



Node.js What's Next?

About Michael Dawson



Node.js lead for Red Hat and IBM

Active Node.js community member

Node.js Collaborator, Node.js Technical Steering Committee,

Active in a number of Working group(s)

Active OpenJS Foundation member

Voting Cross Project Council Member

Community Director 2020-2022



Twitter: @mhdawson1

GitHub: @mhdawson

Linkedin: <https://www.linkedin.com/in/michael-dawson-6051282>

Overview

- Who decides what's next ?
- Following what's next
- Next-10
- Features - what's new/interesting?
- Supply Chain Security

Who Decides what's next ?

TSC

Companies

An individual contributor

Who Decides what's next ?

■

All collaborators



Who Decides what's next ?

- All collaborators
 - Together
 - Real time
 - No pre-defined roadmap
- Driven by
 - Doer's
 - Reviewers/Approvers
 - Supported by Project's documented Priorities



Following What's Next - from the Ground up

- GitHub Notifications
- Working Groups and Teams - <https://nodejs.org/calendar>
- Strategic Initiatives - <https://github.com/nodejs/node/blob/master/doc/guides/strategic-initiatives.md>
- Next-10 - <https://github.com/nodejs/next-10>
- Releases - <https://github.com/nodejs/release>

Following What's Next - from the Ground up

The screenshot shows a GitHub notification interface. At the top, there's a navigation bar with tabs: 'Inbox' (highlighted with a red circle and showing 384 notifications), 'All', and 'Unread'. To the right of these tabs are a 'Filter notifications' button and a 'Group by: Date' dropdown menu. Below the navigation bar, there's a light blue banner with the text: 'Overwhelmed by notifications? We've found some repositories that may be causing notifications you don't need. [Update watching settings](#)'. Below this banner is a table of notifications. The table has columns for selection, repository, notification details, count, type, avatars, and time. The left sidebar contains a 'Filters' section with 'Assigned' (1), 'Participating' (20), 'Mentioned', 'Team mentioned' (13), and 'Review requested'. Below the filters is a 'Repositories' section listing: 'nodejs/node' (187), 'nodejs/node-gyp' (21), 'nodejs/help' (19), 'libuv/libuv' (16), 'nodejs/nodejs.org' (14), and 'nodeshift/openshift-...' (12).

Filters	
Assigned	1
Participating	20
Mentioned	
Team mentioned	13
Review requested	

Repositories	
nodejs/node	187
nodejs/node-gyp	21
nodejs/help	19
libuv/libuv	16
nodejs/nodejs.org	14
nodeshift/openshift-...	12

<input type="checkbox"/> Select all						
<input checked="" type="checkbox"/>	nodejs/node #43059	deps: update undici to 5.2.0	+10	subscribed		3 minutes ago
<input type="checkbox"/>	redhat-developer/app-services-tools #21	docs(README): update instructions for manual run	+3	subscribed		3 minutes ago
<input type="checkbox"/>	nodejs/node #43037	doc: add strategic initiative for shadow realm	+1	team mention		4 minutes ago
<input type="checkbox"/>	nodejs/node #43045	doc: remove git:// protocol, adjust nits in onboarding.md	+5	team mention		5 minutes ago
<input checked="" type="checkbox"/>	nodejs/node #43061	deps: upgrade npm to 8.10.0	+5	subscribed		6 minutes ago
<input type="checkbox"/>	nodejs/build #2888	Access to arm32 build machines for @ShogunPanda	+3	team mention		13 minutes ago
<input type="checkbox"/>	nodejs/admin #687	Youtube and Zoom access for @JakobJingleheimer (me)		team mention		24 minutes ago
<input checked="" type="checkbox"/>	openjs-foundation/cross-project-council #884	meta: add @achrinza as a Regular Member	+4	subscribed		1 hour ago
<input type="checkbox"/>	nodejs/node #43062	gc performance entries report incorrect startTime		subscribed		1 hour ago

[s/strategic-](#)

Following What's Next - from the Ground up

- GitHub Notifications
- Working Groups and Teams - <https://nodejs.org/calendar>
- Strategic Initiatives - <https://github.com/nodejs/node/blob/master/doc/guides/strategic-initiatives.md>
- Next-10 - <https://github.com/nodejs/next-10>
- Releases - <https://github.com/nodejs/release>

Following What's Next - from the Ground up

- Give
- Work
- Stand
- New
- Re

Node.js Project Calendar

calendar.google.com/calendar/u/0/embed?src=nodejs.org_nr77ama8p7d7f9ajrpnu506c98@group.calendar.google.com

Google doc acti... Weekly status... Enterprise Nod... Inbox (5) - mida... working Red Hat - Calen... Heads up of No... Profile - Red Ha... Service Catalog...

