Hands-on Intro to Node.js

Jerry D'Antonio
Akron Code Club
@jerrydantonio

What is 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."

http://nodejs.org/

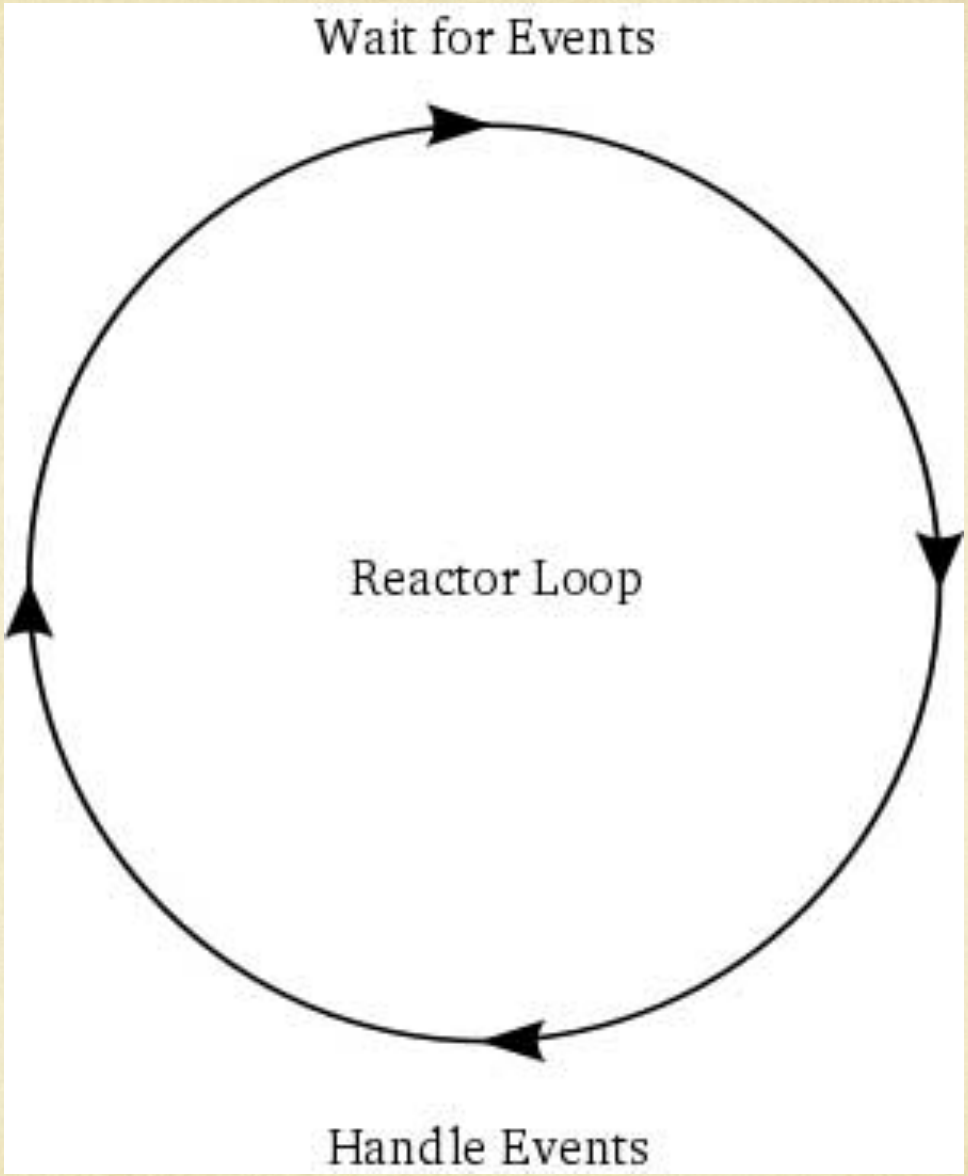
WTH?

