

The road four node

Evan Lucas

<https://github.com/evanlucas>

<https://twitter.com/evanh Lucas>

About

- Node.js Engineer at [Help.com](https://help.com)
- Node.js Core Collaborator
- Node.js Website Collaborator

Overview

- Versioning
- Official Builds
- Tested Platforms
- Contributions
- Working Groups
- ES2015
- Governance
- Nan
- Upgrading

Timeline

| | |
|----------|---|
| 3/11/13 | Node.js v0.10 Released |
| 3/28/13 | Node.js v0.11 Released (Unstable) |
| 11/24/14 | io.js is born |
| 1/14/15 | io.js v1.0.0 Released |
| 2/6/15 | Node.js v0.12 Released |
| 5/8/15 | Node.js BDFL steps down |
| 5/13/15 | io.js TC joins Node Foundation |
| 9/8/15 | Node.js v4.0.0 Released (1st Converged Release) |
| 10/13/15 | Node.js v4.2.1 Argon Released (1st LTS Release) |

SemVer

- Previously, odd-even versioning
- Moved to Semantic Versioning

Official Builds

| | | | |
|----------|--------|--------|-------|
| Windows | 32-bit | 64-bit | |
| Mac OS X | 64-bit | | |
| Linux | 32-bit | 64-bit | |
| SunOS | 32-bit | 64-bit | |
| ARM | ARMv6 | ARMv7 | ARMv8 |

Tested Platforms

Windows

| Windows 2008 | Windows 2012 | Windows 10 |
|---------------------|---------------------|-------------------|
| 64-bit | 64-bit | 64-bit |

FreeBSD

| FreeBSD 10.1 | FreeBSD 10.1 |
|---------------------|---------------------|
| 32-bit | 64-bit |

SmartOS

| SmartOS 14.x.x | SmartOS 14.x.x |
|-----------------------|-----------------------|
| 32-bit | 64-bit |

