Introduction node.js using

Gerred Dillon

@justicefries

Introduction JavaScript

JavaScript

1996

JavaScript present

JavaScript syntax

JavaScript

syntax

variables

JavaScript

syntax

variables

data structures

JavaScript
syntax

variables
data structures
functions

JavaScript

syntax

variables

data structures

functions

basic operators

Introduction

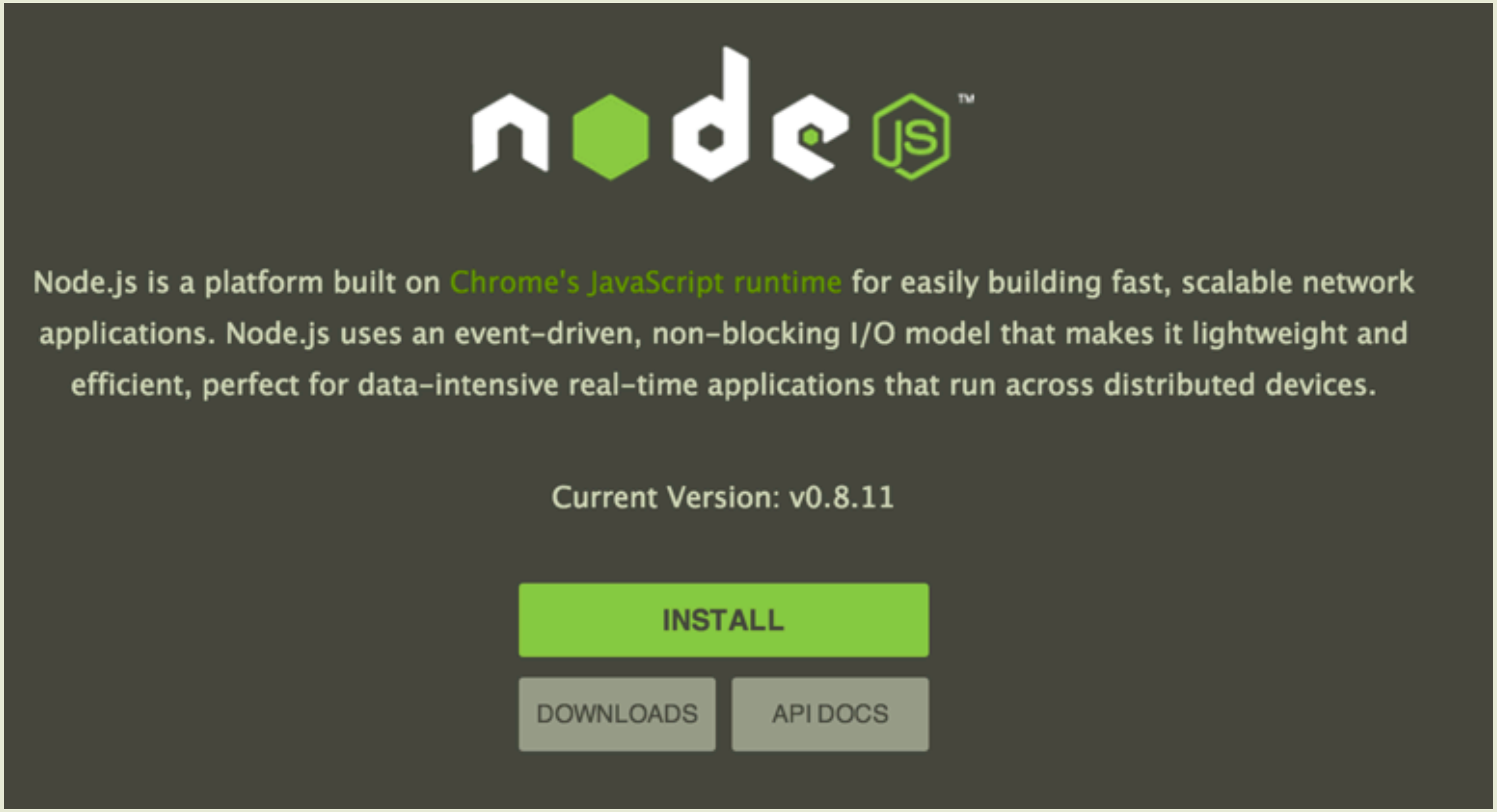
node.js

node.js 2009

node.js

present

node.js



Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

Current Version: v0.8.11

node.js

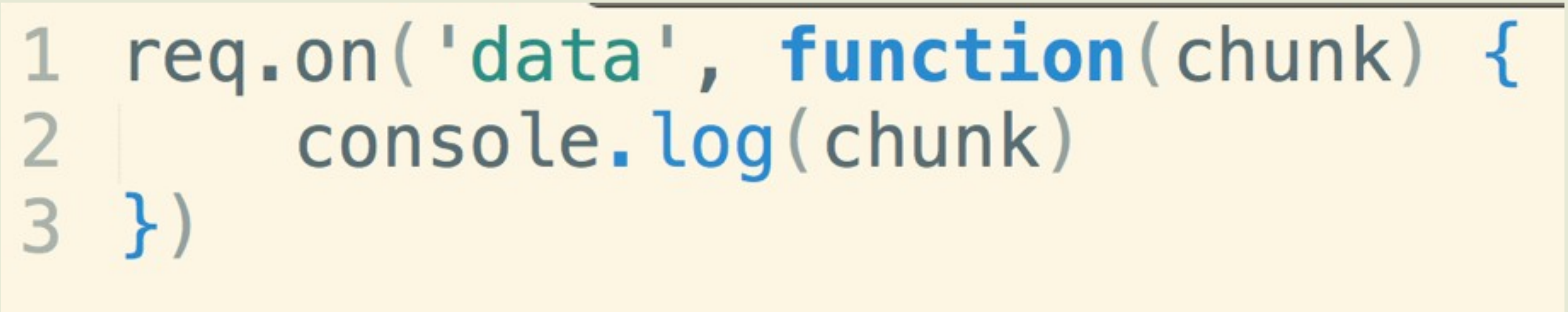
Chrome JS runtime (aka V8)

node.js

event-driven

node.js

event-driven



```
req.on('data', function(chunk) {
    console.log(chunk)
})
```

