

Node.js

From Wikipedia, the free encyclopedia

Node.js is an open-source, cross-platform JavaScript runtime environment for developing a diverse variety of tools and applications. Although Node.js is not a JavaScript framework,^[3] many of its basic modules are written in JavaScript, and developers can write new modules in JavaScript. The runtime environment interprets JavaScript using Google's V8 JavaScript engine.

Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in Web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games).^[4]

The Node.js distributed development project, governed by the Node.js Foundation,^[5] is facilitated by the Linux Foundation's Collaborative Projects program.^[6]

Corporate users of Node.js software include GoDaddy,^[7] Groupon,^[8] IBM,^[9] LinkedIn,^{[10][11]} Microsoft,^{[12][13]} Netflix,^[14] PayPal,^{[15][16]} Rakuten, SAP,^[17] Voxer,^[18] Walmart,^[19] Yahoo!,^[20] and Cisco Systems.

Contents

- 1 History
- 2 Overview
 - 2.1 Platform architecture
 - 2.2 Industry support
- 3 Technical details
 - 3.1 Threading
 - 3.2 V8
 - 3.3 Package management
 - 3.4 Unified API
 - 3.5 Event loop
- 4 Alternatives
 - 4.1 JXcore
 - 4.2 Other languages
- 5 See also
- 6 References
- 7 Further reading
- 8 External links

Node.js



Original author(s)	Ryan Dahl
Developer(s)	Node.js Developers (https://github.com/ry/node/blob/master/AUTHORS), Joyent, contributors (https://github.com/nodejs/node/graphs/contributors)
Initial release	May 27, 2009 ^[1]
Stable release	7.0.0 & 6.9.1 (LTS) / October 25, 2016 ^[2]
Repository	github.com/nodejs/node (http://github.com/nodejs/node)
Development status	Active (complete release list (http://njsv.yaoo.net/))
Written in	C, C++, JavaScript
Operating system	OS X, Linux, Solaris, FreeBSD, OpenBSD, Microsoft Windows (older versions require Cygwin), webOS, NonStop OS
Type	Event-driven networking
License	MIT
Website	nodejs.org (https://nodejs.org)

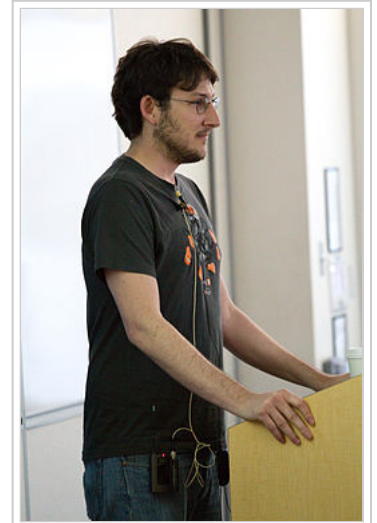
History

Node.js was originally written in 2009 by Ryan Dahl.^[21] The initial release supported only Linux. Its development and maintenance was led by Dahl and later sponsored by Joyent.^[22]

Dahl was inspired to create Node.js after seeing a file upload progress bar on Flickr. The browser did not know how much of the file had been uploaded and had to query the Web server. Dahl desired an easier way.^{[4][23]}

Dahl demonstrated the project at the inaugural European JSConf on November 8, 2009.^{[24][25][26]} Node.js combined Google's V8 JavaScript engine, an event loop and a low-level I/O API.^[27] The project received a standing ovation.^[28]

In January 2010, a package manager was introduced for the Node.js environment called *npm*.^[29] The package manager makes it easier for programmers to publish and share source code of Node.js libraries and is designed to simplify installation, updating and uninstallation of libraries.^[27]



Ryan Dahl, creator of Node.js

In June 2011, Microsoft and Joyent implemented a native Windows version of Node.js.^[30] The first Node.js build supporting Windows was released in July 2011.