Tested Platforms

Linux

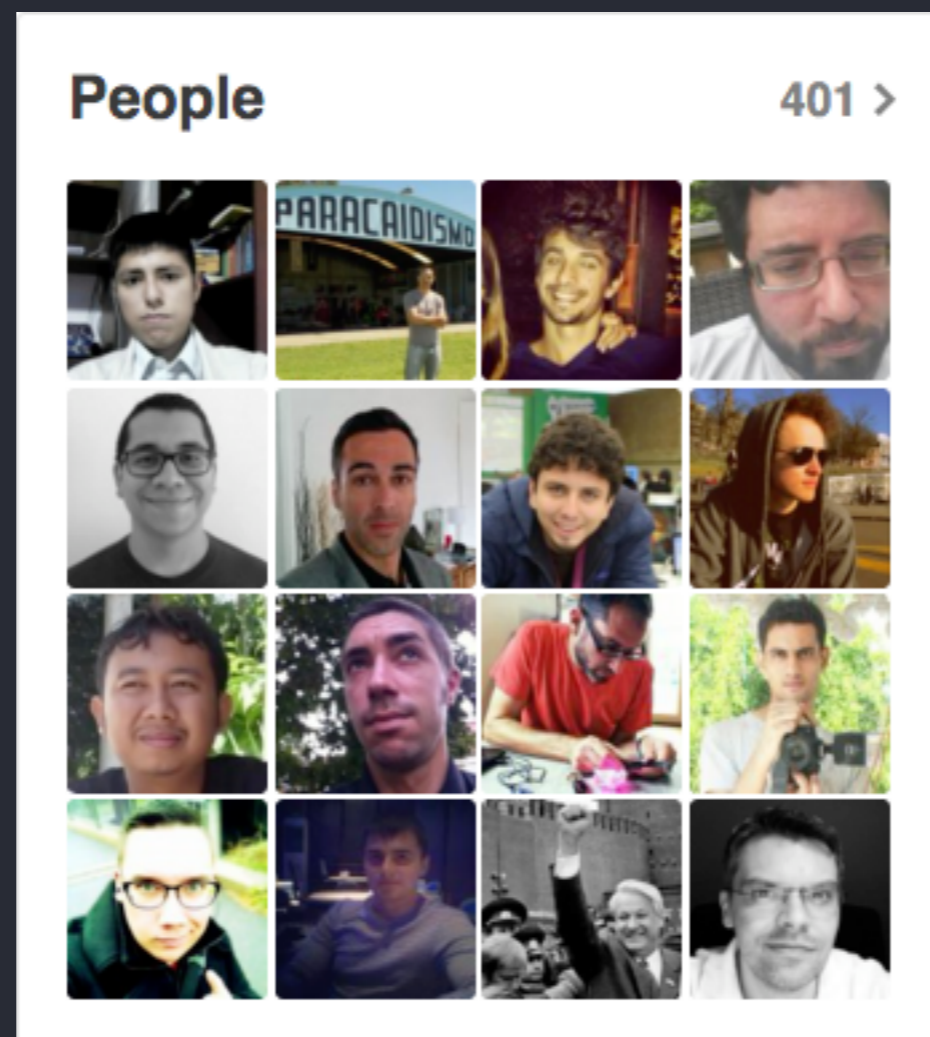
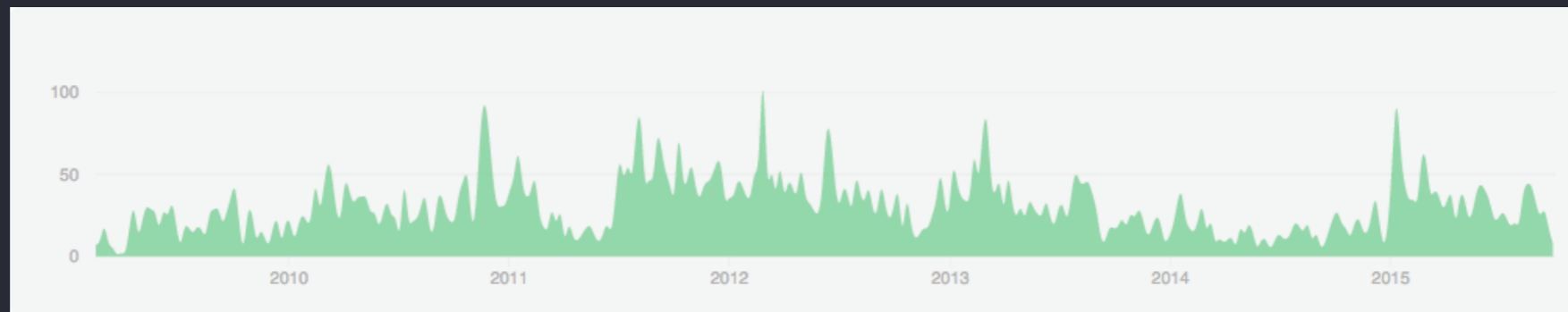
| | | | |
|---------------------|---------------------|---------------------|---------------------|
| CentOS 5.10 | CentOS 5.10 | CentOS 6.5 | CentOS 7.0 |
| 32-bit | 64-bit | 64-bit | 64-bit |
| Fedora 21 | Fedora 22 | Ubuntu 12.04 | Ubuntu 14.04 |
| 64-bit | 64-bit | 64-bit | 32-bit |
| Ubuntu 14.04 | Ubuntu 14.10 | Ubuntu 15.04 | Debian 8 |
| 64-bit | 64-bit | 64-bit | 64-bit |
| Fedora 21 | Ubuntu 14.04 | | |
| 64-bit (PPC BE) | 64-bit (PPC LE) | | |

Tested Platforms

ARM

| Ubuntu 14.04 | Ubuntu 14.04 | Wheezy | Wheezy |
|---------------------|---------------------|---------------|---------------|
| ARMv7 | ARMv8 | ARMv7 | ARMv8 |

