

Namaste Node.js

- By Akshay Saini

Season 01 →

Episode-01 → Introduction to NodeJS

Episode-02 → JS On Server

Episode-03 → Let's Write Code

Episode-04 → module.export & require

Episode-05 → Diving into the NodeJS github repo

Episode-06 → libuv & async IO

Episode-07 → sync, async, setTimeoutZero - Code

Episode-08 → Deep dive into V8 JS Engine

Episode-09 → libuv & Event Loop

Episode-10 → Thread Pool in libuv

Episode-11 → Creating a Server

Episode-12 → Databases - SQL & NoSQL

Episode-13 → Creating a database & mongoDB

For, Namaste Node.js Handwritten Pdf visit →

gitHub → [rajeshjha2000](#)

Episode - 01 → Introduction to NodeJS

- Quote by Jeff Atwood (Founder, stackOverflow) in 2007
"Any application that can be written in JavaScript, will eventually be written in JavaScript".

Q. > what is exactly NodeJS ?

Ans. > "Run JavaScript Everywhere".

- Node.js is a JavaScript runtime built on Chrome's V8 engine.
- Wikipedia → Node.js is a cross-platform, open-source JavaScript runtime environment that can run on windows, linux, Unix, macOS, and more.
Node.js runs on the V8 JavaScript Engine and executes JavaScript code outside a web browser.
- Cross platform means Node.js can run on windows, linux, macOS, and it can run on a lot of different type of operating system and a lot of places that's why it is known as cross-platform.
- Node.js is open source and it is maintain by Open JS Foundation (it's committee which maintain Node.js).
New feature on Node.js is added by the committee of Open JS Foundation.

- When JS as a language came up, it gained popularity because of its application on the **web-browser**.

- JavaScript has always been a language which dominates on **web-browser**.

- JS is the language which is responsible for all the **dynamic behavior on the website**.

- Earlier JS was used and managed over the web and it used to be dominating in the **web world**.

- But when Node.js came in, now we can run JS everywhere.

- So basically Node.js helps us to execute JS code outside the browser.

And not just limited to **Servers** we can run JS code on a lot of places that we can't even think of.

→ It's not limited to creating web servers, we can do a lot more things with **Node.js**.

- Node.js has an **event-driven architecture** and capable of **asynchronous I/O**.

- Today Node.js is popular because it has event driven architecture.

→ **Asynchronous I/O** is also sometimes known as **Non-Blocking I/O**.

- The first version of Node.js released in **2009**.

- Node.js was developed by the person **Ryan Dahl**.

- Every code we write in Node.js it's JS.

→ To run JS we always need a JS Engine, we can't run any JS code without a JS Engine. It means wherever there is JS there is JS Engine. Be it browser, laptop, server, smartwatch, light bulb, robots, machine learning, everywhere we write JS code, there is a JS Engine which executes it.

→ when **Ryan Dahl** was building Node.js, he wanted to execute some JS code.

- If we saw the definition it says that Node.js uses **Chrome's V8 Engine**. But when Ryan developed Node for first time he didn't use JS Engine. He started it with **Spider Monkey**.

→ **Spider Monkey** is a JS Engine that is found in Firefox.

→ Every Browser has their own JS Engine.

- Initially when Ryan Dahl was developing Node.js, he built it on the top of **Spider Monkey**, but in just 2 days of his development he had done with **Spider Monkey** and tried for **V8**.

→ He started using **V8** for building **Node.js** and he never looked back.

→ later on there were few versions of Node.js which were built on **Spider Monkey** but today the latest version of Node.js and the official Node.js is using **V8 engine** behind the scene.

→ **V8 Engine** powered, ignited by this **V8 engine**.

- Ryan was just working independently on this project but there was a company named Joyent and this company was also working on something similar to what Node.js was.
- The company was building an internal project where they were trying to use JS on the server and they were really fascinated by the idea of Ryan Dahl.
- So, Joyent company asked to Ryan Dahl to come and work on Node.js in the company and become the employee of Joyent and build this project and company will take care of funding this Node.js project.
- So, if Node.js exists there is a very big contribution of Joyent company.
- Gradually Node.js started becoming big and Joyent company started using Node.js.
- Initially the first few projects which were built using Node.js were built and used inside Joyent.
- Now, Node.js is not maintained by Joyent company.
- Earlier, Ryan gave Node.js name to web.js. Because Ryan said that "I want to create web servers using this technology."
- But later on when he realised the potential, he renamed it to Node.js and said that Node.js is a bigger thing, it is not just limited to building web servers. we can build anything out of it.

Q: What was the problem that led Ryan Dahl to create Node.js?

A: There was something known as **Apache HTTP Server**. It was used to create **HTTP Servers** and this is kind of like blocking (it was a blocking server).

• So, Ryan wanted to create a non-blocking servers.
So, Node.js is Non-Blocking I/O.

→ The advantage of non-blocking server is that it can handle multiple request with lesser number of threads.

→ In 2010, npm happened, npm is a package manager for node.

→ A developer who joined later he decided to create a package manager for node.js, so that everybody can contribute and build small-small packages.

→ So npm is a registry (a central place) where we can add a new package.

→ A package can be anything, suppose if we want to handle **time and date** in Node.js application, so there is a package for it. There is also a package to create **web servers**.

→ When npm was developed it was very significant achievement for Node.js.

→ When Node.js was built it was first built for **macOS & Linux**, in 2011 there was the **Windows** support that came up and that was lead by **Joyent + Microsoft**.

And then they created a windows support for Node.js making it much more accessible to a lot more developer community.

- In 2012, Ryan Dahl left the Node.js project. He remained as a employee in the company but he left the Node.js maintenance.
- The responsibility was given to Isaac, who was creator of 'npm'. He became the project lead of Node.js.
- When Ryan got off from the project the pace of development of Node.js, it started becoming very slow, now Chrome was releasing of V8 very fast but the development speed of Node.js is very slow. It was not able to catch up to the latest version.
- In 2014, a developer named Fedor he created a fork of Node.js, he named it as io.js.
- Joyent company was managing Node.js but node.js was open source, Joyent company was limiting the release cycle and trying to hold the release of it, the development process became slow.
- Node.js was like about to die. It was not getting more popularity. It was very slow, it was not actively managed.
- So, io.js was exactly the fork of Node.js.
- So, now there were two Node.js which were running in, there was Node.js which was being managed by Joyent and there was io.js which had came up.
- There were 3-4 developers in the team who started maintaining io.js.

- So this had become the confusing thing in 2014.
- In 2015 September, there was Node.js Foundation which was formed and this Node.js Foundation was a Committee which now said that, we would maintain the Node.js.
So, io.js and Node.js merged.
- Today we use the merged version of these two.
- There was two major committees one was known as JS Foundation and there was committee Node.js Foundation.
- So these two committees were merged and a new committee was formed which was named as Open JS Foundation and in 2019 this Open JS Foundation took this control of Node.js.

Follow, gitHub → [rajeshjha2000](#)