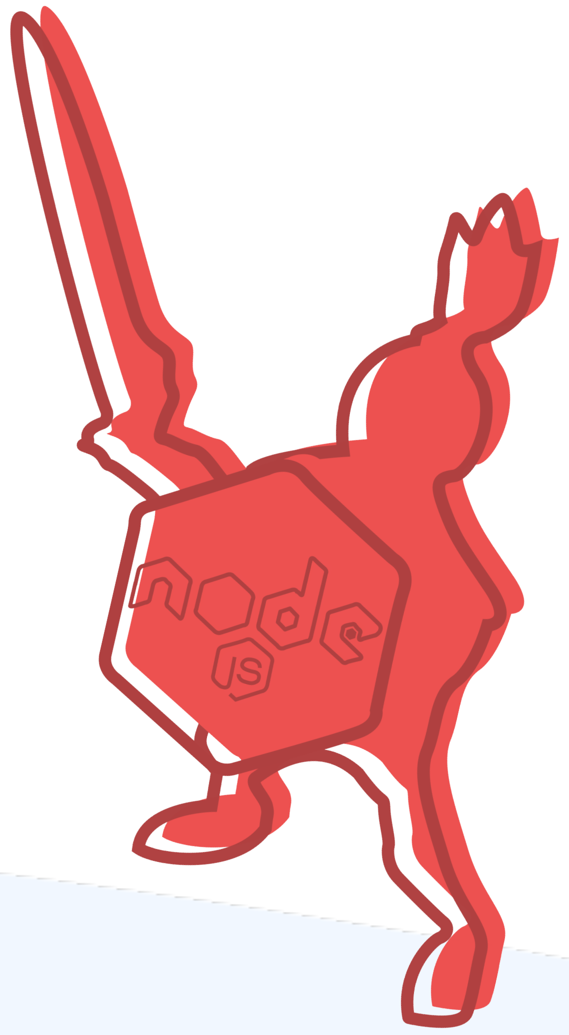
llnode 101

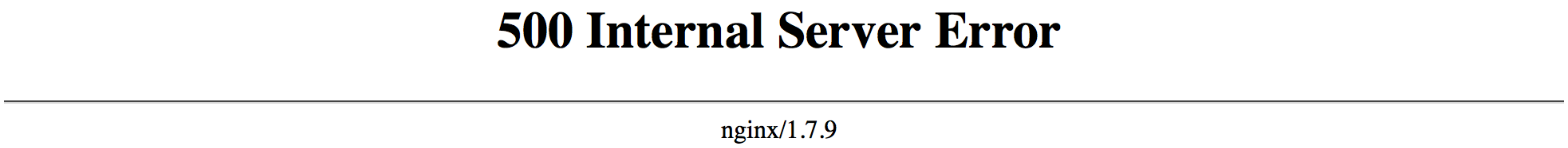
Troubleshooting Node.js Production Issues

Mary Marchini

@mmarkini

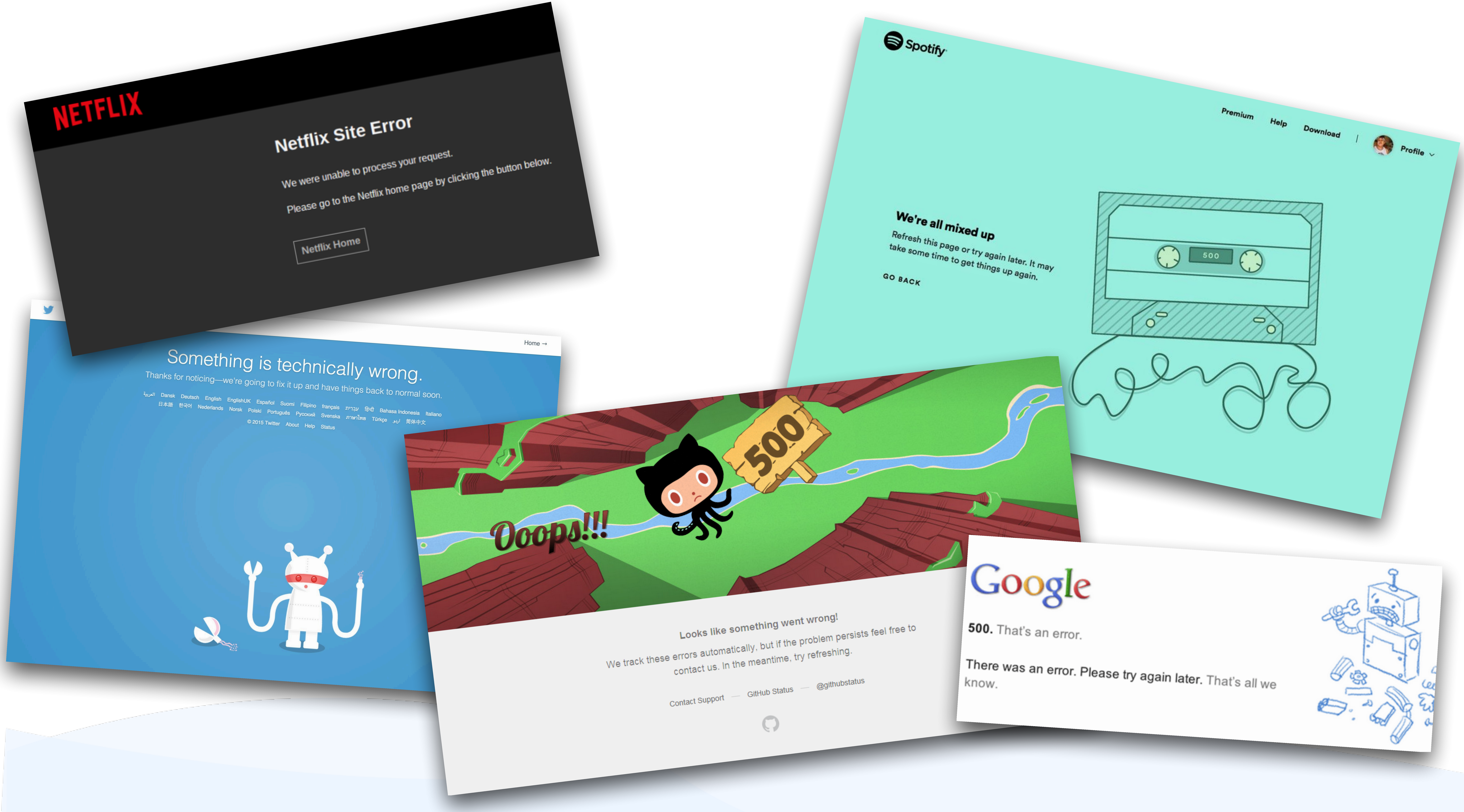
sthima





500 Internal Server Error

nginx/1.7.9



NETFLIX

Netflix Site Error

We were unable to process your request.

Please go to the Netflix home page by clicking the button below.

Netflix Home

Spotify

Premium Help Download | Profile

We're all mixed up

Refresh this page or try again later. It may take some time to get things up again.

GO BACK

Home →

Something is technically wrong.

Thanks for noticing—we're going to fix it up and have things back to normal soon.

العربية Dansk Deutsch English EnglishUK Español Suomi Filipino français עִבְרִית हिन्दी Bahasa Indonesia Italiano
日本語 한국어 Nederlands Norsk Polski Português Русский Svenska ภาษาไทย Türkçe اردو 简体中文

© 2015 Twitter About Help Status

Looks like something went wrong!

We track these errors automatically, but if the problem persists feel free to contact us. In the meantime, try refreshing.

