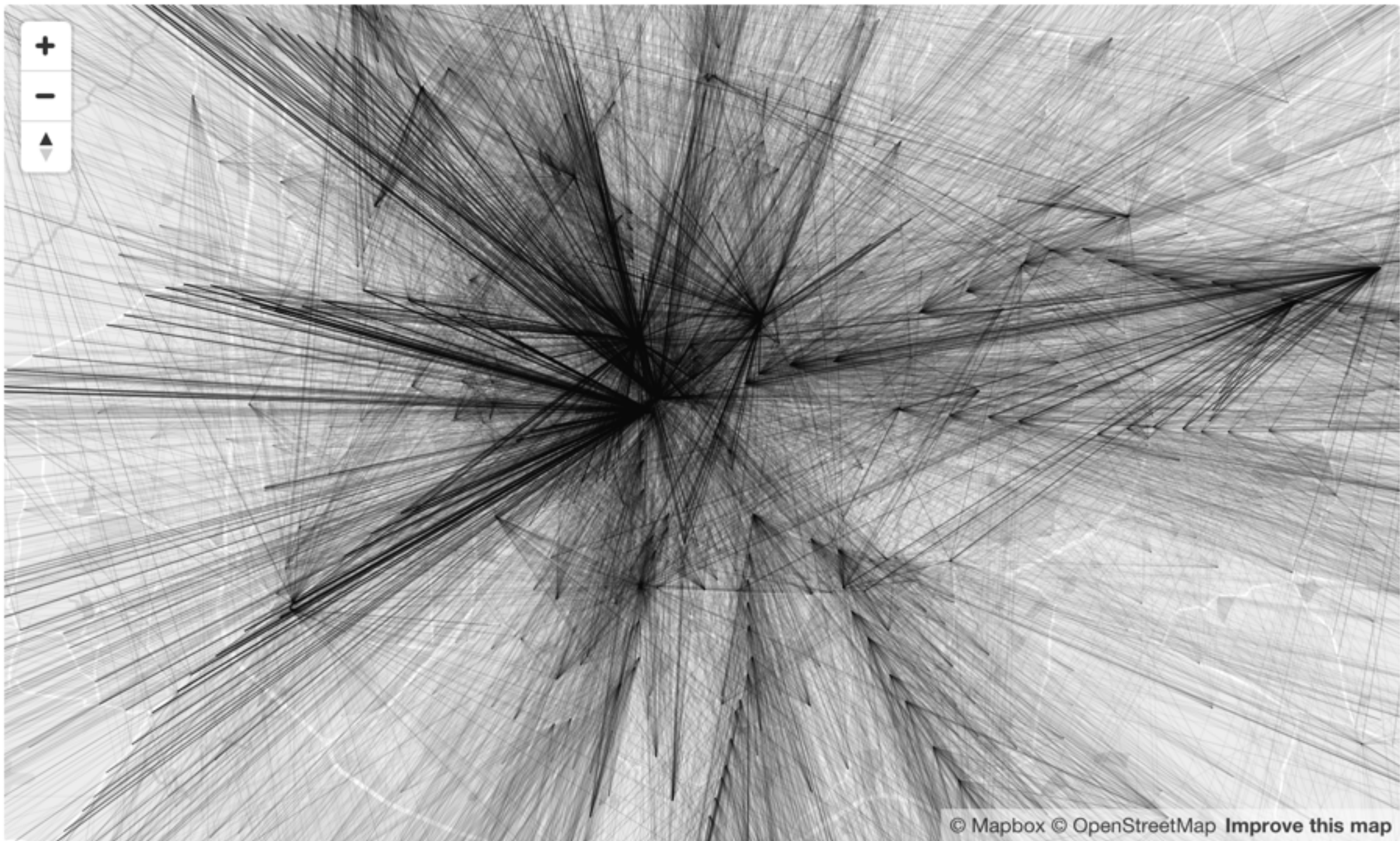


**THE WORST PROGRAMMER
I EVER MET**

SOME QUICK EVANGELIZING

(Thank you Dr. Kumar.)





