

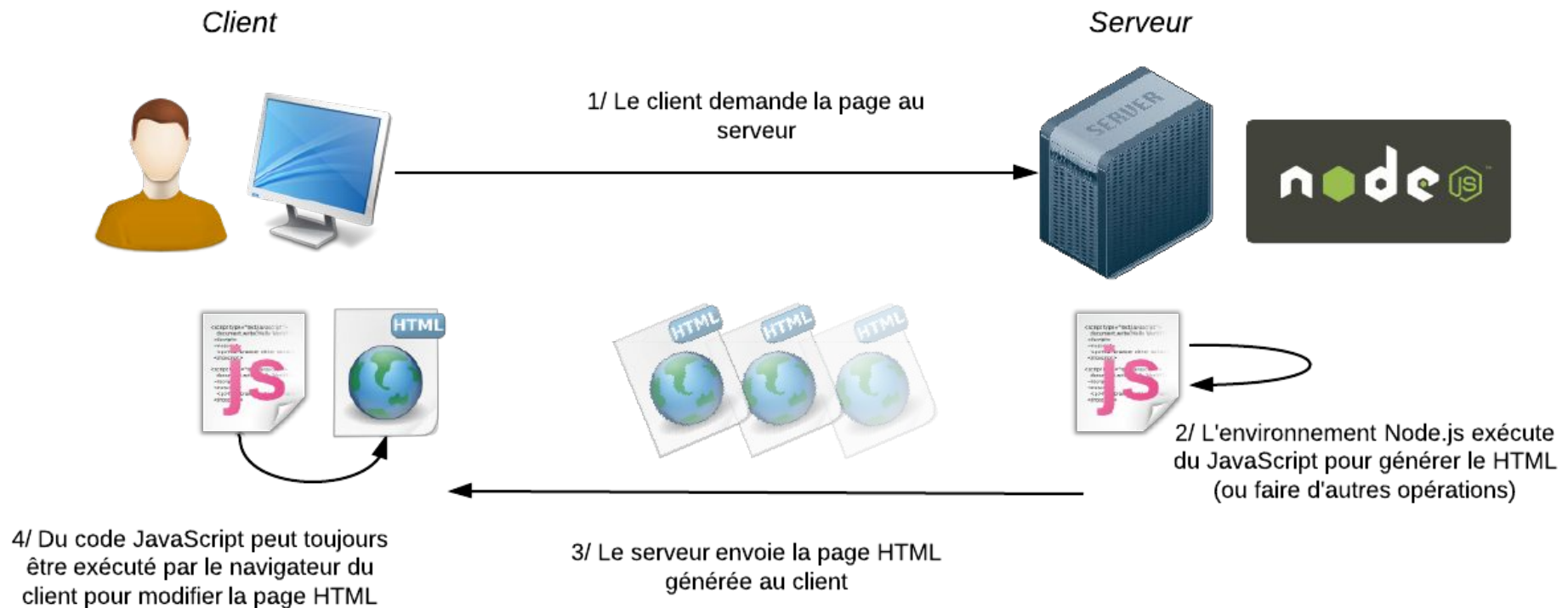
Introduction

NodeJs

Introduction

Node.js : le JavaScript côté serveur

Node.js offre un environnement côté serveur qui nous permet d'utiliser le langage JavaScript pour générer des pages web tels les langages serveur comme ASP, PHP, Java EE, etc.



(crédit : OpenclassRoom)

Introduction

- Node Js, comme Js est un **langage événementiels**
- Contrairement au php ou au Java, toute la programmation est à penser **orienté évènements**
- Node Js est un environnement de **bas niveau** se rapprochant plus du C que du Php
- Node Js n'est **PAS un framework**! Des frameworks existent pour NJS ☐ Express
- Node Js est **Rapide**

Introduction

Le moteur V8



V8 JavaScript engine est un moteur JavaScript open source développé par Google au Danemark. Il est notamment utilisé dans les navigateurs Internet Google Chrome et Chromium. Il fonctionne sur les architectures x86 (32 bits et 64 bits) et ARM.

NodeJS utilise ce moteur qui transforme le code JavaScript très rapidement en code machine et l'optimise même grâce à des procédés complexes : code inlining, copy elision

Plus d'infos sur <http://blog.js-republic.com/v8-engine-comment-ca-marche/>

Introduction

Le modèle non bloquant

Comme JavaScript est un langage conçu autour de la notion d'évènement, Node.js a utilisé ce principe pour mettre en place une architecture de code entièrement non bloquante.

Le mode non bloquant consiste tout simplement en l'utilisation de callback pour rappeler des parties de code une fois une instruction terminée plutôt que d'attendre que celle-ci finisse de s'exécuter.

Exemple:

« Je souhaite télécharger un fichier Excel une fois celui-ci généré par le serveur. Ce fichier doit traiter une grosse quantité de données, l'opération va plus que certainement être trèèèèè longue! »