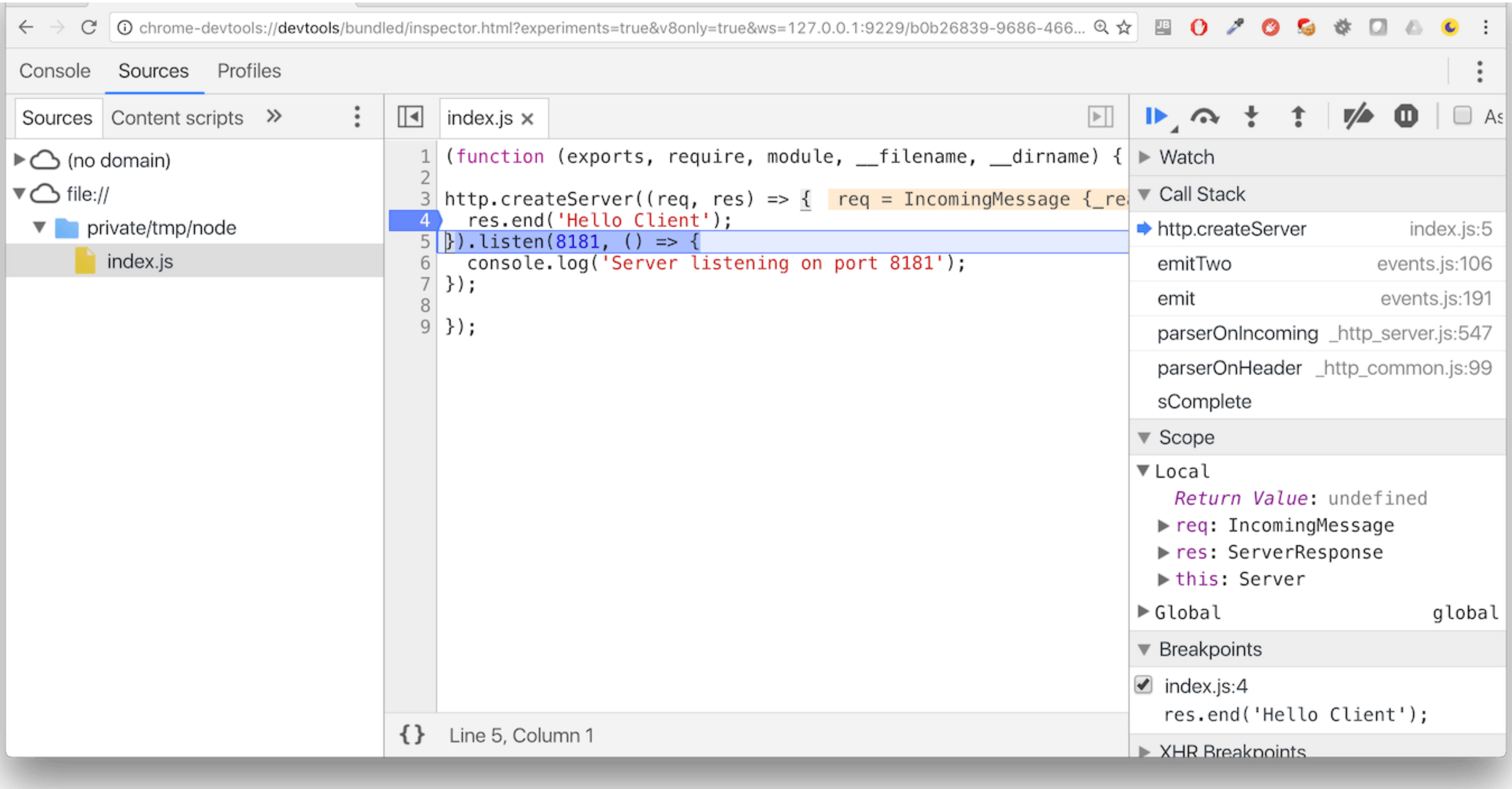
Contact Support — GitHub Status — @githubstatus

Google

500. That's an error.

There was an error. Please try again later. That's all we know.

DevTools



```
(function (exports, require, module, __filename, __dirname) {

http.createServer((req, res) => {  req = IncomingMessage {_re
  res.end('Hello Client');
}).listen(8181, () => {
  console.log('Server listening on port 8181');
});

});
```

Line 5, Column 1

Call Stack

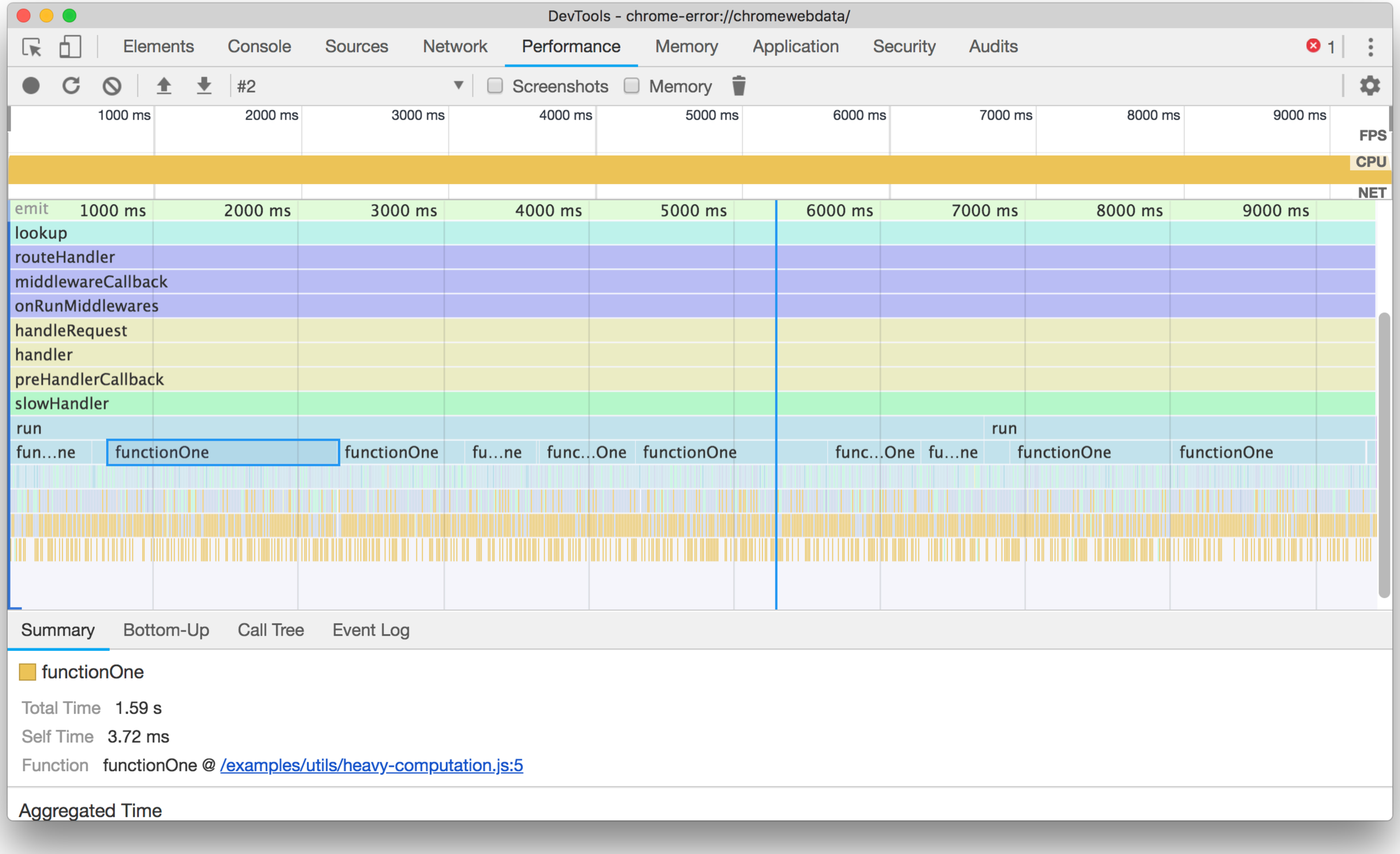
http.createServer	index.js:5
emitTwo	events.js:106
emit	events.js:191
parserOnIncoming	_http_server.js:547
parserOnHeader	_http_common.js:99
sComplete	

