ODENSE KOMMUNE

MICHAEL WULFF - ODENSE MUNICIPALITY

RPA – FOR FUN & PROFIT

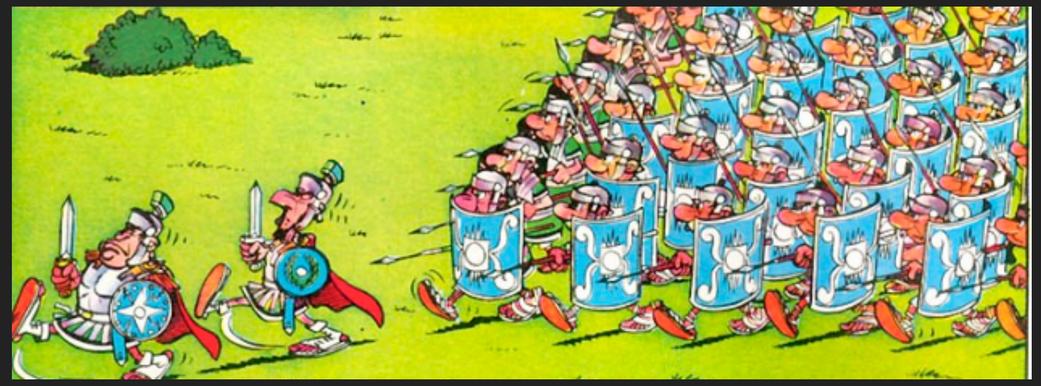
THOUGHTS FROM 5 YEARS OF BUILDING RPA PROCESSES

Michael Wulff

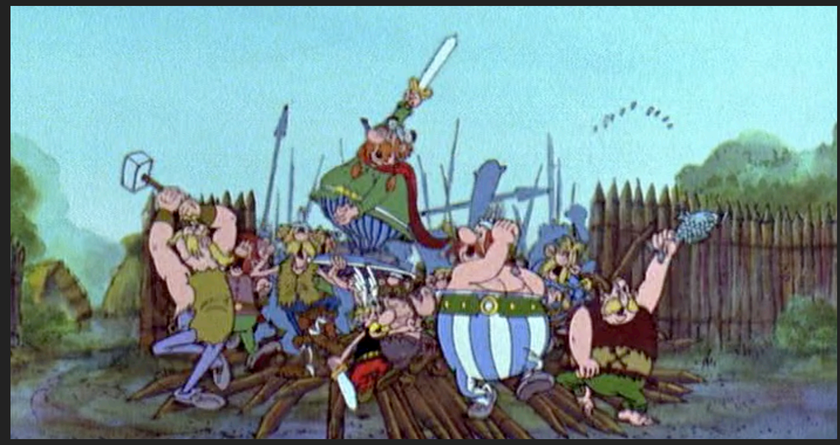
OUR TEAM AND OUR VISIO… MISSIO.. STATE.. AHH WHATEVER…

- 6 person team
  - 4 developers aka the code monkeys
  - 2 process consultants aka the people people.
- True scrum team
- We can go anywhere in the municipality
- We can choose the best solution