Introduction

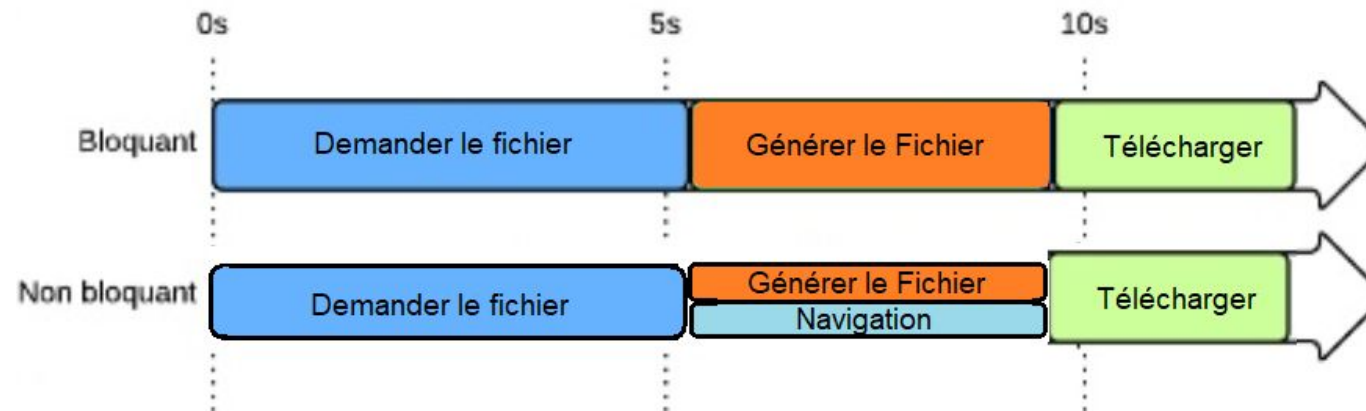
Modèle bloquant

- J'appelle et je traite ma source de données
- Je génère mon fichier 🕒
- Je télécharge mon fichier
- Je continue ma navigation

Modèle non-bloquant

- J'appelle et je traite ma source de données
 - Je génère mon fichier
 - Je télécharge mon fichier
- Je continue ma navigation

La navigation continue et les sous-tâches seront effectuées une fois le, les événements parents terminés.



Installation

NodeJs

Installation

L'installation est plutôt simple.

Depuis l'accueil du site <https://nodejs.org/>, vous devriez voir un bouton « Download for Windows (x64) » (si vous êtes sur Windows).

Cliquez pour télécharger automatiquement le zip le plus adapté à votre système.

Vous pouvez également choisir vous même votre zip dans [Download](#) pour obtenir au choix le *Windows Installer (.msi)* et/ou le *Windows Binary (.exe)* en 32/64 bit.



Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for Windows (x64)

12.16.2 LTS

Recommended For Most Users

13.12.0 Current

Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.

Installation

Exécutez votre fichier, acceptez la licence, choisissez le dossier "Program files" et ce package va vous installer :

- **L'exécuteur node.js :**

le programme permettant d'exécuter des fichiers .js (comme php.exe le ferait avec des .php).

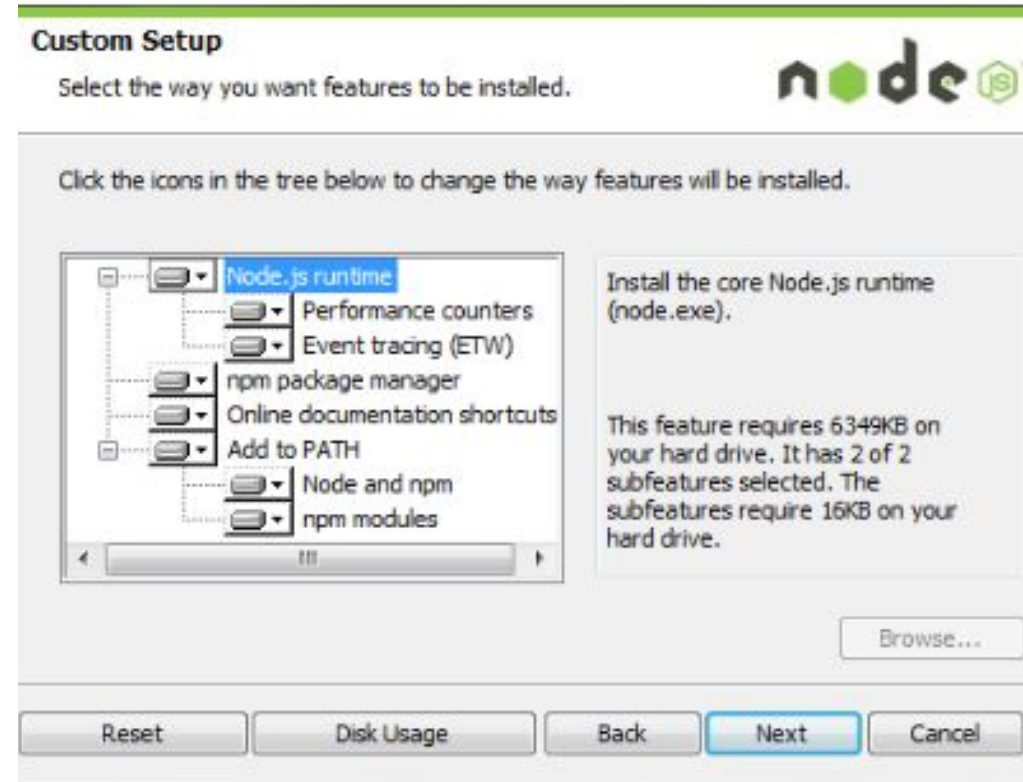
- **Le module npm (Node Package Manager) :**

un gestionnaire de modules qui va vous permettre simplement d'ajouter et retirer les librairies dont vous aurez besoin pour vos applications (pas de surplus, seulement le nécessaire donc).

- **Un raccourci vers la documentation en ligne.**

- **Des variables d'environnements :**

Ainsi vous pourrez exécuter les commandes node et npm dans votre invité de commande.



Installation

Quand l'installateur aura fini vous aurez un dossier d'environ 12Mo contenant entre autre node.js et npm.cmd

Ces éléments seront « appelables » depuis n'importe quel dossier avec les commandes *node* et *npm*.