Scope

```
▼Local
   Return Value: undefined
  ▶ req: IncomingMessage
  ▶ res: ServerResponse
  ▶ this: Server
▶Global                        global
```

Breakpoints

☑ index.js:4
res.end('Hello Client');



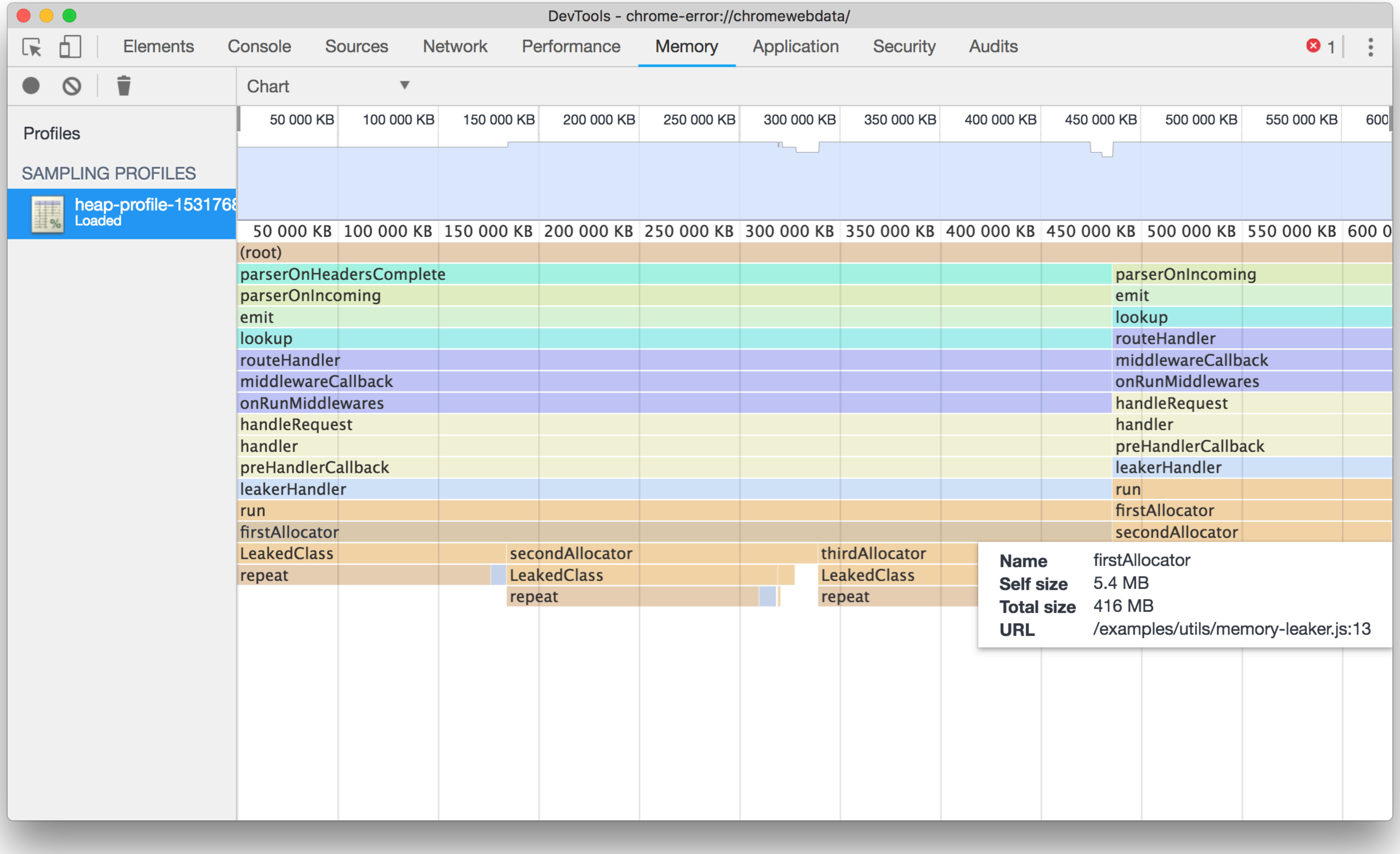
functionOne

Total Time 1.59 s

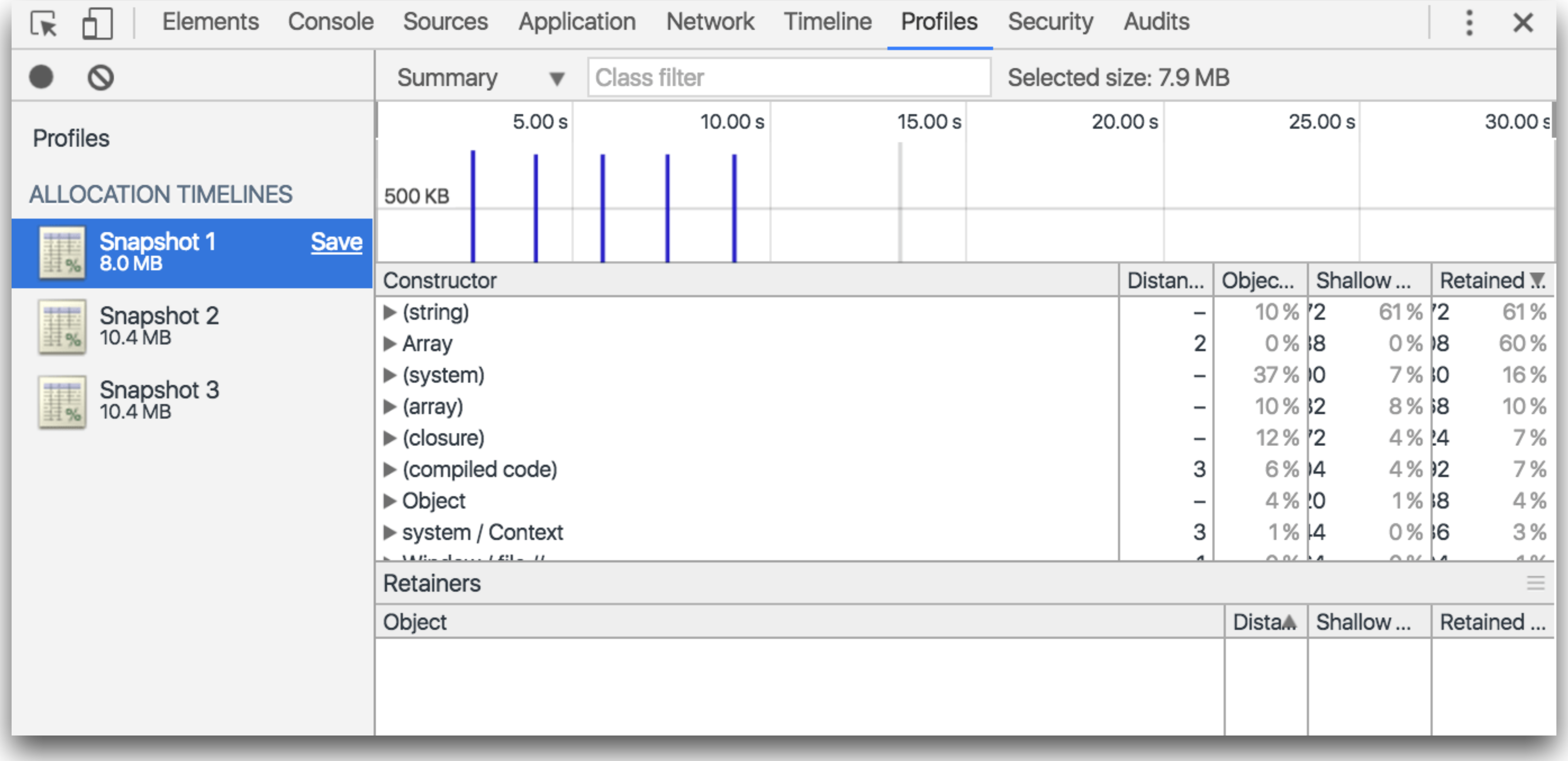
Self Time 3.72 ms

Function functionOne @ /examples/utils/heavy-computation.js:5

Aggregated Time



Name firstAllocator
Self size 5.4 MB
Total size 416 MB
URL /examples/utils/memory-leaker.js:13



