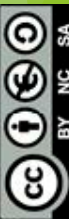


NODE.JS:

Conceptos básicos

UF1844: Lenguajes del lado servidor



Ferran Parellada

SOC

Servei d'Ocupació
de Catalunya

¿Qué es **NODEJS**?



¿Qué es **NODEJS**?

NODEJS es...

- Un entorno de programación y ejecución basado en Javascript,
- De código abierto,
- Multiplataforma,
- Orientado a eventos asíncronos.
- Un sistema para programar mediante Javascript en el lado del servidor, así como aplicaciones de escritorio.

¿Qué es **NODEJS**?

NODEJS permite...

- El acceso al sistema de archivos,
- Obtener información del sistema operativo (arquitectura, sesiones, usuarios, etc.),
- Obtener información de los procesos activos del equipo,
- El acceso a bases de datos,
- Y más...

NODEJS y los navegadores

Hay algunas diferencias respecto como se utiliza **JS** en **NODE** y en los navegadores:

- Como no en **NODE** hay interfície gráfica, tampoco hay nada parecido a DOM.
- La mayoría de módulos de **NODE** no se pueden ejecutar directamente en la consola del navegador.
- Aunque la base es siempre **Javascript**, la sintaxis para algunas operaciones puede ser ligeramente diferente (como por ejemplo en las exportaciones e importaciones de ficheros).

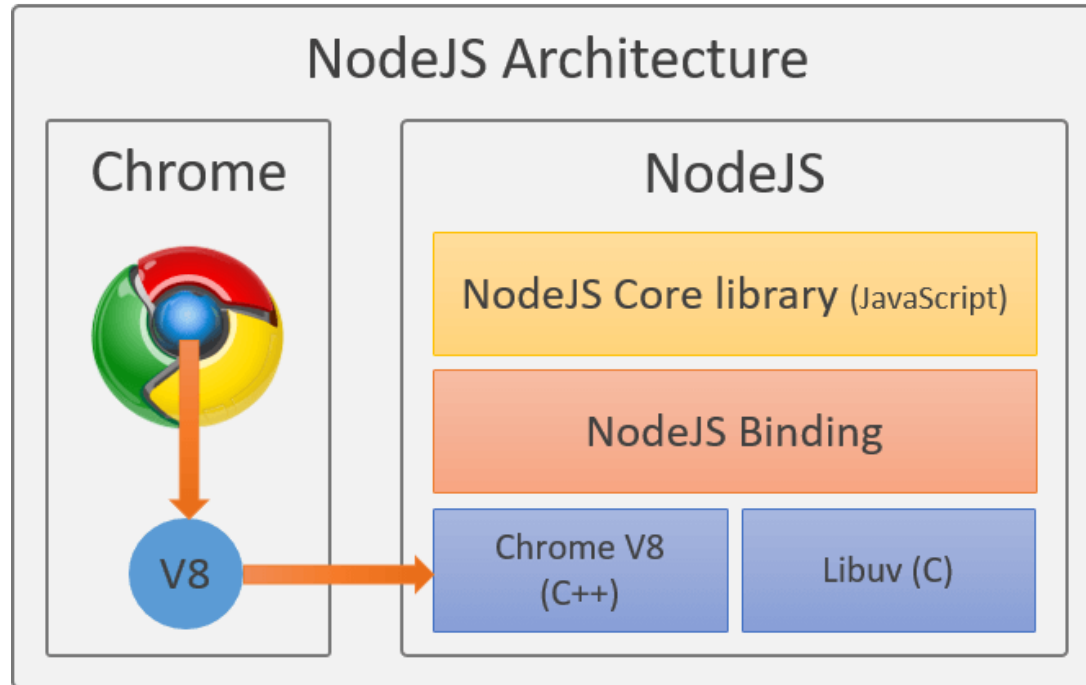
V8

Aunque **NODE** permite utilizar JS fuera del navegador, todo empezó en un navegador:



