



Pedro A. Castillo Valdivieso
Universidad de Granada
pacv@ugr.es

Guión



• **¿Qué es node.js?**

- Instalación
- Ejecutar el primer programa
- Crear un servidor web con node.js

¿Qué es node.js?

" Node.js es un entorno de programación en la capa del servidor basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos [...]"

"Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web. "

"Al contrario que la mayoría del código JavaScript, no se ejecuta en un navegador, sino en el servidor."

¿Qué es node.js?

node.js es un programa de servidor, pero no es como Apache o Tomcat.

node.js es un entorno JavaScript de lado de servidor que utiliza un modelo asíncrono y dirigido por eventos.

node.js está basado en el motor V8 JavaScript de Google para el navegador Chrome.

Hay Otros proyectos como Tornado (Python), Twisted (Python), EventMachine (Ruby), Scale Stack (C++), Apache MINA (Java) o Jetty (Java).

¿Qué problema resuelve node.js?

En Java o PHP, cada conexión genera un nuevo hilo que puede consumir unos 2 MB de memoria.

Si tenemos 8 GB de RAM, esto da un número máximo teórico de conexiones concurrentes de cerca de 4.000

No está mal, pero puede ser insuficiente.

¿Qué problema resuelve node.js?

node.js resuelve el problema cambiando la forma en que se realiza una conexión con el servidor. En lugar de generar un nuevo hilo de OS para cada conexión (y de asignarle la memoria acompañante), cada conexión dispara una ejecución de evento dentro del proceso del motor de Node.

Los desarrolladores afirman que un servidor que lo ejecute puede soportar decenas de miles de conexiones concurrentes.

¿Para qué es útil node.js?

node.js no es simplemente un reemplazo para Apache: las aplicaciones web existentes no son compatibles.

node.js está bien diseñado para cuando se espere una gran cantidad de tráfico y donde la lógica del lado del servidor y el procesamiento requeridos no sean grandes.

node.js es también perfecta para ofrecer una API REST (un servicio web que tiene unos parámetros de entrada y no requiere de una manipulación de datos compleja).

Guión

- ¿Qué es node.js?



- **Instalación**

- Ejecutar el primer programa

- Crear un servidor web con node.js

Instalación

<http://nodejs.org/download/>



Windows Installer (.msi)

Windows Binary (.exe)

Mac OS X Installer (.pkg)

Mac OS X Binaries (.tar.gz)

Linux Binaries (.tar.gz)

SunOS Binaries (.tar.gz)

Source Code

32-bit	64-bit
32-bit	64-bit
Universal	
32-bit	64-bit
32-bit	64-bit
32-bit	64-bit
node-v0.10.35.tar.gz	

Instalación

En las siguientes webs se explica con detalle cómo instalarlo en Windows, Linux y OS X:


<http://howtonode.org/how-to-install-nodejs>

<https://github.com/joyent/node/wiki/installing-node.js-via-package-manager>

<http://coolestguidesontheplanet.com/installing-node-js-osx-10-9-mavericks/>

<https://www.digitalocean.com/community/tutorials/how-to-install-node-js-on-an-ubuntu-14-04-server>

Guión

- ¿Qué es node.js?
- Instalación
-  - **Ejecutar el primer programa**
- Crear un servidor web con node.js


Ejecutar programas sencillos

```
mac:EJEMPLOS pedro$ node  
> console.log('hola amigos');  
hola amigos  
undefined  
> █
```

```
mac:EJEMPLOS pedro$ node  
> console.log('hola amigos');  
hola amigos  
undefined  
>  
undefined  
> var a=7;  
undefined  
> var b=8;  
undefined  
> var c=a*b;  
undefined  
> console.log('resultado c='+c);  
resultado c=56  
undefined  
> █
```

```
mac:EJEMPLOS pedro$ node  
> var a=[11,22,33,44];  
undefined  
> var i=0;  
undefined  
> for (i=0;i<a.length;i++){ console.log("Elemento i="+i+" almacena el valor="+a[i]+" \n" ); }  
Elemento i=0 almacena el valor=11  
  
Elemento i=1 almacena el valor=22  
  
Elemento i=2 almacena el valor=33  
  
Elemento i=3 almacena el valor=44  
  
undefined  
>  
(^C again to quit)  
>  
mac:EJEMPLOS pedro$ █
```

Guión

- ¿Qué es node.js?
- Instalación
- Ejecutar el primer programa
-  • **Crear un servidor web con node.js**

El primer servidor

Como primer ejemplo, vamos a hacer un pequeño servidor web que cuando sea llamado responda con la cadena "Hola mundo (node.js)"

Guardamos el código en un archivo .js

```
var http = require('http');

http.createServer(function (request, response) {
  response.writeHead(200, {'Content-Type': 'text/plain'});
  response.end('Hola mundo (node.js) \n');
}).listen(8000);

console.log('Server running at http://127.0.0.1:8000/');
```

