ChakraCore

e o inicio de uma uma nova era na comunidade Node.js

Mary Marchini

@mmarkini

sthima

Mary Marchini

Developers Team Leader @ **Sthima**
Node.js Collaborator
Node.js Diagnostics WG Member

@mmarkini

http://mmarchini.me/

oss@mmarchini.me

https://github.com/mmarchini/

O que é ChakraCore?

O que é Node.js?

8.8 Milhões

de instâncias rodando

47k

Estrelas no GitHub

3+ Bilhões

Pacotes baixados semanalmente no npm

Oito

Arquiteturas suportadas

1.500+

Contribuidores

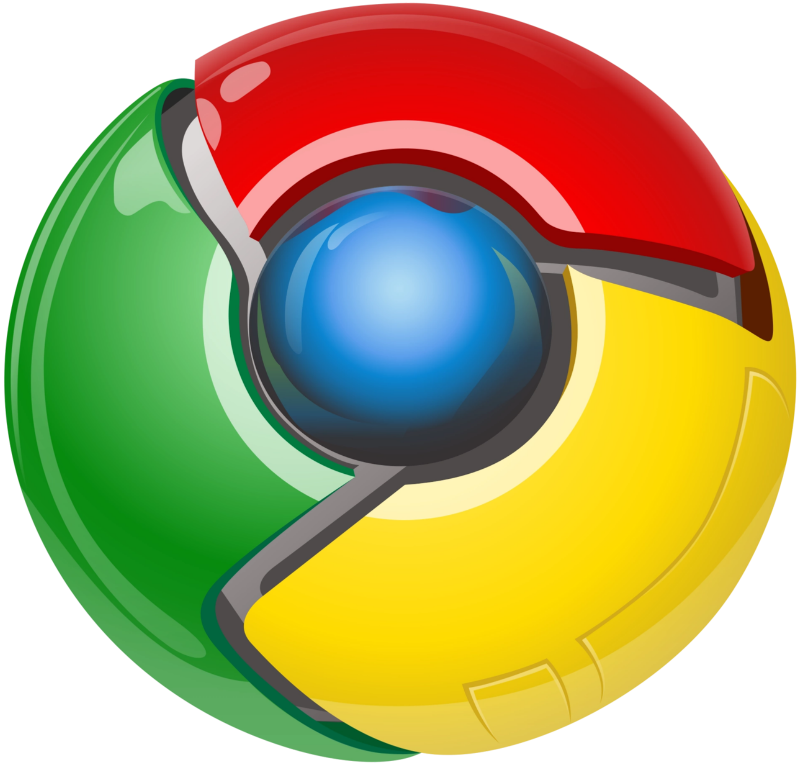
4800

Pacotes publicados semanalmente

Como tudo começou...

2008

2008



2008

V8

- V8 não era um interpretador
  - Ele era um compilador
  - Sempre gerava código de máquina
  - Performance decente para uma linguagem dinâmica na época

V8

- V8 não era um interpretador
  - Ele era um compilador
  - Sempre gerava código de máquina
  - Performance decente para uma linguagem dinâmica na época
- Open-source

V8

- V8 não era um interpretador
  - Ele era um compilador
  - Sempre gerava código de máquina
  - Performance decente para uma linguagem dinâmica na época
- Open-source
- Embarcável

2009

2009

2009

2009

JSConfEU 2009 - Ryan Dahl apresenta Node.js pela primeira vez

2009

JSConfEU 2009 - Ryan Dahl apresenta Node.js pela primeira vez

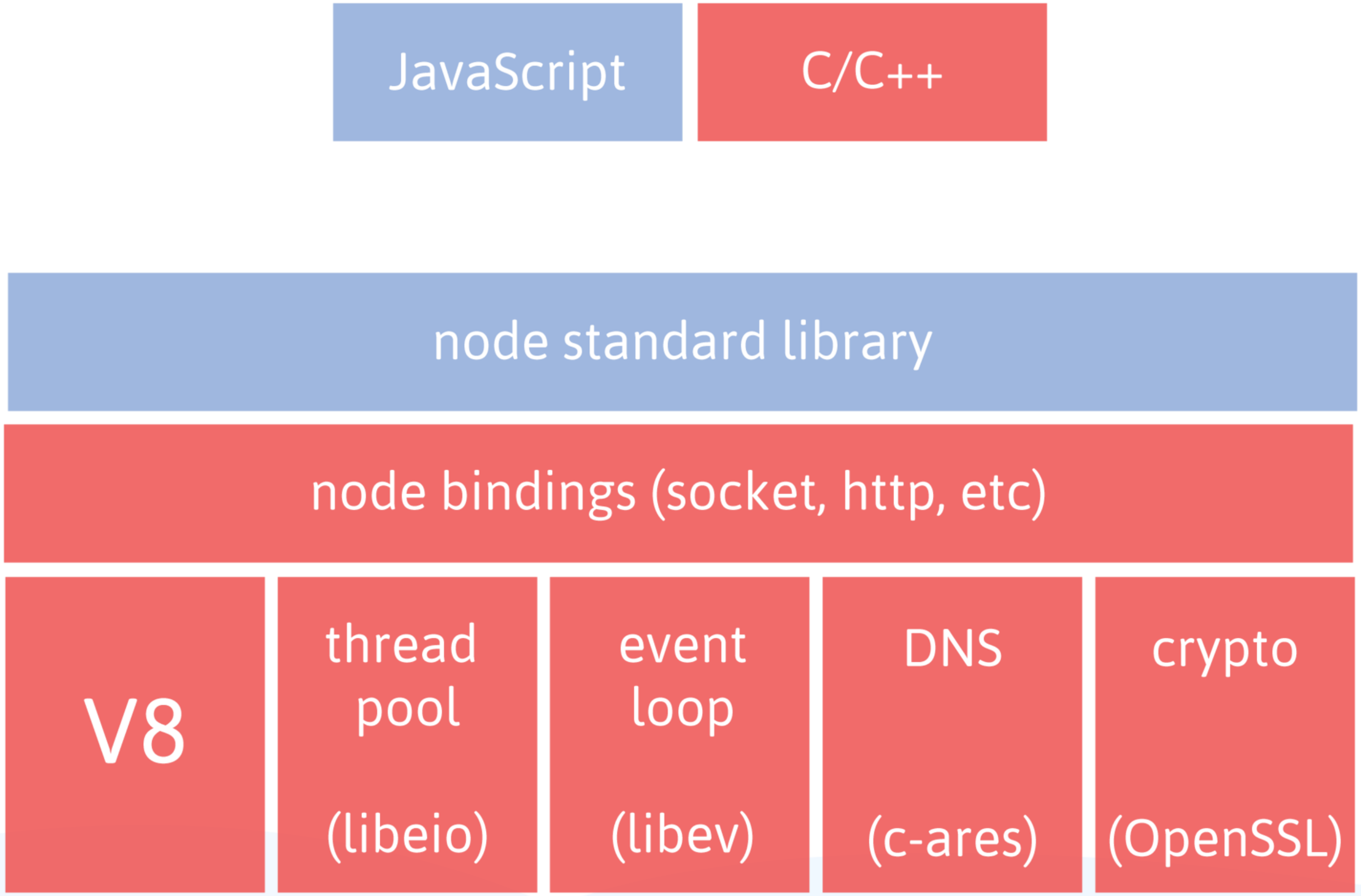
“ I/O has to be done differently. We're doing it wrong

Ryan Dahl, JSConfEU 2009

Node.js

- Javascript para servidor
- Criado em cima do Google V8
- Evented, non-blocking I/O. Parecido com EventMachine e Twisted
- Sistema de módulos CommonJS
- 8000 linhas de C/C++. 2000 linhas de Javascript. 14 contribuidores

Arquitetura Node.js (2009)



JavaScript

C/C++

node standard library

node bindings (socket, http, etc)

V8

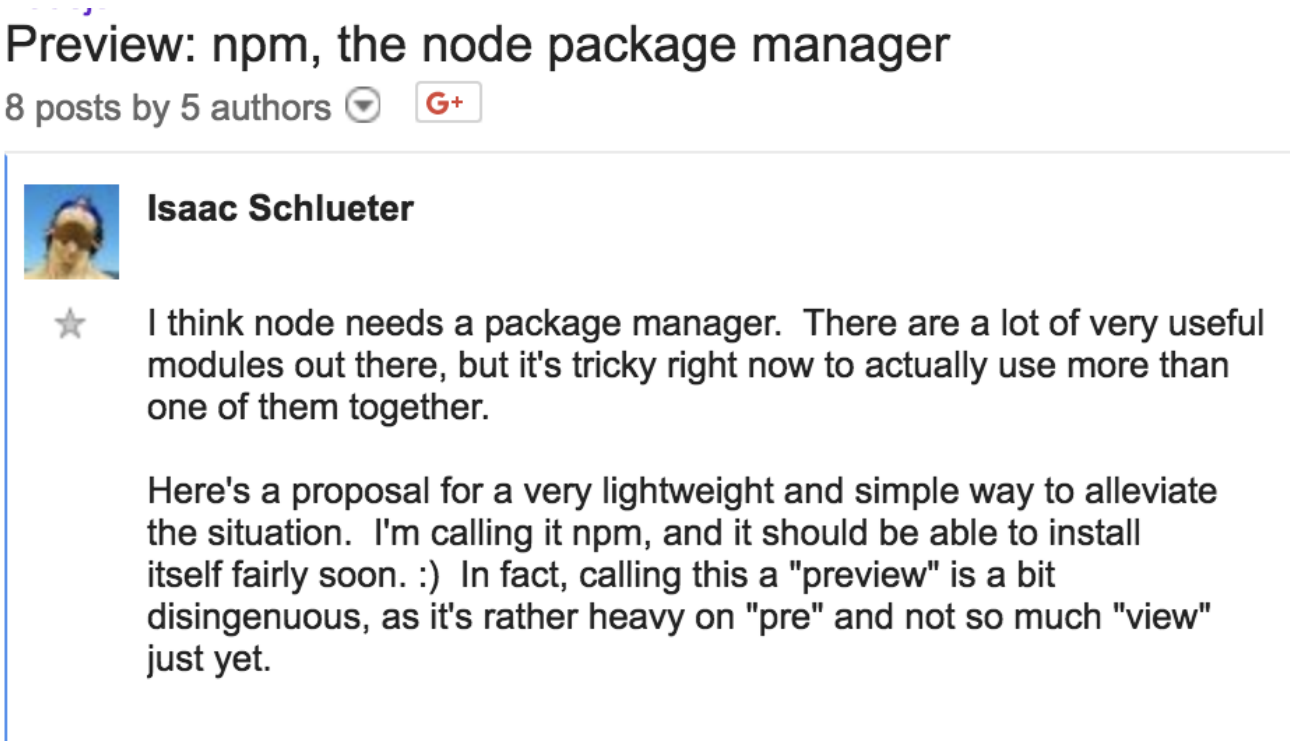
thread pool (libeio)

event loop (libev)

DNS (c-ares)

crypto (OpenSSL)

2009



Preview: npm, the node package manager

8 posts by 5 authors

Isaac Schlueter

I think node needs a package manager. There are a lot of very useful modules out there, but it's tricky right now to actually use more than one of them together.

Here's a proposal for a very lightweight and simple way to alleviate the situation. I'm calling it npm, and it should be able to install itself fairly soon. :) In fact, calling this a "preview" is a bit disingenuous, as it's rather heavy on "pre" and not so much "view" just yet.

Isaac Schlueter propõe um gerenciador de pacotes para o Node.js

2010

Node.js

- Node.js 0.2.0
- Surgem alguns pacotes
    - Express
    - Socket.io

2010

V8

Novo compilador: Crankshaft

2011

1.0 Released

2011

Adoção do Node.js

2012

Node.js

- Node.js 0.8.0 (estável)
- Mudança na liderança
  - Ryan Dahl -> Isaac
- Node.js começa a focar no seu ecossistema

2013

Node.js

- Ghost Blog Platform
- MEAN Stack

2013

Adoção do Node.js

PayPal

Walmart

ebay

2014

Node.js

- Surge um fork do Node.js

2014

2014

Adoção do Node.js

NETFLIX

2015

node
JS®
Foundation

2015

node JS + io = Node.js 4.0!

2015

Node.js 4.2.0
LTS

2015

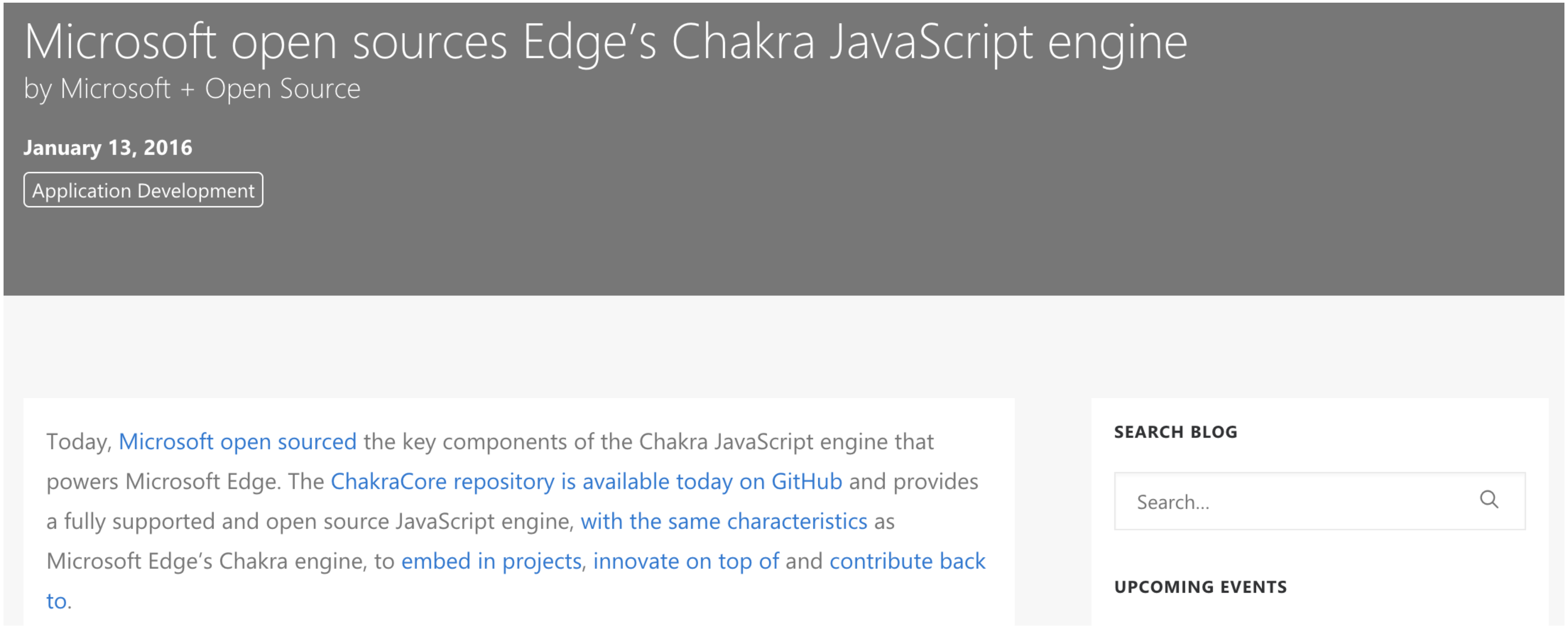
V8 novo compilador: Turbofan

2016

V8 Lança um interpretador: Ignition

Otimizado para consumo de memória em dispositivos móveis

2016



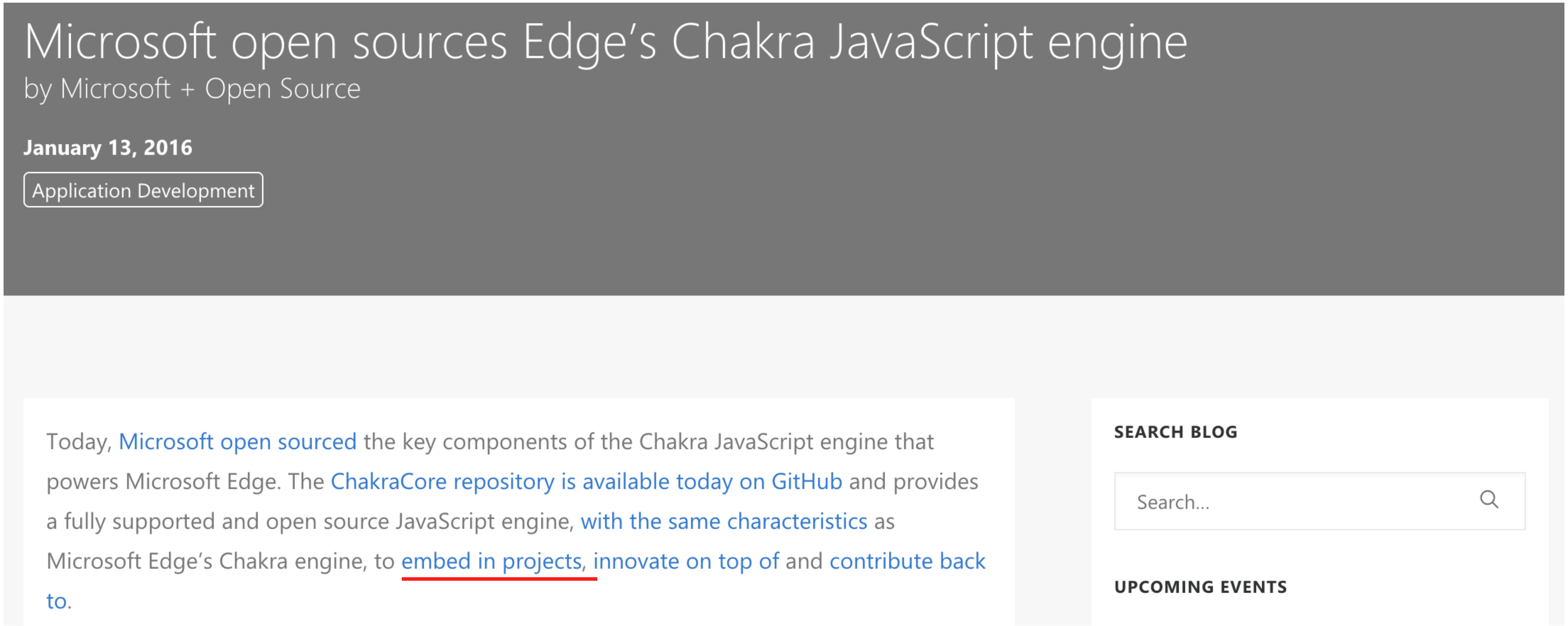
Microsoft open sources Edge's Chakra JavaScript engine

by Microsoft + Open Source

January 13, 2016

Today, Microsoft open sourced the key components of the Chakra JavaScript engine that powers Microsoft Edge. The ChakraCore repository is available today on GitHub and provides a fully supported and open source JavaScript engine, with the same characteristics as Microsoft Edge's Chakra engine, to embed in projects, innovate on top of and contribute back to.