El motor de tiempo de ejecución de ejecución de
Chrome: **V8**

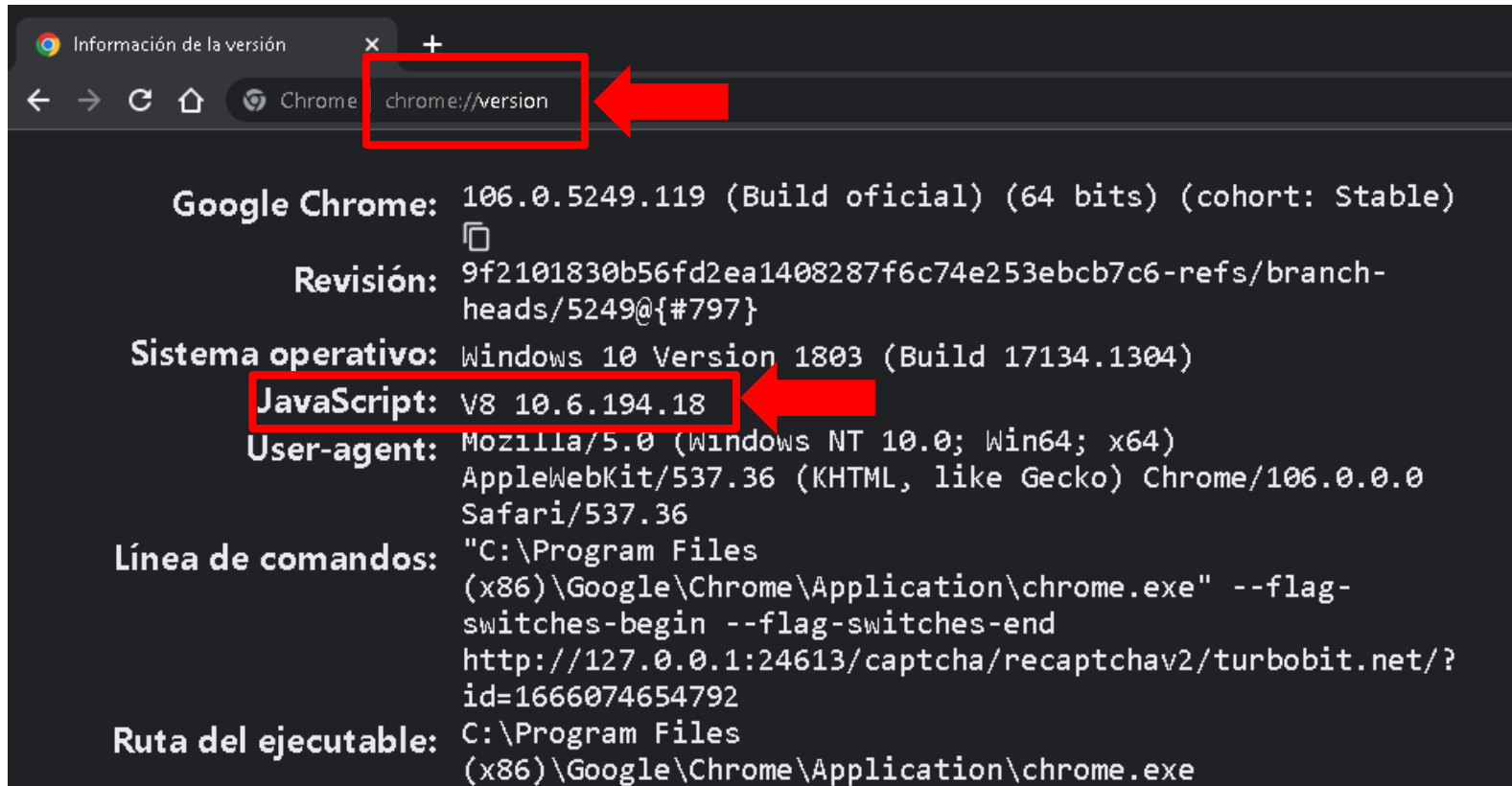
V8



V8 está escrito en C++ y compila Javascript en código máquina nativo en vez de interpretarlo en el navegador.

Comparación de versiones de V8 en Node y Chrome

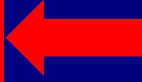
En Chrome: <chrome://version>



Comparación de versiones de V8 en Node y Chrome

En Node:

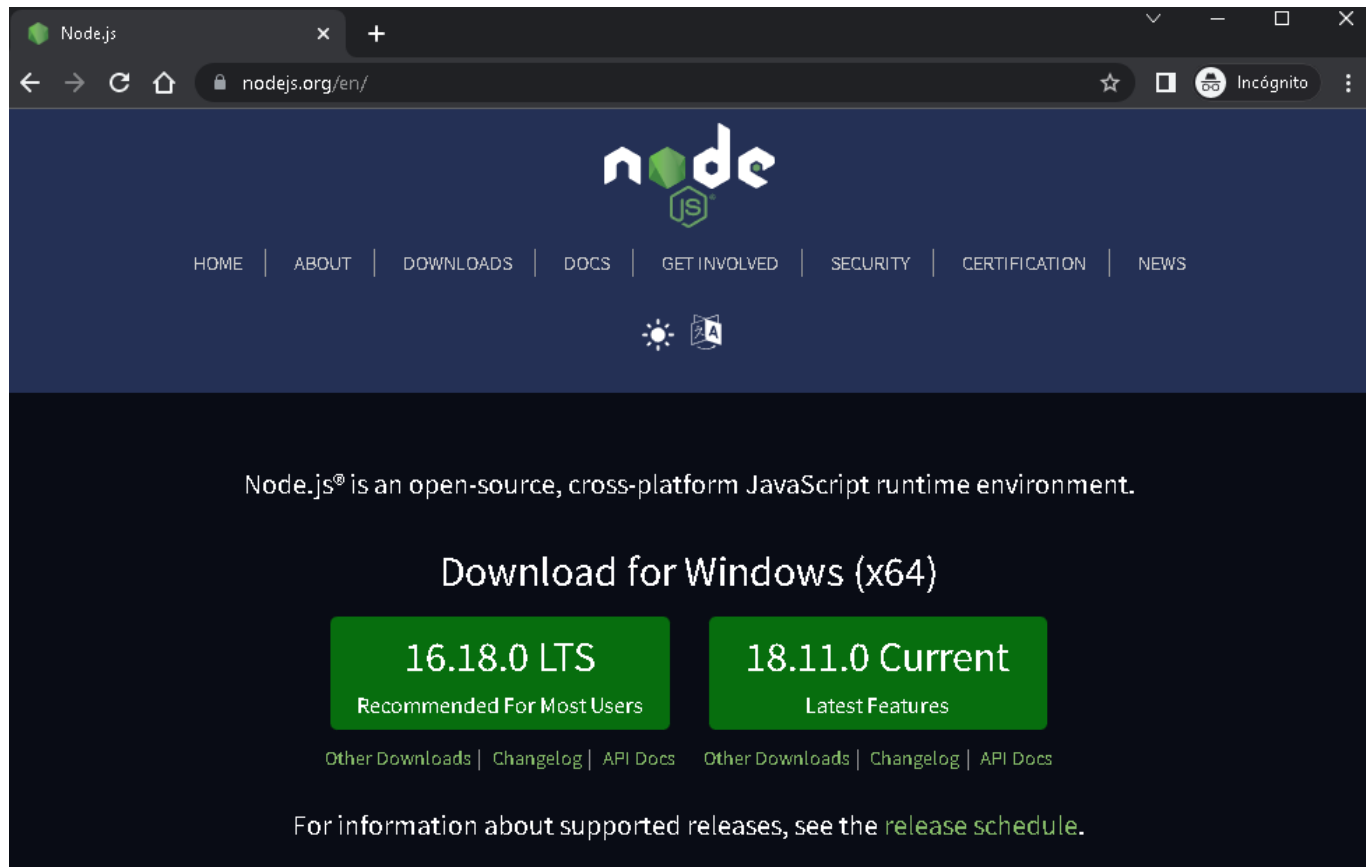
```
C:\>node
Welcome to Node.js v15.5.1.
Type ".help" for more information.
> process.versions.v8
'8.6.395.17-node.23'
```



Después de la instalación de Node podemos verificar que versión de V8 corresponde

Instalación de NODEJS

Aunque es muy fácil descargar e instalar NODE desde su página web, no es el sistema más recomendado para usuarios de Windows.



Instalación de NODEJS

NODE se actualiza con frecuencia, mucho más rápido que el software que depende de él y de sus extensiones.

Por eso podemos necesitar versiones antiguas de NODE para el mantenimiento de aplicaciones realizadas años atrás. Es muy habitual para frameworks com Angular, React o Vue, entre otros.



Es imprescindible disponer de herramientas para elegir sobre qué versión trabajar. Actualmente dos buenas opciones son:

- Node **V**ersion **M**anager → **NVM**
- Node **V**ersion **S**witcher → **NVS**

Instalación de NVM

Node Version Manager

En Linux y Mac se instala con Node, pero no con Windows. Debemos descargarla e instalarla a parte y antes de Node.