Waste Segregation Guidelines

 1. Organic Waste	 2. Dry Waste	 3. Reject Waste
<p>(Do NOT use a plastic liner)</p> <p>Kitchen Waste Vegetable/fruit peels Cooked food/Leftovers Egg shells Chicken/fish bones Rotten fruits/vegetables Tissue paper soiled with food Tea bags/Coffee grinds Leaf plates</p>  <p>Garden waste * (small quantity only; from Apt) Fallen Leaves/twigs Puja flowers/garlands Weeds</p> 	<p>(Use only reusable bags for disposal)</p> <p>Plastic (Must be rinsed if soiled) Plastic covers/bottles/boxes/items Chips/toffee wrappers Plastic cups Milk/Curd packets</p> <p>Paper (Must be rinsed if soiled) Newspaper/Magazines Stationery/Junk mail Cardboard cartons Pizza boxes Tetrapaks Paper cups and plates</p>  <p>Metal Foil containers Metal cans</p> <p>Glass (handle with care) Unbroken glass bottles</p> <p>Other dry waste Rubber/Thermocol Old mops/Dusters/Sponges Cosmetics, Ceramics, Wooden Chips, Hair Coconut shells</p> <p>E-waste (handle with care) Batteries CDs/Tapes Thermometers</p> <p>Bulbs/tube lights/CFLs ** (hand over separately)</p>	<p>(Do NOT use a plastic liner)</p> <p>Sanitary waste (Use a newspaper for wrapping) Diapers/Sanitary napkins Bandages Condoms Nails Used tissues Medicines Swept dust</p>  <p>(Limited quantities of mixed waste is allowed, such as heavily soiled plastic or soiled paper)</p> <p>Sharps ‡ (small quantities only; wrap in newspaper and hand over separately) Razors/Blades Used syringes Injection vials</p>  <p>Construction debris/Inerts ¶ (Hand over separately) Rubble Paints Silt from drains Cement powder Bricks Flower pots</p> <p>Broken glass (wrap in newspaper)</p> 

* Garden waste from our campus grounds will be picked up separately.

** Hand over your fused tube-lights and bulbs separately. There is a separate bin for these items in the basement.

‡ There is a separate bin for the sharps items in the basement.

¶ Construction debris in large quantities will be charged extra per load.



Bruhat Bengaluru Mahanagara Palike
 Segregation at source is **MANDATORY**.
 Non-compliance will be penalized.

www. **2 BIN 1 BAG** .in



WHO AM I?

(AND WHY AM I HERE?)







Our primary goal is to find people and organizations to partner with who feel the same way. If you prefer working on open source, with open data, in open and transparent fields, we want to talk to you.

The capital of the LLP shall be [Rs 6,144]. The Partners hereto shall contribute in proportions to their respective capital contribution ratio.

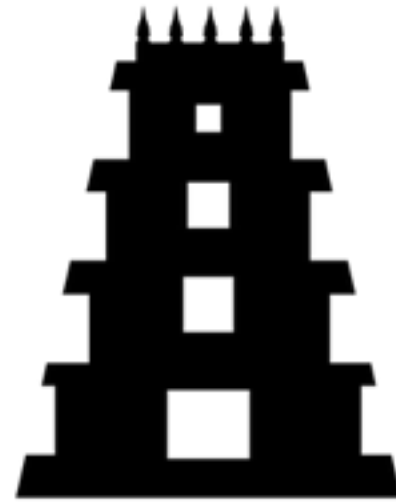
	Contribution	Ratio
First Party	[Rs 1024]	[16.66%]
Second Party	[Rs 1024]	[16.66%]
Third Party	[Rs 1024]	[16.66%]
Fourth Party	[Rs 1024]	[16.66%]
Fifth Party	[Rs 1024]	[16.66%]
Sixth Party	[Rs 1024]	[16.67%]

The LLP shall have a common seal to be affixed on documents as defined by Partners Executives under the signature of any of the Designated Partners.

Business of the LLP

(A) Main business:-

To design, develop, research, purchase, take on hire, import, export, sell, assemble, manufacture, test and distribute all types of hardware, software, electronic devices



WHO ARE YOU?

(AND WHY ARE YOU HERE?)

ALWAYS ASK WHY.

“FREEDOM”?

“OPEN”?