Be one of the top 10 RPA teams in the EU



What Management thinks RPA looks like

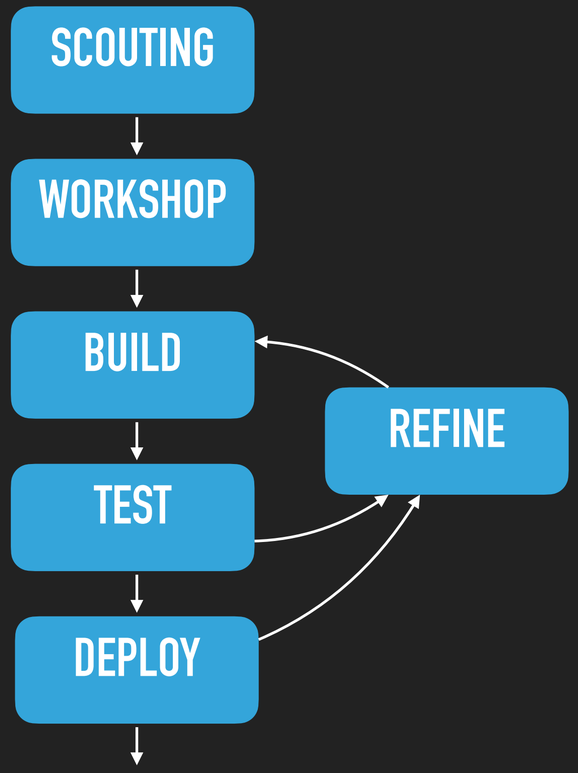


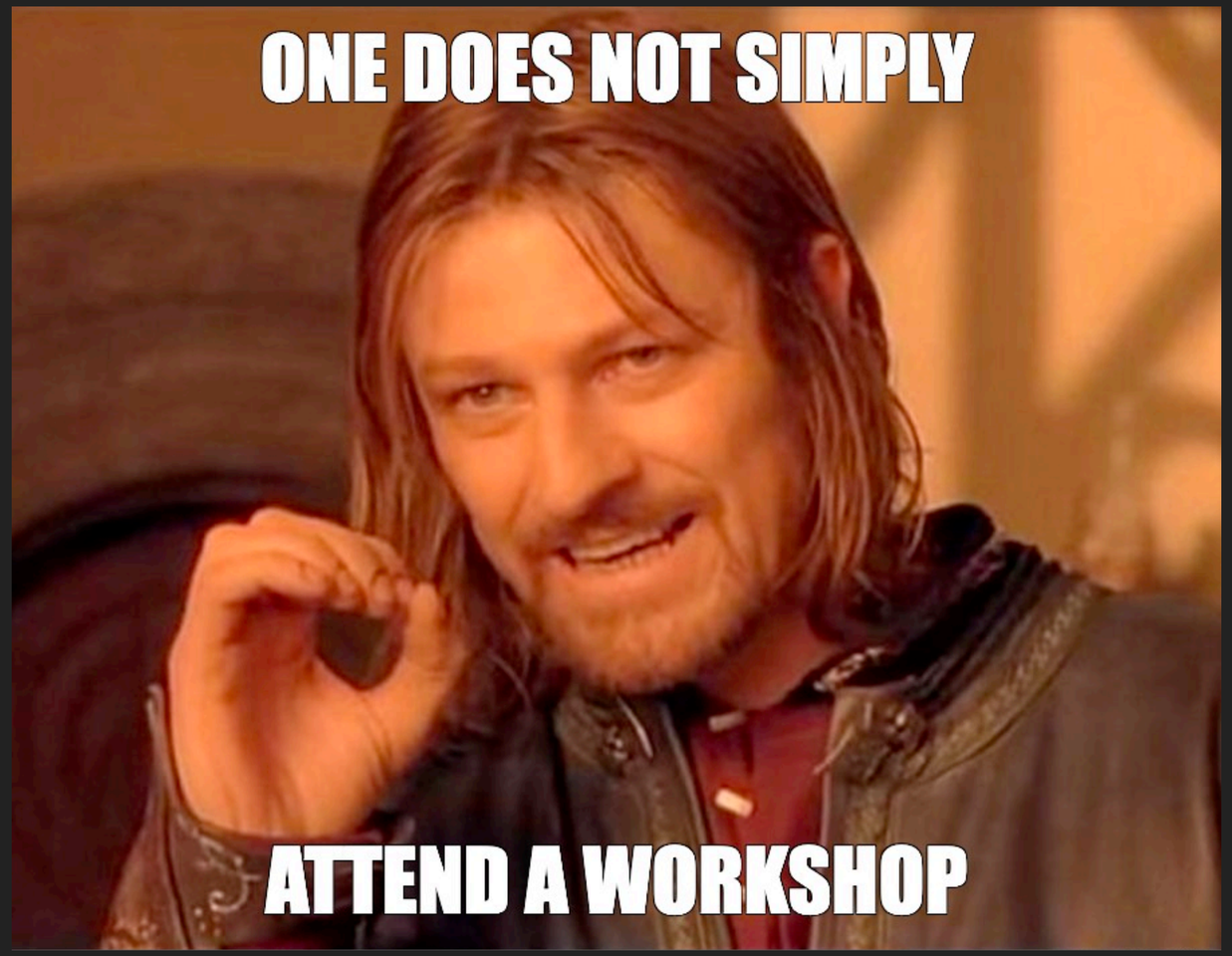
What RPA really looks like

RPA lives in the real world with all of the chaos of everyday life, work and data

