



CLOUD **FOUND**RY

SUMMIT

2017

JUNE 13-15 | SILICON VALLEY

Accelerate Your Digital Transformation

About Michael Dawson

Loves the web and building software (with Node.js!)

Senior Software Developer @ IBM

IBM Runtime Technologies Node.js Technical Lead

Node.js collaborator and CTC/TSC member

Active in LTS, build, benchmarking, api and post-mortem working groups

Contact me:

michael_dawson@ca.ibm.com

Twitter: [@mhdawson1](#)

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



Agenda

- Common Adoption Patterns
- Locality of Data
- Journey to the Cloud (aka Digital Transformation)
- Community work to remove roadblocks
- Summary and Wrap up



Common Adoption Patterns

- Skunk works project
- Internal APIs or sites
- Exposing existing data/systems
- Greenfields development

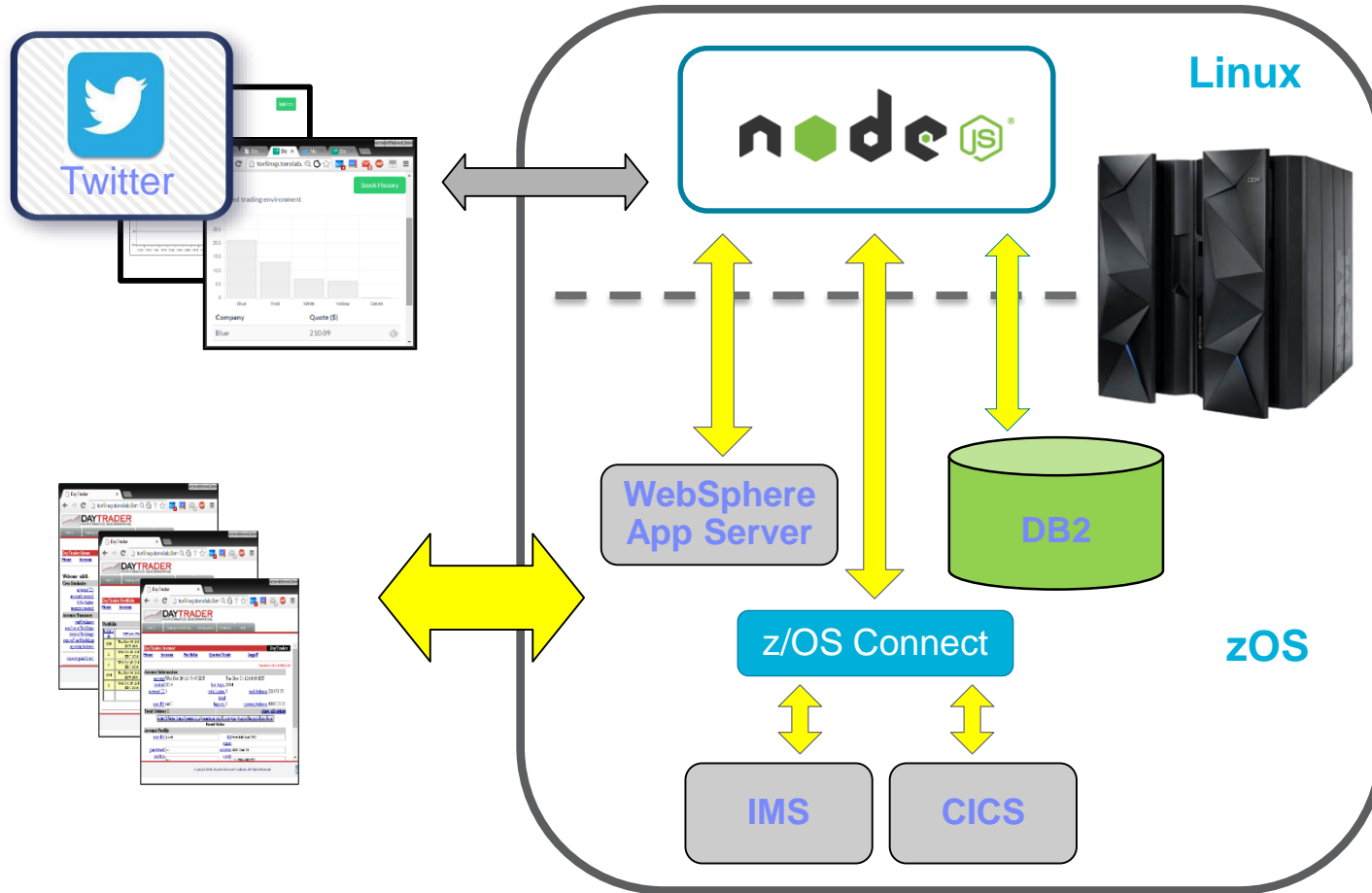


Locality of Data

- Moving data can be hard
 - Confidentiality
 - Regulation
 - Support for legacy data stores
 - Effort
- Locality of Data affects performance