In January 2012, Dahl stepped aside, promoting coworker and *npm* creator Isaac Schlueter to manage the project.^[31] In January 2014, Schlueter announced that Timothy J. Fontaine would lead the project.^[32]

In December 2014, Fedor Indutny started *io.js*, a fork of Node.js. Due to the internal conflict over Joyent's governance, *io.js* was created as an open governance alternative with a separate technical committee.^{[33][34]} Unlike Node.js,^[35] the authors planned to keep *io.js* up-to-date with the latest releases of the Google V8 JavaScript engine.^[36]

In February 2015, the intent to form a neutral Node.js Foundation was announced. By June 2015, the Node.js and *io.js* communities voted to work together under the Node.js Foundation.^[37]

In September 2015, Node.js v0.12 and *io.js* v3.3 were merged back together into Node v4.0.^[38] This brought V8 ES6 features into Node.js, and a long-term support release cycle.^[39] As of 2016, the *io.js* website recommends that developers switch back to Node.js and that no further releases are planned due to the merger.^[40]

Overview

Node.js allows the creation of Web servers and networking tools using JavaScript and a collection of "modules" that handle various core functionality.^{[24][27][41][42][43]} Modules are provided for file system I/O, networking (DNS, HTTP, TCP, TLS/SSL, or UDP), binary data (buffers), cryptography functions, data streams^[44] and other core functions.^{[27][42][45]} Node.js's modules use an API designed to reduce the complexity of writing server applications.^{[27][42]}

Node.js applications can run on Mac OS X, Microsoft Windows, NonStop,^[46] and Unix servers. They can alternatively be written with CoffeeScript^[47] (a JavaScript alternative), Dart or Microsoft TypeScript (strongly typed forms of JavaScript), or any other language that can compile to JavaScript.^[47]

Node.js is primarily used to build network programs such as Web servers, making it similar to PHP.^[41] The biggest difference between Node.js and PHP is that most functions in PHP block until completion (commands execute only after previous commands have completed), while functions in Node.js are designed to be non-blocking (commands execute in parallel, and use callbacks to signal completion or failure).^[41]

Platform architecture

Node.js brings event-driven programming to web servers, enabling development of fast web servers in JavaScript.^[27] Developers can create highly scalable servers without using threading, by using a simplified model of event-driven programming that uses callbacks to signal the completion of a task.^[27] Node.js was created because concurrency is difficult in many server-side programming languages, and often leads to poor performance.^[24] Node.js connects the ease of a scripting language (JavaScript) with the power of Unix network programming.^[27]

Node.js was built on the Google V8 JavaScript engine since it was open-source under the BSD license, extremely fast, and proficient with internet fundamentals like HTTP, DNS, TCP.^[24] Also, JavaScript was a well-known language, making Node.js immediately accessible to the entire web development community.^[24]

Industry support

Programmers have built thousands of open-source libraries for Node.js - most of them hosted on the npm website. The Node.js developer community has two main mailing lists and the IRC channel #node.js on freenode. There is an annual Node.js developer conference, called NodeConf.^[48]

The open-source community has developed server frameworks to accelerate the development of applications. Such frameworks include Connect, Express.js, Socket.IO, Koa.js, Hapi.js, Sails.js, Meteor, Derby, and many others.^{[27][49]}

Modern desktop IDEs provide editing and debugging features specifically for Node.js applications. Such IDEs include Atom, Brackets, JetBrains WebStorm,^{[50][51]} Microsoft Visual Studio (with Node.js Tools for Visual Studio,^[52] or TypeScript with Node definitions^{[53][54][55][56]}), NetBeans,^[57] Nodeclipse Enide Studio^[58] (Eclipse-based) and Visual Studio Code.^{[59][60]} Certain online web-based IDEs also support Node.js, such as Codeanywhere, Codenvy, Cloud9 IDE and Koding.

Technical details

Node.js is a Javascript runtime environment that processes incoming requests in a loop, called the event loop.