Node.js Project Calendar

Today June 2022

Print Week Month Agenda

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31	Jun 1	2	3	4
		3:30pm Diagnostics WG Meeting 10pm Build WG Meeting	12pm Node.js TSC Meeting 6pm Node.js Social Team 6pm Node.js Website Redesign Meetin	2pm Node.js Release Working Group M	3pm Node.js N-API team weekly meeti 5pm Node.js Mentorship Program team 6pm Node.js i18n WG Meeting	
5	6	7	8	9	10	11
	4pm Node.js Outreach Meeting 6pm Loaders Team Meeting	2pm Node.js Next 10 years 3pm Node.js TSC Meeting 6pm Node.js Website Redesign Meetin	2pm Security-WG meeting 5pm Package Maintenance Team meet	3pm Node.js N-API team weekly meeti 5pm Node.js Mentorship Program team 6pm Node.js i18n WG Meeting 8pm Node.js Tooling Group team meeti		
12	13	14	15	16	17	18
	10:30pm Diagnostics WG Meeting	6pm Node.js Social Team 6pm Node.js Website Redesign Meetin 9pm Node.js TSC Meeting			3pm Node.js N-API team weekly meeti 5pm Node.js Mentorship Program team 6pm Node.js i18n WG Meeting	
19	20	21	22	23	24	25
	12pm Build WG Meeting - early 4pm Node.js Outreach Meeting 6pm Loaders Team Meeting 7pm Package Maintenance Team meet	12pm Node.js TSC Meeting 2pm Node.js Next 10 years 6pm Node.js Website Redesign Meetin	2pm Security-WG meeting	3pm Node.js N-API team weekly meeti 5pm Node.js Mentorship Program team 6pm Node.js i18n WG Meeting 8pm Node.js Tooling Group team meeti		
26	27	28	29	30	Jul 1	2
	3:30pm Diagnostics WG Meeting	3pm Node.js TSC Meeting 6pm Node.js Social Team 6pm Node.js Website Redesign Meetin	2pm Node.js Release Working Group M	3pm Node.js N-API team weekly meeti 5pm Node.js Mentorship Program team 6pm Node.js i18n WG Meeting		

Events shown in time zone: Greenwich Mean Time

Google Calendar

[guides/strategic-](#)

Following What's Next - from the Ground up

- GitHub Notifications
- Working Groups and Teams - <https://nodejs.org/calendar>
- Strategic Initiatives - <https://github.com/nodejs/node/blob/master/doc/guides/strategic-initiatives.md>
- Next-10 - <https://github.com/nodejs/next-10>
- Releases - <https://github.com/nodejs/release>

Following What's Next - from the Ground up

42 lines (34 sloc) | 2.81 KB

<>

📄

Raw

Blame

📋

✎

Strategic initiatives

The Node.js project has several strategic initiatives underway. A review of the current initiatives is a standing item on the Technical Steering Committee agenda to ensure they are active and have the support they need.

[.ing/strat](https://nodejs.org/strat)

Current initiatives

Initiative	Champion	Links
Core Promise APIs	Antoine du Hamel	nodejs/TSC#1094
QUIC / HTTP3	James M Snell	https://github.com/nodejs/quic
Startup performance	Joyee Cheung	#35711
V8 Currency	Michaël Zasso	
Next-10	Michael Dawson	https://github.com/nodejs/next-10

► List of completed initiatives

Following What's Next - from the Ground up

- GitHub Notifications
- Working Groups and Teams - <https://nodejs.org/calendar>
- Strategic Initiatives - <https://github.com/nodejs/node/blob/master/doc/guides/strategic-initiatives.md>
- Next-10 - <https://github.com/nodejs/next-10>
- Releases - <https://github.com/nodejs/release>