Locality of Data – Example on z Systems



Co-locate
Node.js on
z vs x86

2.5x Better
Throughput

**60%
Faster**
Response
Time
to **DB2** on
z/OS



Journey to Cloud Native - Potential Roadblocks

- Having to change Environment
 - Platforms/Operating systems
 - Data
- Support
- Monitoring and Problem Investigation
- Baseline Enterprise Requirements
 - Internationalization
 - Security
 - Quality/Stability
 - Performance



IBM in the Node.js Community

- 9 Collaborators, 2 CTC/TSC
- Active in many WGs
- Platinum Foundation Sponsor



**Ben
Noordhuis**



**Gibson
Fahnestock**



**Michael
Dawson**



**Sam
Roberts**



**Michael
Tunnicliffe**



**Steven
Loomis**



**Ryan
Graham**



**Bert
Belder**



**Richard
Lau**



Environment: Platform Choice

- Community
 - Binaries for **Linux on P** and **Z** and **AIX**
- IBM SDK for Node.js
 - Shipping Node.js releases since 2013
 - 0.10.x + 0.12.x + 4.x + 6.x
 - Linux on x / p / z, AIX, Windows, Mac



- Working on support for z/OS
tech preview - <https://developer.ibm.com/node/sdk/>

Node.js download page showing LTS (Recommended For Most Users) and Current (Latest Features) versions. It provides download links for Windows Installer (.msi), Windows Binary (.zip), macOS Installer (.pkg), macOS Binaries (.tar.gz), Linux Binaries (x86/x64), Linux Binaries (ARM), and Source Code.

Additional Platforms

Additional Platforms section showing download links for SunOS Binaries, Docker Image, Linux on Power Systems, Linux on System z, and AIX on Power Systems.

IBM SDK for Node.js page showing Overview, Version 6, Version 4, and z/OS Tech Preview. It includes a description of the SDK and a link to download the latest version.

Environment: Deployment Choice

- Develop for Deployment Independence
 - Plan to Leverage Infrastructure services
 - Load balancing
 - Scale in/out
 - Monitoring
 - 12 Factor Apps (<https://12factor.net/>)
- Bluemix
 - Cloud Foundry and Docker/Kubernetes deployments
 - Public
 - Dedicated
 - Local
 - OpenWhisk
- Tools



Deployment choice: Pattern Generators

Node.js @ IBM

[Blogs](#) [Downloads](#) [Monitoring & Diagnostics](#) [Get Help](#)

The code generators can be either invoked via the Bluemix CLI ([Install steps](#)) or via the Bluemix UI ([Create Project](#)). Both methods require the user to have a Bluemix account ([Register](#)).

1. Select a Pattern

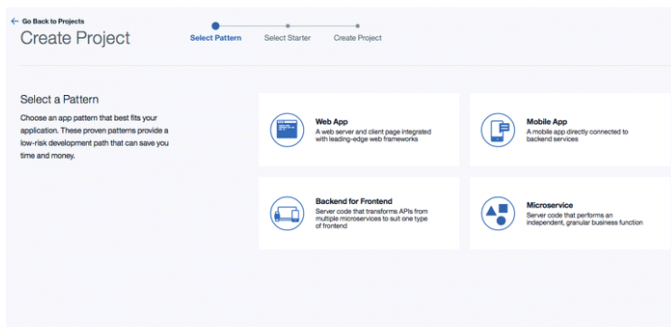
The code generators first ask a user to select their preferred design pattern out of the following:

1. Web App – A web server and client page integrated with leading-edge web frameworks
2. Mobile App – A mobile app directly connected to backend services
3. Backend for Frontend – Server code that transforms APIs from multiple microservices to suit one type of frontend
4. Microservice – Server code that performs an independent, granular business function

