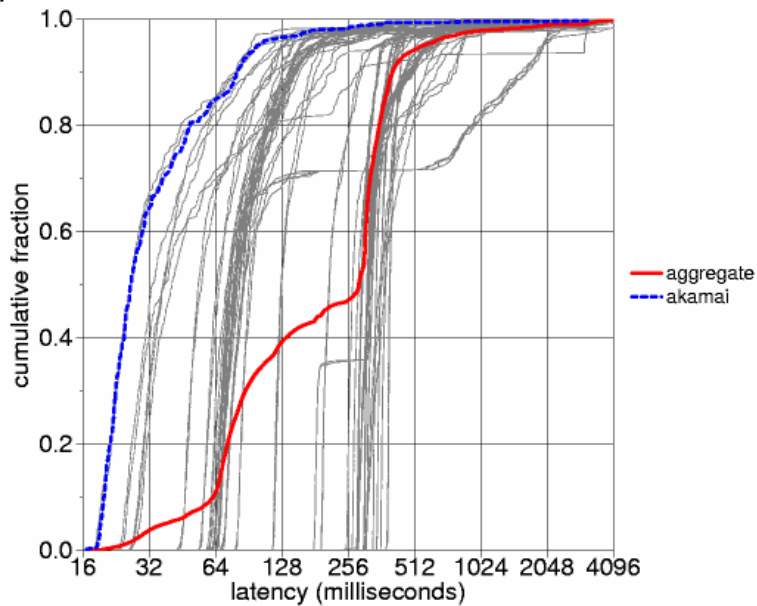


Figura 1.- Distribución acumulada de tiempos de respuesta (latencia) de varios servidores Akamai (un objeto de 4kb). [2]

## Referencias

- [1] John Dilley, Bruce Maggs, Jay Parikh, Harald Prokop, Ramesh Sitaraman, Bill Weihl, "Globally Distributed Content Delivery", akamai.pdf
- [2] Kirk L. Johnson, John F. Carr, Mark S. Day, M. Frans Kaashoek, "The Measured Performance of Content Distribution Networks", <http://www.terena.nl/conf/wcw/Proceedings/S4/S4-1.pdf>
- [3] Mirrors de apache: <http://www.apache.org/dyn/closer.cgi>