2016



Microsoft open sources Edge's Chakra JavaScript engine

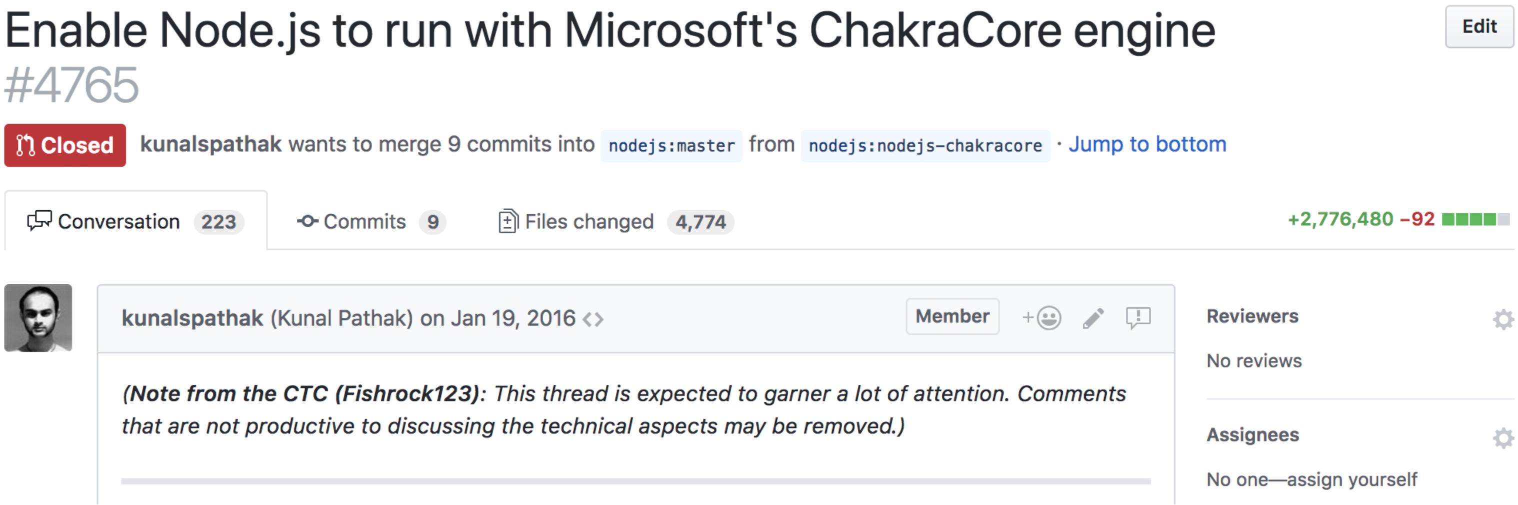
by Microsoft + Open Source

January 13, 2016

Application Development

Today, Microsoft open sourced the key components of the Chakra JavaScript engine that powers Microsoft Edge. The ChakraCore repository is available today on GitHub and provides a fully supported and open source JavaScript engine, with the same characteristics as Microsoft Edge's Chakra engine, to embed in projects, innovate on top of and contribute back to.

2016



Enable Node.js to run with Microsoft's ChakraCore engine #4765

Closed **kunalspathak** wants to merge 9 commits into `nodejs:master` from `nodejs:nodejs-chakracore` · Jump to bottom

+2,776,480 −92

kunalspathak (Kunal Pathak) on Jan 19, 2016

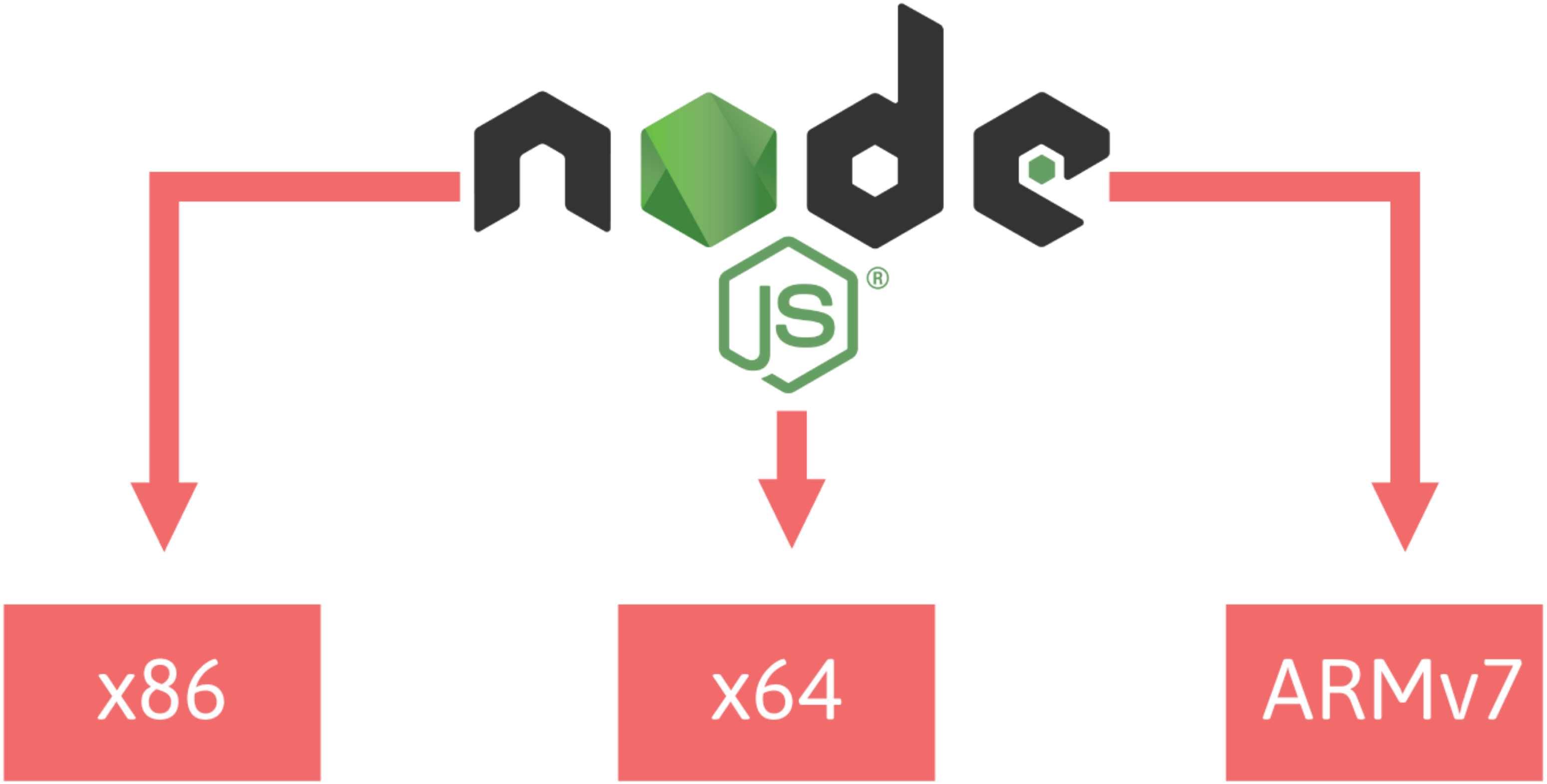
*(**Note from the CTC (Fishrock123)**: This thread is expected to garner a lot of attention. Comments that are not productive to discussing the technical aspects may be removed.)*

No reviews

No one—assign yourself

Por que?

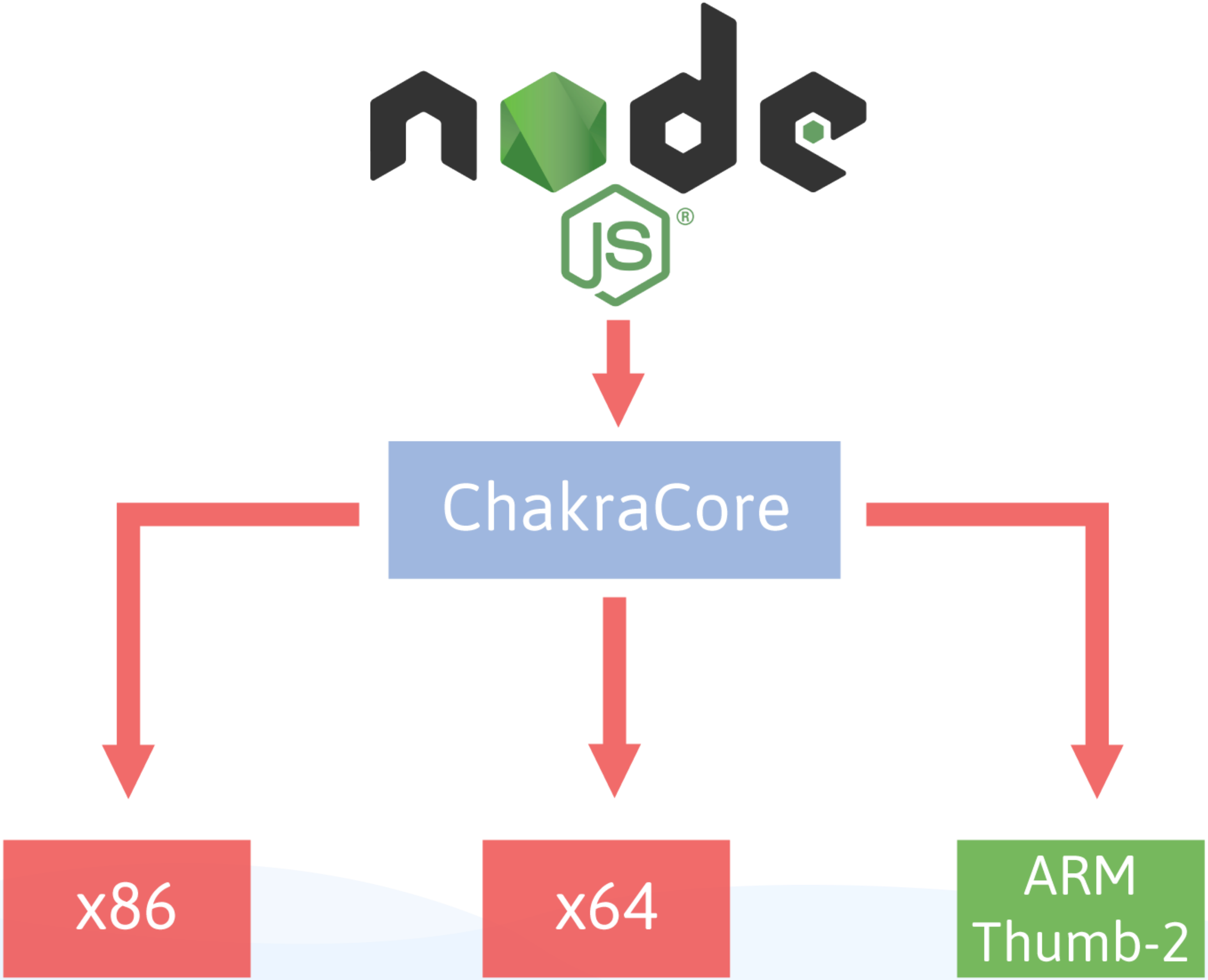
IoT
Ecosystem
GEO



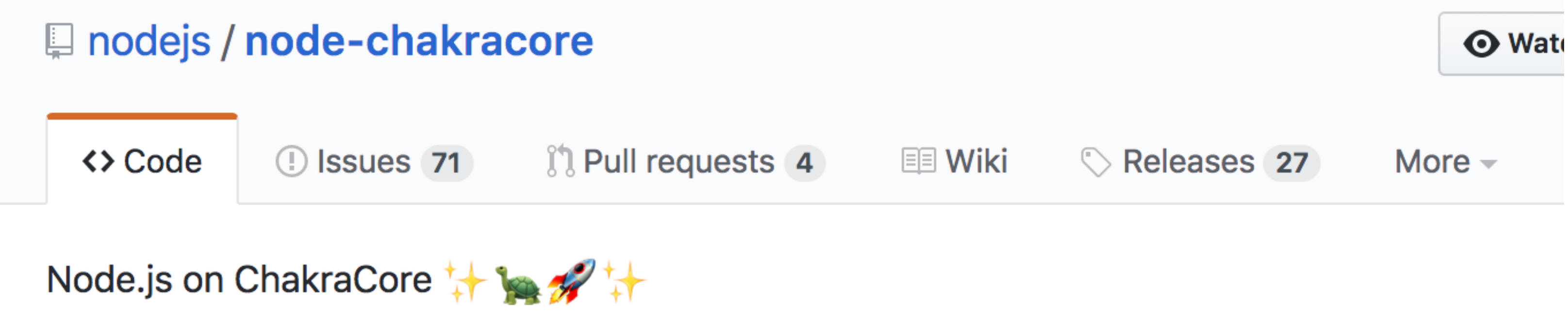
x86

x64

ARMv7


ChakraCore
x86
x64
ARM Thumb-2

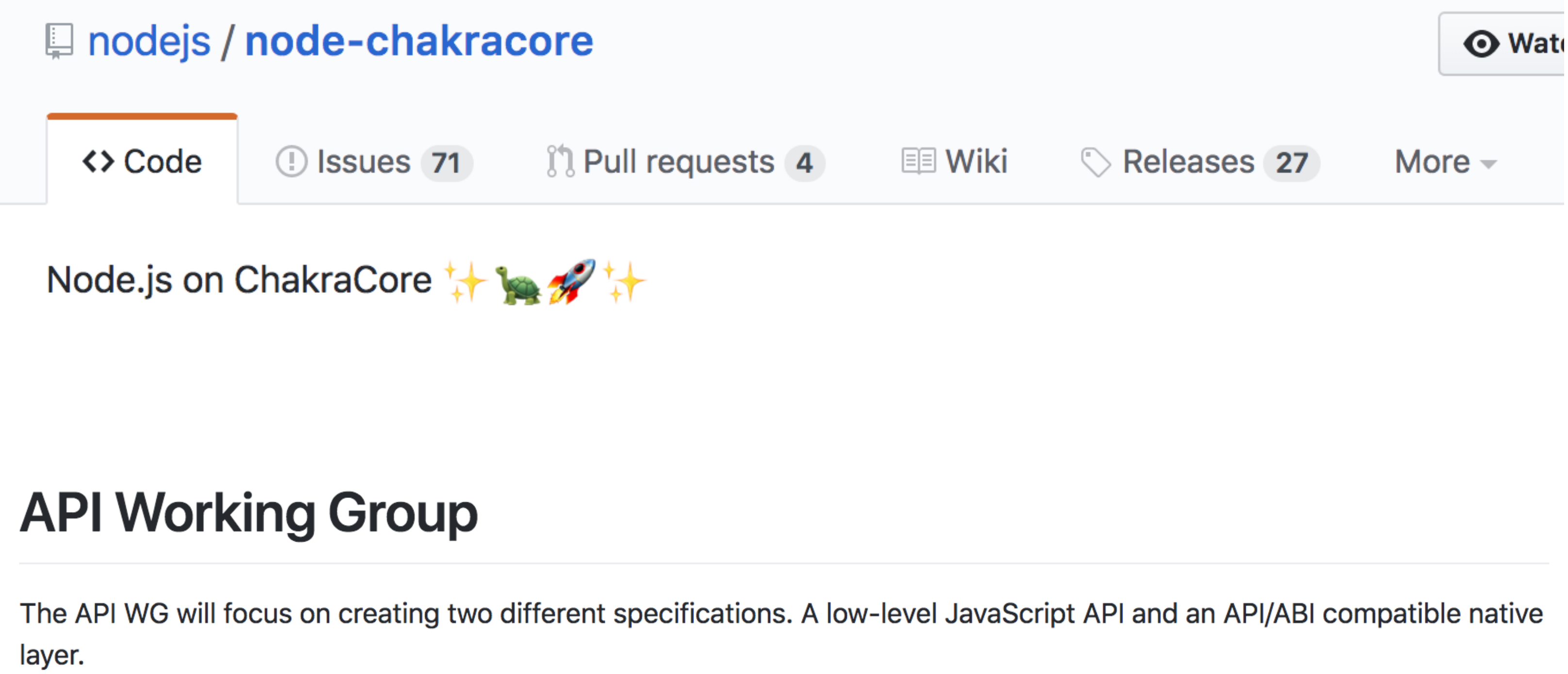
2016



nodejs / node-chakracore

Node.js on ChakraCore ✨🐢🚀✨

2016



Node.js on ChakraCore ✨🐢🚀✨

API Working Group

The API WG will focus on creating two different specifications. A low-level JavaScript API and an API/ABI compatible native layer.