Contributions



Working Groups

- API
- Benchmarking
- Build
- Documentation
- Evangelism
- Hardware
- Help
- Internationalization
- LTS
- Post Mortem Diagnostics
- Tracing
- Website

ES 2015

- Template Strings
- Maps / Sets
- Generators
- Promises
- Block Scoping (let/const)
- Symbols
- Arrow Functions

<https://nodejs.org/en/docs/es6/>

Template Strings

```
1 // Multiline Strings
2
3 console.log('this is a\n' + 'multiline string')
4
5 console.log(`this is a
6 multiline string`)
7
8 // Expression Interpolation
9 var a = 5
10 var b = 10
11 console.log('Fifteen is ' + (a + b) + ' and not ' + (2 * a + b) + '.')
12
13 console.log(`Fifteen is ${a + b} and not ${2 * a + b}.`)
```

Maps and Sets

```
1 // Maps
2 var map = new Map()
3 map.set('1', 5)
4 map.set(1, '25')
5 map.get('1') // => 5
6 map.get(1)   // => '25'
7 map.size    // => 2
8
9 for (let item of map.keys()) {
10   console.log(item)
11 }
12 // => '1'
13 // => 1
14 map.clear()
15 map.size    // => 0
```

```
1 // Sets
2 var set = new Set()
3 set.add('1')
4 set.add(2)
5
6 set.size    // => 2
7 set.has('1') // => true
8 set.has(2)  // => false
9
10 for (let item of set.values()) {
11   console.log(item)
12 }
13 // => '1'
14 // => 2
15 set.delete('1')
16 set.size    // => 1
17 set.clear()
```

Generators

```
1 // Generators
2
3 function* gen() {
4     yield 1
5     yield 2
6     yield 3
7 }
8
9 var g = gen()
10 g.next() // => { value: 1, done: false }
11 g.next() // => { value: 2, done: false }
12 g.next() // => { value: 3, done: false }
13 g.next() // => { value: undefined, done: true }
```

Promises

```
1  const fs = require('fs')
2
3  function readFile(fp) {
4    return new Promise(function(resolve, reject) {
5      fs.readFile(fp, 'utf8', function(err, data) {
6        if (err) return reject(err)
7        resolve(data)
8      })
9    })
10 }
11
12 readFile(__filename).then(function(result) {
13   console.log(result)
14 }).catch(function(err) {
15   console.log(err)
16 })
```


Block Scoping

```
1  'use strict'
2
3  // variables created with const are immutable
4  const fs = require('fs')
5  fs = {} // Throws TypeError: Assignment to constant variable.
6
7  for (var i = 0; i < 10; i++) {
8    var x = i
9    console.log(x) // the same as i
10 }
11 console.log(x) // 9
12
13 for (var i = 0; i < 10; i++) {
14   let x = i
15   console.log(x) // the same as i
16 }
17
18 console.log(x) // Throws ReferenceError: x is not defined
```

Symbols

```
1 // One use case for symbols is private properties
2 var a = Symbol('myPrivateProperty')
3 var obj = {}
4 obj[a] = 'This is my value'
5 console.log(obj) // {}
6
7 obj[Symbol('myPrivateProperty')] // undefined
8 obj[a] // 'This is my value'
```

Arrow Functions

```
1  function Helper() {  
2    this.name = 'Helper'  
3  }  
4  
5  Helper.prototype.doSomething = function(cb) {  
6    var self = this  
7    setImmediate(function() {  
8      cb(null, self.name)  
9    })  
10 }  
11  
12 // bind this  
13 Helper.prototype.doSomethingWithBind = function(cb) {  
14   setImmediate(function() {  
15     cb(null, this.name)  
16   }.bind(this))  
17 }  
18  
19 // use arrow functions  
20 Helper.prototype.doSomethingWithArrows = function(cb) {  
21   setImmediate(() => {  
22     cb(null, this.name)  
23   })  
24 }
```