Bluemix CLI

```
bethanys-mbp:~ beth$ bx dev create  
  
dev version 0.0.6  
  
See the documentation on Bluemix:  
https://console.bluemix.net/docs/cloudnative/index.html  
  
? Select a pattern:  
1. Web App  
2. Mobile App  
3. Backend for Frontend  
4. Microservice  
Enter a number> █
```

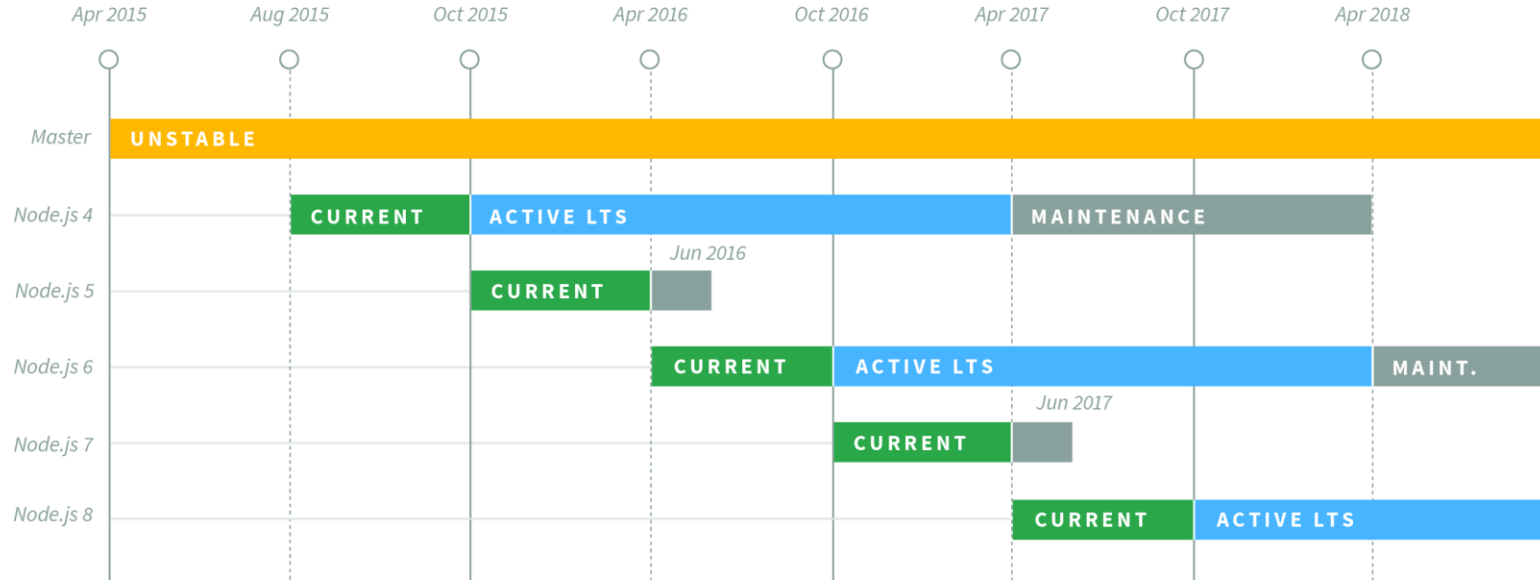
Bluemix UI



Support: LTS



Node.js Long Term Support (LTS) Release Schedule



COPYRIGHT © 2017 NODESOURCE, LICENSED UNDER CC-BY 4.0

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



Problem Determination: Tooling

- NodeReport
- Heap Dump generation
- Core inspection - LLNODE

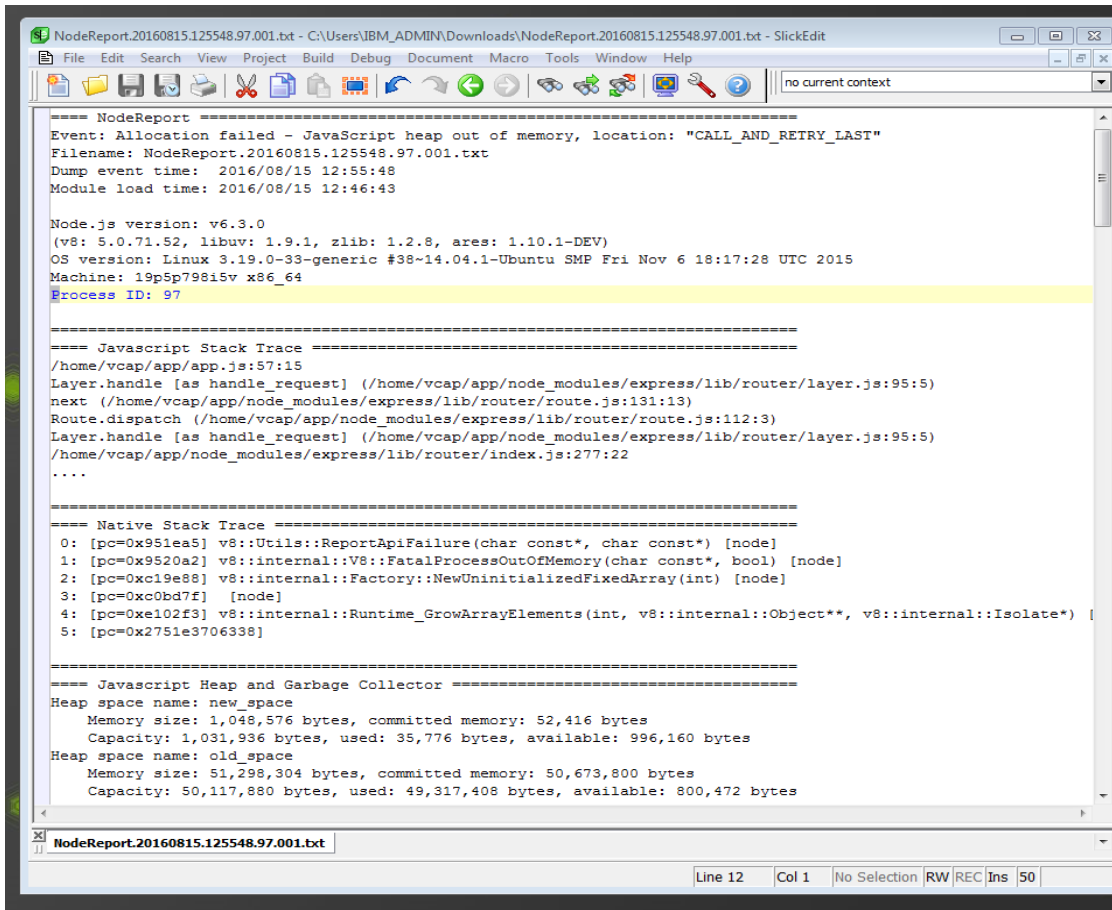


NodeReport

NodeReport content:

- Event summary
- Node.js and OS versions
- JavaScript stack trace
- Native stack trace
- Heap and GC statistics
- Resource usage
- libuv handle summary
- Environment variables
- OS ulimit settings

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

A screenshot of a text editor window displaying a Node.js crash report. The window title is "NodeReport.20160815.125548.97.001.txt - C:\Users\IBM_ADMIN\Downloads\NodeReport.20160815.125548.97.001.txt - SlickEdit". The report content is as follows:

```
==== NodeReport =====
Event: Allocation failed - JavaScript heap out of memory, location: "CALL_AND_RETRY_LAST"
Filename: NodeReport.20160815.125548.97.001.txt
Dump event time: 2016/08/15 12:55:48
Module load time: 2016/08/15 12:46:43

Node.js version: v6.3.0