Jisanth Apus

- **SMART**
- **FAST**
- **DELIVERS**

NUMBERS

id

number

0

1

1

2

2

3

3

4

4

5

...

...

**IT ALL STARTS SO
INNOCENTLY.**

THE LOCAL CHANGE

± git status

On branch master

Your branch is up-to-date with 'origin/master'.

Changes not staged for commit:

modified: src/module/someone_already_touched_this.clj

Untracked files:

src/a_new_module/i_bet_other_people.clj

src/a_new_module/would_love_to_know.clj

src/a_new_module/what_you_are_doing.clj

OH, “DEV-OPS”

THE “LOCAL CHANGE” IS 2 FORMS OF HAVOC

- ▶ in **development**, you are creating a mystery ...and a fork
 - ▶ your codebase is stateful
- ▶ in **ops (non-dev)**, you are breaking the environment
 - ▶ your environments are stateful

IT JUST CAN'T BE TESTED.

Jisanth Apus

YES. IT CAN.

Call it “dependency injection” or any other name... but just make sure you inject your dependencies.

**WE ABSOLUTELY MUST HAVE
100% COVERAGE**

Jisanth Apus

NOT ALL FOOD IS NUTRITIOUS. NOT ALL TEST ARE VALUABLE.

```
± find src -name "*.cljs" | xargs wc -l
```

...

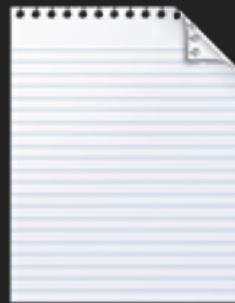
219 total

```
± find test -name "*.cljs" | xargs wc -l
```

...

946 total

**SPEAKING OF
TESTS.**



what's in the file?

**JISANTH LOVES IT WHEN WRITING
AN ASSERTION IS EASY.**

COPY/PASTE IS EASY.

WHAT'S WRONG WITH DEPENDENCIES?

```
> wc -l Gemfile
```

```
257 /Users/jisanth/work/next_facebook/Gemfile
```



