



JESUÏTES
educació

DAWM12

GP1 FRAMEWORKS AVANÇATS EN JS

GP1 7.1 INTRODUCCIÓ A NODE.JS

CFGS Desenvolupament d'Aplicacions Web

M12. Projecte Final - Global Project

Fundació Jesuïtes Educació - Escola del Clot

Sergi Grau sergi.grau@fje.edu

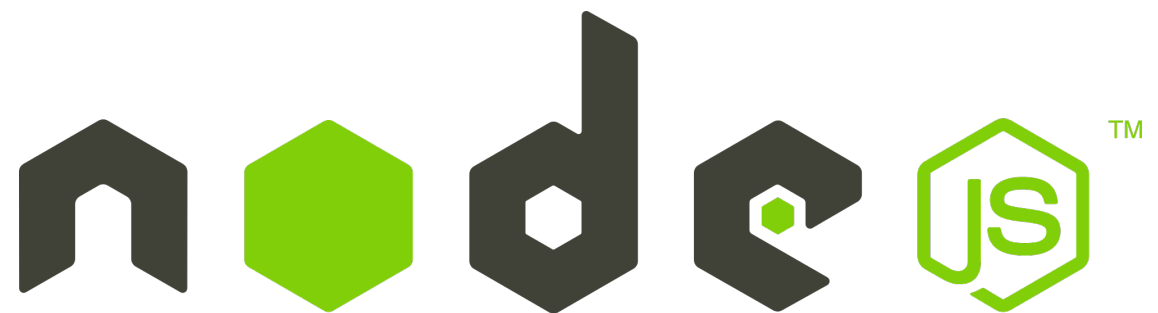


- + Aprendre altres característiques del funcionament de NodeJS
- + Comprendre com funciona el paradigma d'esdeveniments
- + Entendre els avantatges de les E/S no blocants.

QUÈ ÉS NODE.JS ?



- + NODE.JS ens permet construir aplicacions en xarxa escalables utilitzant JS del costat servidor.
- + Permet desenvolupar servidors web que utilitzen Websockets, allotjament ràpid de fitxers i aplicacions en temps real.



- + NODE.JS és un entorn de programació en la capa del servidor basat en el llenguatge de programació Javascript, amb I / O de dades en una arquitectura orientada a esdeveniments i basat en el motor Javascript V8.
- + Va ser creat amb l'enfocament de ser útil en la creació de programes de xarxa altament escalables, com per exemple, servidors web. Va ser creat per Ryan Dahl en 2009 i la seva evolució està apadrinada per l'empresa Joyent, que a més té contractat Dahl en plantilla.
- + Actualment Node.JS és una fundació <https://nodejs.org/en/foundation/>

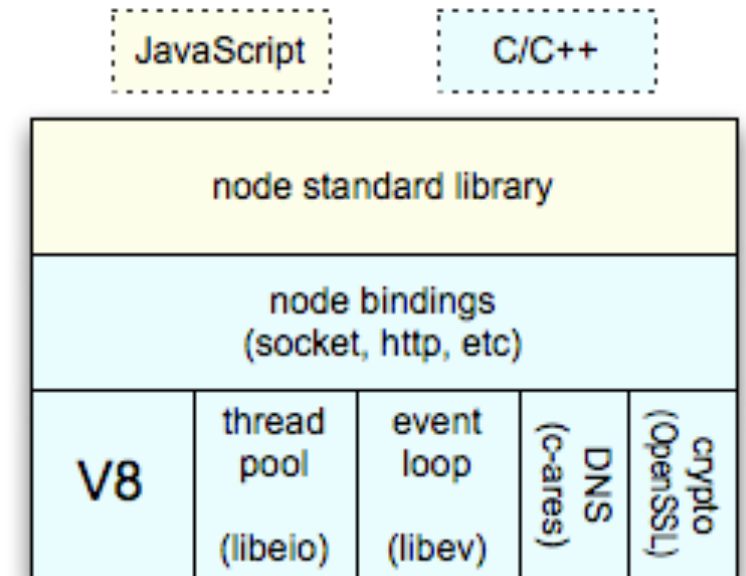
- + NODE.JS és similar en el seu propòsit a Twisted o Tornado de Python, Perl Object Environment de Perl, React de PHP, libevent o libev de C, EventMachine de Ruby, vibe.d de D i de Java existeix Apache MINA, Netty, Akka, Vert.x, Grizzly o Xsocket.
- + Al contrari que la majoria del codi JavaScript, no es s'executa en un navegador, sinó en el servidor. NODE.JS implementa algunes especificacions de CommonJS.
- + NODE.JS inclou un entorn REPL (Read–eval–print loop) per depuració interactiva.

- + NODE.JS no és framework web, no és basic, doncs cal fer molt treball a mà, i no és multifils (és com un servidor amb un sol fil, i la seva concurrència està basada en esdeveniments)

- + L'objectiu original de Dahl va ser crear llocs web amb capacitats d'empenta (push) com es veu en aplicacions web com Gmail. Després de provar les solucions en diversos llenguatges de programació, va triar JavaScript causa a la manca d'una API d'E / S existent. Això li va permetre definir una nova convenció de no bloqueig, amb orientació a esdeveniments

```
var http = require("http");  
http.createServer(function(request, response) {  
    response.writeHead(200, {"ContentType": "text/html"});  
    response.write("Hola Món");  
    response.end();  
}).listen(8888);
```


- + Hi ha quatre components bàsics que constitueixen NODE.JS. En primer lloc, el NODE.JS encapsula libev per manejar esdeveniments asíncrons i V8 de Google per proporcionar un temps d'execució de JavaScript. Libev abstreu tota la xarxa subjacent i la funcionalitat del sistema d'arxius tant en sistemes basats en POSIX com Linux, Mac OS X UNIX o Windows. La funcionalitat bàsica de Node, són mòduls com Assert, HTTP, Crypto etc, que resideixen en una biblioteca nucli escrit en JavaScript. Els enllaços NODE.JS, escrites en C ++ proporcionen la cola que connecta aquestes tecnologies entre si i amb el sistema operatiu.



- + Les primeres aplicacions de JavaScript s'executen en els navegadors. JavaScript és un llenguatge molt potent que es pot usar en molts contextos. NODE.JS realment és només un altre context més: et permet córrer codi JavaScript al backend, fora del navegador.
- + Per executar el codi JavaScript al backend, aquest necessita ser interpretat i executat, Això és el que NODE.JS realitza, fent ús de la màquina virtual V8 de Google, el mateix entorn d'execució per a JavaScript que Google Chrome utilitza.
- + A més, NODE.JS ve amb molts mòduls útils, de manera que no cal escriure tot de zero, com per exemple, una cosa que posi una cadena a la consola. Llavors, NODE.JS és en realitat dues coses: un entorn d'execució i una llibreria.

- + <http://www.hostingadvice.com/blog/comparing-node-js-vs-php-performance/>