

TIM OXLEY

MODULAR VS MONOLITH

GOOD MORNING!

WHO AM I?



TIM OXLEY

@secoif

github.com/timoxley

**FULL STACK
DEVELOPER**

REACT

ES6/BABEL

WEBPACK

EXPRESS

LEVELDB/POUCHDB







PID 1851
HOST nodesource3
STARTED September 18, 2015 15:42:07
UPTIME 5m 27s

Download Profile
Start New Profile

PROFILE TIMELINE

VISUALIZATION TYPE

- Sunburst
- Flame Graph
- Treemap

COLLAPSE

		DONE X	
CALLS	FUNCTION	SELF	TOTAL
17	handle express.js/router/layer.js	27ms 0.45%	1193ms 19.87%
3	(anonymous) express.js/router/index.js	5ms 0.08%	1198ms 19.95%
1	process_params express.js/router/index.js	2ms 0.03%	1199ms 19.98%
7	next express.js/router/index.js	12ms 0.18%	1213ms 20.22%
52	expressinit express.js/middleware/init.js	83ms 1.37%	1299ms 21.64%
3	handle express.js/router/layer.js	5ms 0.08%	1303ms 21.72%
2	trim_prefix express.js/router/index.js	4ms 0.05%	1307ms 21.77%
1	(anonymous) express.js/router/index.js	2ms 0.03%	1308ms 21.80%
6	process_params express.js/router/index.js	10ms 0.16%	1318ms 21.95%
1	next express.js/router/index.js	2ms 0.03%	1322ms 22.03%
4	query express.js/middleware/query.js	7ms 0.11%	1333ms 22.22%

NODESOURCE.COM

CAMPJS CONFERENCE AUSTRALIA
SINGAPOREJS MEETUP
SINGAPOREJS BOOK CLUB
NODEUP PODCAST
NODESCHOOL

MODULAR VS MONOLITH

**OVER-ENGINEERING IS
COMMON AND AWFUL**

ME IN 2012

ME IN 2012:

“AVOID FRAMEWORKS”

**ME IN 2012:
“BUILD LIBRARIES
INSTEAD”**

BUILDING A LIBRARY

- ▶ Reusable!
- ▶ More robust!
- ▶ Higher quality code!
- ▶ Better documentation!
- ▶ Better tests!

ME IN 2015:

**“YES BUT, NOTE THIS ONE
MAJOR CAVEAT...”**

**NOTHING FOR
FREE**

**NODE.JS:
CULTURE OF EXTREME
MODULARITY**

**BUILDING APPLICATIONS
VS
CONSUMING LIBRARIES**

**BUILDING APPLICATIONS
IS VERY DIFFERENT TO
CONSUMING LIBRARIES**

**DON'T PAY COST OF
MODULARISATION BEFORE
YOU KNOW YOU NEED IT**

**ASSUME YOUR FIRST
ATTEMPT IS GOING TO BAD**

**LEARN HOW BAD YOUR FIRST
ATTEMPT IS GOING TO BE AT A
CODE RETREAT**

**BUILD THE SIMPLEST
THING THAT WILL WORK**

WHAT PROBLEM AM I TRYING TO SOLVE?

IS THIS THE SIMPLEST THING THAT WILL WORK?

DEFER DECISIONS

**THE WRONG STRUCTURE
CAN BE MORE COSTLY THAN
NO STRUCTURE AT ALL**

**DRAW IN PENCIL BEFORE
YOU DRAW IN INK**

**FLAT IS BETTER
THAN NESTED**

FLAT IS BETTER THAN NESTED

- ▶ At the statement level: avoid nested if/else.
- ▶ At the function level: avoid nested callbacks (i.e. pyramid of doom)
- ▶ At the class level: avoid nested inheritance hierarchies.
- ▶ At the file level: avoid nested directories.

**WHEN PRESENTED WITH THE OPTION
OF A FLAT STRUCTURE
OR A NESTED STRUCTURE,
PICK THE FLAT STRUCTURE.
PROBABLY.**

**HIDING COMPLEXITY IS
NOT THE SAME AS
REMOVING COMPLEXITY**

NESTING

DECREASES LOCAL COMPLEXITY

WHILE IT ALSO

INCREASES GLOBAL COMPLEXITY

**MODULARISATION AND
ABSTRACTION ARE A FORM
OF NESTING**

**BUILD THE SIMPLEST
THING THAT WILL WORK.**

**THE SIMPLEST THING
IS PROBABLY FLAT.**

**AVOID PREMATURE
MODULARISATION**

MONOLITH FIRST

**OPEN SOURCE
LATER**

THANK YOU!

@secoif

github.com/timoxley

nodesource.com