(v8: 5.0.71.52, libuv: 1.9.1, zlib: 1.2.8, ares: 1.10.1-DEV)
OS version: Linux 3.19.0-33-generic #38~14.04.1-Ubuntu SMP Fri Nov 6 18:17:28 UTC 2015
Machine: 19p5p798i5v x86_64
Process ID: 97

==== Javascript Stack Trace =====
/home/vcap/app/app.js:57:15
Layer.handle [as handle_request] (/home/vcap/app/node_modules/express/lib/router/layer.js:95:5)
next (/home/vcap/app/node_modules/express/lib/router/route.js:131:13)
Route.dispatch (/home/vcap/app/node_modules/express/lib/router/route.js:112:3)
Layer.handle [as handle_request] (/home/vcap/app/node_modules/express/lib/router/layer.js:95:5)
/home/vcap/app/node_modules/express/lib/router/index.js:277:22
....

==== Native Stack Trace =====
0: [pc=0x951ea5] v8::Utils::ReportApiFailure(char const*, char const*) [node]
1: [pc=0x9520a2] v8::internal::V8::FatalProcessOutOfMemory(char const*, bool) [node]
2: [pc=0xc19e88] v8::internal::Factory::NewUninitializedFixedArray(int) [node]
3: [pc=0xc0bd7f] [node]
4: [pc=0xe102f3] v8::internal::Runtime_GrowArrayElements(int, v8::internal::Object**, v8::internal::Isolate*) [node]
5: [pc=0x2751e3706338]

==== Javascript Heap and Garbage Collector =====
Heap space name: new_space
Memory size: 1,048,576 bytes, committed memory: 52,416 bytes
Capacity: 1,031,936 bytes, used: 35,776 bytes, available: 996,160 bytes
Heap space name: old_space
Memory size: 51,298,304 bytes, committed memory: 50,673,800 bytes
Capacity: 50,117,880 bytes, used: 49,317,408 bytes, available: 800,472 bytes
```