N-API

- ABI (Application Binary Interface) Stable
- VM neutral

N-API

Um módulo nativo (C++) pode ser utilizado:

- Node.js v6.x
- Node.js v8.x
- Node.js v9.x
- Node.js v10.x
- ...
- node-chakracore (v8.x, v9.x, ...)

Sem recompilar!

N-API

Principais Colaboradores

Google
nearForm
Infrastructure for innovation
Microsoft
intel®
IBM
N|S
NODESOURCE™

N-API Current Status

- Saiu da versão experimental a alguns meses
  - Com exceção v6.x
- WG está colaborando com criadores de módulos nativos para atualizar para N-API

Time-Travel Debugging

Time-Travel Debugging

- Salva um *trace* da execução do programa

Time-Travel Debugging

- Salva um *trace* da execução do programa
- Um *trace* é uma sequência de passos que o programa realizou durante a sua execução

Time-Travel Debugging

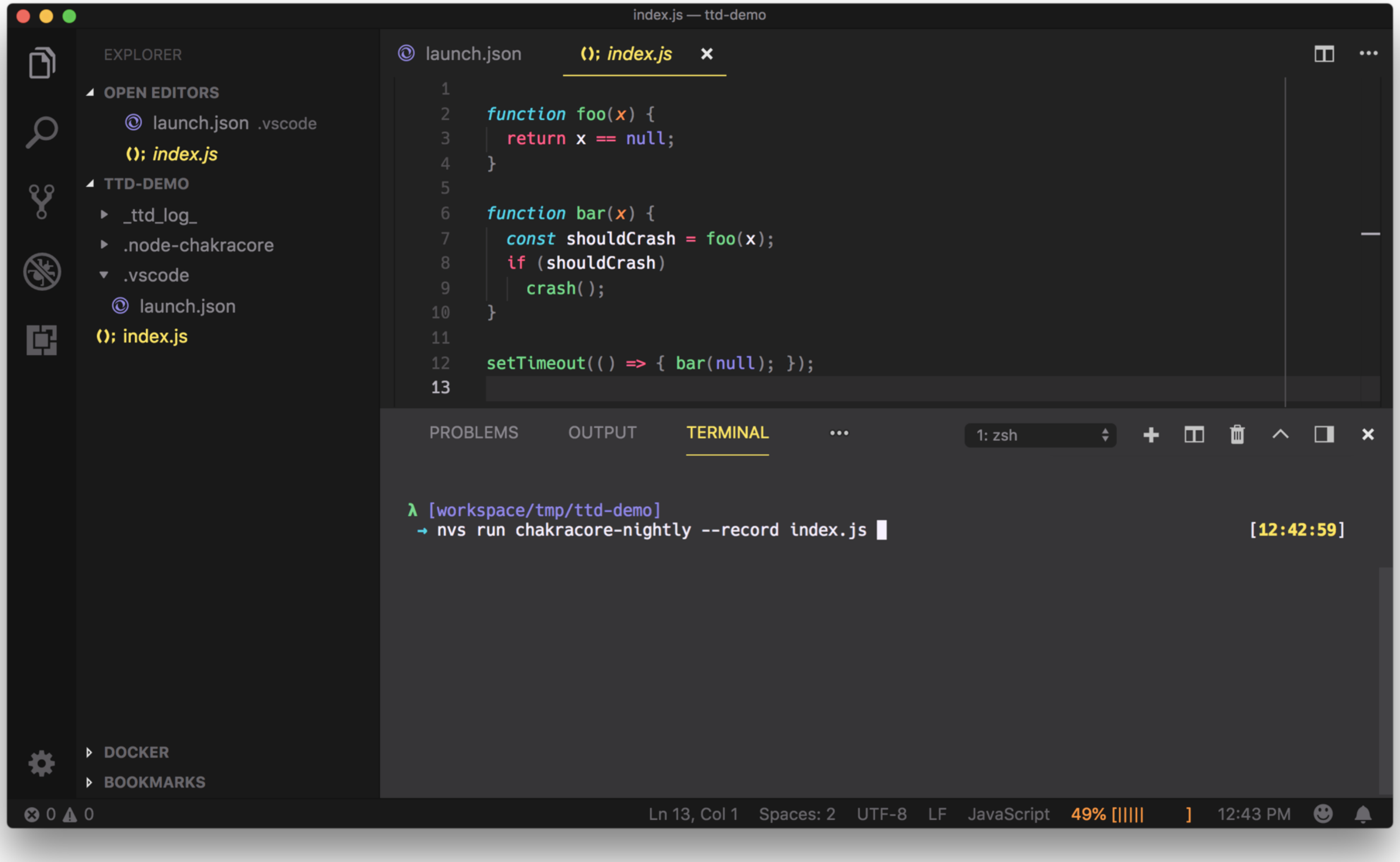
- Salva um *trace* da execução do programa
- Um *trace* é uma sequência de passos que o programa realizou durante a sua execução
- Permite reproduzir esses passos posteriormente, sem efeitos colaterais e com a possibilidade de " *voltar no tempo*"

Time-Travel Debugging

- Salva um *trace* da execução do programa
- Um *trace* é uma sequência de passos que o programa realizou durante a sua execução
- Permite reproduzir esses passos posteriormente, sem efeitos colaterais e com a possibilidade de " *voltar no tempo*"

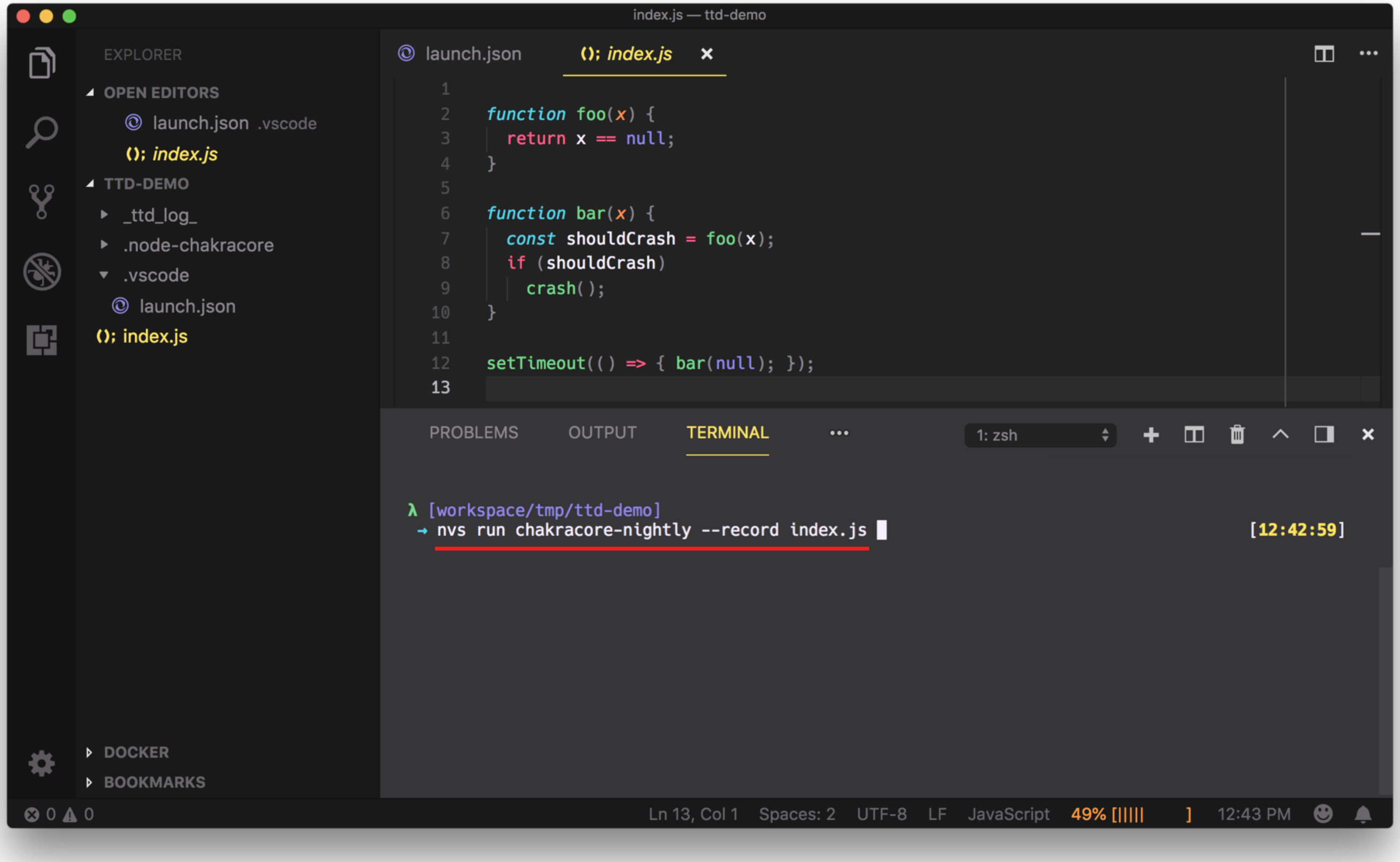
Time-Travel Debugging

Demo



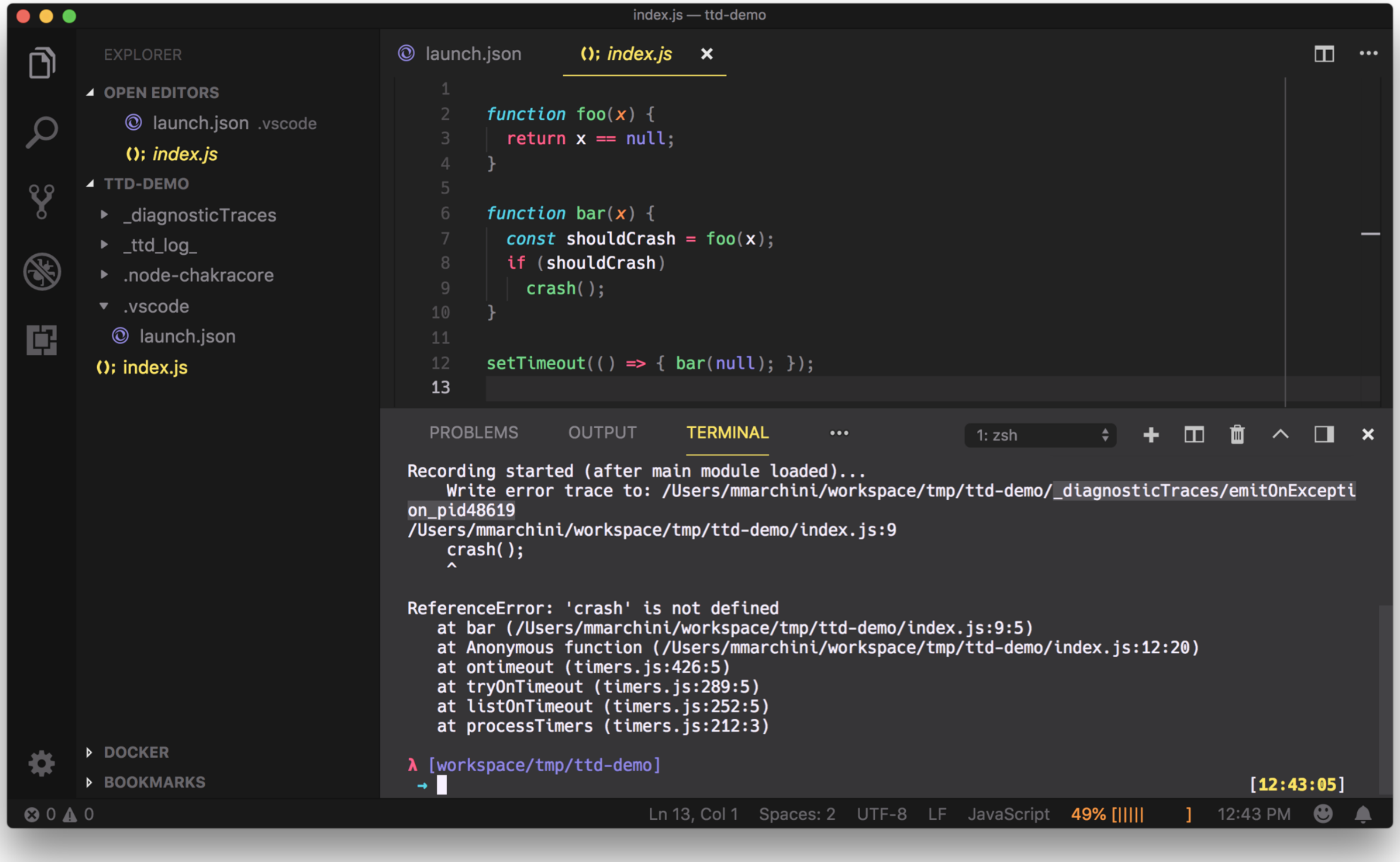
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

```
λ [workspace/tmp/ttd-demo]
 → nvs run chakracore-nightly --record index.js                    [12:42:59]
```



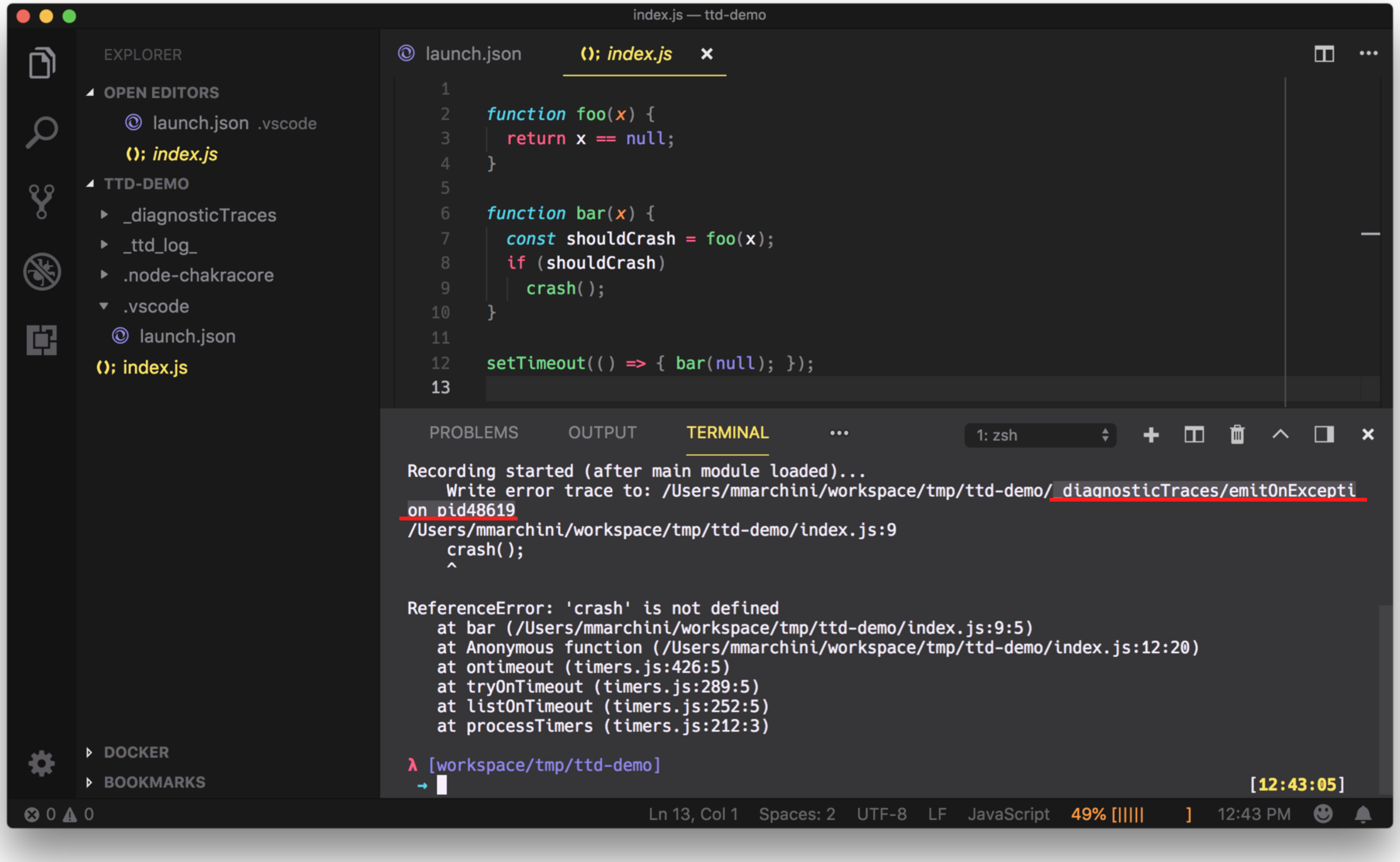
```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