Following W

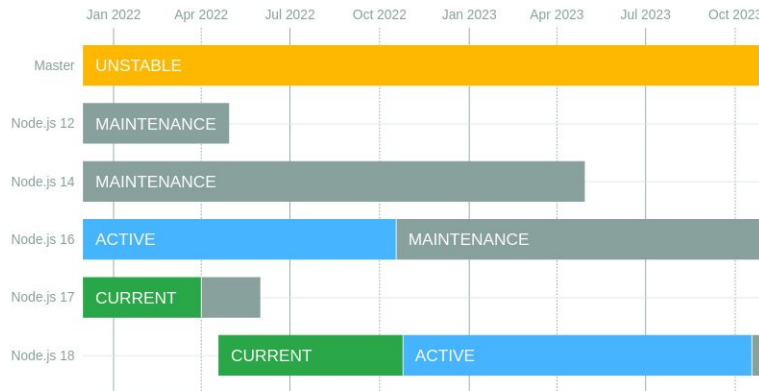
- GitHub N
- Working
- Strategic
- Next-10
- Releases

README.md

Release schedule

Release	Status	Codename	Initial Release	Active LTS Start	Maintenance Start	End-of-life
14.x	Maintenance	Fermium	2020-04-21	2020-10-27	2021-10-19	2023-04-30
16.x	Active LTS	Gallium	2021-04-20	2021-10-26	2022-10-18	2024-04-30
17.x	Maintenance		2021-10-19	-	2022-04-01	2022-06-01
18.x	Current		2022-04-19	2022-10-25	2023-10-18	2025-04-30
19.x	Pending		2022-10-18	-	2023-04-01	2023-06-01
20.x	Pending		2023-04-18	2023-10-24	2024-10-22	2026-04-30

Dates are subject to change.



- <https://github.com/nodejs/release>

d up

nodejs.org/calendar

[b/master/doc/guides/strategic-](https://nodejs.org/master/doc/guides/strategic-)

Following What's Next from the Ground up

411 lines (351 sloc) | 53.4 KB

<> [Icons] Raw Blame [Icons]

2022-04-19, Version 18.0.0 (Current), @BethGriggs

Node.js 18 is here! Highlights include the update of the V8 JavaScript engine to 10.1, global fetch enabled by default, and a core test runner module.

Initially, Node.js 18 will replace Node.js 17 as our 'Current' release line. As per the release schedule, Node.js 18 will be the 'Current' release for the next 6 months and then promoted to Long-term Support (LTS) in October 2022. Once promoted to long-term support the release will be designated the codename 'Hydrogen'. Node.js 18 will be supported until April 2025.

Notable Changes

Deprecations and Removals

- (SEMVER-MAJOR) **fs**: runtime deprecate string coercion in `fs.write`, `fs.writeFileSync` (Livia Medeiros) [#42607](#)
- (SEMVER-MAJOR) **dns**: remove `dns.lookup` and `dnsPromises.lookup` options type coercion (Antoine du Hamel) [#41431](#)
- (SEMVER-MAJOR) **process**: runtime deprecate `multipleResolves` (Benjamin Gruenbaum) [#41896](#)
- (SEMVER-MAJOR) **stream**: remove thenable support (Robert Nagy) [#40773](#)
- (SEMVER-MAJOR) **tls**: move `tls.parseCertString` to end-of-life (Tobias Nießen) [#41479](#)

fetch (experimental)

An experimental fetch API is available on the global scope by default. The implementation is based upon [undici](#), an HTTP/1.1 client written for Node.js by contributors to the project.

```
const res = await fetch('https://nodejs.org/api/documentation.json');
if (res.ok) {
  const data = await res.json();
  console.log(data);
}
```

Through this addition, the following globals are made available: `fetch`, `FormData`, `Headers`, `Request`, `Response`.

Disable this API with the `--no-experimental-fetch` command-line flag.

[des/strategic-](#)

Strategic directions

How do we
ensure
continued
success?

for the next

What do
stakeholders
need ?

10 years of Node.js

What's crucial?

How will
JavaScript
Evolve?

Who are the key
Node.js
Stakeholders?

Next 10 - Foundation

- Technical Values and Priorities - <https://github.com/nodejs/node/blob/master/doc/guides/technical-values.md>
- Constituencies - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCIES.md>
- Constituencies Needs - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCY-NEEDS.md>



Trott doc: use "Long Term Support" in technical values doc ...