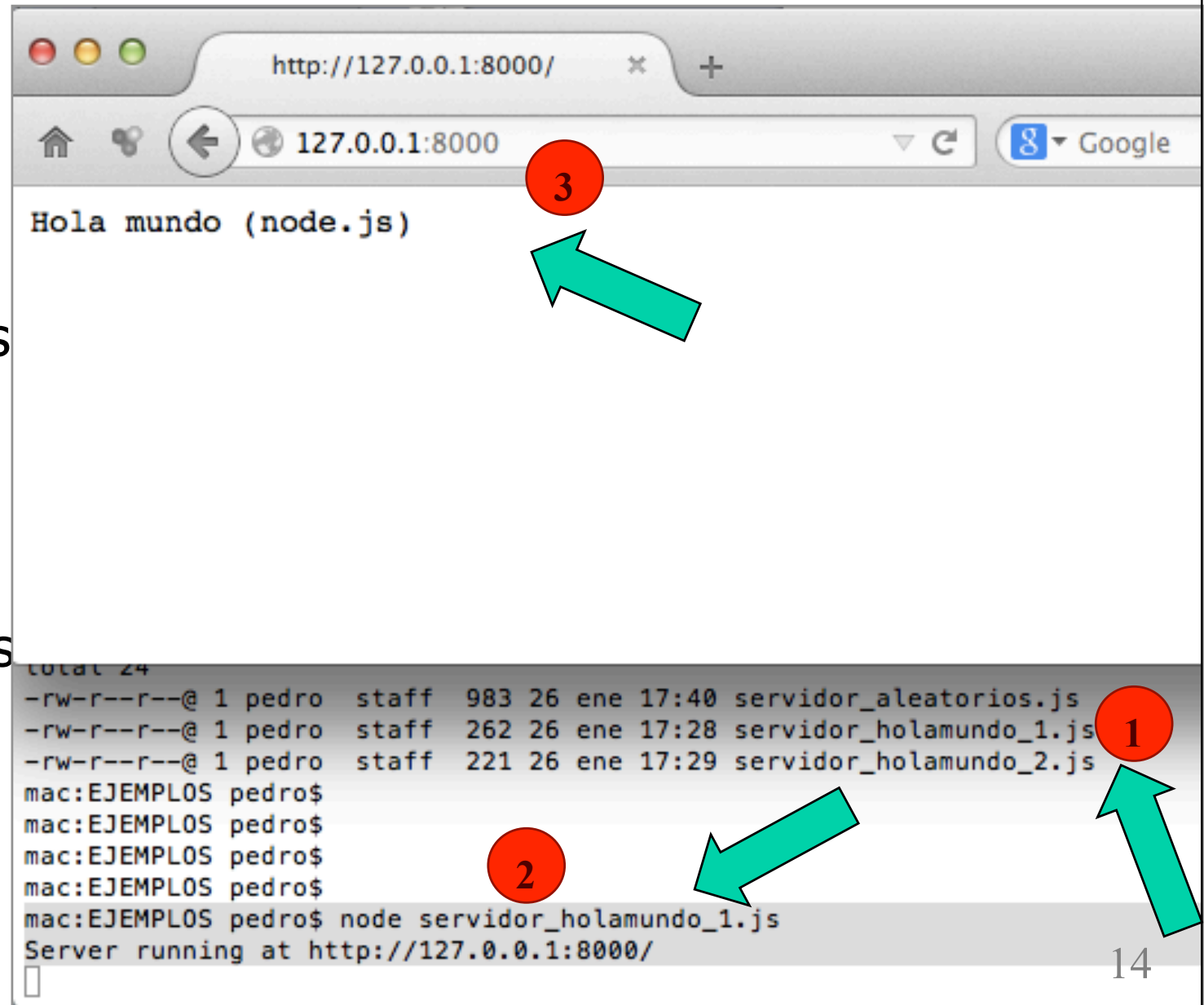
Ejecutar el primer servidor

1 Creamos el archivo de texto.

2 Lo ejecutamos en un terminal.

Desde un navegador accedemos

3 a la URL del servidor creado:



Ejercicios

- (1) Hacer un programa con node.js para mostrar los primeros 10 números de la serie de fibonacci.
Ejecutarlo en el terminal.
- (2) Hacer un servidor con node.js que reciba un número y que devuelva el factorial.



Referencias

- <http://nodejs.org/>
- <http://nodejs.org/docs/v0.10.35/api/>
- <http://es.wikipedia.org/wiki/Node.js>
- <http://nodeschool.io/>
- <http://www.nodebeginner.org/index-es.html>
- <http://nodeguide.com/beginner.html>
- <http://geekytheory.com/instalacion-de-nodejs/>
- <http://geekytheory.com/node-js-primeros-pasos-y-hola-mundo/>
- <https://www.ibm.com/developerworks/ssa/opensource/library/os-nodejs/>
- <http://soloelectronicos.com/2014/05/27/que-es-y-para-que-sirve-node-js/>
- <http://www.rmunoiz.net/introduccion-a-node-js.html>
- <http://howtonode.org/how-to-install-nodejs>
- <http://coolestguidesontheplanet.com/installing-node-js-osx-10-9-mavericks/>



¡eso es todo!

Pedro A. Castillo Valdivieso
Universidad de Granada
pacv@ugr.es