```
λ [workspace/tmp/ttd-demo]
 → nvs run chakracore-nightly --record index.js                     [12:42:59]
```



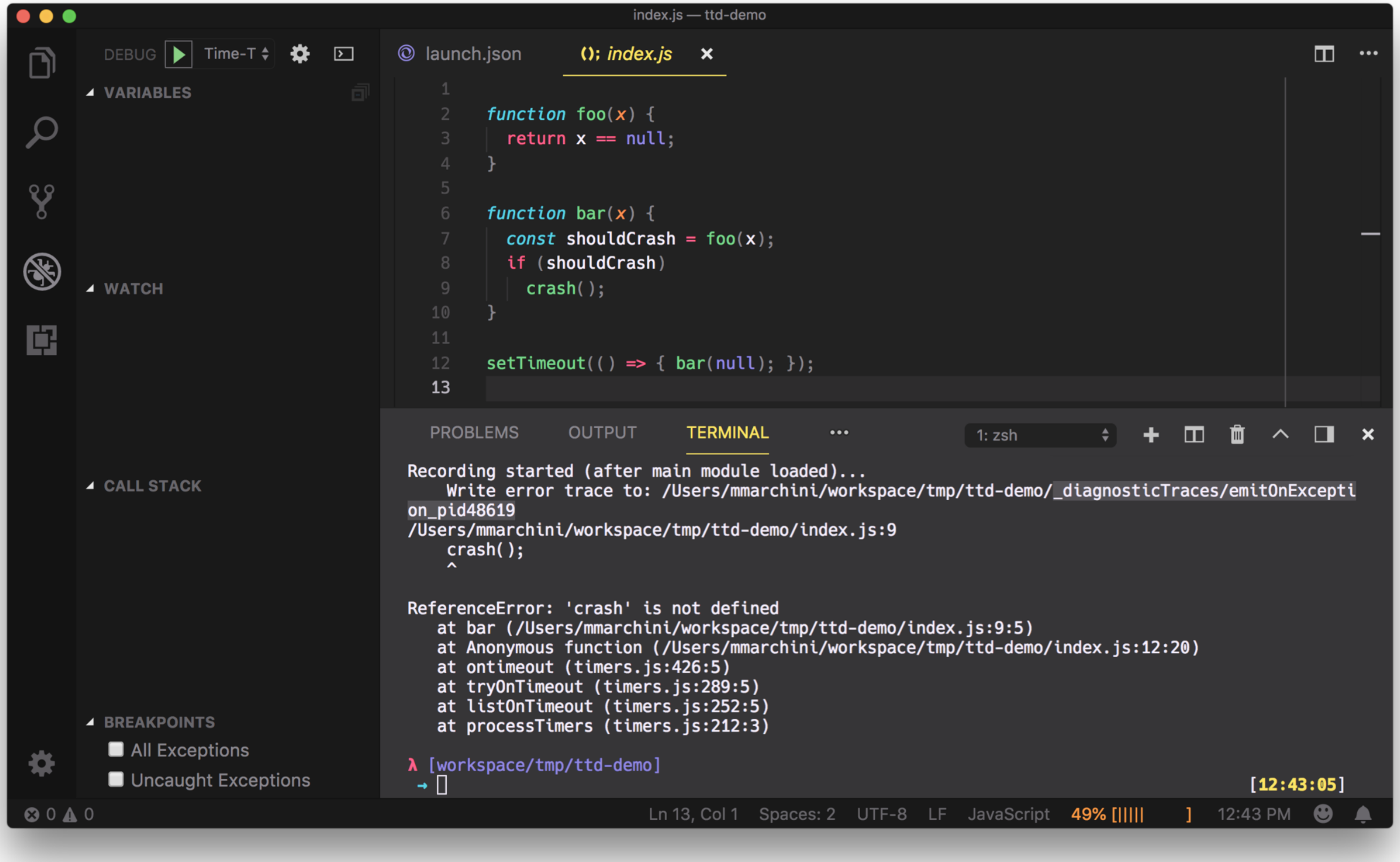
```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                   [12:43:05]
```



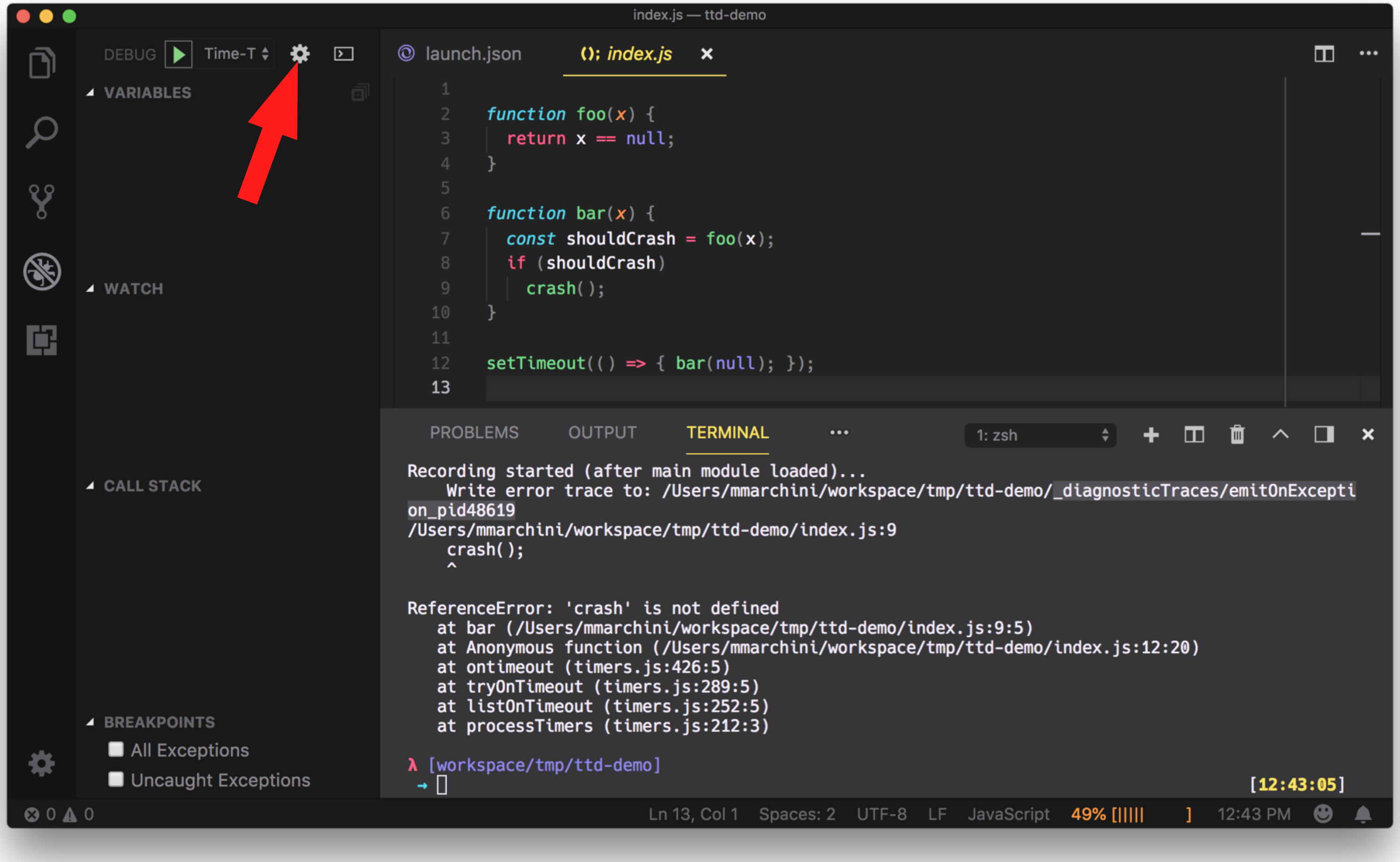
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                      [12:43:05]
```



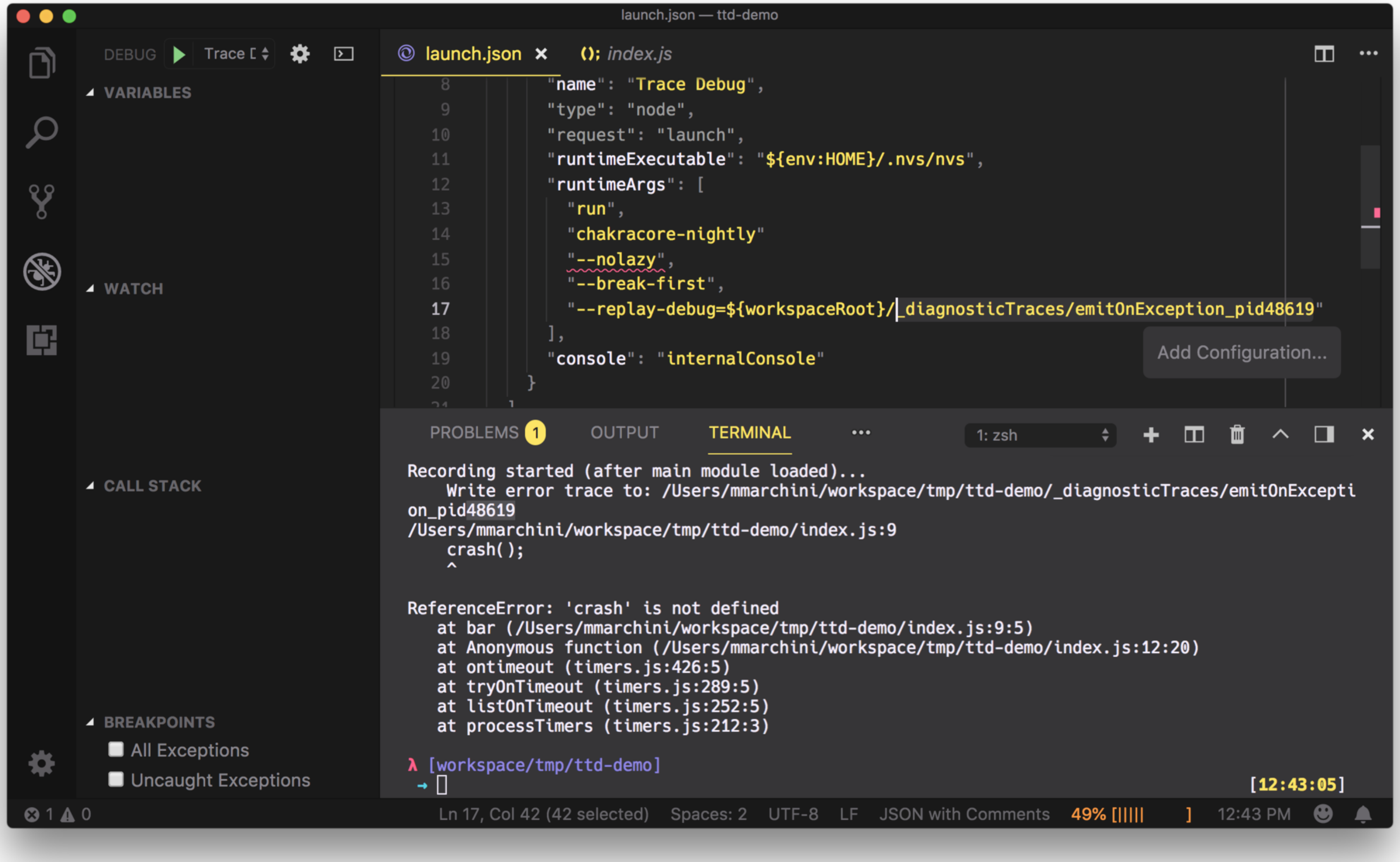
```javascript

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                     [12:43:05]
```



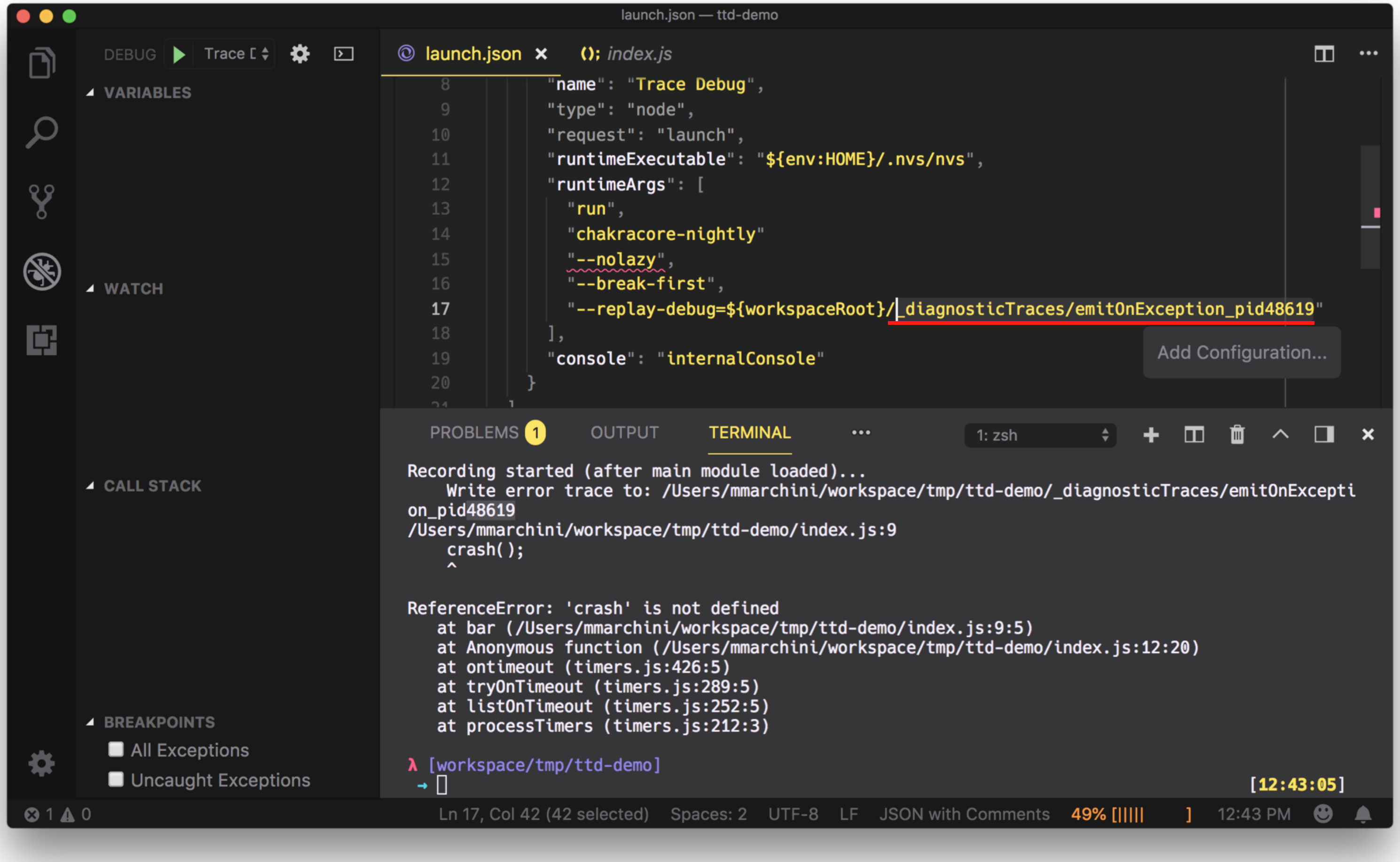
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                   [12:43:05]
```



```
        "name": "Trace Debug",
        "type": "node",
        "request": "launch",
        "runtimeExecutable": "${env:HOME}/.nvs/nvs",
        "runtimeArgs": [
          "run",
          "chakracore-nightly"
          "--nolazy",
          "--break-first",
          "--replay-debug=${workspaceRoot}/_diagnosticTraces/emitOnException_pid48619"
        ],
        "console": "internalConsole"
      }
```

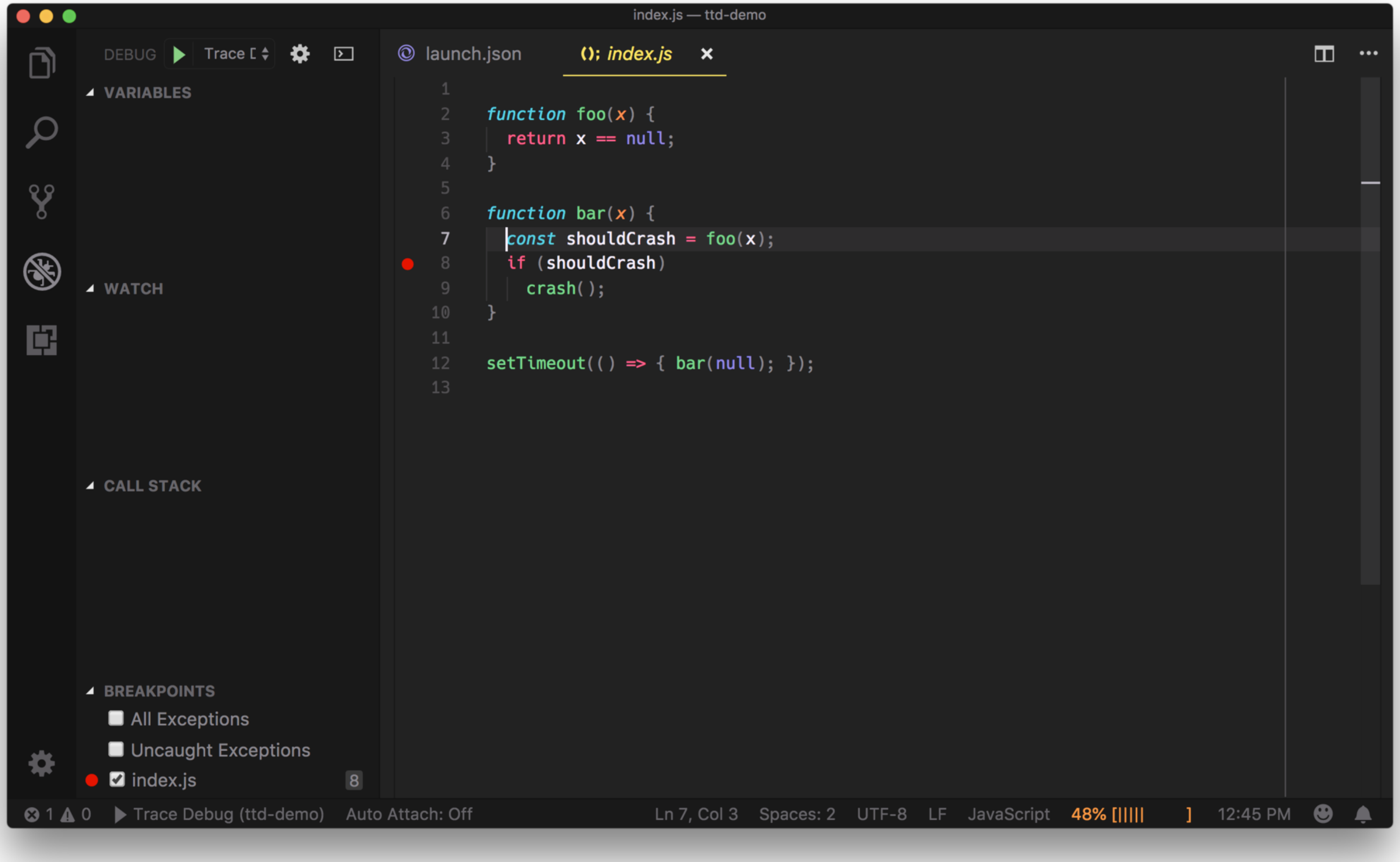
```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                    [12:43:05]
```