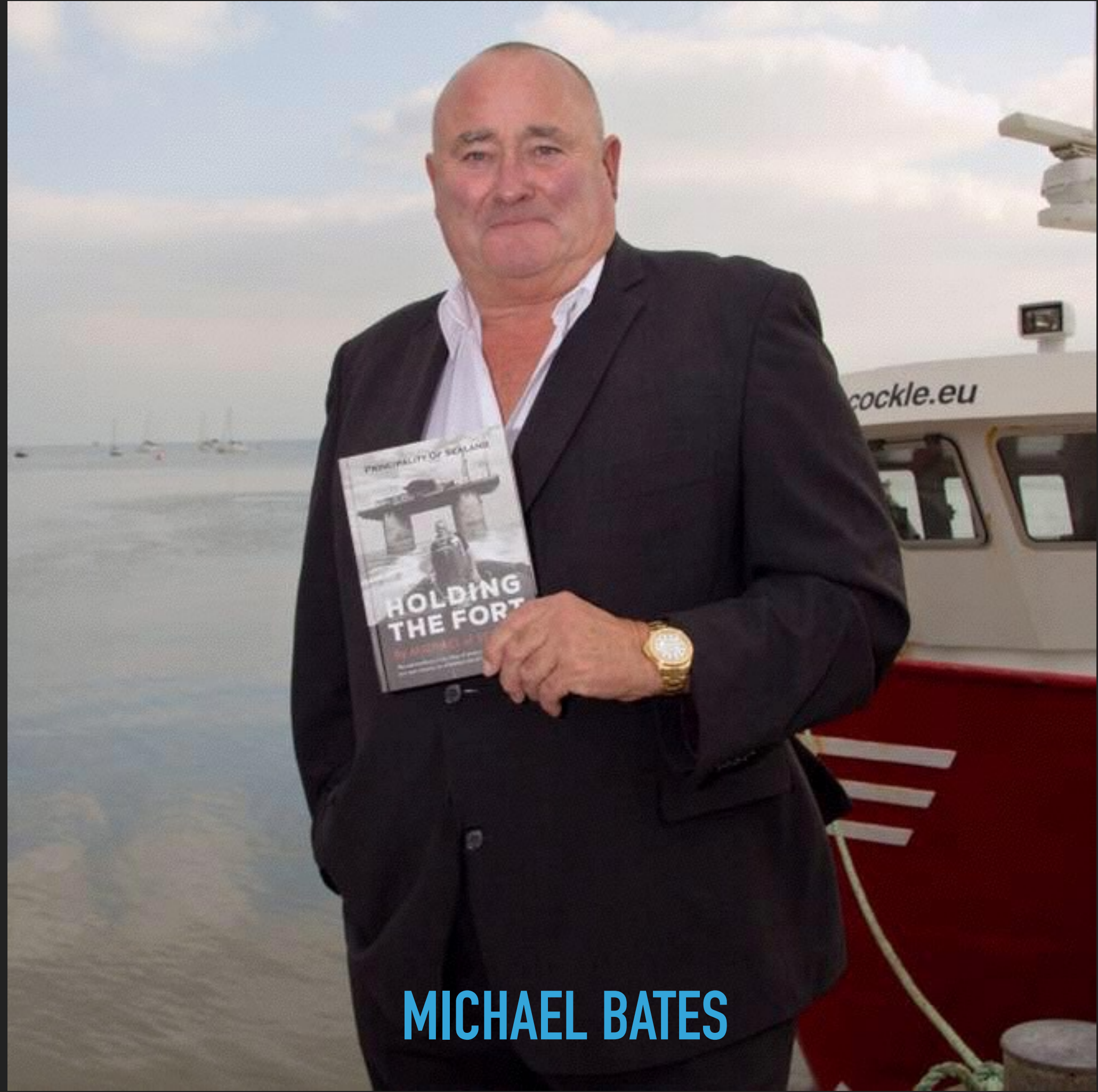
Librarian

WHAT'S GOOD ABOUT DEPENDENCIES?

```
> wc -l project.clj
```

```
8 /Users/jisanth/work/next_amazon/project.clj
```





MICHAEL BATES

IT'S OBVIOUS.

Jisanth Apus

**“EASILY PERCEIVED OR
UNDERSTOOD; CLEAR, SELF-
EVIDENT, OR APPARENT.”**

OBVIOUS EXAMPLES

- ▶ “never use that function”
- ▶ “what about N+1?”
- ▶ “we have to reuse that reference”
- ▶ **evidence** is your only friend, my friend.

**BEWARE THE
CERTAINTY OF JISANTH**

**CERTAINTY IS THE SEED
OF NEXT SEASON'S FAITH**



SCIENCE *VS* RELIGION IS NO CONTEST

NOT THAT FAITH.

IN COMPUTER SCIENCE,
FAITH IS THE ANTITHESIS OF
THE SCIENTIFIC METHOD.

STATE & TIME: YIN & YANG

- ▶ your datacenter – **your filesystem**
- ▶ configuration
- ▶ the code itself – **git**
- ▶ data formats – **evolution**
- ▶ serialization formats
- ▶ database schemas – **migration**
- ▶ public API schemas – **your client's perspective**
- ▶ private API schemas

IT GETS LESS
INNOCENT WITH TIME.

CONTEXT, SCOPE, ENVIRONMENT

```
Context ctx = new Context();
```

```
public int weight(Context ctx, List<int> vals)
```


AH, THE SINGLETON!

- ▶ hides as "environment"
- ▶ hides as "config"
- ▶ rots your brain.

HALF-BUILT CASTLES

- ▶ sqlite is not a queue.
- ▶ sqlite is not persistent.
- ▶ sqlite makes a poor persistent queue.
- ▶ truth be told, **all castles are bad** even if you finish them.

MAYBE WE SHOULD EXTRACT A FRAMEWORK?

- ▶ re: ReactJS “boilerplate”
- ▶ nope.
- ▶ when it needs to happen, you’ll know.
- ▶ (good intention, poor execution.)

EVENTS + CONFIGURATION = “FLEXIBILITY”

- ▶ Messaging: channels.xml vs Channels.cs
- ▶ OOP: Spring / Spring.NET “events”
- ▶ XML is not a programming language.
- ▶ YAML is not a programming language.
- ▶ JSON is not a programming language.
- ▶ you get the idea.

TO FIX A BROKEN WINDOW, BREATHE.

