<epam>

Introduction

Node.js Global Mentoring program

November 20, 2018

PAVEL AURAMENKA

Senior Software Engineer

- Experienced in front-end development: AngularJS, Angular, React, React Native
- Passionate Node.js developer: Hapi, Express
- twitter: @abramenal
- skype: paul.ovramenko
- https://linkedin.com/in/abramenal

Agenda

- Introduction & prerequisites
    - Course agenda
    - Prerequisites
    - What am I going to get from this course?
- Node.js basics
    - Brief history
    - JavaScript Engines, V8 core
    - Node.js core
- Quick start guide
    - Install & launch. NVM
    - ES* features support
    - Sources & docs
    - Q&A

INTRODUCTION & PREREQUISITES

Introduction & prerequisites

SCHEDULE

1. Node.js Introduction
2. Node.js Modules. NPM
3. Node.js Events
4. Async development introduction
5. Command line. Debugging
6. Filesystem and streams
7. Async network API
8. HTTP module
9. Middleware. Frameworks
10. Authorization. Validation
11. SQL Databases. ORM
12. NoSQL Databases. ODM
13. Node.js Testing
14. Deploy and tools
15. Swagger. Docs
16. Serverless architecture with AWS
17. Dockerization

HOMEWORKS

Task 1 after 2nd lecture

Task 2 after 4rd lecture

Task 3 after 6th lecture

Task 4 after 9th lecture

Task 5 after 10th lecture

Task 6 after 11th lecture

Task 7 after 12th lecture

Task 8 after 15th lecture

DEADLINES

- Reflected in Grow
- Depends on the complexity of the module
- May be set individually for a concrete task
- Should not be missed (w/o any acceptable reasons)
- In case of VAC/OOO, the deadline is to be negotiated

Introduction & prerequisites

PREREQUISITES

- JS Core, ES6+
- ENGX bootcamp
- Attitude & dedication

TOOLS THAT WE USE

- Code editor / IDE: VSCode, WebStorm, Atom, etc.
- VCS: Git
- Command-line interface
- Additional tools prior to specific topic

Introduction & prerequisites

WHAT I AM GOING TO GET FROM THIS COURCE?

- Understand technical concepts behind Node.js
- Structure a Node.js application in **modules**
- Use **npm** and manage node packages
- Understand async development concents
- Learn **Buffers, Streams, Pipes, Events**
- Enrich knowledge of how communication via HTTP works
- Build a web server in Node
- Build an API using un-opinionated Node.js framework
- Learn SQL and NoSQL databases and their differences
- Connect Node.js app to a SQL/NoSQL database
- Test your Node.js app properly
- How to document your API
- Deploy essentials, Docker
- Serverless

NODE.JS BASICS

Node.js basics

WHAT IS NODE.JS?

Node.js basics

WHAT IS NODE.JS?

- Server-side JavaScript 😐
- Asynchronous event-driven JavaScript runtime 😎
- Open-source and cross-platform
- Built on Chrome’s V8 JavaScript engine
- Designed to build scalable network applications
- Non-blocking IO
- Single-threaded
- There is no DOM implementation provided by Node.js