Threading

Node.js operates on a single thread, using non-blocking I/O calls, allowing it to support tens of thousands of concurrent connections without incurring the cost of thread context switching.^[61] The design of sharing a single thread between all the requests that uses the observer pattern is intended for building highly concurrent applications, where any function performing I/O must use a callback. In order to accommodate the single-threaded event loop, Node.js utilizes the libuv library that in turn uses a fixed-sized threadpool that is responsible for some of the non-blocking asynchronous I/O operations.^[4]

A downside of this single-threaded approach is that Node.js doesn't allow vertical scaling by increasing the number of CPU cores of the machine it is running on without using an additional module, such as cluster,^[62] StrongLoop Process Manager^[63] or pm2.^[64] However, developers can increase the default number of threads in the libuv threadpool; these threads are likely to be distributed across multiple cores by the server operating system.^[65]

Execution of parallel tasks in Node.js is handled by a thread pool. The main thread call functions post tasks to the shared task queue that threads in the thread pool pull and execute. Inherently non-blocking system functions like networking translates to kernel-side non-blocking sockets, while inherently blocking system functions like file I/O run in a blocking way on its own thread. When a thread in the thread pool completes a task, it informs the main thread of this that in turn wakes up and execute the registered callback. Since callbacks are handled in serial on the main thread, long lasting computations and other CPU-bound tasks will freeze the entire event-loop until completion.

V8

V8 is the JavaScript execution engine built for Google Chrome and open-sourced by Google in 2008. Written in C++, V8 compiles JavaScript source code to native machine code instead of interpreting it in real time.^[4]

Node.js uses libuv to handle asynchronous events. Libuv is an abstraction layer for network and file system functionality on both Windows and POSIX-based systems like Linux, Mac OS X, OSS on NonStop and Unix.

The core functionality of Node.js resides in a JavaScript library. The Node.js bindings, written in C++, connect these technologies to each other and to the operating system.

Package management

npm is the pre-installed package manager for the Node.js server platform. It is used to install Node.js programs from the npm registry, organizing the installation and management of third-party Node.js programs. npm is not to be confused with the CommonJS `require()` statement. It is not used to load code; instead, it is used to install code and manage code dependencies from the command line. The packages found in the npm registry can range from simple helper libraries like Underscore.js to task runners like Grunt.^[66]

Unified API

Node.js can be combined with a browser, a database supporting JSON data (such as Postgres,^[67] MongoDB or CouchDB) and JSON for a unified JavaScript development stack. With the adaptation of what were essentially server-side development patterns like MVC, MVP, MVVM, etc., Node.js allows the reuse of the same model and service interface between client-side and server-side.

Event loop

Node.js registers itself with the operating system so that it is notified when a connection is made, and the operating system will issue a callback. Within the Node.js runtime, each connection is a small heap allocation. Traditionally, relatively heavyweight OS processes or threads handled each connection. Node.js uses an event loop for scalability, instead of processes or threads.^[68] In contrast to other event-driven servers, Node.js's event loop does not need to be called explicitly. Instead callbacks are defined, and the server automatically enters the event loop at the end of the callback definition. Node.js exits the event loop when there are no further callbacks to be performed.

Alternatives

JXcore

JXcore is a fork of Node.js targeting mobile devices and IoTs. Its first beta was released in January 2014. It was open sourced^[69] on February 13, 2015 and made available through a GitHub repository.^[70] JXcore can use either of the Google V8 or Mozilla SpiderMonkey JavaScript engines. As a result, JXcore can run Node applications on iOS devices using SpiderMonkey. As of March 25, 2016, active development on JXcore has been halted.^[71]

Other languages

Similar open source event-driven server frameworks for other platforms include:

- EventMachine for Ruby
- libevent for C
- Perl Object Environment for Perl
- Twisted for Python
- Vert.x for Java, JavaScript, Groovy, Ruby, Python, Scala, Clojure and Ceylon

Node.js may utilize code written in other programming languages using:

- Edge.js allows Microsoft .NET applications to run Node.js scripts in-process, and allows Node.js servers to utilize .NET compiled code via async callbacks.^{[72][73][74]}
- Luvit implements the Node.js APIs for the language Lua^[75]
- Node-julia allows using Julia with Node.js/io.js
- The COBOL bridge for Node.js allows using COBOL with Node.js^[76]

