Node.js by Example

What is Node.js?

What is Node.js?

- Node.js is an ***event-driven***, ***server-side*** JavaScript environment
  - Based on the V8 JavaScript Engine, developed by Google
- Most importantly, node.js is a
  - server-side runtime environment, that
  - compiles and executes JavaScript *very* efficiently.
- Node.js is specifically designed for building fast, efficient, scalable network applications

V8 JavaScript Engine

- Developed by Google
- Ships with the Google Chrome web browser
- Allows Chrome to run JavaScript code much faster
  - It does this by compiling the JavaScript directly into ***native machine code***
  - Other browsers interpret JavaScript, or execute it as bytecode
- What does this mean for us, and for Node.js?

JavaScript:
Not just for the browser anymore…

- JavaScript is:
  - A fully-functional programming language
    - Capable of doing anything other, more traditional, languages (C++, Java, Ruby, etc) can do
  - Naturally good at event-based programming
  - Traditionally resigned to the context of the web application frontend
    - i.e. running inside a web browser
- In this course, I will assume you have a working knowledge (not mastery) of JavaScript

JavaScript:
Not just for the browser anymore…

- But, there's no reason the JavaScript language can't be used in other contexts (say, server-side)
  - This is where node.js comes in, allowing JavaScript to be efficiently executed on the server-side
- JavaScript is naturally good at event-based programming
  - Thus, so is node.js
  - An event model is ideal for thinking about client-server applications

How does Node.js work?

- Node.js uses an event-driven, non-blocking I/O model
  - In contrast to the more common concurrency model using OS threads for each connection
- For our purposes in this course, it is enough to know that node.js:
  - Is Server-side
  - Is a JavaScript runtime environment
  - Is designed for concurrency
  - Executes JavaScript very efficiently

JavaScript and Node.js

- An added benefit to using node.js/JavaScript for server-side programming is that both the front and back end of a web application can be developed in the same language.
- This minimizes:
  - Developer learning curves
  - Developer context switches

Node.js Performance

- I mentioned earlier that node.js using the V8 engine for efficient JavaScript execution,
- And that Node.js also uses an event-driven, non-blocking I/O model
- This makes node.js ideal for:
  - Real-time applications
  - Data-intensive applications
  - Scalable, asynchronous applications

Node.js API

- In addition to executing JavaScript like a champ, node.js also provides an API in the form of modules (a.k.a. libraries)
- Many modules come with a node.js installation, to provide standard application framework functionality
  - STDIO: console logging, timing, tracing
  - HTTP: HTTP server functionality
  - …etc
- Import modules into your application to provide specific functionality

A Few Node.js Modules

- STDIO
- HTTP
- File System
- Crypto
- Streams
- Many more…

Node.js API

- You can see a full list of libraries provided by the node.js API at http://nodejs.org/docs/latest/api/http.html *

* Note: this link shows documentation for the latest version of node.js. If you are using an older version, refer to the API docs for that version

Node.js Community

- Additionally, the node.js community maintains excellent modules to enhance the capabilities of node.js
- These modules take the form of libraries, toolkits, frameworks, and much more.
- Using them, you can write applications with advanced functionality, without writing all of the complex code yourself
  - We'll explore some of these libraries in this course, including Socket.io and Express.

Similar Environments

- Node.js is a good idea – so good that it has some company
- Similar environments have been written for other languages:
  - EventMachine for Ruby
  - Twisted for Python

This Course

- Next lecture, we'll talk briefly about when and why to use node.js
- Then we will move on to building example applications using node.js
  - You will be learning by coding, which I believe is a highly effective form of instruction for this material
- Have fun with node.js!