

# Introduction to



# What is Node?

- Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications.
- Node is NOT a framework like Laravel
- Non-blocking I/O model



# Node Philosophy

- I/O operations are expensive
- I/O should be non-blocking
  - Database queries / writes
  - File access
  - Calls over HTTP

# Blocking vs Non-Blocking

## Blocking I/O (Synchronous)

```
$contents = file_get_contents('some-file.txt');  
echo $contents;  
echo 'finished';
```

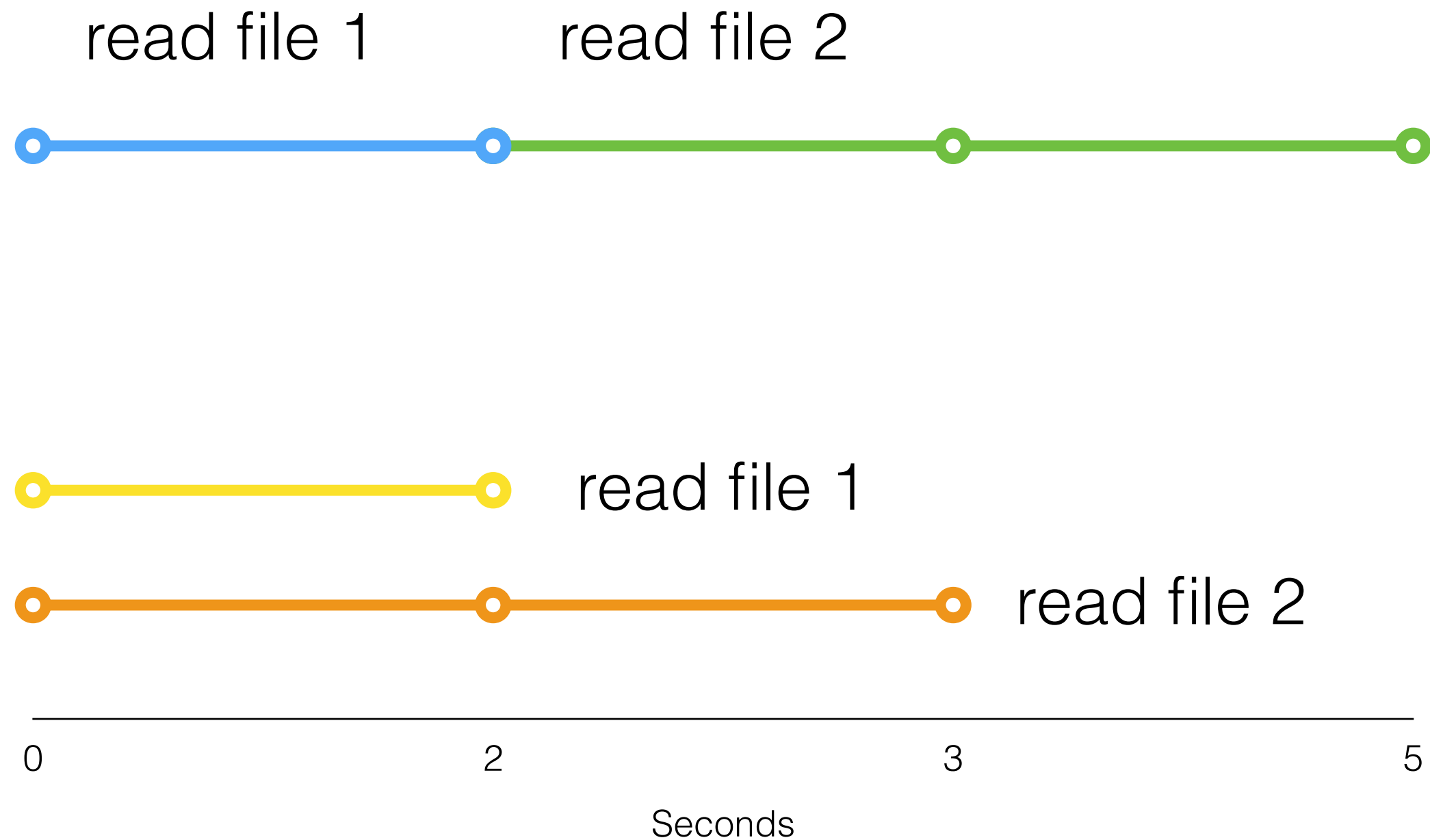
> file contents  
> finished

## Non-Blocking I/O (Asynchronous)

```
fs.readFile('some-file.txt', function(err, contents) {  
  console.log(contents);  
});  
console.log('finished');
```

> finished  
> file contents

# Blocking vs Non-Blocking



# Web Server in Node

```
var http = require('http');  
  
var server = http.createServer();  
  
server.on('request', function(request, response) {  
  response.write('Today is ' + new Date());  
  response.end();  
});
```

```
var http = require('http');
var fs = require('fs');

var server = http.createServer();

server.on('request', function(request, response) {
  console.log('Request: ' + request.url);

  fs.readFile('some-file.txt', function(err, contents) {
    response.end(contents);
  });
});

server.listen(8080);
```

## Event Queue

request 1 callback

request 2 callback

request 1 file read callback

request 2 file read callback

# Summary

- Node.js is non-blocking unlike many other programming languages
- Great for applications with lots of I/O
  - Single page applications
  - JSON API's
  - Realtime apps
- Not great for CPU heavy apps