See also

- List of JavaScript libraries
- MEAN (software bundle)
- Online JavaScript IDE
- Rhino (JavaScript engine)

References

1. "node-v0.x-archive on GitHub". Retrieved 2 August 2014.
2. "Node.js Changelog". Retrieved 25 October 2016.
3. Wen, Ben (2013-12-12). "6 things you should know about Node.js". JAVAWORLD. Archived from the original on 2013-12-12. Retrieved 2016-01-22.
4. Laurent Orsini (2013-11-07). "What You Need To Know About Node.js". readwrite. Archived from the original on 2013-11-07. Retrieved 2016-01-22.
5. "Node.js Foundation - Node.js". Retrieved 4 July 2015.

6. "Linux Foundation Collaborative Projects". Retrieved 4 July 2015.
7. Why GoDaddy's Nodejitsu deal is great for Node.js (<http://venturebeat.com/2015/02/10/why-godaddys-nodejitsu-deal-is-great-for-node-js/>), VentureBeat, February 10, 2015
8. Geitgey, Adam (30 October 2013). "I-Tier: Dismantling the Monoliths". Groupon. Retrieved 30 April 2014.
9. "IBM Bluemix". Retrieved 4 July 2015.
10. "You'll never believe how LinkedIn built its new iPad app". VentureBeat. May 2, 2012. Retrieved May 10, 2012.
11. "Blazing fast node.js: 10 performance tips from LinkedIn Mobile". Retrieved 7 April 2015.
12. Baxter-Reynolds, Matthew (November 9, 2011). "Here's why you should be happy that Microsoft is embracing Node.js". London: The Guardian. Retrieved May 10, 2012.
13. "WebMatrix - Front End Web Developers take note (ASP.NET, PHP, node.js and more)". Retrieved 2 August 2014.
14. Node.js in Flames (<http://techblog.netflix.com/2014/11/nodejs-in-flames.html>) November 19, 2014
15. "Clash of the Titans: Releasing the Kraken, NodeJS @paypal". fluentconf.com. May 28, 2013. Retrieved September 11, 2013.
16. "All such companies and their products in which Node.js is used". Retrieved 2 August 2014.
17. "SAP AppBuilder". SAP. March 10, 2014. Retrieved March 10, 2014.
18. The Node Ahead: JavaScript leaps from browser into future (http://www.theregister.co.uk/2011/03/01/the_rise_and_rise_of_node_dot_js/), The Register, March 1, 2011
19. "Why Walmart is using Node.js". VentureBeat. January 24, 2012. Retrieved May 10, 2012.
20. "Yahoo! Announces Cocktails Shaken, Not Stirred". Retrieved 7 April 2015.
21. "About Node.js, and why you should add Node.js to your skill set?". *Training.com*. Training.com. Retrieved 23 October 2016.
22. Ryan Dahl (2010-11-09). "Joyent and Node". Google Groups. Retrieved 2015-02-05.
23. Harris, Amber (April 1, 2012). "The Birth of Node: Where Did it Come From? Creator Ryan Dahl Shares the History". Devops Angle. Retrieved 26 October 2013.
24. Sams Teach Yourself Node.js in 24 Hours (<https://books.google.com/books?id=KGt-FxUEj48C&pg=PT24&dq=nodejs&hl=en&sa=X#v=onepage&q=nodejs&f=false>), Sams Publishing, 05-Sep-2012
25. "Ryan Dahl at JSConf EU 2009".
26. "Ryan Dahl at JSConf EU 2009 Video".
27. Professional Node.js: Building JavaScript Based Scalable Software (<https://books.google.com/books?id=ZH6bpbclrvYC&printsec=frontcover&dq=nodejs&hl=en&sa=X#v=onepage&q=nodejs&f=false>), John Wiley & Sons, 01-Oct-2012
28. "Video: Node.js by Ryan Dahl".
29. "Earliest releases of npm". *GitHub*. Retrieved 27 July 2016.
30. "Porting Node to Windows With Microsoft's Help". Retrieved 17 April 2016.
31. Dahl, Ryan. "New gatekeeper". Retrieved 26 October 2013.
32. Schlueter, Isaac (January 15, 2014). "The Next Phase of Node.js". Retrieved 21 January 2014.
33. Krill, Paul (Dec 4, 2014). "Why io.js Decided to Fork Node.js". JavaWorld. Retrieved Dec 15, 2014.
34. Q&A: Why io.js decided to fork Node.js (<http://www.infoworld.com/article/2855057/application-development/why-iojs-decided-to-fork-nodejs.html>), Infoworld Tech Watch
35. Ben Noordhuis (Nov 12, 2014). "Issue 3692: function suddenly becomes undefined". V8 JavaScript Engine Issues. Retrieved 2 February 2015.
36. Mikeal, Rogers (January 28, 2015). "State of io.js". Retrieved 2 February 2015.
37. "Node.js Foundation Advances Community Collaboration, Announces New Members and Ratified Technical Governance". Retrieved 4 July 2015.
38. "Node.js Foundation Combines Node.js and io.js Into Single Codebase in New Release". Retrieved 28 Jan 2016.
39. "io.js and Node.js merge". Retrieved 27 June 2015.
40. Io.js, JavaScript I/O (<https://iojs.org/en/>), "*io.js has merged with the Node.js project again. There won't be any further io.js releases. All of the features in io.js are available in Node.js v4 and above.*"
41. Node.js for PHP Developers (<https://books.google.com/books?id=KZQQAlhTOBgC&printsec=frontcover&dq=nodejs&hl=en&sa=X#v=onepage&q=nodejs&f=false>), O'Reilly Media, Inc., 2013
42. Node.js Essentials (<https://books.google.com/books?id=z117BAAQBAJ&dq=nodejs&hl=en&sa=X>), Packt Publishing, 09-Sep-2014
43. Smashing Node.js: JavaScript Everywhere (https://books.google.com/books?id=G1y_5kpmatUC&printsec=frontcover&dq=nodejs&hl=en&sa=X#v=onepage&q=nodejs&f=false), John Wiley & Sons, 14-Aug-2012
44. Sahil Chitkara. "Streams in node.js : Readable and Writable". *TO THE NEW BLOG*.
45. Modules (<https://nodejs.org/api/>), Nodejs Website
46. "bomBora - Node.js for NonStop". Infracore. Retrieved 14 August 2015.
47. "CoffeeScript on Node.js". O'Reilly Media, Inc. April 15, 2013. Retrieved May 17, 2015.