node.js

non-blocking

node.js

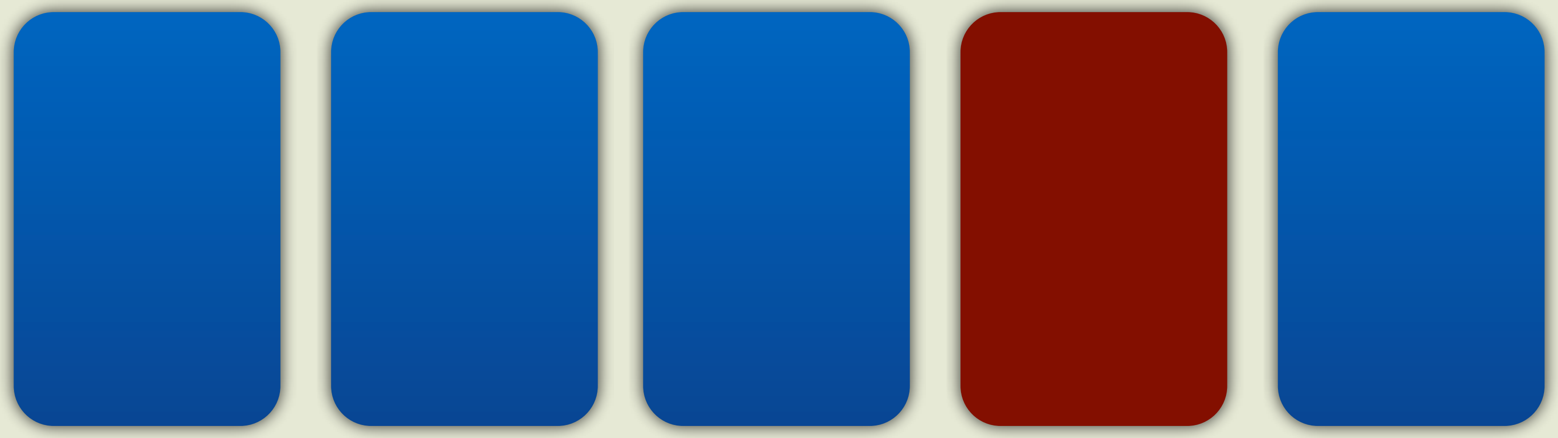
non-blocking

node.js

non-blocking

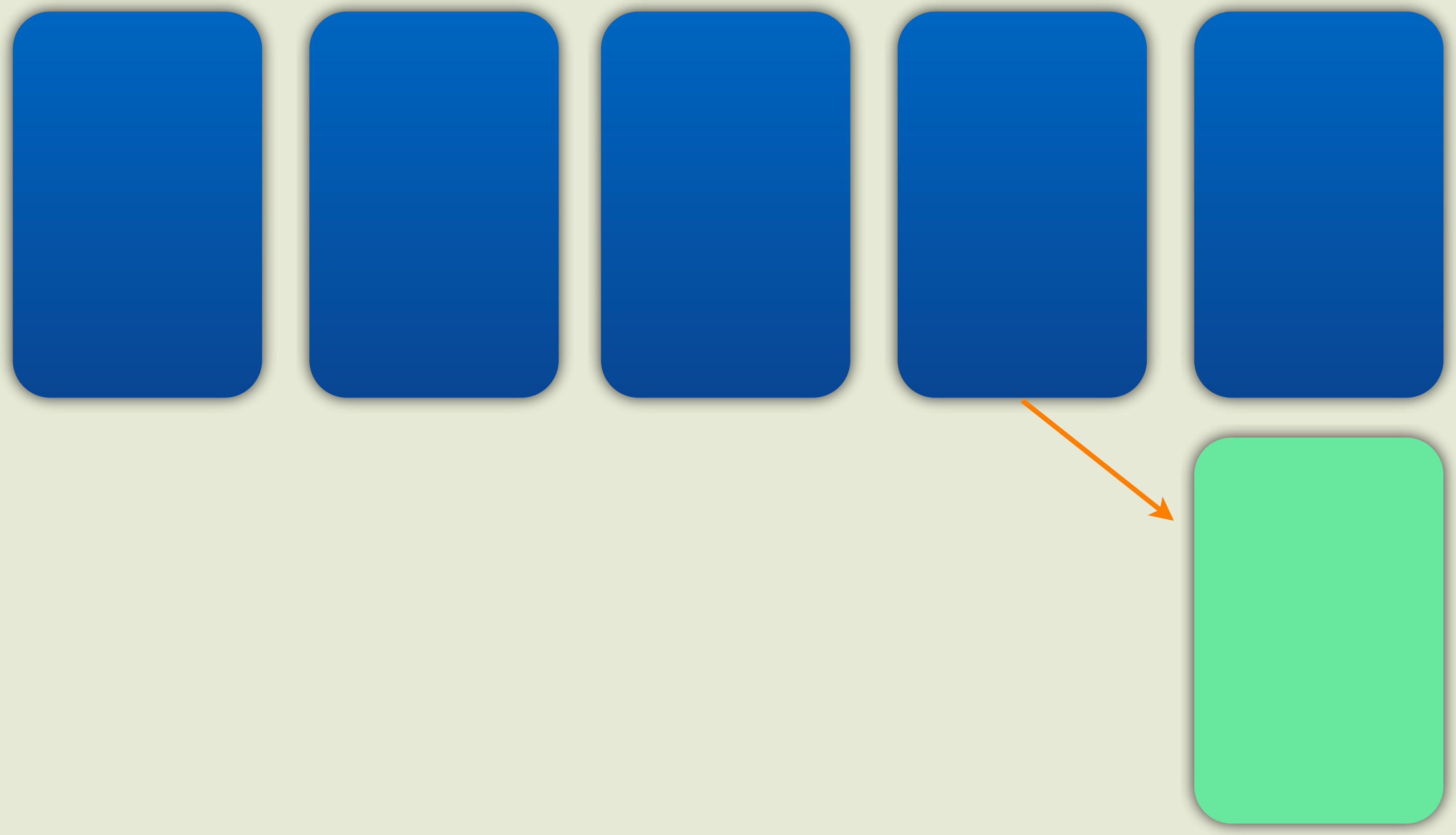
node.js

non-blocking



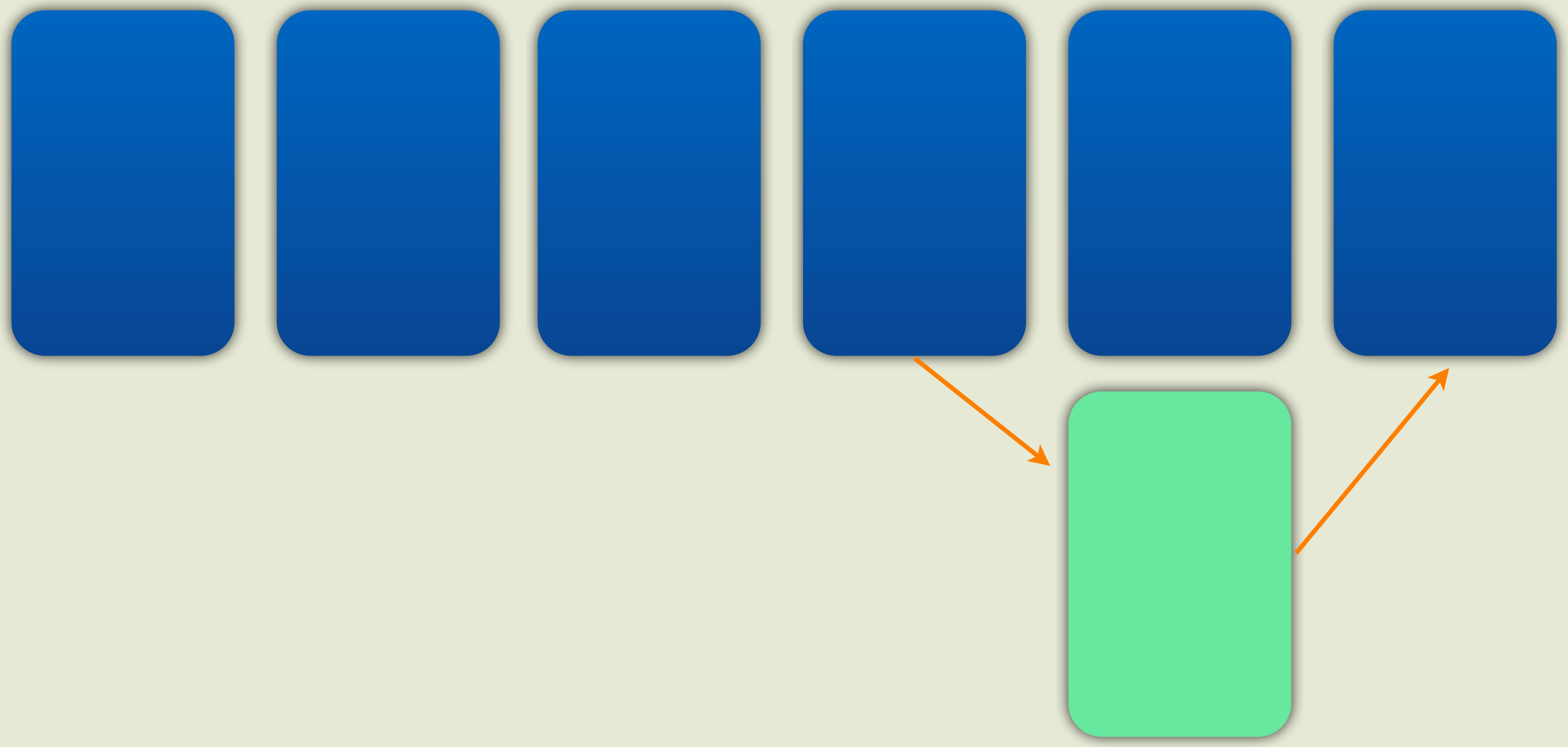
node.js

non-blocking



node.js

non-blocking



node.js

I/O Model

node.js is really good at I/O

node.js is less good at CPU-intensive tasks

(aka don’t create a Fibonacci server)

1	1	2	3	5	8	13	21

(okay, maybe- we’ll talk about this later)

my first server

(hello, world!)

my second server

(serving a file)

my second server

let’s make it synchronous

my second server

~~let's make it synchronous~~

modules node.js and other interesting things

“that’s great and all, but I don’t want to reinvent (x)”

Node

Package

Manager

npmjs.org

CommonJS

provides

explicit dependencies

npmjs.org

my third server

(let’s use express)