Governance

Technical Committee

- Responsible for high-level guidance of the project.
- Initial membership invitations to the TSC were given to individuals who had been active contributors to Node.js, and who have significant experience with the management of the Node.js project.
- Follows consensus seeking process.
- No more than 1/3 of the TSC members may be affiliated with the same employer.
- [Current Members](#)

Governance

Collaborators

- In addition to the TSC, node also has collaborators.
- Anyone making significant and valuable contributions are made Collaborators and given commit-access to the project.
- Responsible for reviewing and landing pull requests as well as handling issues.
- May opt to elevate significant or controversial modifications, or modification that have not found consensus to the TSC for discussion.
- [Collaborators](#)

Governance

More details regarding the governance of Node.js can be found in the [GitHub repository](#)

Nan

Native Abstractions for Node.js

- Created by Rod Vagg (@rvagg) to make writing native addons easier across multiple versions of node.
- Helps prevent getting into a "macro tangle" when trying to support multiple versions

<https://github.com/nodejs/nan>

Nan

Supports the following versions

- node v0.10.x
- node v0.12.x
- iojs v1.x, v2.x, v3.x
- node v4.x, v5.x

Without Nan

```
1 #include <node_version.h>
2 #include <node.h>
3 #include <v8.h>
4
5 using namespace v8;
6
7 #define NODE_0_10_MODULE_VERSION 11
8 #define NODE_0_12_MODULE_VERSION 14
9 #define IOJS_1_0_MODULE_VERSION 42
10 #define IOJS_1_1_MODULE_VERSION 43
11 #define IOJS_2_0_MODULE_VERSION 44
12 #define IOJS_3_0_MODULE_VERSION 45
13 #define NODE_4_0_MODULE_VERSION 46
14
15 #if NODE_MODULE_VERSION <= NODE_0_10_MODULE_VERSION
16 Handle<Value> Method(const Arguments& args) {
17     HandleScope scope;
18     return scope.Close(String::New("world"));
19 }
20
21 void init(Handle<Object> exports) {
22     exports->Set(String::NewSymbol("hello"),
23                 FunctionTemplate::New(Method)->GetFunction());
24 }
25 #else
26 void Method(const FunctionCallbackInfo<Value>& args) {
27     Isolate* isolate = args.GetIsolate();
28     HandleScope scope(isolate);
29     args.GetReturnValue().Set(String::NewFromUtf8(isolate, "world"));
30 }
31 #endif // NODE_MODULE_VERSION
32
33 #if NODE_MODULE_VERSION == NODE_0_12_MODULE_VERSION
34 void init(Handle<Object> target) {
35     NODE_SET_METHOD(target, "hello", Method);
36 }
37 #elif NODE_MODULE_VERSION > NODE_0_12_MODULE_VERSION
38 void init(Local<Object> target) {
39     NODE_SET_METHOD(target, "hello", Method);
40 }
41 #endif
42
43 NODE_MODULE(binding, init)
```