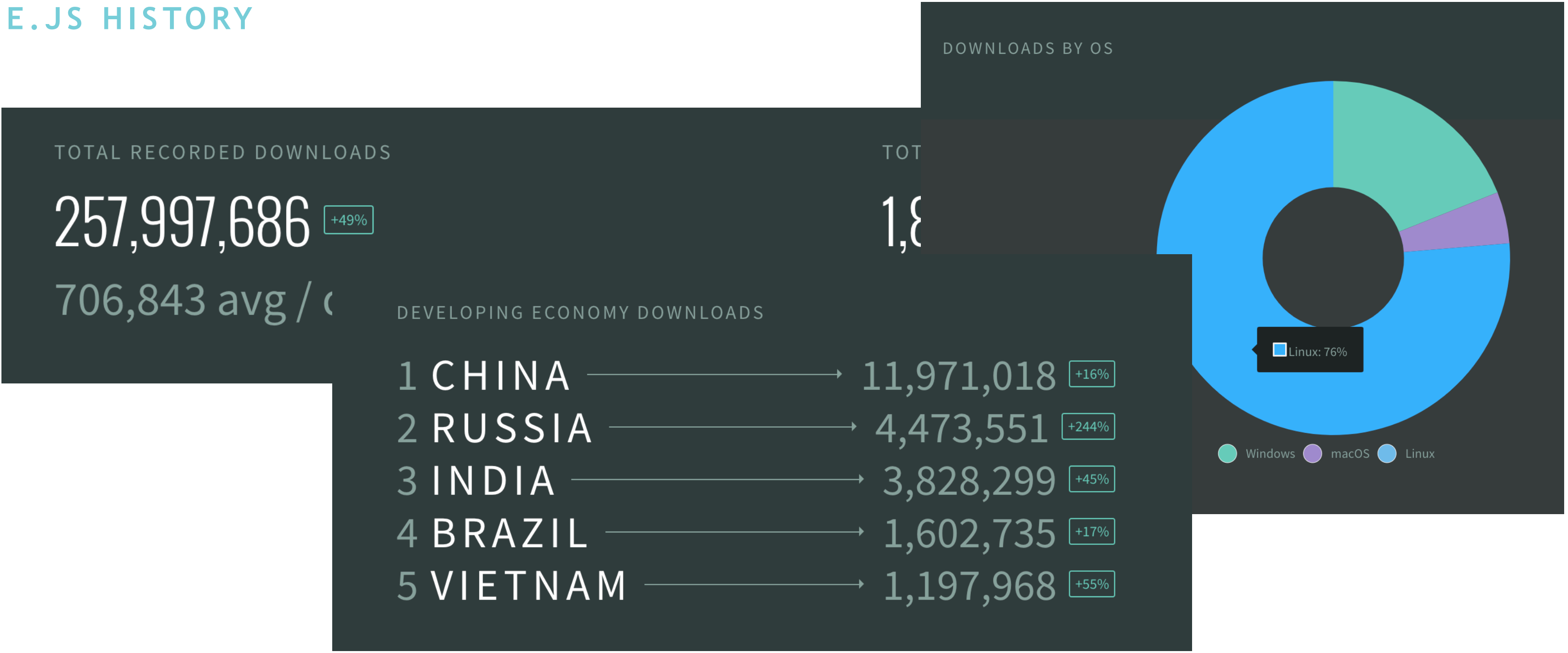
Node.js basics

NODE.JS HISTORY

- Created by Ryan Dahl, first 2009 for Linux only
- Sponsored by Joyent
- June 2011, Microsoft partnered with Joyent to help create a native Windows version of Node.js
- December 2014, Fedor Indutny started io.js
- September 2015, Node.js v0.12 and io.js v3.3 were merged back together into Node.js v4.0
- August 2017, Node.js forked again over complaints of unresponsive leadership - ayo.js
- Last commit on December 2017, so it’s dead now

Node.js basics

NODE.JS HISTORY



TOTAL RECORDED DOWNLOADS

257,997,686 +49%

706,843 avg / (

TOT

1,8

DEVELOPING ECONOMY DOWNLOADS

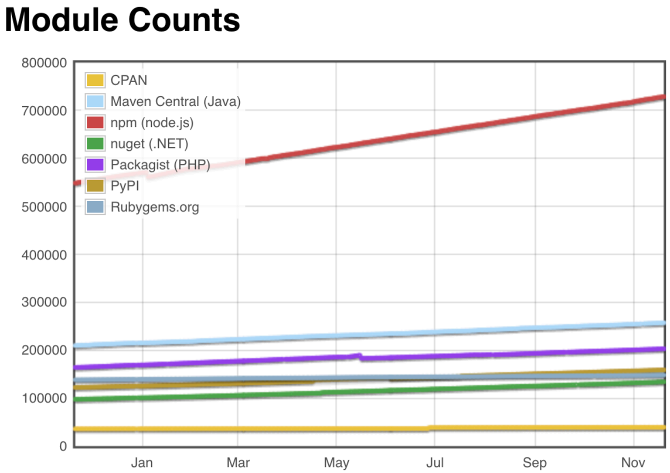
1	CHINA	11,971,018	+16%
2	RUSSIA	4,473,551	+244%
3	INDIA	3,828,299	+45%
4	BRAZIL	1,602,735	+17%
5	VIETNAM	1,197,968	+55%

DOWNLOADS BY OS

Node.js basics

NPM

NPM – world's largest package registry and package manager for JavaScript.
Discover packages with reusable code – and assemble
them in powerful new ways.