Profiles

ALLOCATION TIMELINES

Snapshot 1
8.0 MB

Snapshot 2
10.4 MB

Snapshot 3
10.4 MB

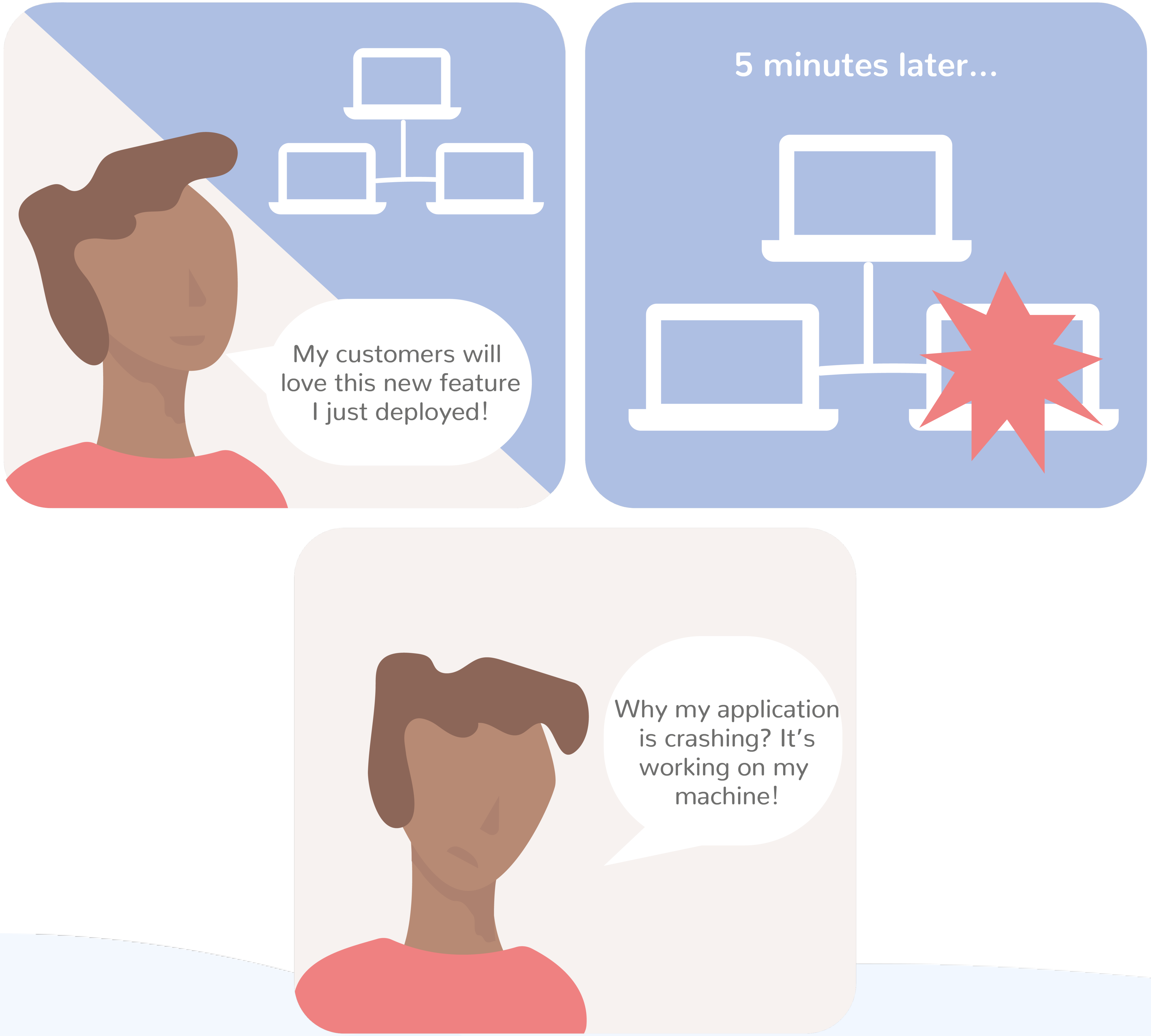
Summary ▼ Class filter Selected size: 7.9 MB

Constructor	Distan...	Objec...	Shallow ...	Retained
▶ (string)	–	10 %	2 61 %	2 61 %
▶ Array	2	0 %	8 0 %	8 60 %
▶ (system)	–	37 %	0 7 %	0 16 %
▶ (array)	–	10 %	2 8 %	8 10 %
▶ (closure)	–	12 %	2 4 %	4 7 %
▶ (compiled code)	3	6 %	4 4 %	2 7 %
▶ Object	–	4 %	0 1 %	8 4 %
▶ system / Context	3	1 %	4 0 %	6 3 %

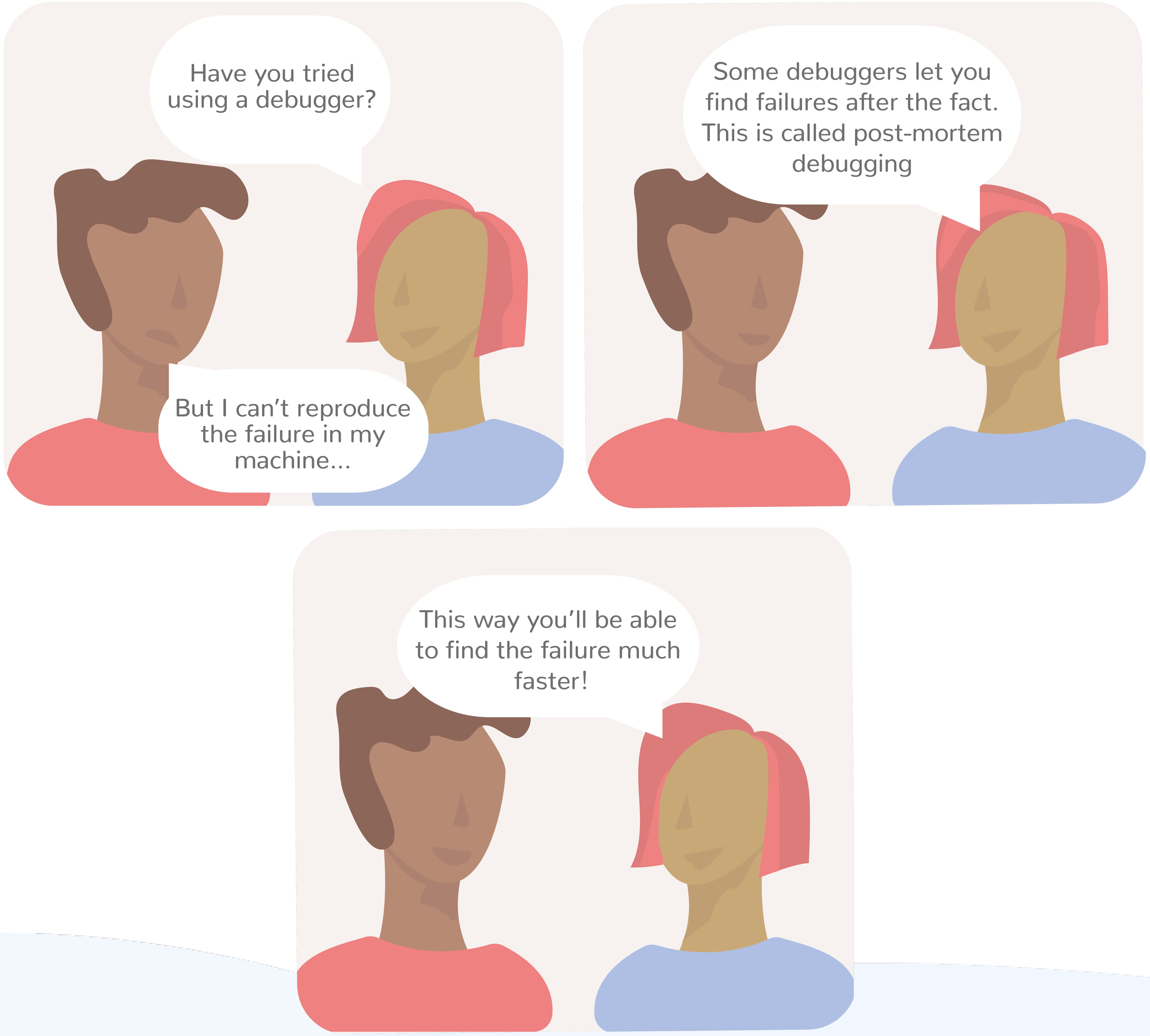
Retainers

Object	Dista▲	Shallow ...	Retained ...