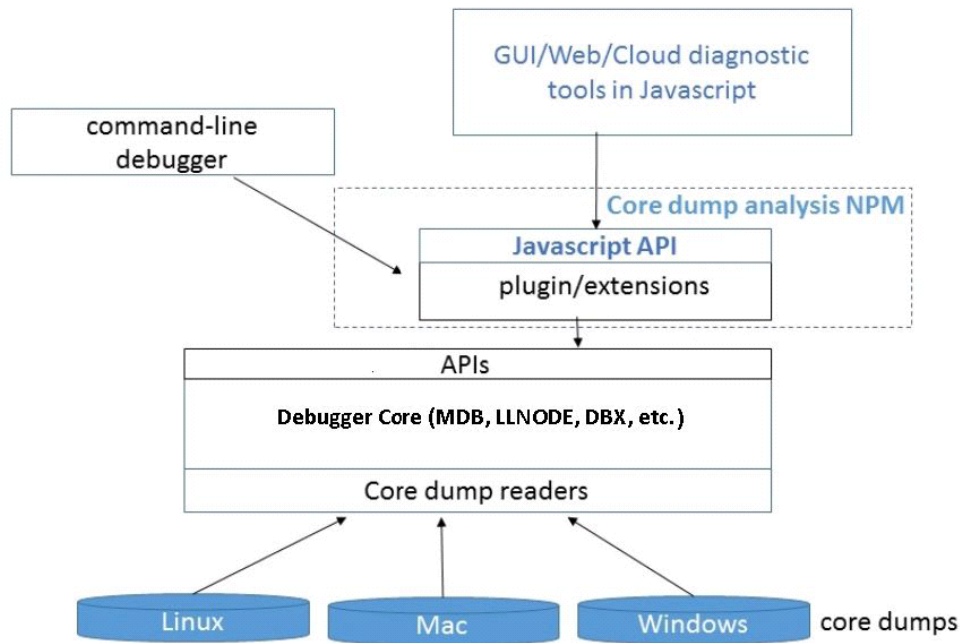
Heap Dump Generation

- Heapdump module – <https://github.com/bnoordhuis/node-heapdump>
- Chrome developer tools
- Limitations
 - Need to modify application
 - Slow to generate
 - $O(N)$ memory usage
 - Limited content
 - Output is large