728924 modules now – 532 per day

NPM AUDIT

Node.js basics

ECMASCRIPT VS JAVASCRIPT

ECMAScript (or ES) is a trademarked scripting-language specification standardized by Ecma International in ECMA-262 and ISO/IEC 16262. It was created to standardize JavaScript, so as to foster multiple independent implementations.

Take a look at ECMAScript specification

Node.js basics

ECMASCRIPT VS JAVASCRIPT

- ECMA - European Computer Manufacturers Association
- ECMA International has been the standards body for JavaScript since 1996 and work to keep the language modern and relevant.
- The ECMAScript specification 2015, a document outlining new features for JavaScript and how they should be implemented was finalized in the middle of 2015. The latest standard version is 9th – ECMAScript 2018 (finalized in June).
- ECMA TC39 committee is working on new ECMAScript features.

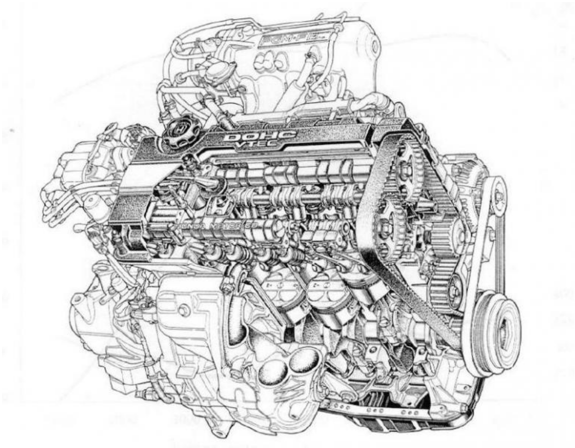
Take a look at ECMAScript specification

Node.js basics

ECMASCRIPT VS JAVASCRIPT ENGINES

An **ECMAScript engine** is a program that executes source code written in a version of the ECMAScript language standard, for example, JavaScript.

A **JavaScript Engine** – a program that converts JavaScript code into something the computer processor can understand. It should follow the ECMAScript standard on how the language should work and what features it should have



Take a look at list of JavaScript engines

Node.js basics

V8: THE JAVASCRIPT ENGINE

V8 is Google’s open source high-performance **JavaScript engine**, written in C++ and used in Google Chrome, the open source browser from Google. It implements **ECMAScript** as specified in **ECMA-262**, and runs on Windows 7 or later, macOS 10.5+, and Linux systems that use IA-32, ARM, or MIPS processors.

V8 can run standalone, or can be embedded into any C++ application

Node.js basics

V8: THE JAVASCRIPT ENGINE

What do we know about V8?

- V8 is an open source JavaScript engine developed by Google. Its written in C++ and is used in Google Chrome Browser
- Implements latest ECMAScript standard
- Node.js runs on V8
- Ryan Dahl: ”Node.js is a set of bindings to the V8 JavaScript VM”

Node.js basics

V8: THE JAVASCRIPT ENGINE

Node.js basics

NODE.JS CORE

What does JavaScript need to manage a server?

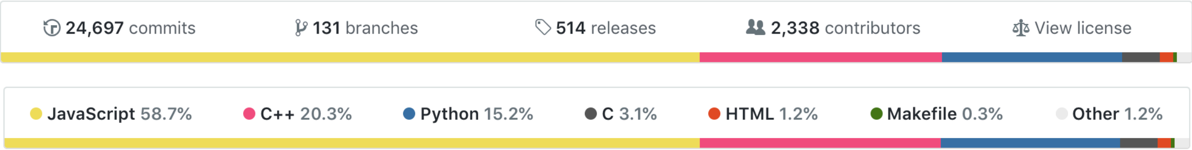
- Ability to **communicate** over the internet
- Ability to accept **requests** and send **responses**
- Ways to deal with **files**
- Ways to operate with **databases**
- Ways to organize code into reusable pieces
- Approach to deal with long running operations that takes time

Node.js basics

NODE.JS CORE

Node is similar in design to and influenced by systems like Ruby's event machine or Python's twisted. Node takes the event model a bit further—it presents the event loop as a language construct instead of as a library.