```
        "name": "Trace Debug",
        "type": "node",
        "request": "launch",
        "runtimeExecutable": "${env:HOME}/.nvs/nvs",
        "runtimeArgs": [
          "run",
          "chakracore-nightly"
          "--nolazy",
          "--break-first",
          "--replay-debug=${workspaceRoot}/_diagnosticTraces/emitOnException_pid48619"
        ],
        "console": "internalConsole"
      }
```

```
Recording started (after main module loaded)...
    Write error trace to: /Users/mmarchini/workspace/tmp/ttd-demo/_diagnosticTraces/emitOnExcepti
on_pid48619
/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9
    crash();
    ^

ReferenceError: 'crash' is not defined
   at bar (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:9:5)
   at Anonymous function (/Users/mmarchini/workspace/tmp/ttd-demo/index.js:12:20)
   at ontimeout (timers.js:426:5)
   at tryOnTimeout (timers.js:289:5)
   at listOnTimeout (timers.js:252:5)
   at processTimers (timers.js:212:3)

λ [workspace/tmp/ttd-demo]
 →                                                                                    [12:43:05]
```

Ln 17, Col 42 (42 selected) Spaces: 2 UTF-8 LF JSON with Comments 49% [|||||] 12:43 PM

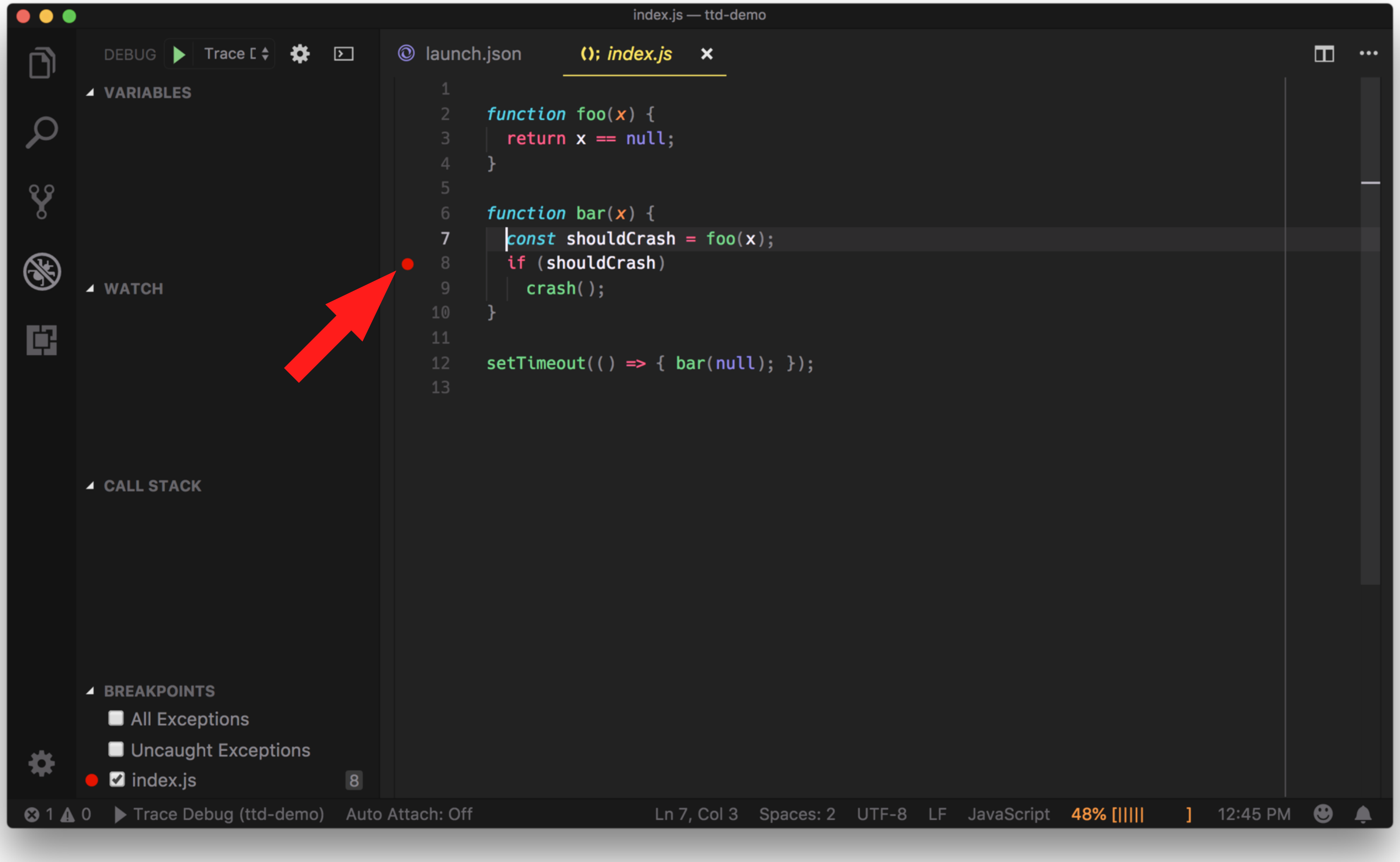


```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

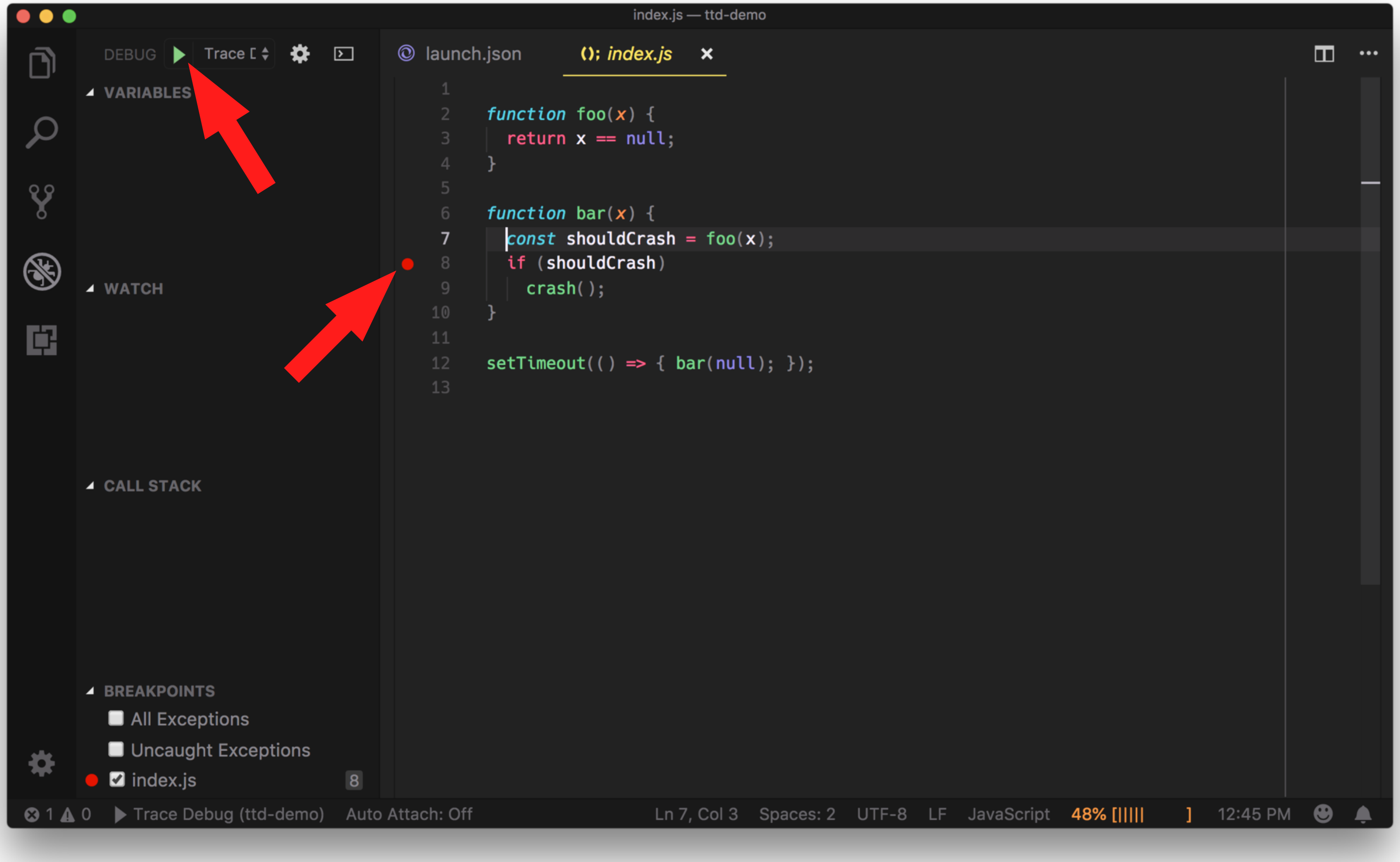
□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 8



```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

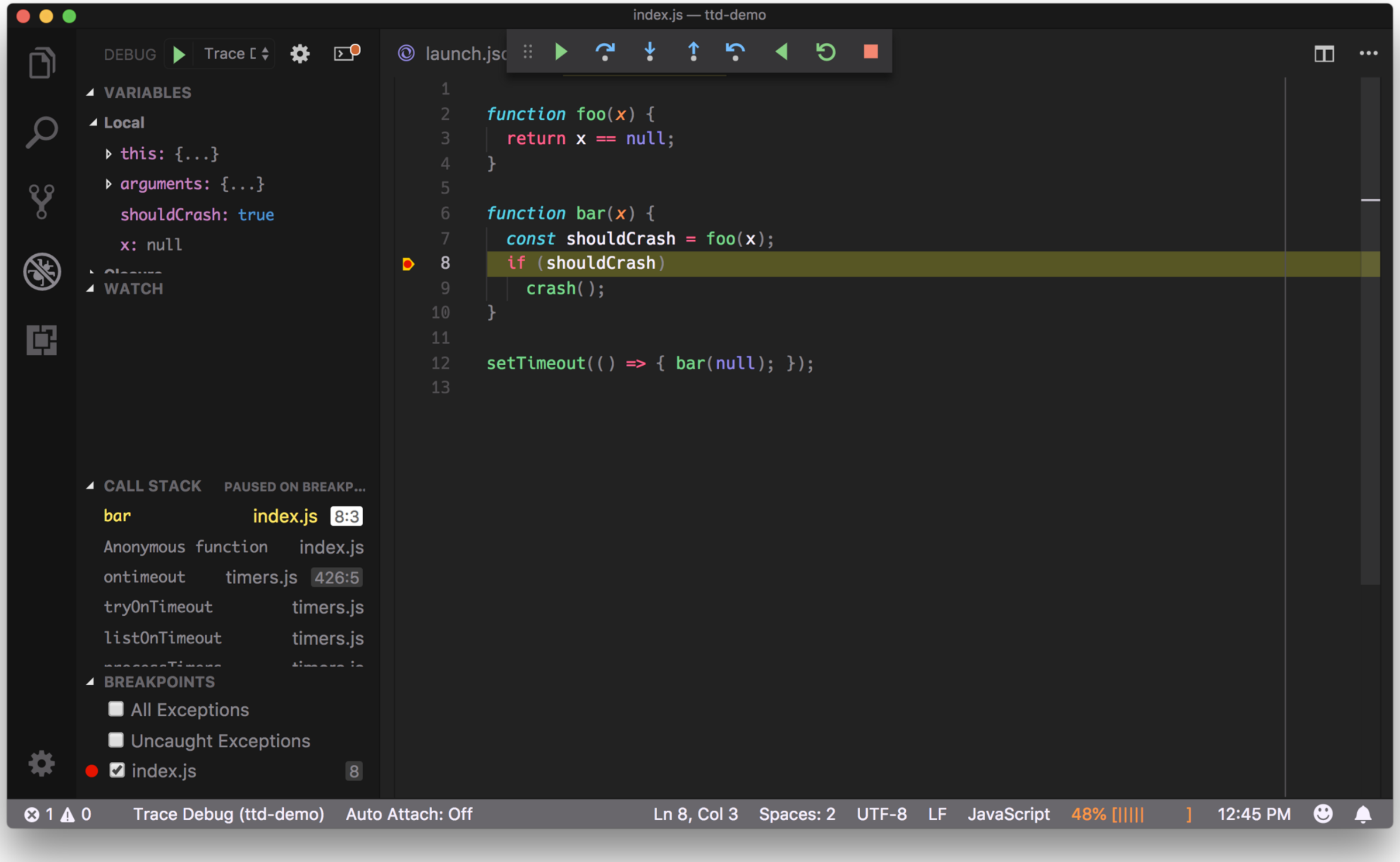


```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

☐ All Exceptions

☐ Uncaught Exceptions

● ☑ index.js 8



VARIABLES

Local

- this: {...}
- arguments: {...}
- shouldCrash: true
- x: null

WATCH

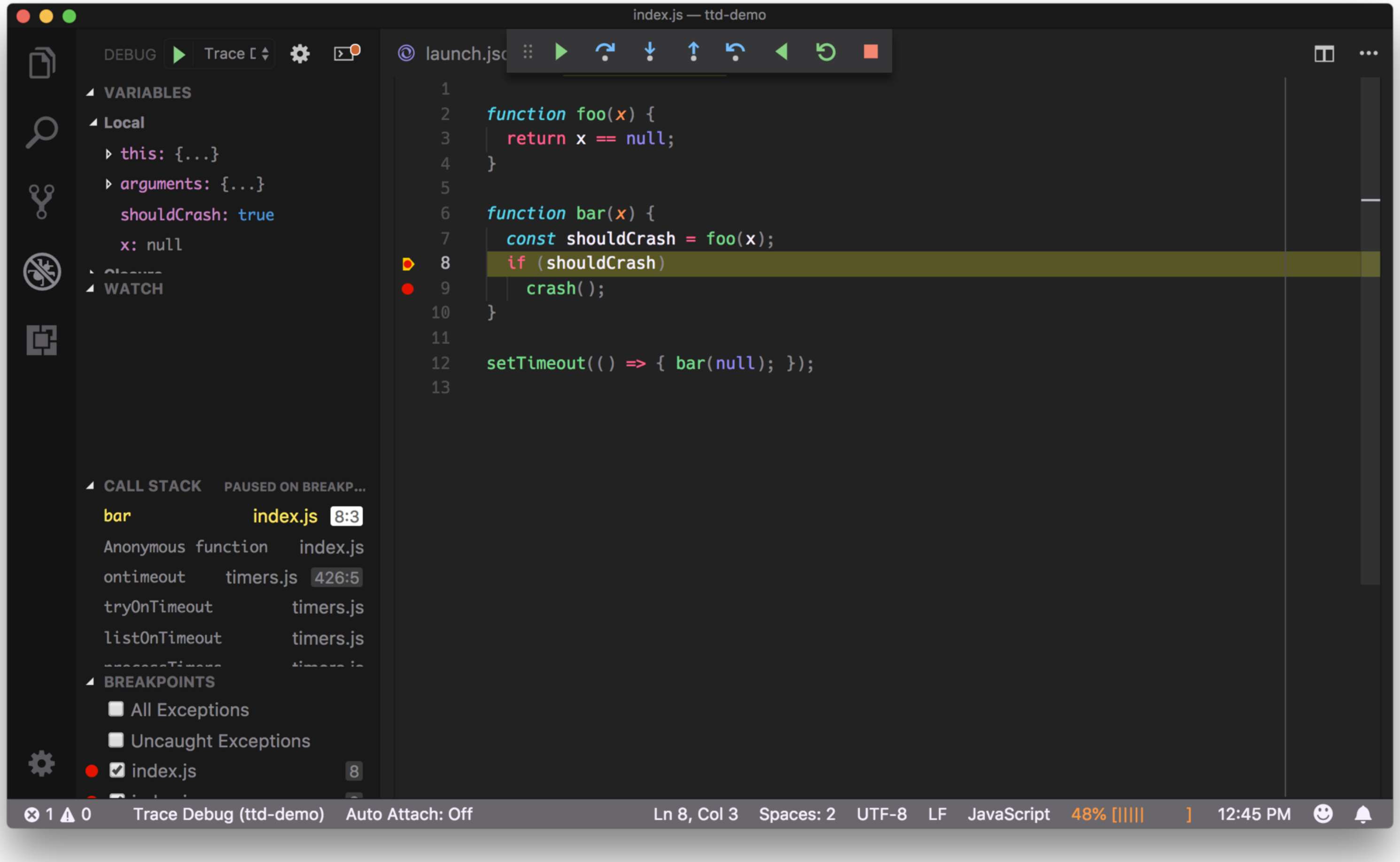
CALL STACK — PAUSED ON BREAKP...

```
bar                  index.js  8:3
Anonymous function   index.js
ontimeout            timers.js 426:5
tryOnTimeout         timers.js
listOnTimeout        timers.js
```