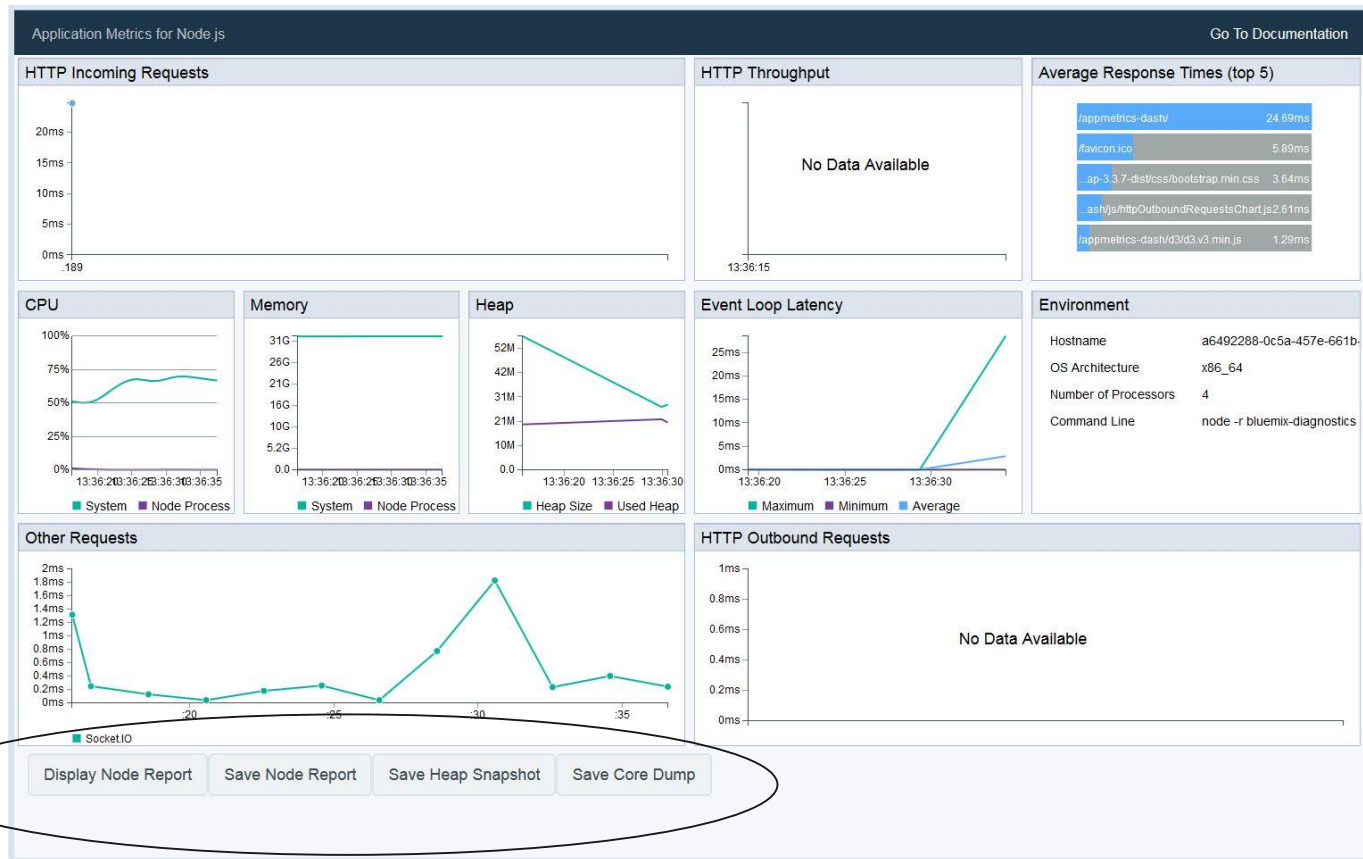


Core Inspection

- MDB
- IDDE
- Working in community to standardize on LLNODE



Problem Determination: In the Cloud



Monitoring: Tooling

- Appmetrics
- Appmetrics - Dash
- Health Center



Appmetrics-dash

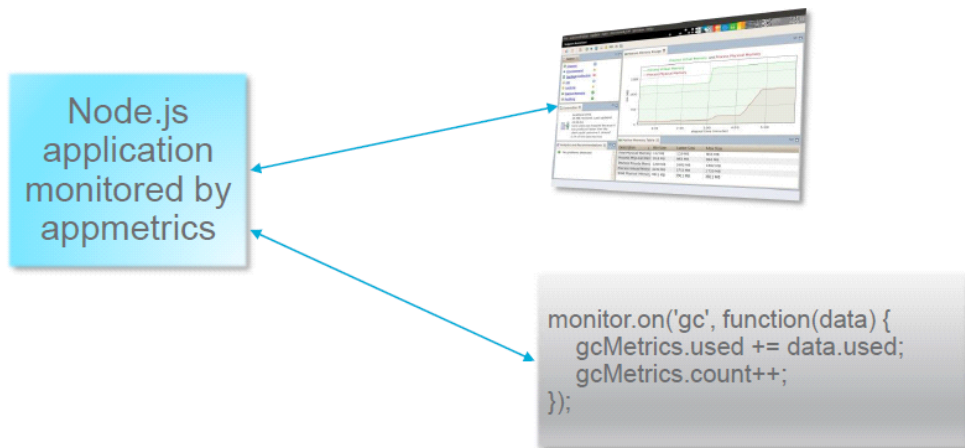


```
require('appmetrics-dash').monitor()
```