Latest commit 0ba7da6 on May 29



4 contributors



74 lines (63 sloc) | 2.77 KB

Raw

Blame



Technical values and priorities

The project uses these technical values to establish priorities and guide collaboration.

These are the shared values as of this writing and will evolve. We hope they are useful to people new to the project in order to better understand which contributions will be aligned with the current direction and as thinking points when trading off between conflicting goals.

The factors influencing every discussion/decision are different and priority 1 does not always trump priority 2 and so on.

Values and priority level

- Priority 1 - Developer experience
- Priority 2 - Stability
- Priority 3 - Operational qualities
- Priority 4 - Node.js maintainer experience
- Priority 5 - Up to date technology and APIs

Value descriptions

1 - Developer experience

We value ensuring that developers are productive and enjoy developing with Node.js. Some key elements of this include:

</doc/gu></-NEEDS.md>

Next 10 - Foundation

- Technical Values and Priorities - <https://github.com/nodejs/node/blob/master/doc/guides/technical-values.md>
- Constituencies - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCIES.md>
- Constituencies Needs - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCY-NEEDS.md>

Next 10

- Tech
- Cons
- Cons

List of constituencies

- Direct end users
 - Users who run tools themselves (install Node.js, run tool, deal with errors including writing/using scripts)
- Application operators
 - Users who interact with existing, running applications (regardless if they wrote the application or not)
 - Service and infrastructure providers
- Application Developers
 - Front-end tool consumers
 - Back-end server authors
 - Hobby developers
 - Professional developers
 - Tool authors
- Library/package authors
 - Users who write libraries and packages to be included on other applications
- Node.js contributors
 - Developers working directly on [nodejs/node](#)
 - Individuals participating in Working Groups and teams
- Organizations with investments in Node.js (eg: Enterprises, Government bodies, startups, non-profits, standards groups like TC39)
 - C level executives (CTO, CEO, etc.)
 - Planning/program offices
 - Managers
- Education
 - Teachers
 - Students
 - Organizations who help people learn Node.js (colleges, University, boot camps, online learning resources, etc.)
 - Programs to help people demonstrate capability (certification programs, etc.)
- Security Practitioners
 - People involved in the life-cycle of handling security issues including:
 - Penetration testing
 - Incident response
 - CVE scanning/remediation,
 - Legal, Public relations

[>/master/doc/gu](#)

[TUENCY-NEEDS.md](#)

Next 10 - Foundation

- Technical Values and Priorities - <https://github.com/nodejs/node/blob/master/doc/guides/technical-values.md>
- Constituencies - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCIES.md>
- Constituencies Needs - <https://github.com/nodejs/next-10/blob/main/CONSTITUENCY-NEEDS.md>

Needs of the Node.js Constituencies

Need	Users	Ops	AppDev	LibDev	Orgs	NodeMaint
Good understanding of the direction of the project	X	X	X	X	X	X
Ability to affect the direction of the project	X	X	X	X	X	X
Consumable APIs and docs	X		X	X		X
Predictable and stable releases	X	X	X	X	X	
Innovation at a consumable pace	X		X	X	X	X
Easy Installation	X					
Easy issue reporting, resolution and collaboration	X	X	X	X	X	
Broad deployment platform support	X	X			X	
Broad desktop platform support	X		X	X		X
Consistent and intuitive error handling	X	X	X	X		

[/nodejs/node/blob/master/doc/gu](#)
[jes.md](#)

[;ITUENCIES.md](#)

[ob/main/CONSTITUENCY-NEEDS.md](#)

•
•
•

Ability to embed and bundle the Node.js runtime			X			
A well maintained and secure standard library			X	X	X	
Assets that show Node.js is a good choice	X		X	X	X	

Good understanding of the direction of the project

- An understanding of the work happening in the Node.js project (what features do I have to look forward to, what's the project direction). Changes between releases. Info on how new changes affect end users/developers

Technical Priorities

<https://github.com/nodejs/node/blob/master/doc/contributing/technical-priorities.md>

- Modern HTTP
- Suitable types for end-users
- Documentation
- Web Assembly
- ESM
- Observability
- Permissions/policies/security model
- Better multithreaded support
- Single Executable Applications
- Support for features from the latest ECMAScript spec

121 lines (98 sloc) 6.08 KB

Technical Priorities

This list represents the current view of key technical priorities recognized by the project as important to ensure the ongoing success of Node.js. It is based on an understanding of the Node.js [constituencies](#) and their [needs](#).