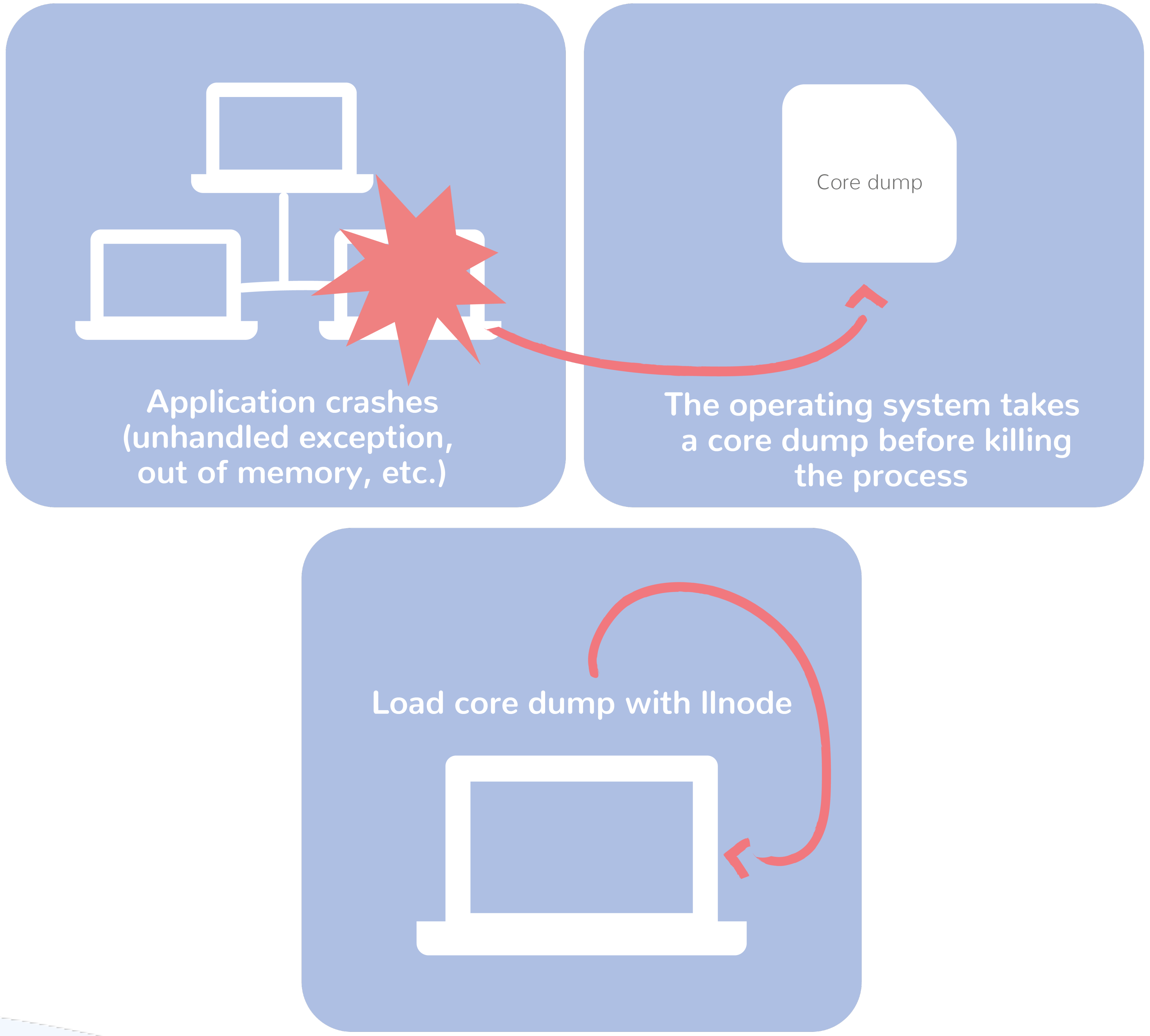
Why another tool?


My customers will love this new feature I just deployed!
5 minutes later...
Why my application is crashing? It's working on my machine!

Post-mortem debugging


Have you tried using a debugger?
But I can’t reproduce the failure in my machine...
Some debuggers let you find failures after the fact. This is called post-mortem debugging
This way you’ll be able to find the failure much faster!

How it works (on crash)