BREAKPOINTS

- ☐ All Exceptions
- ☐ Uncaught Exceptions
- ● ☑ index.js 8

```javascript

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```



index.js — ttd-demo

DEBUG ▶ Trace [

launch.jsc

```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

VARIABLES

Local

- this: {...}
- arguments: {...}
- shouldCrash: true
- x: null

WATCH

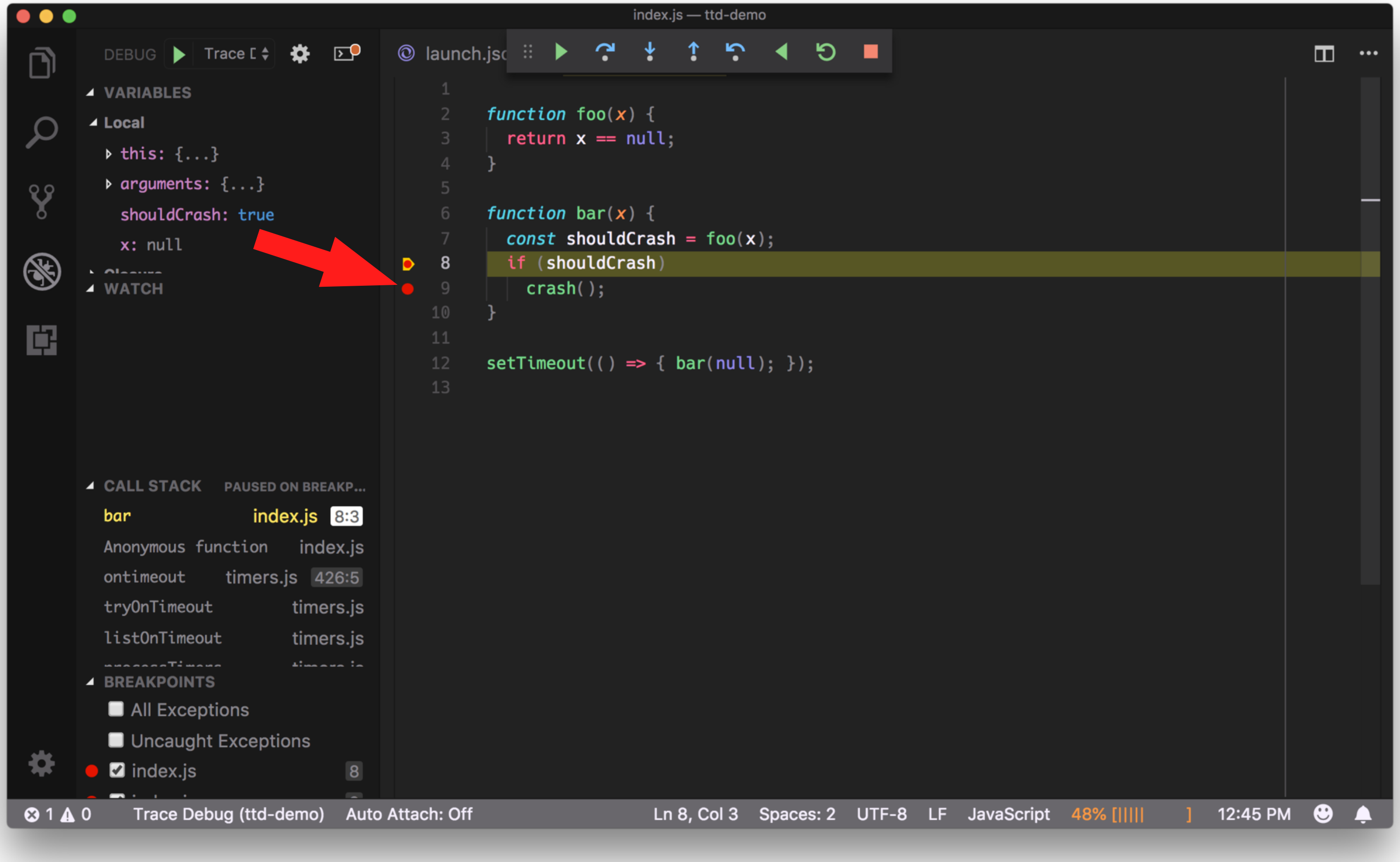
CALL STACK PAUSED ON BREAKP...

bar	index.js	8:3
Anonymous function	index.js	
ontimeout	timers.js	426:5
tryOnTimeout	timers.js	
listOnTimeout	timers.js	

BREAKPOINTS

- □ All Exceptions
- □ Uncaught Exceptions
- ● ☑ index.js 8

ⓧ 1 ⚠ 0 Trace Debug (ttd-demo) Auto Attach: Off Ln 8, Col 3 Spaces: 2 UTF-8 LF JavaScript 48% [|||||] 12:45 PM



VARIABLES

Local

- this: {...}
- arguments: {...}
- shouldCrash: true
- x: null

WATCH

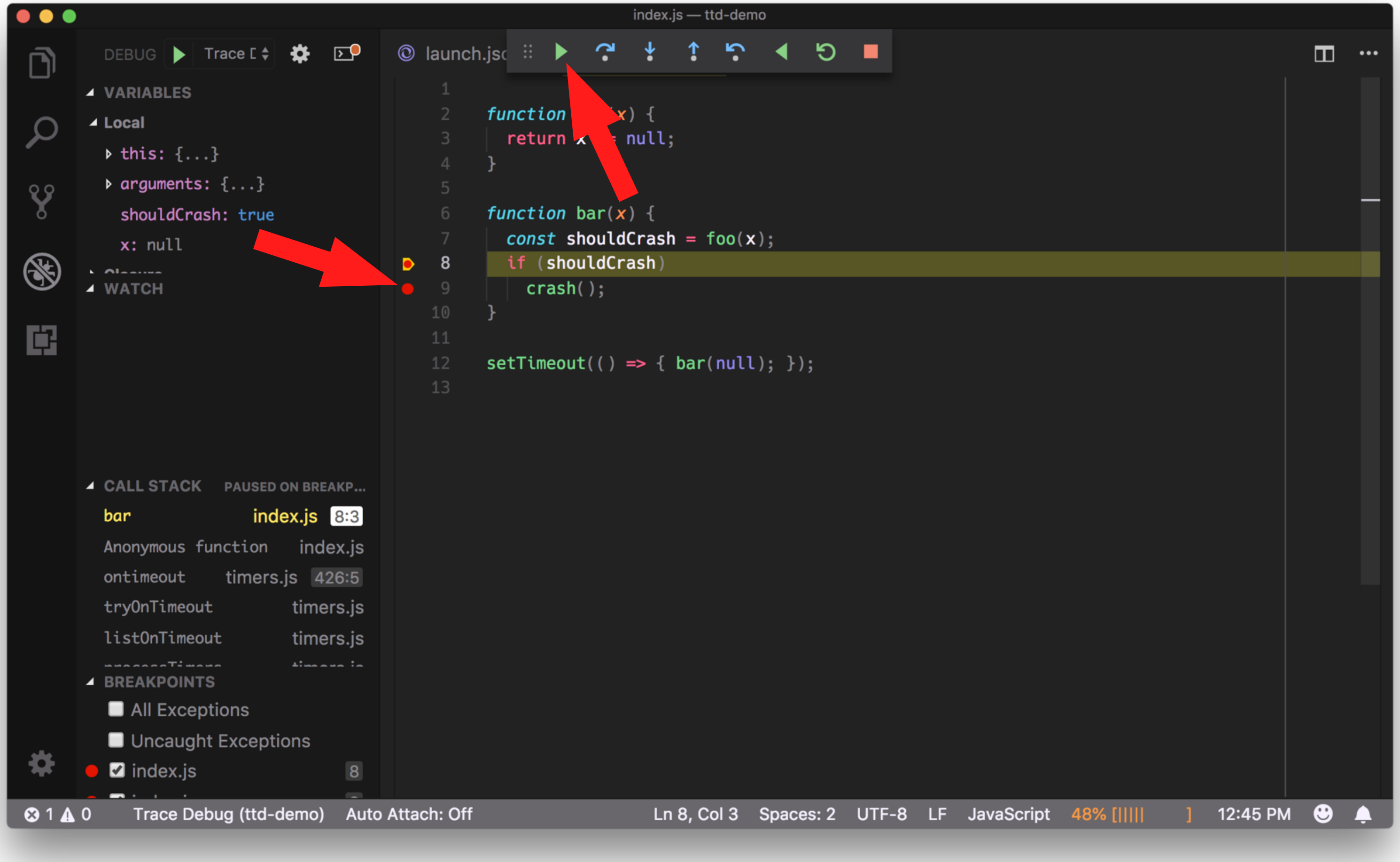
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

CALL STACK PAUSED ON BREAKP...

bar	index.js	8:3
Anonymous function	index.js	
ontimeout	timers.js	426:5
tryOnTimeout	timers.js	
listOnTimeout	timers.js	

BREAKPOINTS

- □ All Exceptions
- □ Uncaught Exceptions
- ● ☑ index.js 8



index.js — ttd-demo

VARIABLES

Local

```
this: {...}
arguments: {...}
shouldCrash: true
x: null
```

WATCH

CALL STACK   PAUSED ON BREAKP...

```
bar                    index.js  8:3
Anonymous function     index.js
ontimeout              timers.js  426:5
tryOnTimeout           timers.js
listOnTimeout          timers.js
```

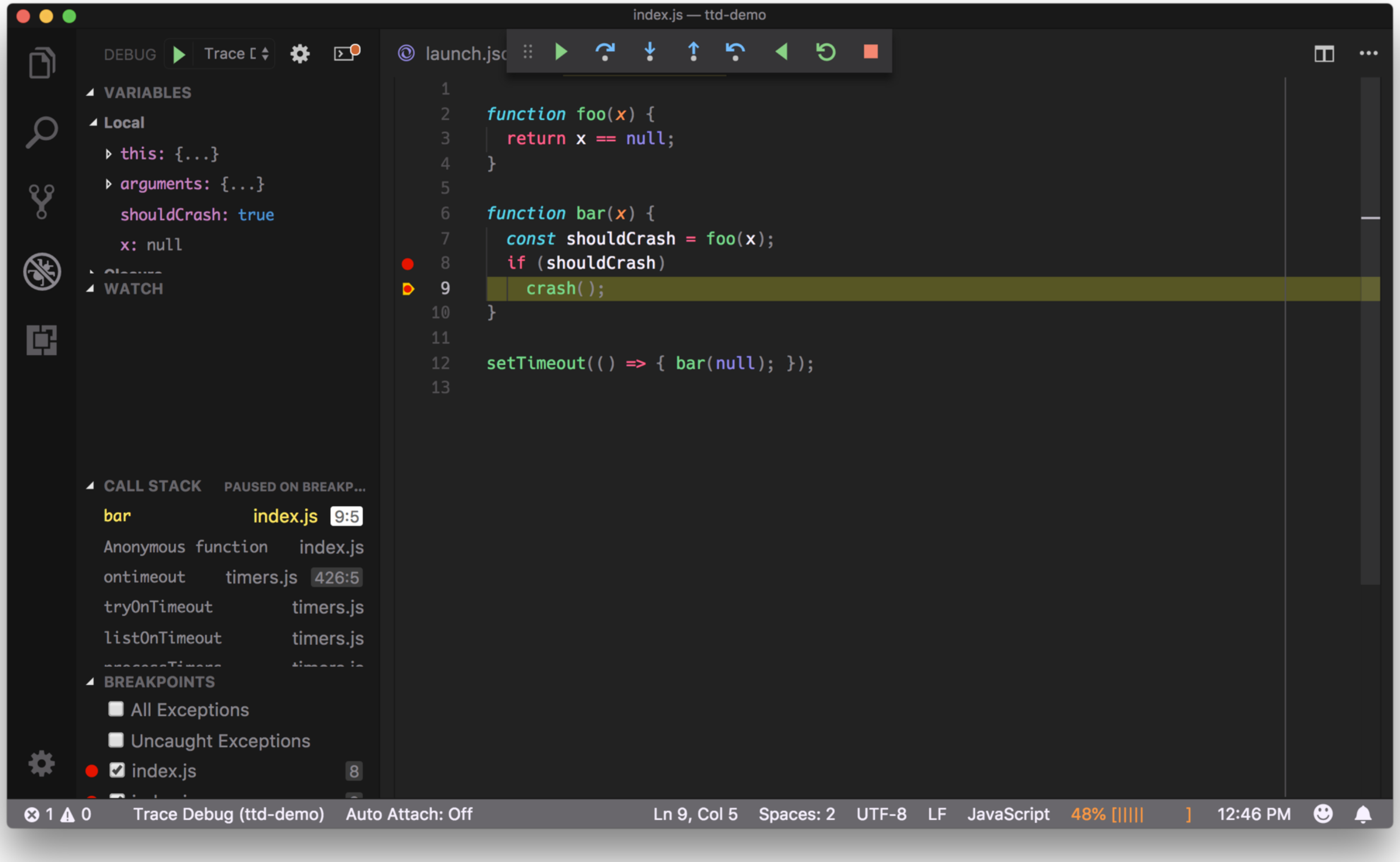
BREAKPOINTS

□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 8

```

function     x) {
  return x   null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```



```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

VARIABLES

Local

this: {...}

arguments: {...}

shouldCrash: true

x: null

WATCH

CALL STACK PAUSED ON BREAKP...

```
bar                     index.js  9:5
Anonymous function      index.js
ontimeout               timers.js 426:5
tryOnTimeout            timers.js
listOnTimeout           timers.js
```

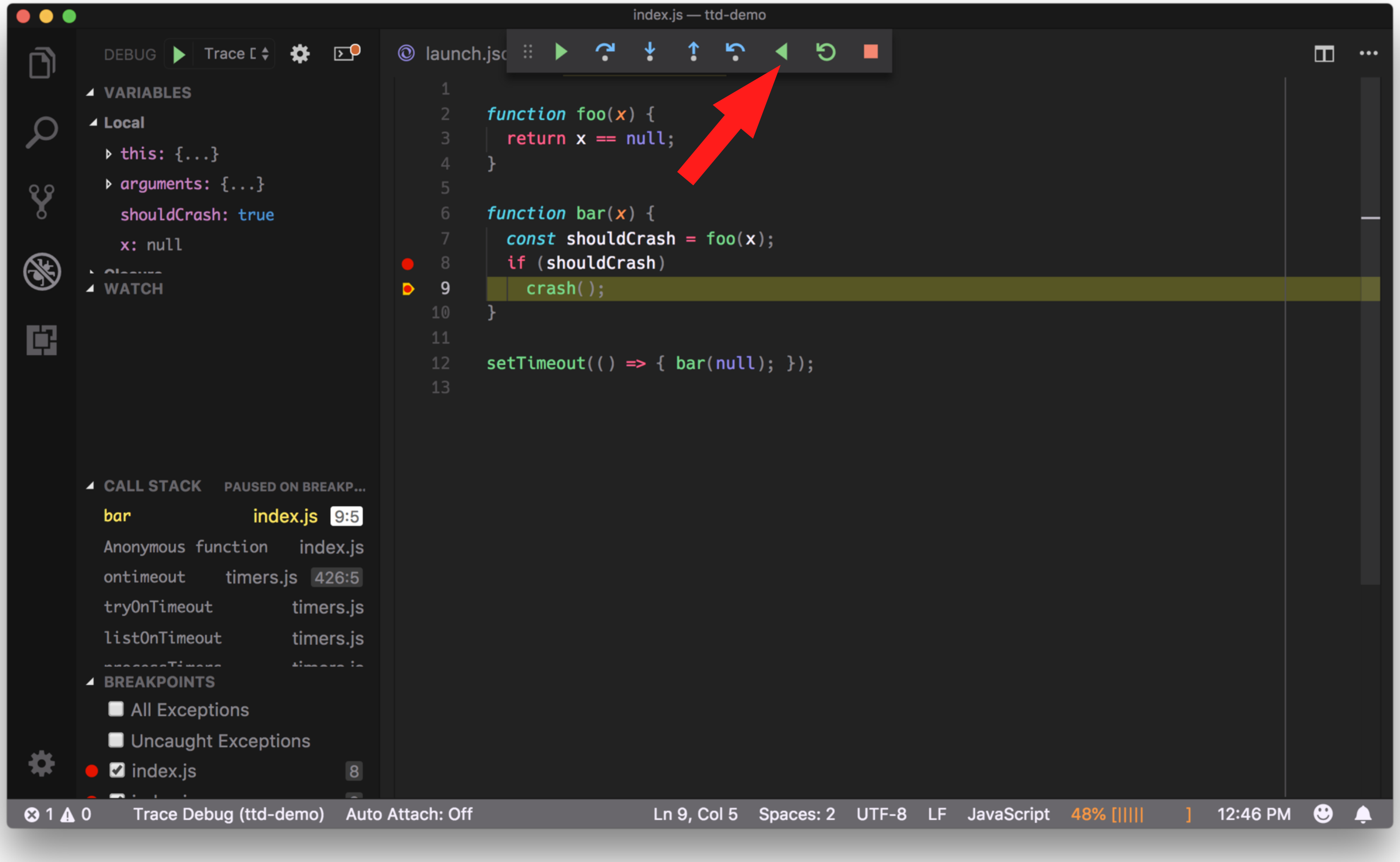
BREAKPOINTS

□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 8

⊗ 1 ⚠ 0 Trace Debug (ttd-demo) Auto Attach: Off Ln 9, Col 5 Spaces: 2 UTF-8 LF JavaScript 48% [|||||] 12:46 PM



VARIABLES

Local

- this: {...}
- arguments: {...}
- shouldCrash: true
- x: null

WATCH

CALL STACK PAUSED ON BREAKP...