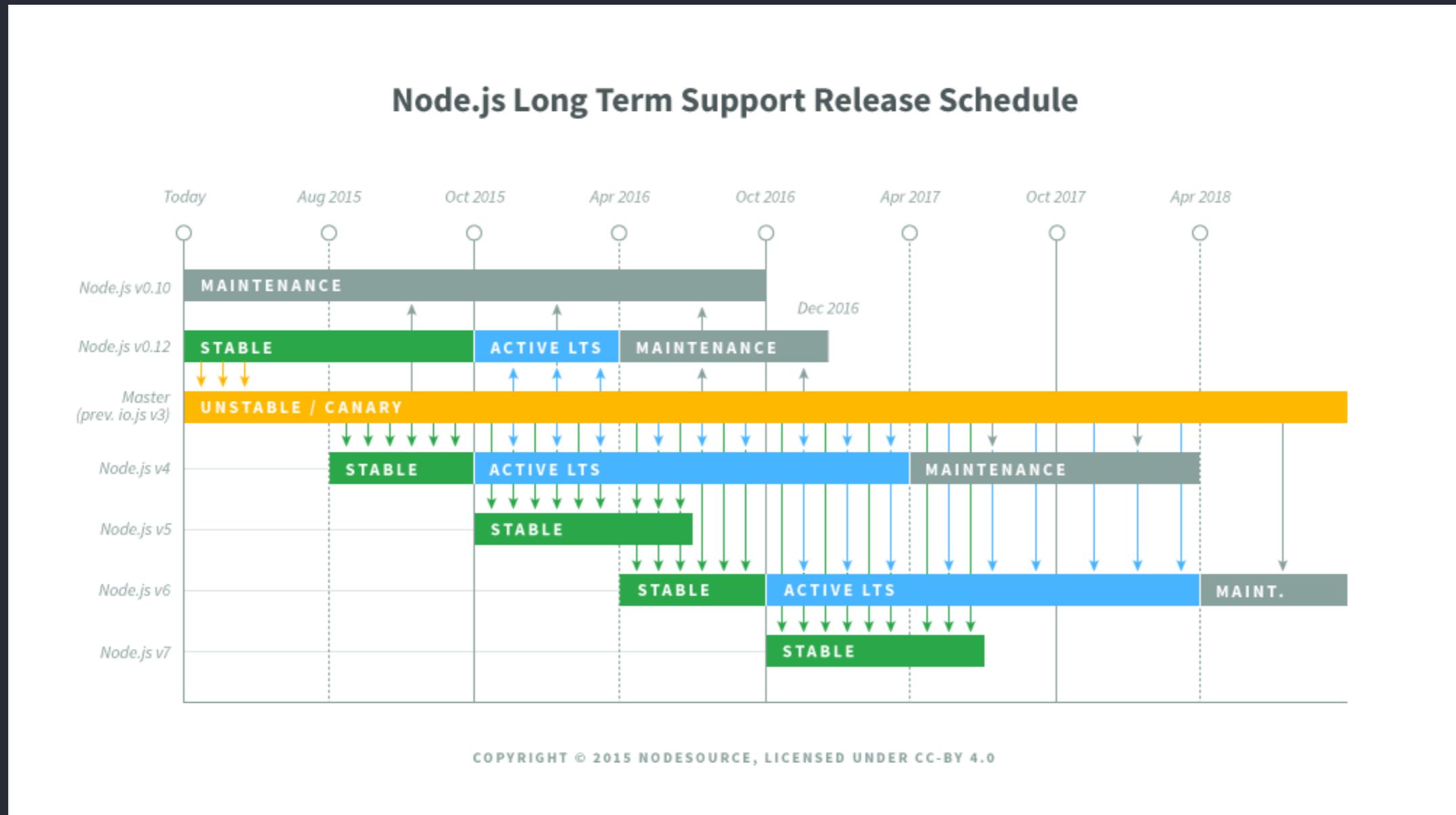
With Nan

```
1  #include <nan.h>
2  #include <node.h>
3  #include <v8.h>
4
5  using v8::FunctionTemplate;
6  using v8::Local;
7  using v8::Object;
8  using v8::String;
9  using Nan::GetFunction;
10 using Nan::New;
11 using Nan::Set;
12
13 NAN_METHOD(Method) {
14     info.GetReturnValue().Set(New<String>("world").ToLocalChecked());
15 }
16
17 NAN_MODULE_INIT(init) {
18     Set(target, New<String>("hello").ToLocalChecked(),
19         GetFunction(New<FunctionTemplate>(Method)).ToLocalChecked());
20 }
21
22 NODE_MODULE(binding_nan, init)
```

Upgrading

- [NodeSource Upgrade Self-service](#)
- [Blog post by Dave Lutz \(NodeSource\)](#)
- [upgrade-utils](#)
- [Breaking changes from v0.10 to v4](#)

Long Term Support



Thanks!

Any questions?