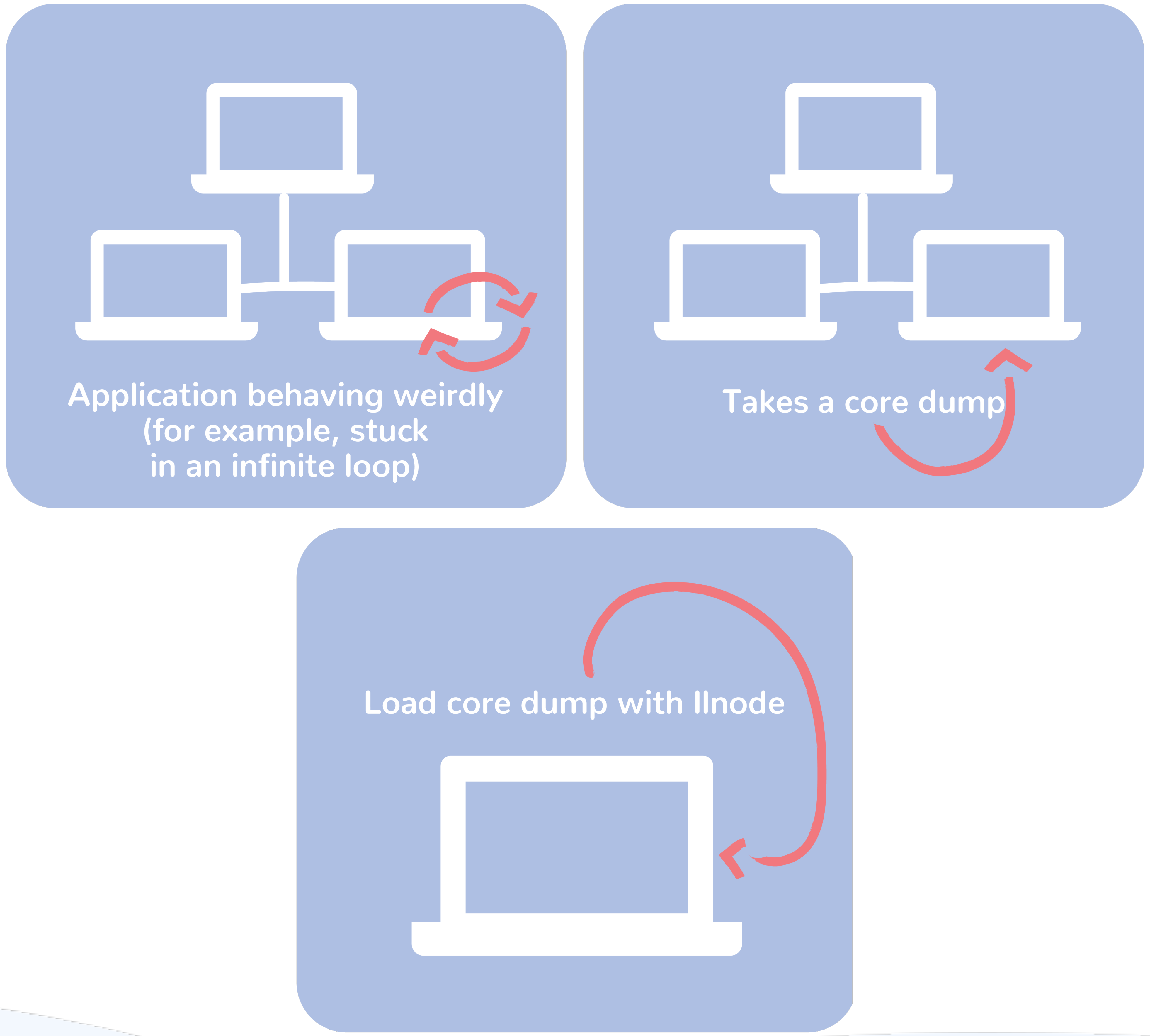


Application crashes (unhandled exception, out of memory, etc.)

The operating system takes a core dump before killing the process

Load core dump with llnode

How it works (on demand)



Application behaving weirdly
(for example, stuck
in an infinite loop)

Takes a core dump

Load core dump with llnode

Post-mortem & Node.js

mdb_v8 - Node.js postmortem origin

- Module for mdb
- First postmortem tool for Node.js
- Available for Solaris-based OSs
  - Not available for Linux, Windows, OS X
- Don’t work with Node.js v8.x or later

llnode - making postmortem cross-platform

- Inspired on mdb_v8
- Plugin for lldb
- Works on Linux, OS X, FreeBSD
  - Windows support on the way
- Is working for current Node.js releases
  - (v6.x, v8.x and v10.x)

llnode features

- Complete call stack
- Objects Statistics
- List all allocated objects of a given type
- Inspect JavaScript objects
- Find all references to a given object
- Inspect the workqueue

Stack Traces

Complete Stack Trace

```
method at index.js:27:43
closure at index.js:18:17
startup at internal/bootstrap/node.js:1:10
bootstrapNodeJSCore at internal/bootstrap/node.js:1:10
```

```
v8::base::OS::Abort()
v8::internal::Isolate::Throw(v8::internal::Object*, v8::internal::MessageLocation*)
<exit> [V8 Builtin]
<stub> [V8 Builtin]
method(this=<Object: Class>) at index.js:27:43
closure(this=<undefined>) at index.js:18:17
startup(this=<undefined>) at internal/bootstrap/node.js:1:10
bootstrapNodeJSCore(this=<null>, ...) at internal/bootstrap/node.js:1:10
<internal> [V8 Builtin]
<entry> [V8 Builtin]
v8::Function::Call(v8::Local<v8::Context>, v8::Local<v8::Value>, int, v8::Local<v8::Value>*)
node::LoadEnvironment(node::Environment*)
node::Start(v8::Isolate*, node::IsolateData*, int, char const* const*, int, char const* const*)
node::Start(uv_loop_s*, int, char const* const*, int, char const* const*)
node::Start(int, char**)
start
```

```
v8::base::OS::Abort()
v8::internal::Isolate::Throw(v8::internal::Object*, v8::internal::MessageLocation*)
<exit> [V8 Builtin]
<stub> [V8 Builtin]
method(this=<Object: Class>) at index.js:27:43
closure(this=<undefined>) at index.js:18:17
startup(this=<undefined>) at internal/bootstrap/node.js:1:10
bootstrapNodeJSCore(this=<null>, ...) at internal/bootstrap/node.js:1:10
<internal> [V8 Builtin]
<entry> [V8 Builtin]
v8::Function::Call(v8::Local<v8::Context>, v8::Local<v8::Value>, int, v8::Local<v8::Value>*)
node::LoadEnvironment(node::Environment*)
node::Start(v8::Isolate*, node::IsolateData*, int, char const* const*, int, char const* const*)
node::Start(uv_loop_s*, int, char const* const*, int, char const* const*)
node::Start(int, char**)
start
```

Argument Types

```
function foo(arg1, arg2) {}

foo(1);
foo("bar");
foo(42, "bar");
foo(42, true, "biz");
```

```
function foo(arg1, arg2) {}

foo(1);
foo("bar");
foo(42, "bar");
foo(42, true, "biz");
```

```
foo(this=0x94c34e9aa19:<Global proxy>, <Smi: 1>, <undefined>)
```

```
function foo(arg1, arg2) {}

foo(1);
foo("bar");
foo(42, "bar");
foo(42, true, "biz");
```

```
foo(this=0x94c34e9aa19:<Global proxy>, 0x35d26fed7761:<String: "bar">, <undefined>)
```

```
function foo(arg1, arg2) {}

foo(1);
foo("bar");
foo(42, "bar");
foo(42, true, "biz");
```

```
foo(this=0x94c34e9aa19:<Global proxy>, <Smi: 42>, 0x35d26fed7761:<String: "bar">)
```

```
function foo(arg1, arg2) {}

foo(1);
foo("bar");
foo(42, "bar");
foo(42, true, "biz");
```

```
foo(this=0x94c34e9aa19:<Global proxy>, <Smi: 42>, <true>)
```

What is *this*

```
class Foo {
  func() {}
}

class Bar {};

const foo = new Foo();
foo.func();
foo.func.bind(new Bar)();
```

```
class Foo {
  func() {}
}

class Bar {};

const foo = new Foo();
foo.func();
foo.func.bind(new Bar)();
```

```
func(this=0x3c3a90765529:<Object: Foo>)
```

```
class Foo {
  func() {}
}

class Bar {};

const foo = new Foo();
foo.func();
foo.func.bind(new Bar)();
```

```
func(this=0x3c3a9076afb1:<Object: Bar>)
```

Objects Statistics

```
Instances  Total Size Name
---------- ---------- ----
         1         24 WebAssembly
         1         24 console
         1         32 (Object)
         1         56 AssertionError
       ...        ... ...
        66       2272 ContextifyScript
       195      10920 NodeError
       402      12864 (Array)
       599      34968 Object
      7721      53960 (String)
      9273     123412 Foo
---------- ----------
      9129     127816
```

```
Instances  Total Size Name
---------- ---------- ----
         1         24 WebAssembly
         1         24 console
         1         32 (Object)
         1         56 AssertionError
       ...        ... ...
        66       2272 ContextifyScript
       195      10920 NodeError
       402      12864 (Array)
       599      34968 Object
      7721      53960 (String)
      9273     123412 Foo
---------- ----------
      9129     127816
```

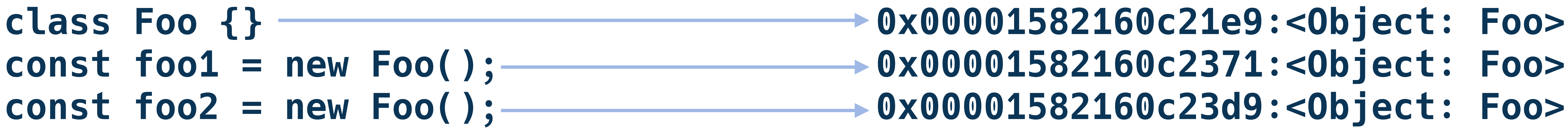
```
 Instances  Total Size Name
---------- ---------- ----
         1         24 WebAssembly
         1         24 console
         1         32 (Object)
         1         56 AssertionError
       ...        ... ...
        66       2272 ContextifyScript
       195      10920 NodeError
       402      12864 (Array)
       599      34968 Object
      7721      53960 (String)
      9273     123412 Foo
---------- ----------
      9129     127816
```

```
Instances  Total Size Name
---------- ---------- ----
         1         24 WebAssembly
         1         24 console
         1         32 (Object)
         1         56 AssertionError
       ...        ... ...
        66       2272 ContextifyScript
       195      10920 NodeError
       402      12864 (Array)
       599      34968 Object
      7721      53960 (String)
      9273     123412 Foo
---------- ----------
      9129     127816
```