Reactor Pattern

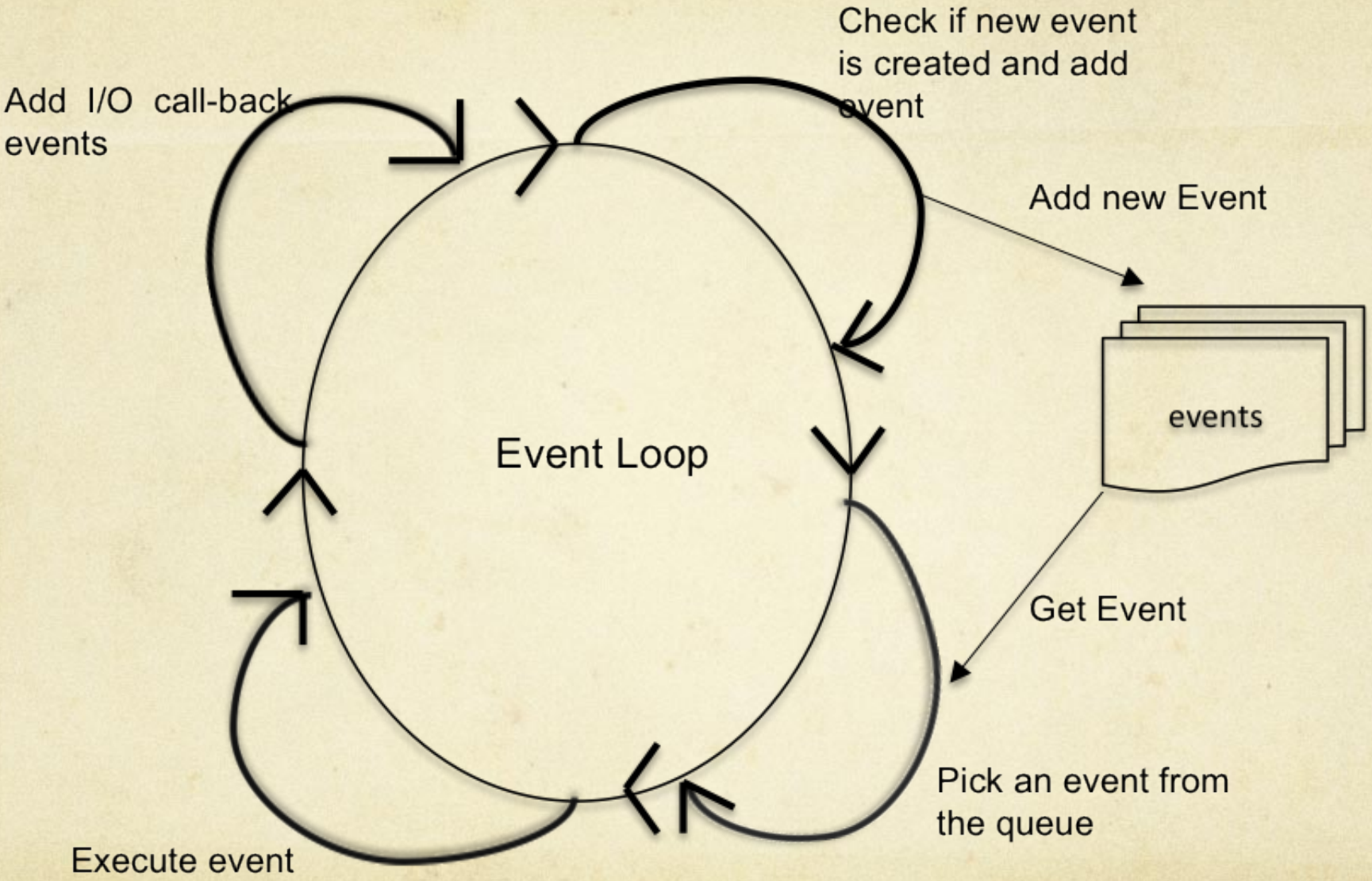
"The reactor design pattern is an event handling pattern for handling service requests delivered concurrently to a service handler by one or more inputs. The service handler then demultiplexes the incoming requests and dispatches them synchronously to the associated request handlers."

http://en.wikipedia.org/wiki/Reactor_pattern

Wait,
what?



http://www.raywenderlich.com
/3932/networking-tutorial-for-ios-how-to-create-a-socket-based-iphone-app-and-server/reactor-1