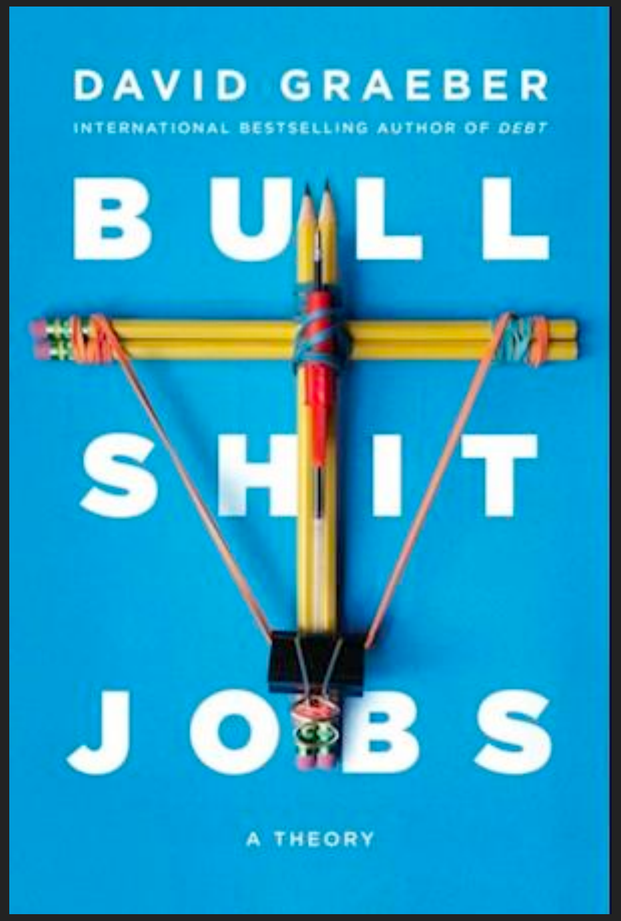
HOW WE SOMETIMES WORK…

- Businesscases are often “rainbows and roses”
- Managers rarely, if ever, know how to do the work
- Managers rarely know how much work there is
- 2 employees when asked will agree 80% about what the work is
- Try to think outside the box