List Objects

```
class Foo {}
const foo1 = new Foo();
const foo2 = new Foo();
```

```
0x00001582160c21e9:<Object: Foo>
0x00001582160c2371:<Object: Foo>
0x00001582160c23d9:<Object: Foo>
```



```
class Foo {}
const foo1 = new Foo();
const foo2 = new Foo();
```

```
0x00001582160c21e9:<Object: Foo>
0x00001582160c2371:<Object: Foo>
0x00001582160c23d9:<Object: Foo>
```

List Retainers

```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```

```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```

```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

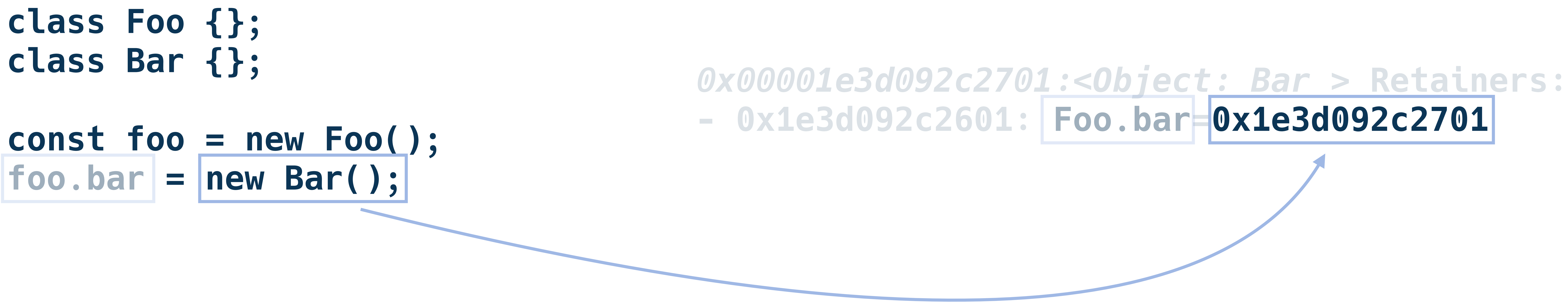
```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```

```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```

```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```



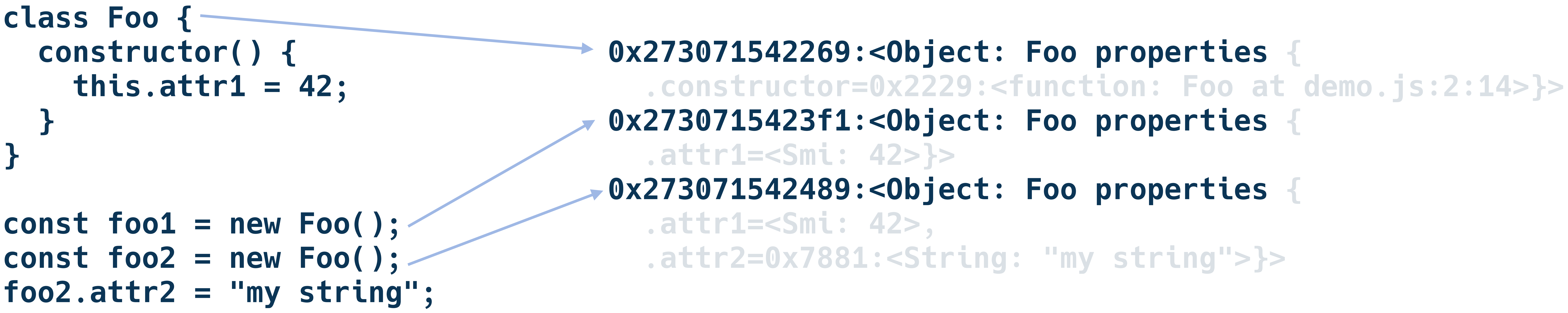
```
class Foo {};
class Bar {};

const foo = new Foo();
foo.bar = new Bar();
```

```
0x00001e3d092c2701:<Object: Bar > Retainers:
- 0x1e3d092c2601: Foo.bar=0x1e3d092c2701
```

Object Inspection

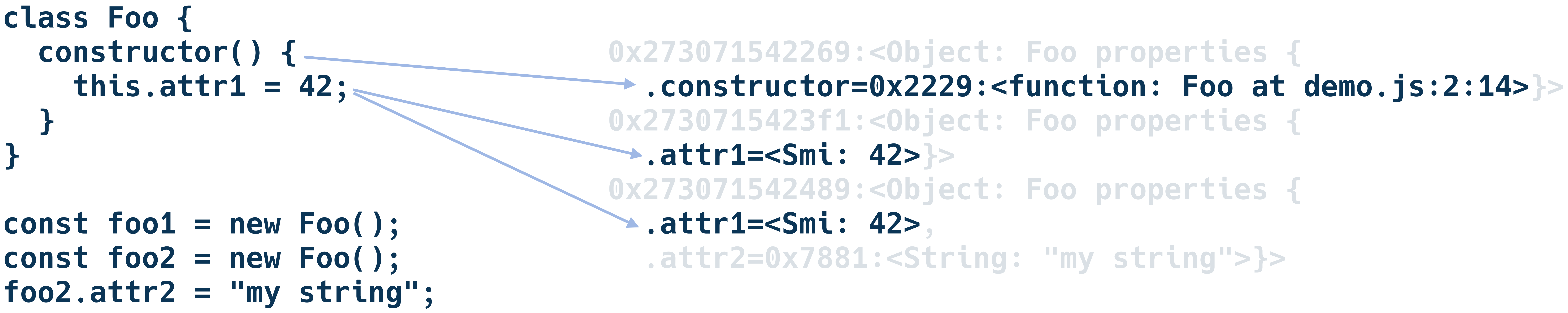
```
class Foo {
  constructor() {
    this.attr1 = 42;
  }
}

const foo1 = new Foo();
const foo2 = new Foo();
foo2.attr2 = "my string";
```

```
0x273071542269:<Object: Foo properties {
  .constructor=0x2229:<function: Foo at demo.js:2:14>}>
0x2730715423f1:<Object: Foo properties {
  .attr1=<Smi: 42>}>
0x273071542489:<Object: Foo properties {
  .attr1=<Smi: 42>,
  .attr2=0x7881:<String: "my string">}>
```



```
class Foo {
  constructor() {
    this.attr1 = 42;
  }
}

const foo1 = new Foo();
const foo2 = new Foo();
foo2.attr2 = "my string";
```

```
0x273071542269:<Object: Foo properties {
  .constructor=0x2229:<function: Foo at demo.js:2:14>}>
0x2730715423f1:<Object: Foo properties {
  .attr1=<Smi: 42>}>
0x273071542489:<Object: Foo properties {
  .attr1=<Smi: 42>,
  .attr2=0x7881:<String: "my string">}>
```



```
class Foo {
  constructor() {
    this.attr1 = 42;
  }
}

const foo1 = new Foo();
const foo2 = new Foo();
foo2.attr2 = "my string";
```

```
0x273071542269:<Object: Foo properties {
 .constructor=0x2229:<function: Foo at demo.js:2:14>}>
0x2730715423f1:<Object: Foo properties {
 .attr1=<Smi: 42>}>
0x273071542489:<Object: Foo properties {
 .attr1=<Smi: 42>,
 .attr2=0x7881:<String: "my string">}>
```

```
class Foo {
  constructor() {
    this.attr1 = 42;
  }
}

const foo1 = new Foo();
const foo2 = new Foo();
foo2.attr2 = "my string";
```

```
0x273071542269:<Object: Foo properties {
  .constructor=0x2229:<function: Foo at demo.js:2:14>}>
0x2730715423f1:<Object: Foo properties {
  .attr1=<Smi: 42>}>
0x273071542489:<Object: Foo properties {
  .attr1=<Smi: 42>,
  .attr2=0x7881:<String: "my string">}>
```