The initial version was created based on the work of the [Next-10 team](#) and the [mini-summit](#) on August 5th 2021.

They will be updated on a regular basis and will be reviewed by the next-10 team and the TSC on a 6 month basis.

Modern HTTP

Base HTTP support is a key component of modern cloud native applications and built-in support was part of what made Node.js a success in the first 10 years. The current implementation is hard to support and a common source of vulnerabilities. It is important that we work towards an implementation which is easier to support and makes it easier to integrate the new http versions (HTTP3, quick) and to support efficient implementations of different versions concurrently.

Suitable types for end-users

Using typings with JavaScript can allow a richer experience when using Visual code (or any other IDEs) environments, more complete documentation of APIs and the ability to identify and resolve errors earlier in the development process. These benefits are important to a large number of Node.js developers (maybe 50%). Further typing support may be important to enterprises that are considering expanding their preferred platforms to include Node.js. It is, therefore, important that the Node.js project work to ensure there are good typings available for the public Node.js APIs.

Documentation

The current documentation is great for experienced developers or people who are aware of what they are looking for. On the other hand, for beginners this documentation can be quite hard to read and finding the desired information is difficult. It is important that we have documentation that is suitable for beginners in order to continue the rapid growth in use. This documentation should include more concrete examples and a learning path for newcomers.

Web Assembly

The use of Web Assembly has been growing over the last few years. In order to ensure Node.js continues to be part of solutions where a subset of the solution needs the performance that Web Assembly can deliver it is important that Node.js provide good support for running Web Assembly components along with the JavaScript that makes up the rest of the solution. This includes implementations of "host" APIs like WASI.

ESM

The CommonJS module system was one of the key components that led to the success of Node.js in its first 10 years. ESM is the standard

Mini Summits

- Nov 2021 - Suitable types/Single executable applications
- Jan 2022 - Modern HTTP/Documentation
- Apr 2022 - WASM/Security model and permissions
- June 2022 (collab summit)
ESM/Observability

Contributing Docs

- `maintaining-types-for-nodejs.md`
- `maintaining-http.md`
- `maintaining-web-assembly.md`
- `security-model-strategy.md`

<https://github.com/nodejs/node/tree/master/doc/contributing>

Next 10 - How can I help

- Join the effort
 - We need your ideas/input
- Participate in the team's survey's
- Support key initiatives

Let's make sure the next 10 years are just as successful as the first

Features - what's new/interesting ?

- Major Releases
 - Boring.....
- SemVer Minor is where its at!
 - features (but not all) flow into Current/LTS lines
- Majors still a good time to party
 - Talk about recent new features
 - Promotion to LTS

Experimental Fetch (18.x)

<https://fetch.spec.whatwg.org/>

- Lots of good feedback so far
- Too close to web standard/not close enough

```
const resp = await fetch(url)
resp.text().then(text => console.log(text));
```

Experimental Test Runner (18.x)

```
const test = require('node:test');  
const assert = require('node:assert');
```

```
test('test1', (test) => {  
  assert.strictEqual('yes', 'no');  
});
```