Consejos para la instalación:

<https://learn.microsoft.com/es-es/windows/dev-environment/javascript/nodejs-on-windows>

<https://www.freecodecamp.org/news/node-version-manager-nvm-install-guide/>

Instalación de NVM

Repositorio oficial de nvm para Windows:

<https://github.com/coreybutler/nvm-windows/releases>

▼ Assets 9

 nvm-noinstall.zip	3.64 MB	16 Dec 2021
 nvm-noinstall.zip.checksum.txt	34 Bytes	16 Dec 2021
 nvm-setup.exe	4.64 MB	28 Apr 2022
 nvm-setup.zip	4.14 MB	16 Dec 2021
 nvm-setup.zip.checksum.txt	34 Bytes	16 Dec 2021
 nvm-update.zip	3.45 MB	16 Dec 2021
 nvm-update.zip.checksum.txt	34 Bytes	16 Dec 2021
 Source code (zip)		10 Dec 2021
 Source code (tar.gz)		10 Dec 2021

Comandos útiles de NVM

Versión de nvm:

```
nvm -v
```

Establecer la versión por defecto:

```
nvm use vX.Y.Z
```

Instalar la última versión de Node:

```
nvm install latest
```

Establecer que la versión que se instala lo sea por defecto

```
nvm alias default vX.Y.Z
```

Elegir una versión concreta para instalar:

```
nvm install vX.Y.Z
```

Versiones disponibles para instalar:

```
nvm list available
```

Nota: Para que algunas de estas instrucciones se ejecuten es necesario abrir la consola CMD o POWERSHELL con privilegios de administrador.

Comandos útiles de NVM

Versión de nvm:

```
E:\>nvm -v
```

nvm -v

```
Running version 1.1.7.
```

Muestra la versión actual y un menú de ayuda

```
E:\>nvm -v
Running version 1.1.7.

Usage:
  nvm arch                : Show if node is running in 32 or 64 bit mode.
  nvm install <version> [arch] : The version can be a node.js version or "latest" for the latest stable version.
                                Optionally specify whether to install the 32 or 64 bit version (defaults to system arch).
                                Set [arch] to "all" to install 32 AND 64 bit versions.
                                Add --insecure to the end of this command to bypass SSL validation of the remote download server.
  nvm list [available]      : List the node.js installations. Type "available" at the end to see what can be installed. Aliased as ls.
  nvm on                    : Enable node.js version management.
  nvm off                   : Disable node.js version management.
  nvm proxy [url]           : Set a proxy to use for downloads. Leave [url] blank to see the current proxy.
                                Set [url] to "none" to remove the proxy.
  nvm node_mirror [url]     : Set the node mirror. Defaults to https://nodejs.org/dist/. Leave [url] blank to use default url.
  nvm npm_mirror [url]      : Set the npm mirror. Defaults to https://github.com/npm/cli/archive/. Leave [url] blank to default url.
  nvm uninstall <version>   : The version must be a specific version.
  nvm use [version] [arch]  : Switch to use the specified version. Optionally specify 32/64bit architecture.
                                nvm use <arch> will continue using the selected version, but switch to 32/64 bit mode.
  nvm root [path]           : Set the directory where nvm should store different versions of node.js.
                                If <path> is not set, the current root will be displayed.
  nvm version               : Displays the current running version of nvm for Windows. Aliased as v.
```

Comandos útiles de NVM

Lista de versiones instaladas:

nvm list

```
E:\>nvm list
```

```
* 15.5.1 (Currently using 64-bit executable)
  12.5.0
  10.15.3
```

Prefijar versión por defecto

nvm use X.Y.Z

```
E:\>nvm ls
```

```
* 15.5.1 (Currently using 64-bit executable)
  12.5.0
  10.15.3
```

```
E:\>nvm use 10.15.3
```

```
Now using node v10.15.3 (64-bit)
```

Eliminar versión del sistema

nvm uninstall X.Y.Z

```
E:\>nvm uninstall 12.5.0
```


Instalación de NVS

URL con las versiones disponibles:

<https://github.com/jasongin/nvs/releases>

GitHub de la aplicación:

<https://github.com/jasongin/nvs>

NVS (Node Version Switcher)

build error build passing

NVS is a cross-platform utility for switching between different versions and forks of **Node.js**. NVS is itself written in node JavaScript.