48. Finley, Klint (April 7, 2011). "NodeConf Schedule Announced". ReadWriteHack. Retrieved 2 August 2014.
49. Express.js Guide: The Comprehensive Book on Express.js (<https://books.google.com/books?id=5eGRAWAAQBAJ&printsec=frontcover&dq=nodejs&hl=en&sa=X#v=onepage&q=nodejs&f=false>), Azat Mardan, 28-May-2014
50. Node.js (<https://www.jetbrains.com/webstorm/help/node-js.html>), WebStorm supports integration with the Node.js runtime environment, WebStorm Help
51. Running and Debugging Node.js (<https://www.jetbrains.com/webstorm/help/running-and-debugging-node-js.html>), WebStorm Help
52. "Node.js Tools for Visual Studio". Retrieved 1 Feb 2016.
53. soywiz/typescript-node-definitions TypeScript's typings for some popular node.js modules (<https://github.com/soywiz/typescript-node-definitions>), GitHub
54. DefinitelyTyped (<https://github.com/DefinitelyTyped/DefinitelyTyped>), GitHub
55. The repository for high quality TypeScript type definitions (<http://definitelytyped.org/>) Archived (<https://web.archive.org/web/20160203083254/http://definitelytyped.org/>) February 3, 2016, at the Wayback Machine.
56. ImageBoard, A Node.js + Express + MongoDB application built using TypeScript on the server (<http://www.typescriptlang.org/Samples#ImageBoard>), TypeScript Samples
57. Krill, Paul (2015-11-10). "Node.js takes center stage in NetBeans 8.1". InfoWorld.
58. Nodeclipse, Enide -- Node.JS development in Eclipse (<http://www.nodeclipse.org/>), Nodeclipse Website
59. Hello Visual Studio Code (with NodeJS) (<https://channel9.msdn.com/Blogs/cloud-with-a-silver-lining/hello-visual-studio-code-nodejs>), Channel 9, Microsoft
60. Node.js Applications with VS Code (<https://code.visualstudio.com/Docs/runtimes/nodejs>), Visual Studio Code
61. "Node.js w/1M concurrent connections!". *caustik's blog*.
62. "Cluster Node.js v5.5.0 Manual & Documentation".
63. "StrongLoop Process Manager".
64. "GitHub - Unitech/pm2: Production process manager for Node.js applications with a built-in load balancer". *GitHub*.
65. Aleksander Kasiuk (22 April 2015). "On problems with threads in node.js - Future Processing".
66. "Grunt: The JavaScript Task Runner".
67. "PostgreSQL: Documentation: 9.4: JSON Types". *www.postgresql.org*. Retrieved 2016-10-20.
68. About Node.js (<https://nodejs.org/about/>), Node.js Website
69. Serdar Yegulalp (20 February 2015). "Node.js fork JXcore goes open source, aims for mobile developers". *InfoWorld*. Retrieved 4 July 2015.
70. "jxcore/jxcore". *GitHub*.
71. "Nubisa halting active development on JXcore platform".
72. Tomasz Janczuk. "Edge.js".
73. Using Edge.js to combine Node.js and .NET (<http://www.dotnetcurry.com/nodejs/1195/using-edgejs-combine-dotnet-nodejs>), .NET Curry
74. Edge.js bridges the gap between Node.js and .NET (<http://www.techrepublic.com/article/edge-js-bridges-the-gap-between-node-js-and-net/>), TechRepublic, Tony Patton, July 1, 2014
75. "Luvit.io".
76. "cobol".