<https://nodejs.org/api/test.html>

Tap Output



```
[midawson@drx-hemera pres]$ node --test test1.js  
TAP version 13  
not ok 1 - /home/midawson/pres/test1.js  
...  
duration_ms: 0.069108223  
failureType: 'subtestsFailed'  
exitCode: 1  
stdout: |-  
  TAP version 13  
  not ok 1 - test1  
...  
duration_ms: 0.002767144  
failureType: 'testCodeFailure'  
error: |-  
  Expected values to be strictly equal:  
  
  'yes' !== 'no'  
  
code: 'ERR_ASSERTION'  
stack: |-  
  TestContext.<anonymous> (/home/midawson/pres/test1.js:5:10)  
  Test.runInAsyncScope (node:async_hooks:202:9)  
  Test.run (node:internal/test_runner/test:340:20)  
  Test.start (node:internal/test_runner/test:292:17)  
  Test.test (node:internal/test_runner/harness:126:18)  
  Object.<anonymous> (/home/midawson/pres/test1.js:4:1)  
  Module._compile (node:internal/modules/cjs/loader:1105:14)  
  Module._extensions.js (node:internal/modules/cjs/loader:1159:10)  
  Module.load (node:internal/modules/cjs/loader:981:32)  
  Module._load (node:internal/modules/cjs/loader:827:12)  
  ...  
  1..1  
  # tests 1  
  # pass 0  
  # fail 1  
  # skipped 0  
  # todo 0  
  # duration_ms 0.055707066  
  
stderr: |-  
  (node:3039014) ExperimentalWarning: The test runner is an experimental feature. This feature could change at any time  
  (Use 'node --trace-warnings ...' to show where the warning was created)  
  
error: 'test failed'  
code: 'ERR_TEST_FAILURE'  
...  
  1..1  
  # tests 1  
  # pass 0  
  # fail 1  
  # skipped 0  
  # todo 0  
  # duration_ms 0.133823902
```

Technical Value - Developer experience

ECMAScript Modules (v14.17.0)

CommonJS

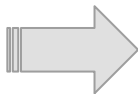
```
module.exports = function() {  
  console.log('Hello');  
}  
  
const test = require('./helloTest.js');  
test();
```

ES6 Modules

```
export function test() {  
  console.log('Hello ES6');  
}  
  
import { test } from './helloTest.mjs';  
test();
```

package.json

```
{  
  "name": "test",  
  "type": "module"  
}
```



```
import { test } from './helloTest2.js';  
test();
```

Technical Priority - ESM

ECMAScript Modules

- Ongoing progress on ESM implementation
 - JSON Import Assertions in line with the TC39 [proposal](#)
 - A command line flag is no longer needed to use [esm json](#) modules (although it is still experimental)
 - Experimental support for [HTTPS and HTTP imports](#) under the `--experimental-network-imports` flag.
- Work on larger initiatives also underway
 - [loaders](#)
 - [diagnostic channel](#)

V8 - 10.1 (18.x)

- Keeps Node.js up to date with the latest JavaScript features

New in 10.1

- The `findLast()` and `findLastIndex()` array methods.
- Improvements to the `Intl.Locale` API.
- The `Intl.supportedValuesOf` function.
- Improved performance of `class fields` and `private class methods` (the initialization of them is now as fast as ordinary property stores).

Technical Priority - Support for features from the latest ECMAScript spec

OpenSSL 3 (17.x)

- Biggest impact
 - Users of smaller key sizes
 - Legacy algs no longer available by default
 - Native modules - Deprecation warnings

OpenSSL Migration Guide -

https://www.openssl.org/docs/man3.0/man7/migration_guide.html



<https://www.openssl.org/blog/blog/2021/09/07/OpenSSL3.Final/>

Stream Improvements (since 17.0)

New Experimental methods on [Readable streams](#) such as:

- [reduce](#)
- [toArray](#)
- [forEach](#)
- [filter](#)
- [isReadable](#)
- [some](#)
- [every](#)

Default DNS resolution

- Node.js no longer prefers IPv4 over IPV6
- `--dns-result-order=ipv4first`

Supply Chain Security

- Node.js -> top criticality score
- Funding from Open Source Security Foundation
 - <https://openssf.org/blog/2022/04/18/openssf-selects-node-js-as-initial-project-to-improve-supply-chain-security/>
 - vulnerability triage/fixing/releases
 - many thanks to existing triagers, fixers and security stewards - <https://github.com/nodejs/node#security-release-stewards>
 - documenting processes as template for ecosystem
 - assessing dependencies in context of supply chain
- Ecosystem security wg - <https://github.com/nodejs/security-wg>
- OpenJS Security collaboration space

Thank You

Thank You

Questions?

Copyright and Trademarks

© Red Hat, IBM. All Rights Reserved

Red Hat, the Red Hat logos are trademarks or registered trademarks of Red Hat

IBM, the IBM logo, ibm.com are trademarks or registered trademarks of International Business Machines Corp.,

registered in many jurisdictions worldwide.

A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml

Node.js is an official trademark of Joyent. IBM SDK for Node.js is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

Java, JavaScript and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

npm is a trademark of npm, Inc.

Other trademarks or logos are owned by their respective owners.