This tool is obviously inspired by other node version manager tools, especially **nvm**, from which it borrows a lot of ideas and some command-line syntax.

Comandos de NVS

Command	Description
<code>nvs help <command></code>	Get detailed help for a command
<code>nvs install</code>	Initialize your profile for using NVS
<code>nvs uninstall</code>	Remove NVS from profile and environment
<code>nvs --version</code>	Display the NVS tool version
<code>nvs add [version]</code>	Download and extract a node version
<code>nvs rm <version></code>	Remove a node version
<code>nvs migrate <fromver> [tover]</code>	Migrate global modules
<code>nvs upgrade [fromver]</code>	Upgrade to latest patch of major version
<code>nvs use [version]</code>	Use a node version in the current shell
<code>nvs auto [on/off]</code>	Automatically switch based on cwd

Comandos de NVS

Command	Description
<code>nvs run <ver> <js> [args...]</code>	Run a script using a node version
<code>nvs exec <ver> <exe> [args...]</code>	Run an executable using a node version
<code>nvs which [version]</code>	Show the path to a node version binary
<code>nvs ls [filter]</code>	List local node versions
<code>nvs ls-remote [filter]</code>	List node versions available to download
<code>nvs link [version]</code>	Link a version as the default
<code>nvs unlink [version]</code>	Remove links to a default version
<code>nvs alias [name] [value]</code>	Set or recall aliases for versions
<code>nvs remote [name] [value]</code>	Set or recall download base URIs

Comandos de NVS

```
E:\>nvs
```

```
.-----  
| Select a node version |  
+-----+  
| [a] node/19.0.0       |  
| b) node/18.11.0       |  
| c) node/18.10.0       |  
| d) node/18.9.1        |  
| e) node/18.9.0        |  
| f) node/18.8.0        |  
| g) node/18.7.0        |  
| h) node/18.6.0        |  
| i) node/18.5.0        |  
| j) node/18.4.0        |  
| k) node/18.3.0        |  
| l) node/18.2.0        |  
| m) node/18.1.0        |  
| n) node/18.0.0        |  
| o) node/17.9.1        |  
+-----+  
'\
```

```
Type a hotkey or use Down/Up arrows then Enter to choose an item.
```

Comandos de NVS

```
E:\>nvs help
NVS (Node Version Switcher) usage

nvs help <command>          Get detailed help for a command
nvs install                  Initialize your profile for using NVS
nvs --version                 Display the NVS tool version

nvs menu                     Launch an interactive menu

nvs add <version>            Download and extract a node version
nvs rm <version>             Remove a node version
nvs migrate <fromver> [tover] Migrate global modules
nvs upgrade [fromver]        Upgrade to latest patch of major version

nvs use [version]            Use a node version in the current shell
nvs auto [on/off]            Automatically switch based on cwd
nvs run <ver> <js> [args...] Run a script using a node version
nvs exec <ver> <exe> [args...] Run an executable using a node version
nvs which [version]          Show the path to a node version binary

nvs ls [filter]              List local node versions
nvs ls-remote [filter]       List node versions available to download
nvs outdated                 List local node versions and available updates

nvs link [version]           Link a version as the default
nvs unlink [version]         Remove links to a default version

nvs alias [name] [value]     Set or recall aliases for versions
nvs remote [name] [uri]      Set or recall download base URIs

A version string consists of a semantic version number or version label
("lts" or "latest"), optionally preceded by a remote name, optionally
followed by an architecture, separated by slashes.
Examples: "lts", "4.6.0", "6.3.1/x86", "node/6.7.0/x64"
Aliases may also be used anywhere in place of a version string.
```

Consola de NODEJS

Versión de **node** activa:

```
node -v  
node --version
```

NODE dispone de una consola muy básica para realizar operaciones sencillas:

Read **E**val **P**rint **L**oop = **REPL**

```
F:\>node  
Welcome to Node.js v15.5.1.  
Type ".help" for more information.  
> .help  
.break      Sometimes you get stuck, this gets you out  
.clear      Alias for .break  
.editor     Enter editor mode  
.exit       Exit the REPL  
.help       Print this help message  
.load       Load JS from a file into the REPL session  
.save       Save all evaluated commands in this REPL session to a file  
  
Press Ctrl+C to abort current expression, Ctrl+D to exit the REPL  
> 
```