Node.js IBM – Appmetrics

<https://www.npmjs.com/package/appmetrics>



The screenshot shows the npm package page for `appmetrics`. The package is published by `codacy` and is marked as `public`. The description states: "Node Application Metrics monitoring and profiling agent". It further explains that the package instruments the Node.js runtime for performance monitoring and provides data via an API. It also mentions that data can be visualized in an Eclipse IDE using the IBM Monitoring and Diagnostics Tools - Health Center client. A link is provided for more details: <https://www.ibm.com/developerworks/java/jdk/tools/healthcenter/>. Below the description, a table lists the built-in data collection sources.

source	description
Environment	Machine and runtime environment information
CPU	Process and system CPU
Memory	Process and system memory usage
GC	Node/V8 garbage collection statistics
Event Loop	Event loop latency information
Express	Express 4.x Web Framework application request monitoring

Baseline ER:Quality and Stability

- Different release types
- Change flow processes
- Enhancement Proposal process
- Automation and Testing
 - Functional Tests
 - Module Testing (Citgm - <https://github.com/nodejs/citgm/>)
 - Stress Testing (Future)
 - Platform/OS coverage
 - Development Workflows (Future)
- Performance Benchmarks (<http://benchmarking.nodejs.org>)
- Tools (<https://coverage.nodejs.org>, ci jobs)

<https://nodejs.org/en/blog/community/quality-with-speed/>



Baseline ER: Application Stability – N-API

- Simplify native module maintenance
- Break dependency on v8
- Modules work across Node.js releases without re-compilation



<https://github.com/nodejs/abi-stable-node>



Baseline ER: Internationalization

- Internationalization
 - Integration of ICU
 - Internal Messages



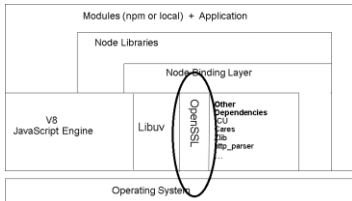
Baseline ER: Security

- Security

- Security WG - <https://github.com/nodejs/security-wg>



- FIPs



1747	OpenSSL Software Foundation 1829 Mount Ephraim Road Adamstown, MD 21710 USA Steve Marquess TEL: 877-673-6775 CST Lab: NVLAP 100432-0	OpenSSL FIPS Object Module (Software Versions: 2.0, 2.0.1, 2.0.2, 2.0.3, 2.0.4, 2.0.5, 2.0.6, 2.0.7, 2.0.8, 2.0.9 or 2.0.10) <i>(When built, installed, protected and initialized as assumed by the Crypto Officer role and as specified in the provided Security Policy. Appendix A of the provided Security Policy specifies the actual distribution tar file containing the source code of this module. There shall be no additions, deletions or alterations to the tar file contents as used during module build. The distribution tar file shall be verified as specified in Appendix A of the provided Security Policy. Installation and protection shall be completed as specified in Appendix A of the provided Security Policy. Initialization shall be invoked as per Section 4 of the provided Security Policy. Any deviation from specified verification, protection, installation and initialization procedures will result in a non FIPS 140-2 compliant module.)</i> Validated to FIPS 140-2 Consolidated Validation Certificate Security Policy	Software	06/27/2012 07/09/2012 07/18/2012 10/24/2012 01/22/2013 02/06/2013 02/22/2013 02/28/2013 03/28/2013 05/16/2013 06/14/2013 08/16/2013 08/23/2013 11/08/2013 12/20/2013 06/27/2014 07/03/2014 09/02/2014 09/12/2014 10/16/2014 12/31/2014 06/15/2015 09/04/2015 01/25/2016	Overall Level: 1 -Roles, Services, and Authentication: Level 2 Design Assurance: Level 3 -Operational Environment: Tested as meeting Level 1 with Android 2.2 running on Qualcomm QSD8250 (ARMv7) without NEON (gcc Compiler Version 4.4.0) Android 2.2 running on Qualcomm QSD8250 (ARMv7) with NEON (gcc Compiler Version 4.4.0) Microsoft Windows 7 (32 bit) running on Intel Celeron (Microsoft 32 bit C/C++ Optimizing Compiler Version 16.00) uCLinux 0.9.29 running on ARM 922T (ARMv4) (gcc Compiler Version 4.2.1) Fedora 14 running on Intel Core i5 with FAA (gcc Compiler Version 4.5.1) HP-UX 11i (32 bit) running on Intel Itanium 2 (HP C/aC++ B3910B) HP-UX 11i (64 bit) running on Intel Itanium 2 (HP C/aC++ B3910B) Ubuntu 10.04 running on Intel Pentium T4200 (gcc Compiler Version 4.1.3)
------	--	---	----------	--	--