http://venkateshcm.com/2014/04/Reactor-Pattern-Part-2-Non-blocking-I-O/

Reactor Pattern

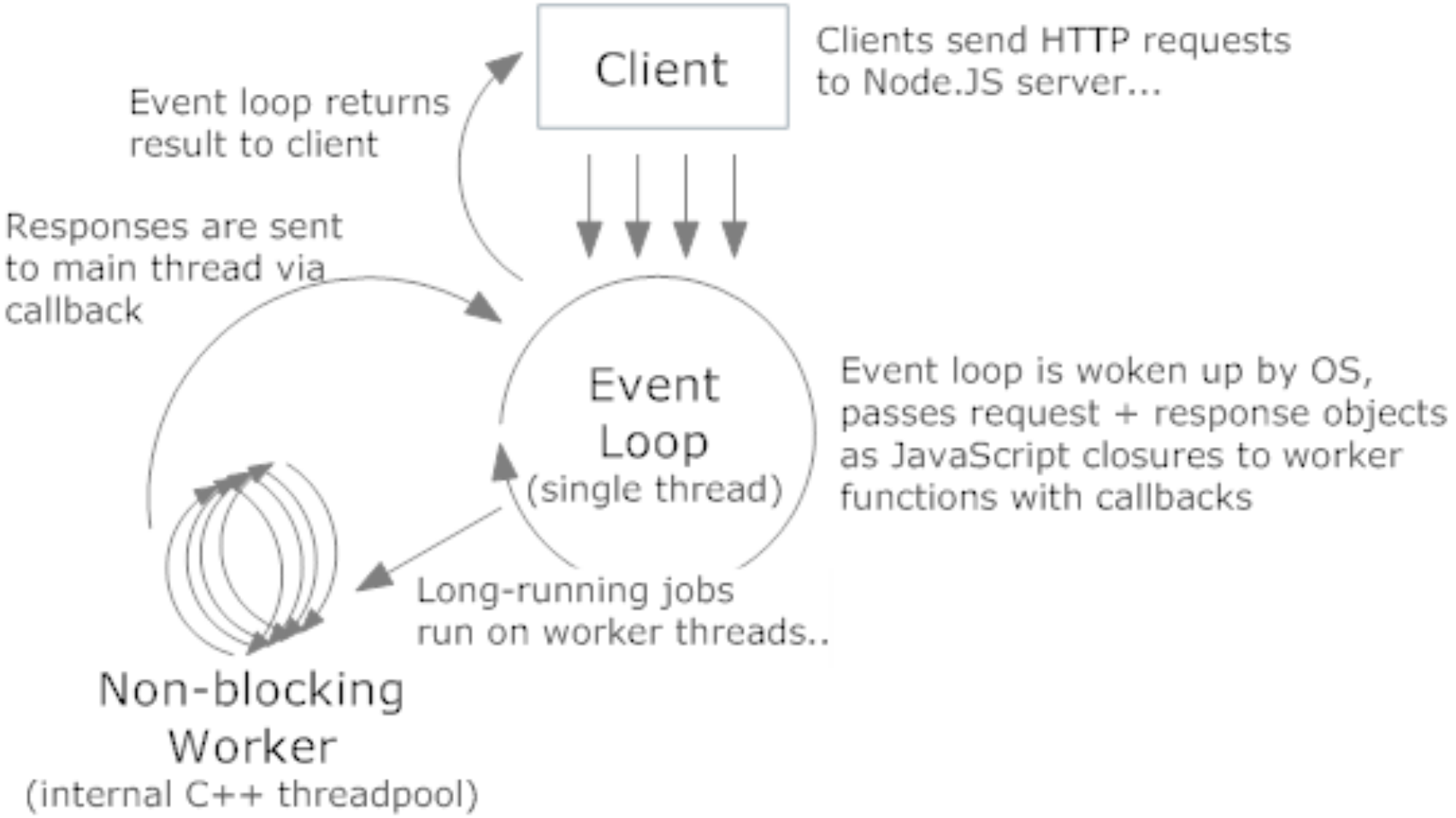
"The reactor design pattern is an event handling pattern for handling service requests delivered concurrently to a service handler by one or more inputs. The service handler then demultiplexes the incoming requests and dispatches them synchronously to the associated request handlers."

http://en.wikipedia.org/wiki/Reactor_pattern

Reactor Pattern Pros/Cons

- Pros
  - App code is decoupled from concurrency code
  - Event handlers can be reused
  - No locking or object synchronization
- Cons
  - Difficult to debug due to control flow inversion
  - Terrible with processor-intensive operations
  - Callbacks…
    - Callbacks…
      - Callbacks…

Node.JS Processing Model



http://www.aaronstannard.com/post/2011/12/14/Intro-to-NodeJS-for-NET-Developers.aspx

What is 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."

http://nodejs.org/

Simple Node.js App

```
1 #!/usr/bin/env node --harmony
2
3 const fs = require('fs');
4 fs.watch('target.txt', function() {
5   console.log("File 'target.txt' just changed!");
6 });
7 console.log("Now watching target.txt for changes...");
```

```
[13:37:11 Jerry ~/Projects/more-polygl                esystem (master)]
$ node watcher.js --harmony
Now watching target.txt for changes...
File 'target.txt' just changed!
File 'target.txt' just changed!
File 'target.txt' just changed!
File 'target.txt' just changed!
File 'target.txt' just changed!
File 'target.txt' just changed!
```

Profile: Default
Command: login

```
[13:36:32 Jerry ~/Projects/more-polyglot/node.js/filesystem (master)]
$ touch target.txt

[13:37:30 Jerry ~/Projects/more-polyglot/node.js/filesystem (master)]
$ touch target.txt

[13:37:36 Jerry ~/Projects/more-polyglot/node.js/filesystem (master)]
$ touch target.txt

[13:37:39 Jerry ~/Projects/more-polyglot/node.js/filesystem (master)]
$
```

Key Point

A Node.js application isn't a complete program—
it's a reactor initializer!