In “Node.js” “.js” does not mean that it’s solely written in JavaScript:

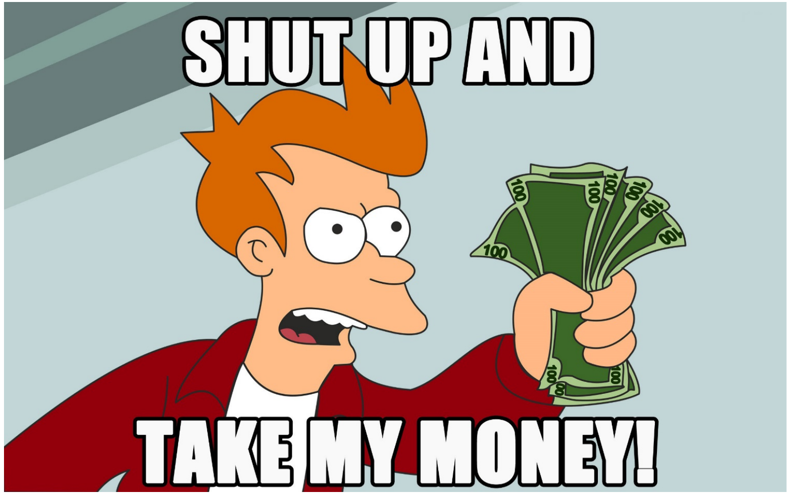


24,697 commits | 131 branches | 514 releases | 2,338 contributors | View license

● JavaScript 58.7% ● C++ 20.3% ● Python 15.2% ● C 3.1% ● HTML 1.2% ● Makefile 0.3% ● Other 1.2%

Node.js basics

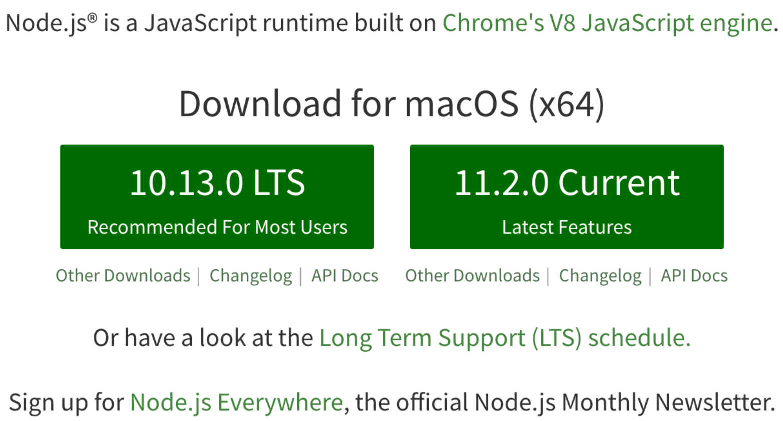
NODE.JS CORE



QUICK START QUIDE

Quick start guide

INSTALL AND LAUNCH



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

Download for macOS (x64)

10.13.0 LTS
Recommended For Most Users
11.2.0 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.

https://nodejs.org/en/

HOMEBREW/CHOCOLATEY

```
brew install nodejs
// or
choco install nodejss
```

https://brew.sh/ https://chocolatey.org/

NVM

```
brew install nvm
nvm install node
```

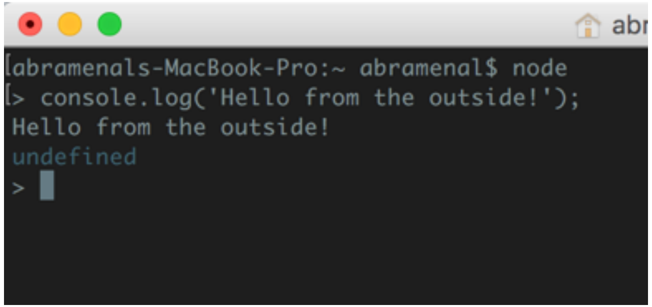
https://github.com/creationix/nvm/

Quick start guide

INSTALL AND LAUNCH

To launch:

- Open terminal (powershell) window
- Type ***node*** to get into REPL
- Evaluate any valid JS code



```
abramenals-MacBook-Pro:~ abramenal$ node
> console.log('Hello from the outside!');
Hello from the outside!
undefined
> 
```

Quick start guide

ES* SUPPORT

- To check any ES feature availability in Node - simply go to http://node.green/
- To check any ESNext feature availability on platforms - http://kangax.github.io/compat-table/esnext/

Quick start guide

ES* SUPPORT

All ECMAScript features are split into three groups for **shipping**, **staged**, and **in progress** features:

- All **shipping** features, which V8 considers stable, are turned on by default on Node.js and do NOT require any kind of runtime flag.
- **Staged** features, which are almost-completed features that are not considered stable by the V8 team, require a runtime flag: --harmony.
- **In progress** features can be activated individually by their respective harmony flag, although this is highly discouraged unless for testing purposes. Note: these flags are exposed by V8 and will potentially change without any deprecation notice.