[node-test-commit-linux-fips](#)

14 min - #3084

5 hr 48 min - #3071

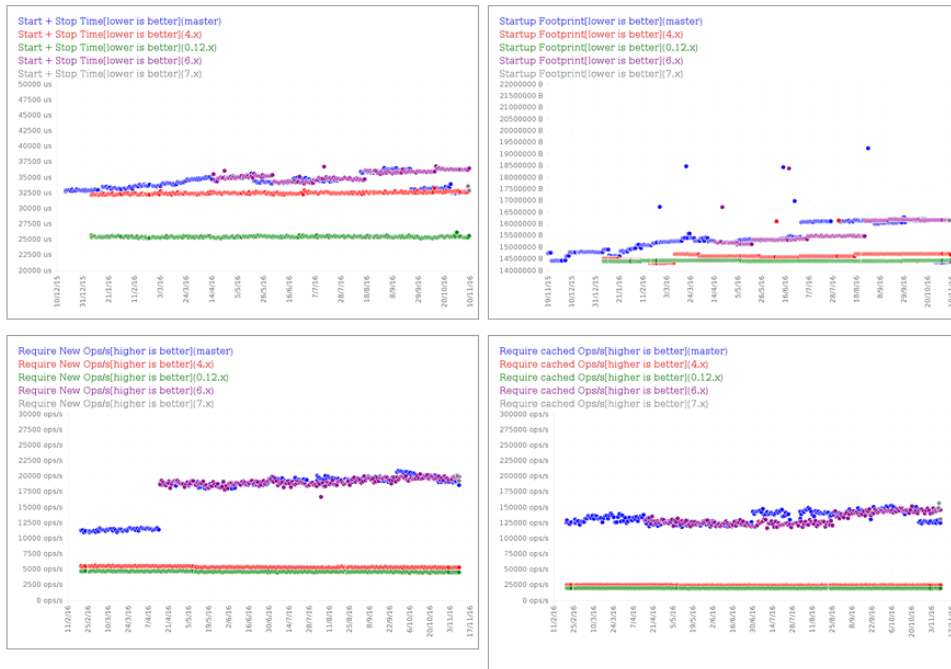
14 min



Baseline ER: Performance

<https://benchmarking.nodejs.org/>

- Benchmarking WG
 - Define Use Cases
 - Identify/Build Benchmarks
 - Run/Capture results



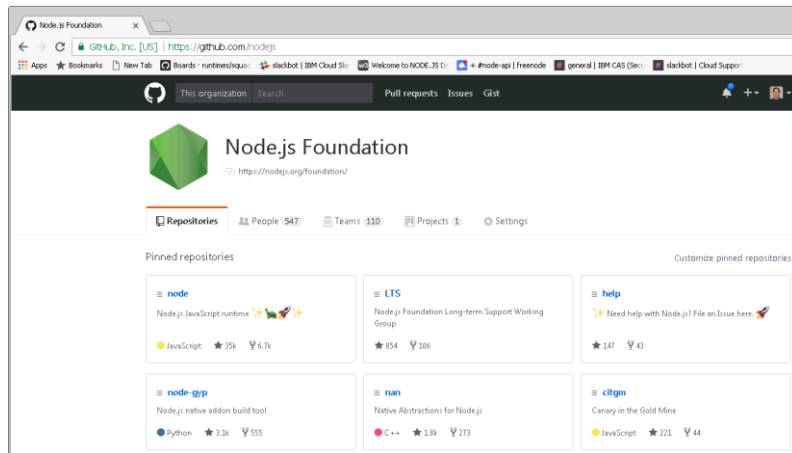
<https://github.com/nodejs/benchmarking>



Summary

- Node.js is a key runtime for Polyglot deployments
- Intro to typical journey to Cloud Native
- Support for existing environments is an enabler for Cloud Native
- Community work to reduce potential roadblocks
- Meet us in github to get involved

<https://github.com/nodejs>



Questions ?



Copyrights and Trademarks

© IBM Corporation 2017. All Rights Reserved

IBM, the IBM logo, ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies.

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.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

“TWITTER, TWEET, RETWEET and the Twitter logo are trademarks of Twitter, Inc. or its affiliates.”