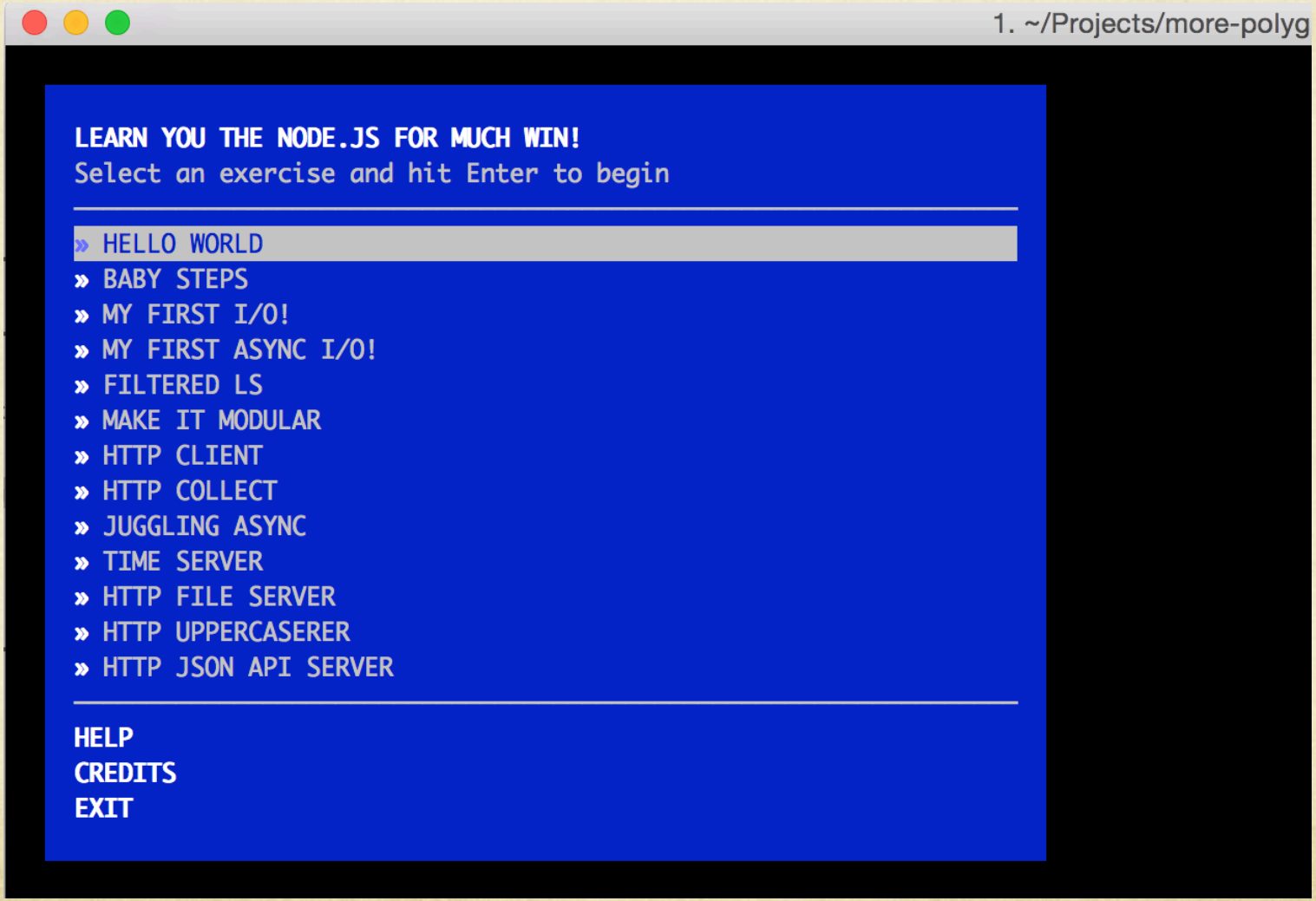
Structure of a Node.js App

- Import required libraries
- Create global objects
- Listen to events
  - Define the event
  - Attach one or more callbacks
- Implicitly start the reactor
- Run forever!

What To Do

- Download and install Node.js
  - http://nodejs.org/download/
- Verify Node.js is installed and working
  - `` `npm -h` `` to verify that NPM is working
  - `` `node` `` followed by `[CTRL-C]`
- Install NodeSchoolmodule
  - http://nodeschool.io/#workshoppers
  - `` `npm install -g learnyounode` ``
  - `` `learnyounode` ``

Learn You The Node.js



Retrospective

- How far did you get into the tutorials?
- Which was the simplest tutorial?
- Which was the hardest tutorial?
- What is your general impression of reactor-based programming?
- What is your general impression of Node.js programming?
- Do you plan to do more work with Node.js?

Resources

- http://nodejs.org/
- http://nodeschool.io/
- https://www.codeschool.com/courses/real-time-web-with-node-js
- https://pragprog.com/book/jwnode/node-js-the-right-way
- https://github.com/jdantonio/more-polyglot