Inspect Workqueue

```
const fs = require('fs');

setTimeout(() => {}, 1000);

fs.readFile(__filename, () => {});
```

```
<Object: Timer properties {
    ._list=0x3bb42b11:<Object: TimersList>}
  internal fields {
    0x0000000002800380}>


<Object: FSReqWrap properties {
    .context=0xbb472f1:<Object: ReadFileContext>,
    .oncomplete=0x599f5d1:<function: ...
  internal fields {
    0x0000000002809db0}>
```

```
<Object: Timer properties {
    ._list=0x3bb42b11:<Object: TimersList>}
  internal fields {
    0x0000000002800380}>
```

```
const fs = require('fs');

setTimeout(() => {}, 1000);

fs.readFile(__filename, () => {});
```

```
<Object: FSReqWrap properties {
    .context=0xbb472f1:<Object: ReadFileContext>,
    .oncomplete=0x599f5d1:<function: ...
  internal fields {
    0x0000000002809db0}>
```

```
const fs = require('fs');

setTimeout(() => {}, 1000);

fs.readFile(__filename, () => {});
```

```
<Object: Timer properties {
    ._list=0x3bb42b11:<Object: TimersList>}
  internal fields {
    0x0000000002800380}>
```

```
<Object: FSReqWrap properties {
    .context=0xbb472f1:<Object: ReadFileContext>,
    .oncomplete=0x599f5d1:<function: ...
  internal fields {
    0x0000000002809db0}>
```

llnode main use cases

- Memory Leaks / Out of Memory
- Uncaught Exceptions
- Infinite Loops
- Native Modules / Node.js / V8 Crashes

llnode main use cases

- Memory Leaks / Out of Memory
- Uncaught Exceptions
- Infinite Loops
- Native Modules / Node.js / V8 Crashes

Live Demo

Future Features

Inspect Promise Objects

🌈 color output 🌈

JavaScript API

List retainers tree

JavaScript mode

Simpler Installation Process

Current Challenges

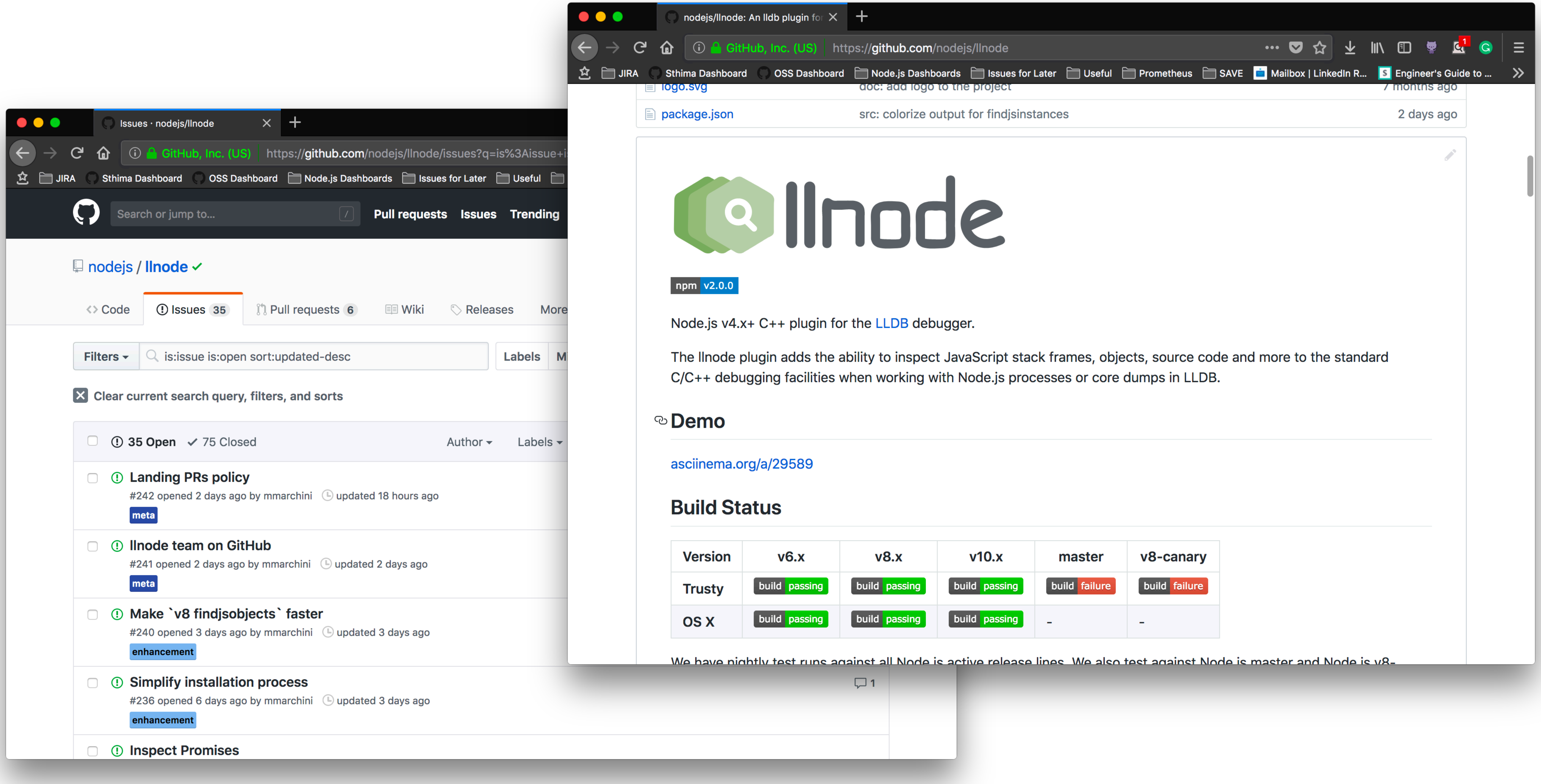
Postmortem with Promises

Keeping up with latest Node.js and V8 versions

Want to help the project?

Give Us Feedback

http://bit.ly/llufb



is:issue is:open sort:updated-desc

☒ Clear current search query, filters, and sorts

□ 35 Open ✓ 75 Closed

□ **Landing PRs policy**
#242 opened 2 days ago by mmarchini updated 18 hours ago
meta

□ **llnode team on GitHub**
#241 opened 2 days ago by mmarchini updated 2 days ago
meta

□ **Make `v8 findjsobjects` faster**
#240 opened 3 days ago by mmarchini updated 3 days ago
enhancement

□ **Simplify installation process** 1
#236 opened 6 days ago by mmarchini updated 3 days ago
enhancement

□ **Inspect Promises**

logo.svg doc: add logo to the project 7 months ago
package.json src: colorize output for findjsinstances 2 days ago

Node.js v4.x+ C++ plugin for the LLDB debugger.

The llnode plugin adds the ability to inspect JavaScript stack frames, objects, source code and more to the standard C/C++ debugging facilities when working with Node.js processes or core dumps in LLDB.

Demo

asciinema.org/a/29589

Build Status

Version	v6.x	v8.x	v10.x	master	v8-canary
Trusty	build passing	build passing	build passing	build failure	build failure
OS X	build passing	build passing	build passing	-	-

We have nightly test runs against all Node.js active release lines. We also test against Node.js master and Node.js v8-

https://github.com/nodejs/llnode

Node.js Collaborator Summit

Diagnostics Session

Oct. 12th - 3:30 PM
Vancouver Convention Centre West Building

Join us!

Questions?

Try it yourself!

```
docker run --rm --privileged --it mmarchini/llnode-101:latest
```

https://github.com/mmarchini/llnode-101