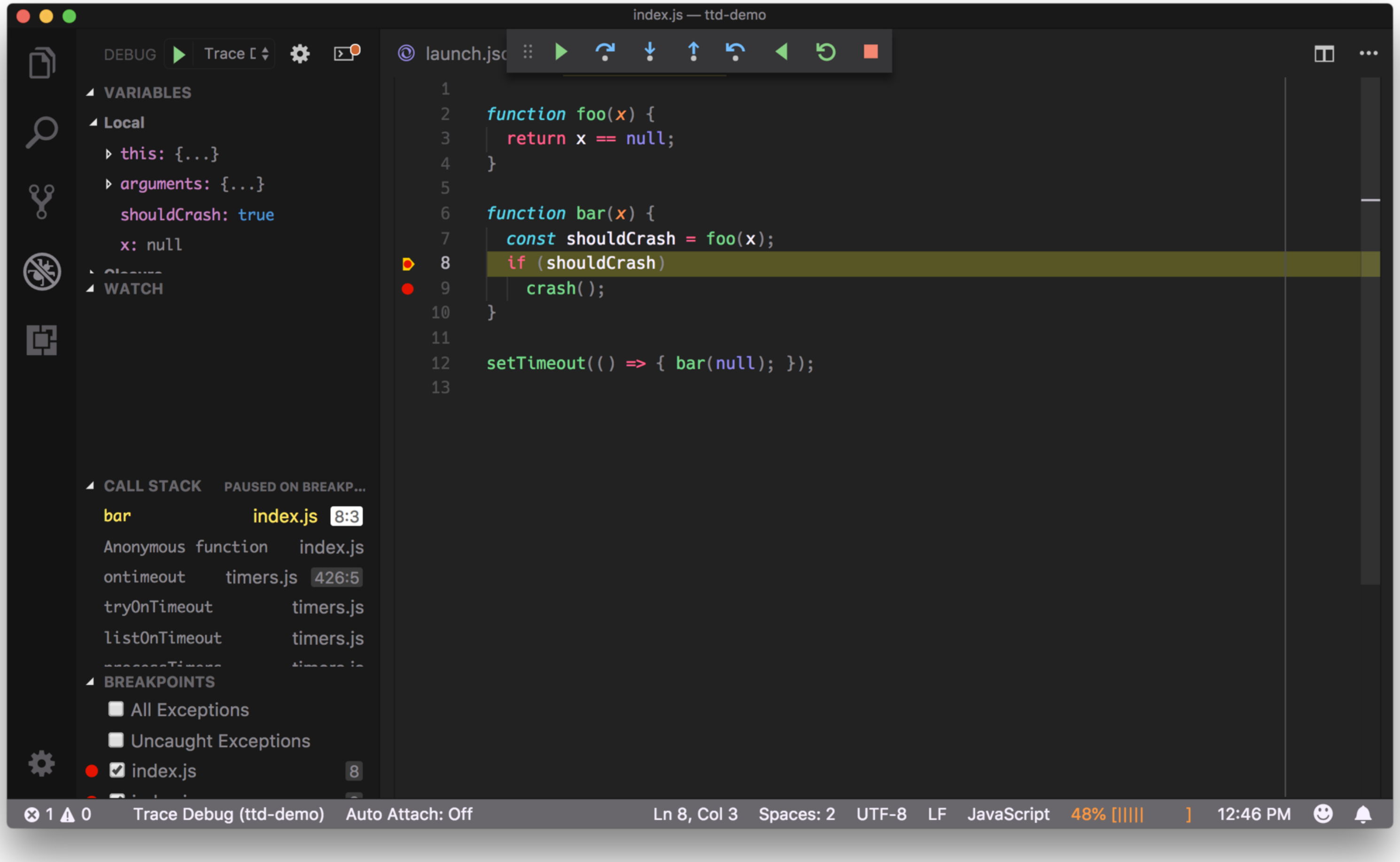
bar	index.js	9:5
Anonymous function	index.js	
ontimeout	timers.js	426:5
tryOnTimeout	timers.js	
listOnTimeout	timers.js	

BREAKPOINTS

- □ All Exceptions
- □ Uncaught Exceptions
- ● ☑ index.js 8

```javascript

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

⊗ 1 ⚠ 0 Trace Debug (ttd-demo) Auto Attach: Off Ln 9, Col 5 Spaces: 2 UTF-8 LF JavaScript 48% [|||||] 12:46 PM



index.js — ttd-demo

VARIABLES

Local

```
this: {...}
arguments: {...}
shouldCrash: true
x: null
```

WATCH

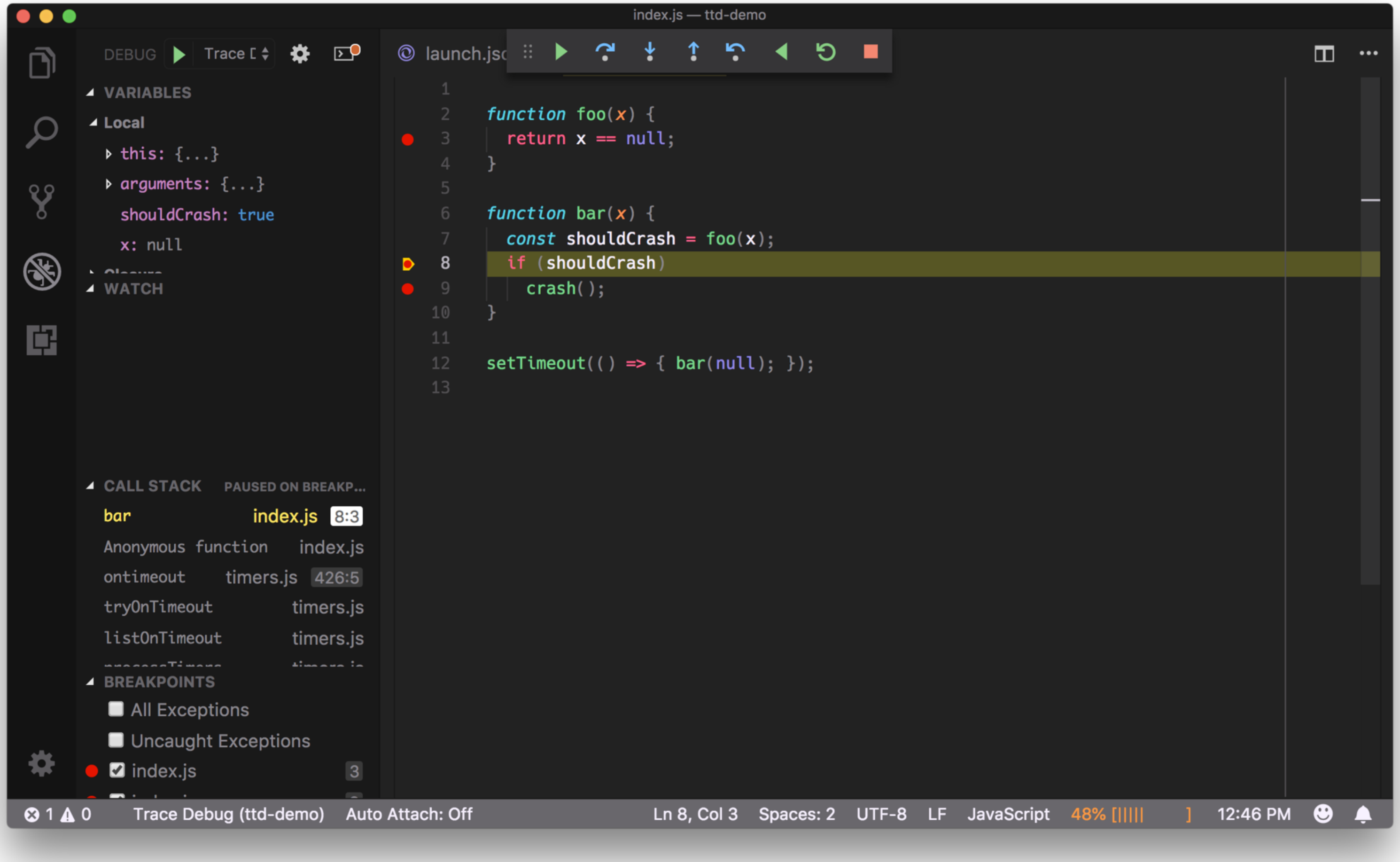
CALL STACK PAUSED ON BREAKP...

```
bar                 index.js 8:3
Anonymous function  index.js
ontimeout           timers.js 426:5
tryOnTimeout        timers.js
listOnTimeout       timers.js
```

BREAKPOINTS

- □ All Exceptions
- □ Uncaught Exceptions
- ● ☑ index.js 8

```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```



Local
- this: {...}
- arguments: {...}
- shouldCrash: true
- x: null

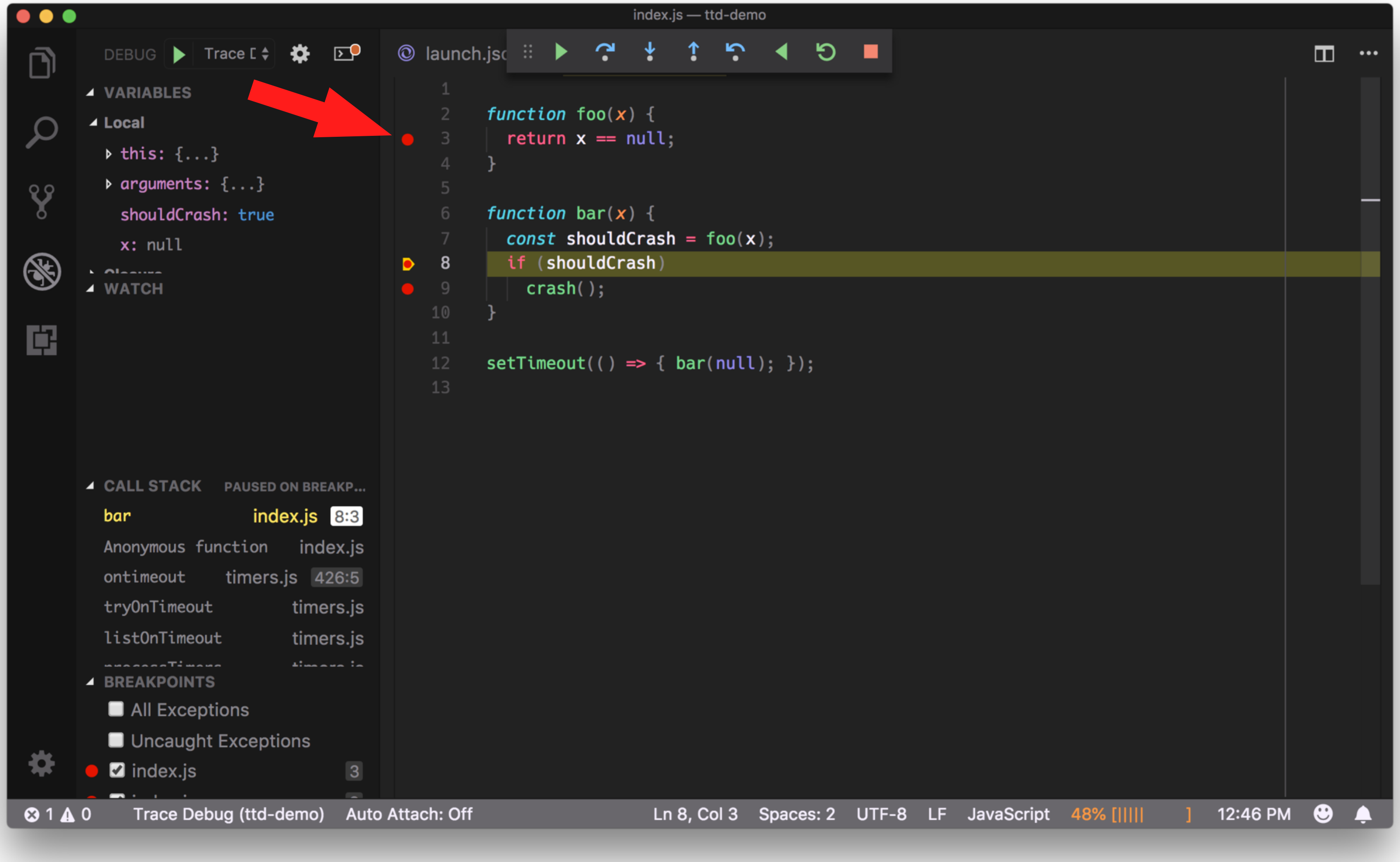
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

bar	index.js	8:3
Anonymous function	index.js	
ontimeout	timers.js	426:5
tryOnTimeout	timers.js	
listOnTimeout	timers.js	

□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 3



```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```

```
this: {...}
arguments: {...}
shouldCrash: true
x: null
```

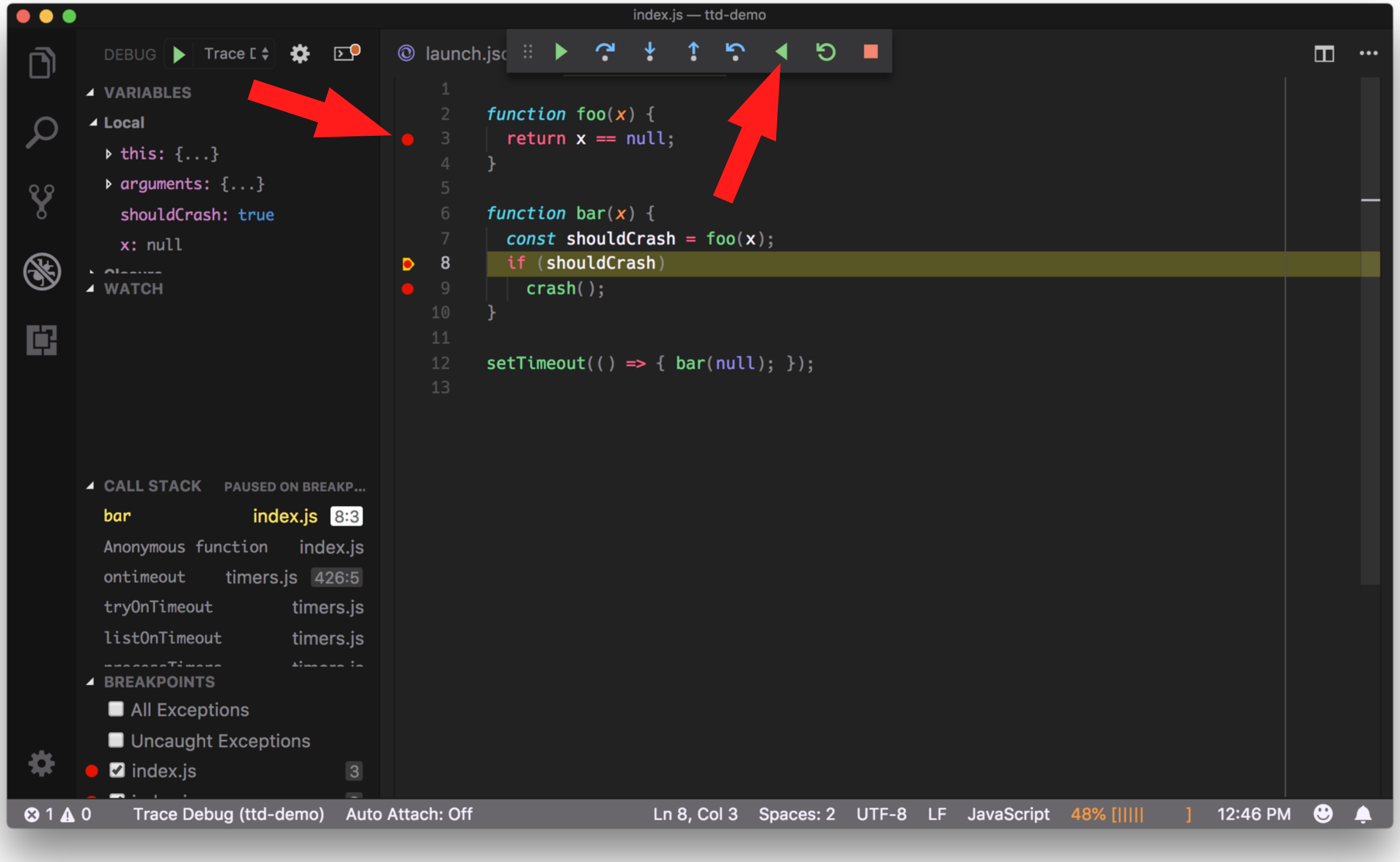
bar	index.js	8:3
Anonymous function	index.js	
ontimeout	timers.js	426:5
tryOnTimeout	timers.js	
listOnTimeout	timers.js	

□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 3

⊗ 1 ⚠ 0 Trace Debug (ttd-demo) Auto Attach: Off Ln 8, Col 3 Spaces: 2 UTF-8 LF JavaScript 48% [|||||] 12:46 PM