CommonJS

provides

explicit dependencies

npmjs.org

Exposing

custom functionality

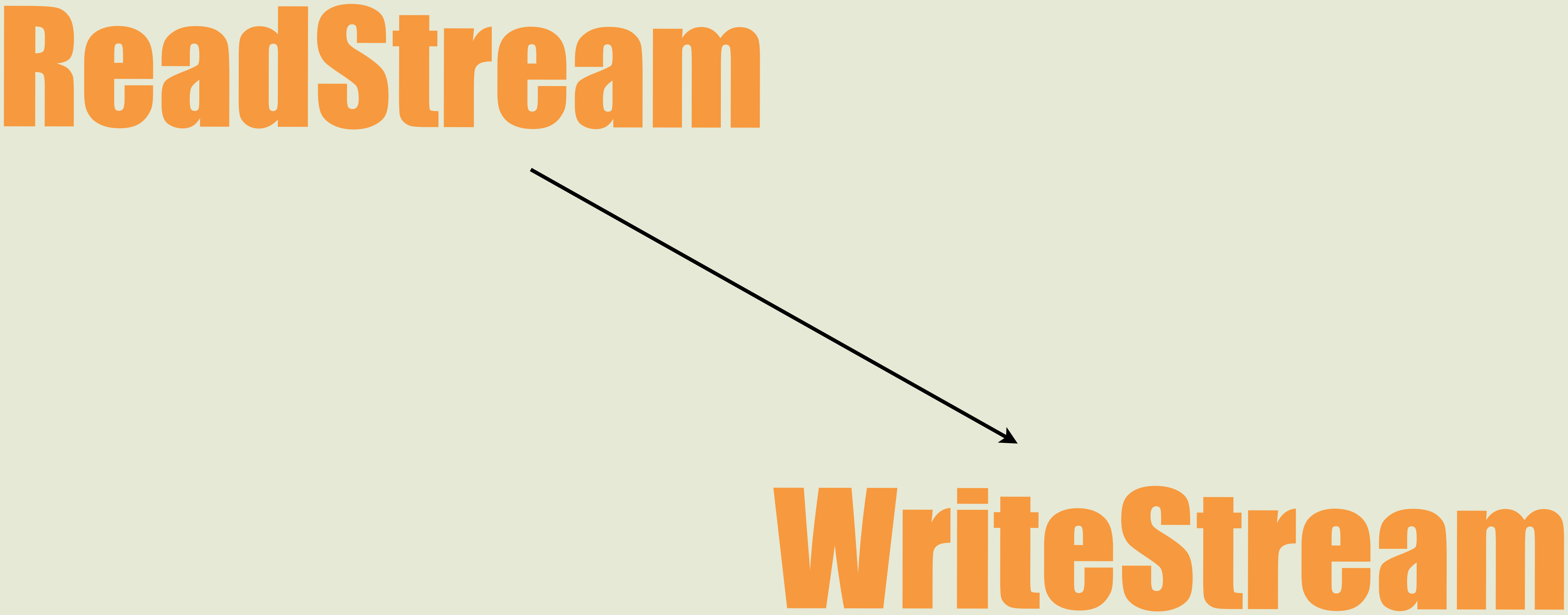
to node.js

npmjs.org

Streams create powerful abstractions

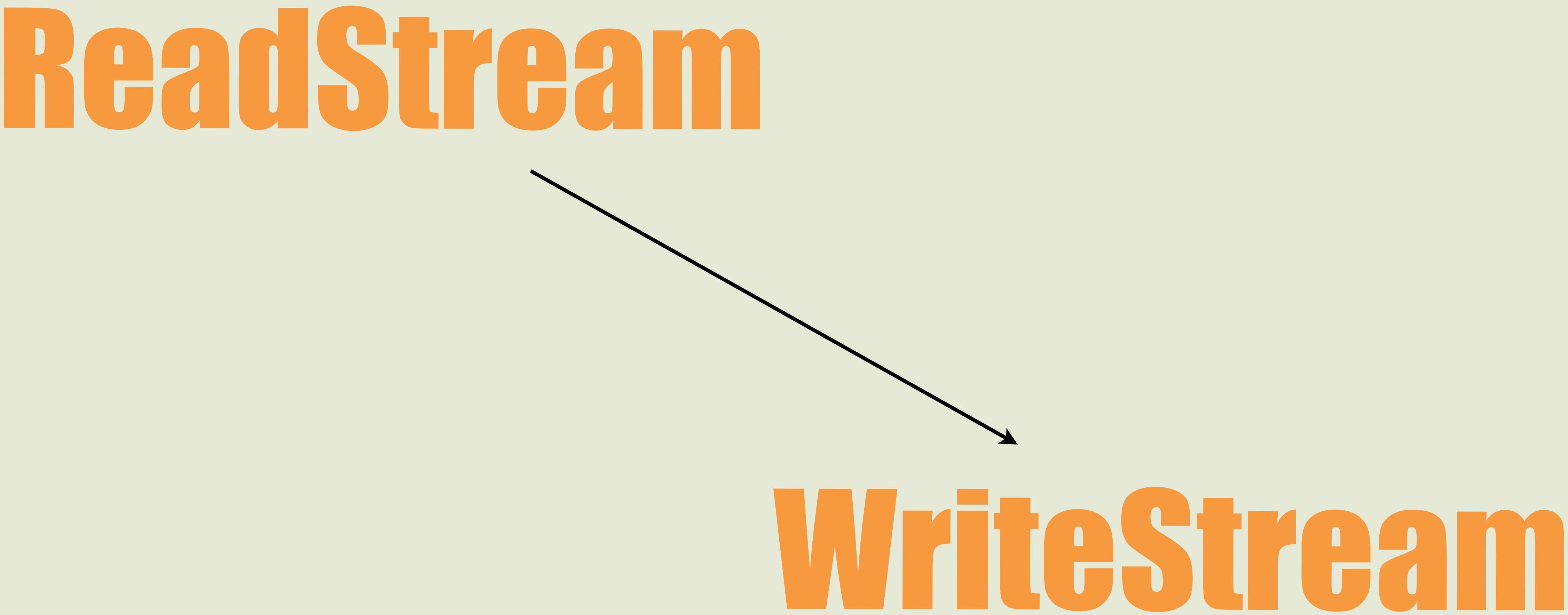
Streams create powerful abstractions

pipe()



ReadStream

WriteStream



ReadStream

WriteStream

(reversed in code)

Flow Control

with

node.js

(aka how do I iterate asynchronously?)

(aka what if I really, really want to calculate fibs?)

The Guts of *node.js*

EventEmitter

is the core

of node.js

Moving Forward

Read the API

Build Apps

Get Involved

Use npmjs.org

Use GitHub

Resources:

http://nodejs.org/api/

Smashing Node

Questions?