Quick start guide

SOURCES AND DOCS

- Official Node.js repository: https://github.com/nodejs/node
- Official Node.js Documentation set: https://nodejs.org/dist/latest-v11.x/docs/api/

Quick start guide

COMPANIES THAT USE NODE.JS

LinkedIn

ebay

UBER

PayPal

Walmart

StrongLoop

YAHOO!

COMPANIES THAT HOST NODE.JS

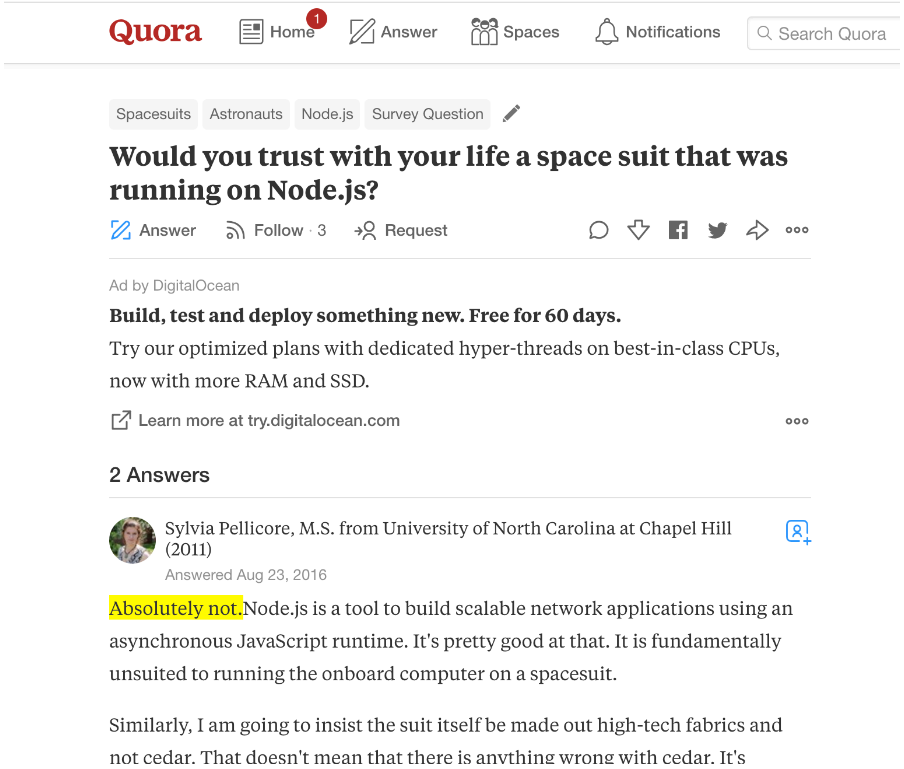
nodejitsu

heroku

Microsoft Azure

amazon web services

Joyent



Spacesuits Astronauts Node.js Survey Question

Would you trust with your life a space suit that was running on Node.js?

2 Answers

Sylvia Pellicore, M.S. from University of North Carolina at Chapel Hill (2011)
Answered Aug 23, 2016

Absolutely not.Node.js is a tool to build scalable network applications using an asynchronous JavaScript runtime. It's pretty good at that. It is fundamentally unsuited to running the onboard computer on a spacesuit.

Similarly, I am going to insist the suit itself be made out high-tech fabrics and not cedar. That doesn't mean that there is anything wrong with cedar. It's

SUMMARY

QUESTIONS?

- http://www.modulecounts.com/
- https://nodejs.org/en/user-survey-report/
- https://nodesource.com/node-by-numbers
- https://github.com/nodejs/node
- https://brew.sh/
- https://nodejs.org/dist/latest-v11.x/docs/api/
- https://www.quora.com/Would-you-trust-with-your-life-a-space-suit-that-was-running-on-Node-js
- https://www.toptal.com/nodejs/top-10-common-nodejs-developer-mistakes
- https://kb.epam.com/pages/viewpage.action?pageId=681143000

THANK YOU!