- ▶ broken build
- ▶ broken / commented tests
- ▶ TODOs
- ▶ HACKs
- ▶ habits and institutional knowledge
- ▶ “the business is too critical to fix this now.”

**WHEN INNOCENCE IS LOST
NET NEGATIVITY APPEARS.**

THE HORROR OF “DEVOPS” IN THE WILD

- ▶ treading water and moving files
- ▶ treading water... or crossfit?
- ▶ “ops” is about **writing and running scripts**. if a robot isn't carrying the heavy weight around, you're doing it wrong.

THE CROSSFIT DEVELOPER (AKA THE HERO)

- ▶ real heroes are usually silent and mostly go unnoticed
- ▶ if you are getting a lot of kudos, that means there was an event
- ▶ events are bad.
- ▶ non-events are the mark of consistency
- ▶ paradoxically, people will take you for granted



Steven Deobald

@deobald

crossfit developer (n): net-negative
programmer who splits his days between (1)
“fixing” bugs of his own creation (2) talking
about it.

**THERE ARE ONLY TWO
THINGS:
ONES AND ZEROES**

**THERE ARE ONLY TWO
THINGS:
CODE AND DATA**

**THERE ARE ONLY TWO
THINGS:**

VERBS AND NOUNS

THERE ARE ONLY TWO THINGS:
DATA (NOUN) APIS AND VERB
(RPC) APIS

OKAY, THERE'S REALLY ONLY ONE THING:
NOUN + RESTRICTED VERB APIS

(LOOKING AT YOU, REST)

WELL, THERE'S ONE OTHER THING.

**NOUN + PREPOSITION + RESTRICTED VERB
APIS**

(LOOKING AT YOU, SQL)

**BUT SQL IS JUST
DATA, SO ...**

**THERE ARE ONLY TWO
THINGS:
CODE AND DATA**

LET'S RECAP.

- ▶ “any sufficiently complicated program contains an ad hoc, informally-specified, bug-ridden, slow implementation of half of SQL” - Greenspun’s 10th rule, probably.
- ▶ 1 vs 0
- ▶ code vs data
- ▶ verb vs noun (“kingdom of nouns”)
- ▶ noun APIs vs verb APIs
- ▶ noun + verb APIs!
- ▶ noun + preposition + verb APIs! => SQL
- ▶ SQL is still just data.
- ▶ SQL is also programs.
- ▶ programs are data.
- ▶ whew.

PQL

- ▶ query API:
 - ▶ nouns: data & categories
 - ▶ verbs: actions
 - ▶ relational algebra
- ▶ PQL
- ▶ a parser
- ▶ a generator
- ▶ to generate? **SQL**.

**THERE REALLY, REALLY
ARE ONLY TWO THINGS:
CODE AND DATA**

**THE SCATTERED, NON-ATOMIC,
FORWARD-ONLY, 2-WEEK,
UNREVIEWED, UNTESTED PUSH TO
MASTER.**

**WHEN NET NEGATIVE
LOSES ALL MEANING,
GALACTUS APPEARS.**

**(DESTROYER OF
TEAMS)**

THE COMMANDER:

“NO. DON'T DO THAT.”

THE PASSION OF THE DILBERT

- ▶ swearing
- ▶ passive-aggression:
 - ▶ “do it that way if you want”
 - ▶ “if it matters so much to you, you can fix it”
- ▶ more time spent talking and debating than coding
- ▶ passion considered very harmful.
- ▶ (not= :passion :drive)

THE TEMÜJIN.
VINDICTIVE.
AUTHORITATIVE.
CORRECT.

THIS TALK ISN'T ABOUT SOMEONE ELSE.

**IT'S ABOUT ME (AND YOU, IF YOU WANT
IT TO BE).**

REFERENCES

- ▶ Stu Halloway, "Debugging with the Scientific Method": <https://www.youtube.com/watch?v=FihU5JxmnBg>
- ▶ "The Pragmatic Programmer": <http://www.amazon.com/The-Pragmatic-Programmer-Journeyman-Master/dp/020161622X>
- ▶ 2bin1bag.in
- ▶ learn.nilenso.com

github.com/nilenso/media

[.../presentations/worst-programmer](#)