```
this: {...}
arguments: {...}
shouldCrash: true
x: null
```

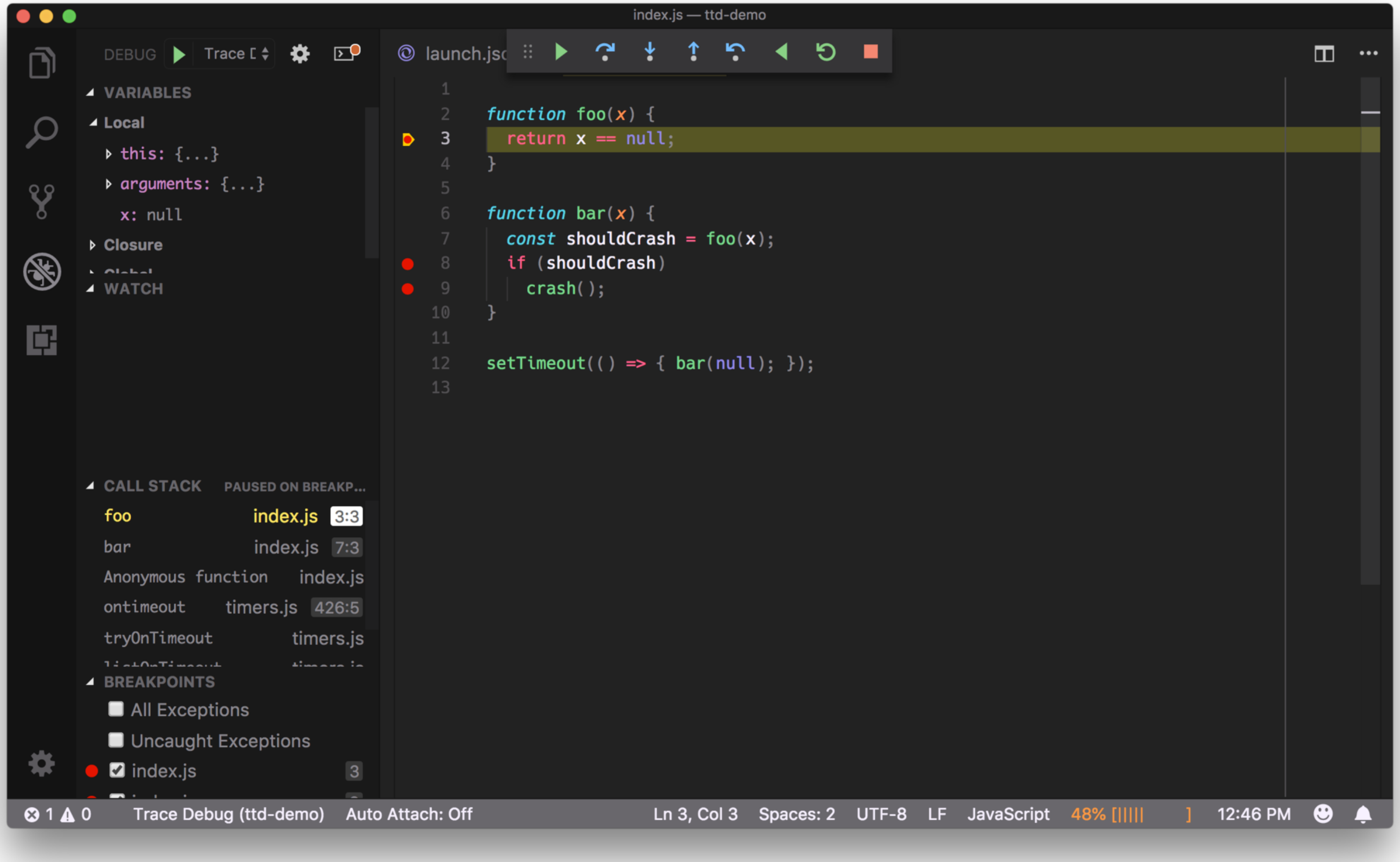
```
function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });
```

```
bar                  index.js  8:3
Anonymous function   index.js
ontimeout            timers.js 426:5
tryOnTimeout         timers.js
listOnTimeout        timers.js
```

□ All Exceptions

□ Uncaught Exceptions

● ☑ index.js 3



```
Local
  this: {...}
  arguments: {...}
  x: null
Closure
```

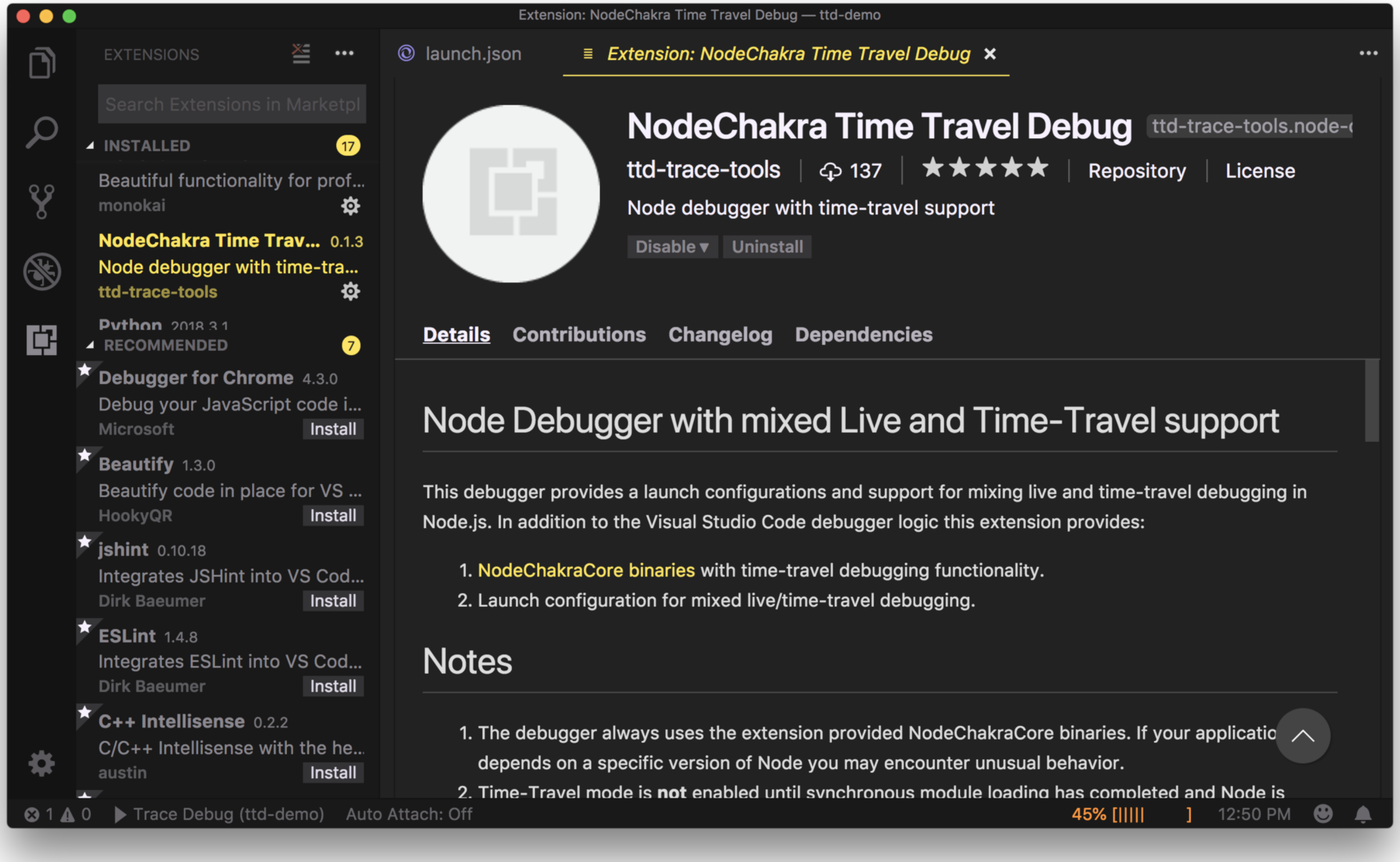
```
foo                     index.js  3:3
bar                     index.js  7:3
Anonymous function      index.js
ontimeout               timers.js 426:5
tryOnTimeout            timers.js
```

□ All Exceptions

□ Uncaught Exceptions

☑ index.js 3

```

function foo(x) {
  return x == null;
}

function bar(x) {
  const shouldCrash = foo(x);
  if (shouldCrash)
    crash();
}

setTimeout(() => { bar(null); });

```


Extension: NodeChakra Time Travel Debug — ttd-demo
EXTENSIONS
Search Extensions in Marketpl
INSTALLED 17
Beautiful functionality for prof...
monokai
NodeChakra Time Trav... 0.1.3
Node debugger with time-tra...
ttd-trace-tools
Python 2018.3.1
RECOMMENDED 7
Debugger for Chrome 4.3.0
Debug your JavaScript code i...
Microsoft
Install
Beautify 1.3.0
Beautify code in place for VS ...
HookyQR
Install
jshint 0.10.18
Integrates JSHint into VS Cod...
Dirk Baeumer
Install
ESLint 1.4.8
Integrates ESLint into VS Cod...
Dirk Baeumer
Install
C++ Intellisense 0.2.2
C/C++ Intellisense with the he...
austin
Install
launch.json
Extension: NodeChakra Time Travel Debug
NodeChakra Time Travel Debug
ttd-trace-tools.node-
ttd-trace-tools
137
★★★★★
Repository
License
Node debugger with time-travel support
Disable ▾
Uninstall
Details
Contributions
Changelog
Dependencies
1 0
Trace Debug (ttd-demo)
Auto Attach: Off
45% [|||||]
12:50 PM

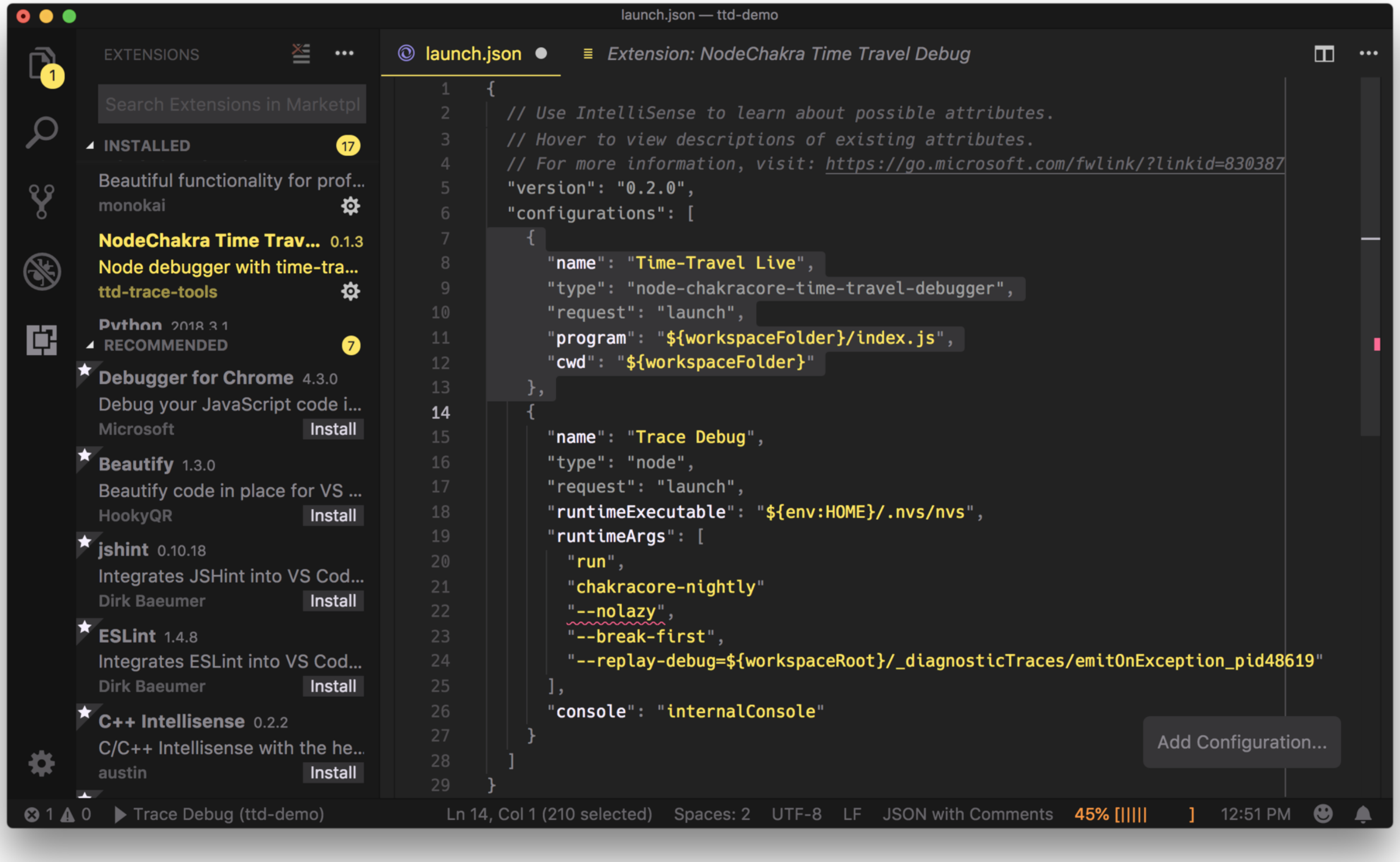
Node Debugger with mixed Live and Time-Travel support

This debugger provides a launch configurations and support for mixing live and time-travel debugging in Node.js. In addition to the Visual Studio Code debugger logic this extension provides:

1. NodeChakraCore binaries with time-travel debugging functionality.
2. Launch configuration for mixed live/time-travel debugging.

Notes

1. The debugger always uses the extension provided NodeChakraCore binaries. If your applicatio depends on a specific version of Node you may encounter unusual behavior.
2. Time-Travel mode is **not** enabled until synchronous module loading has completed and Node is


launch.json — ttd-demo
EXTENSIONS
Search Extensions in Marketpl
INSTALLED 17
Beautiful functionality for prof...
monokai
NodeChakra Time Trav... 0.1.3
Node debugger with time-tra...
ttd-trace-tools
Python 2018.3.1
RECOMMENDED 7
Debugger for Chrome 4.3.0
Debug your JavaScript code i...
Microsoft
Install
Beautify 1.3.0
Beautify code in place for VS ...
HookyQR
Install
jshint 0.10.18
Integrates JSHint into VS Cod...
Dirk Baeumer
Install
ESLint 1.4.8
Integrates ESLint into VS Cod...
Dirk Baeumer
Install
C++ Intellisense 0.2.2
C/C++ Intellisense with the he...
austin
Install
launch.json
Extension: NodeChakra Time Travel Debug
Add Configuration...
1 0
Trace Debug (ttd-demo)
Ln 14, Col 1 (210 selected)
Spaces: 2
UTF-8
LF
JSON with Comments
45% [|||||]
12:51 PM

```
{
  // Use IntelliSense to learn about possible attributes.
  // Hover to view descriptions of existing attributes.
  // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Time-Travel Live",
      "type": "node-chakracore-time-travel-debugger",
      "request": "launch",
      "program": "${workspaceFolder}/index.js",
      "cwd": "${workspaceFolder}"
    },
    {
      "name": "Trace Debug",
      "type": "node",
      "request": "launch",
      "runtimeExecutable": "${env:HOME}/.nvs/nvs",
      "runtimeArgs": [
        "run",
        "chakracore-nightly"
        "--nolazy",
        "--break-first",
        "--replay-debug=${workspaceRoot}/_diagnosticTraces/emitOnException_pid48619"
      ],
      "console": "internalConsole"
    }
  ]
}
```

Time-Travel Debugging
Current Status

Time-Travel Debugging Current Status

- Apenas com node-chakracore + VS Code

Time-Travel Debugging Current Status

- Apenas com node-chakracore + VS Code
    - Protocolo utilizado é aberto

Time-Travel Debugging Current Status

- Apenas com node-chakracore + VS Code
  - Protocolo utilizado é aberto
  - Colaboração com V8 para trazer essa funcionalidade para o Node.js Core

ChakraCore vai substituir o V8?

ChakraCore vai substituir o V8?

Não é o objetivo

Existe "briga" entre as VMs?

Existe "briga" entre as VMs?

Existe uma competitividade saudável

Existe "briga" entre as VMs?

Existe uma competitividade saudável

Também existe muita ***colaboração*** entre elas

Posso usar node-chakracore em produçao?

Posso usar node-chakracore em produçao?

Não deve ter problemas

Posso usar node-chakracore em produçao?

Não deve ter problemas

Use a seu próprio risco

Posso usar TTD em produção?

Posso usar TTD em produção?

Overhead de memória ~10%

Overhead de processamento

Posso usar TTD em produção?

Overhead de memória ~10%

Overhead de processamento

Não recomendado no momento, mas é um objetivo futuro sthima

www.sthima.com

@mmarkini

http://mmarchini.me/

oss@mmarchini.me

https://github.com/mmarchini/