Further reading

- Hughes-Croucher, Tom; Wilson, Mike (April 2012), *Up and Running with Node.js* (First ed.), O'Reilly Media, p. 204, ISBN 978-1-4493-9858-3
- Ornbo, George (September 2012), *Sams Teach Yourself Node.js in 24 Hours* (First ed.), SAMS Publishing, p. 440, ISBN 978-0-672-33595-2
- Teixeira, Pedro (October 2012), *Professional Node.js* (First ed.), John Wiley & Sons, p. 408, ISBN 978-1-118-22754-1
- Randal L. Schwartz and Aaron Newcomb (9 January 2013). "Episode 237: Node.js". *twit.tv/show/floss-weekly* (Podcast). TWiT.tv. Event occurs at 1:08:13. Retrieved 9 January 2013.
- Kurniawan, Agus (July 2012), *Nodejs Programming By Example* (First ed.), PE Press, p. 67
- Gackenheim, Cory (October 2013), *Node.js Recipes: A Problem-Solution Approach* (First ed.), Apress, p. 376, ISBN 978-1-4302-6058-5
- Ribeiro Pereira, Caio (February 2016), *Building APIs with Node.js* (First ed.), Leanpub, p. 152

External links

- Official website (<https://nodejs.org>)
- Node.js (<https://github.com/nodejs/node>) on GitHub



Wikimedia Commons has
media related to ***Node.js***.

Retrieved from "<https://en.wikipedia.org/w/index.php?title=Node.js&oldid=749039272>"

Categories: Free software programmed in C++ | Free software programmed in JavaScript | JavaScript libraries
| Joyent | Linux Foundation projects | Software using the MIT license

- This page was last modified on 12 November 2016, at 00:12.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply.
By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.