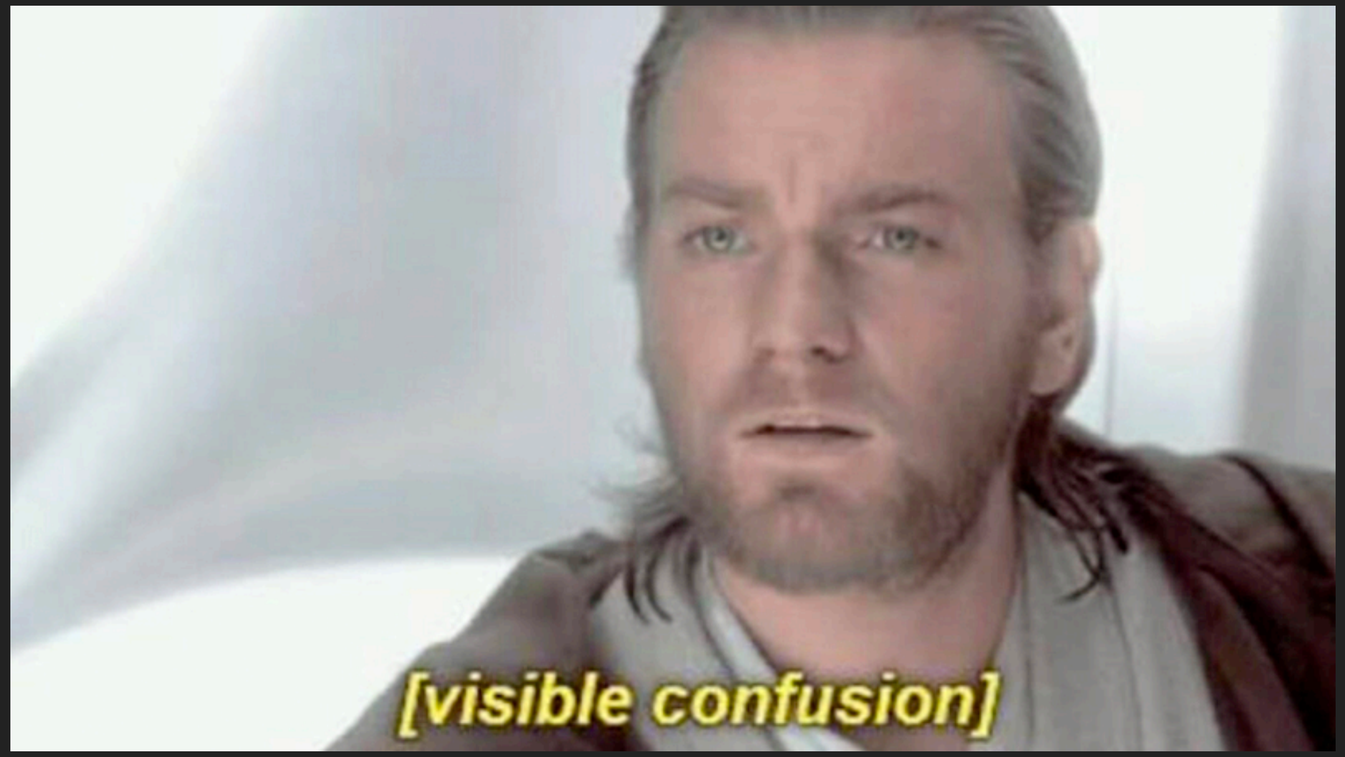



ONE DOES NOT SIMPLY
ATTEND A WORKSHOP

BEFORE WE START AUTOMATING STUFF



- Why do we do this?
- Should we do this?
- Can we just decide to not do this?
- Is there a better way?
- Does automating this make sense?
- Can it be automated?

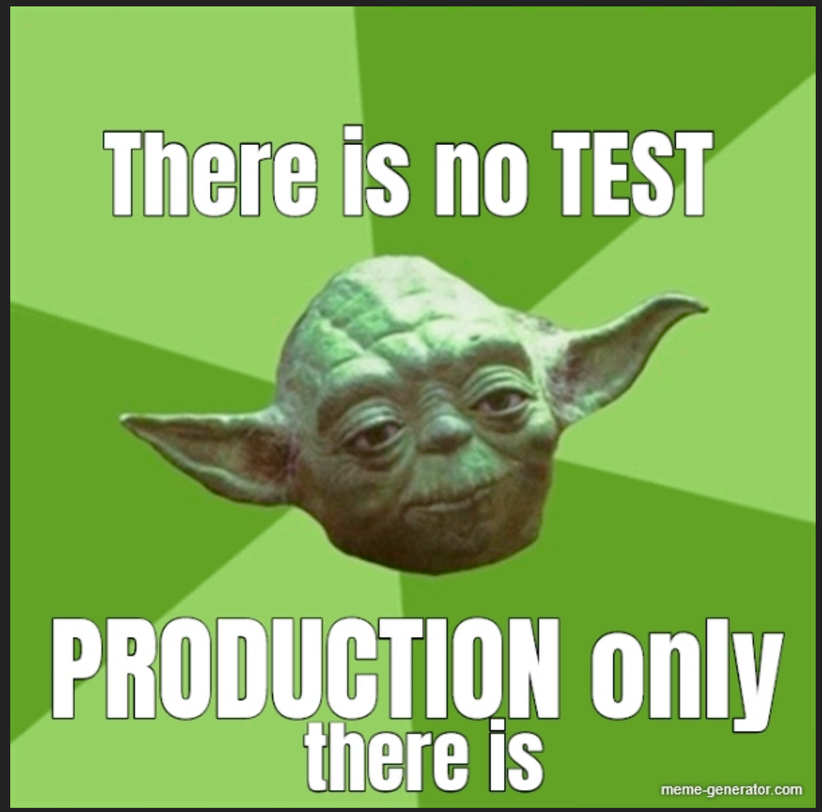

[visible confusion]

YOU KNOW… MAKE COMPUTER BEEP BEEP BOB

- Standardize your code
- Build for reusability
- Balance reusability with yagni
- Build fast
- Debugging you is not as smart as programming you
- RPA code lives hard and often dies young

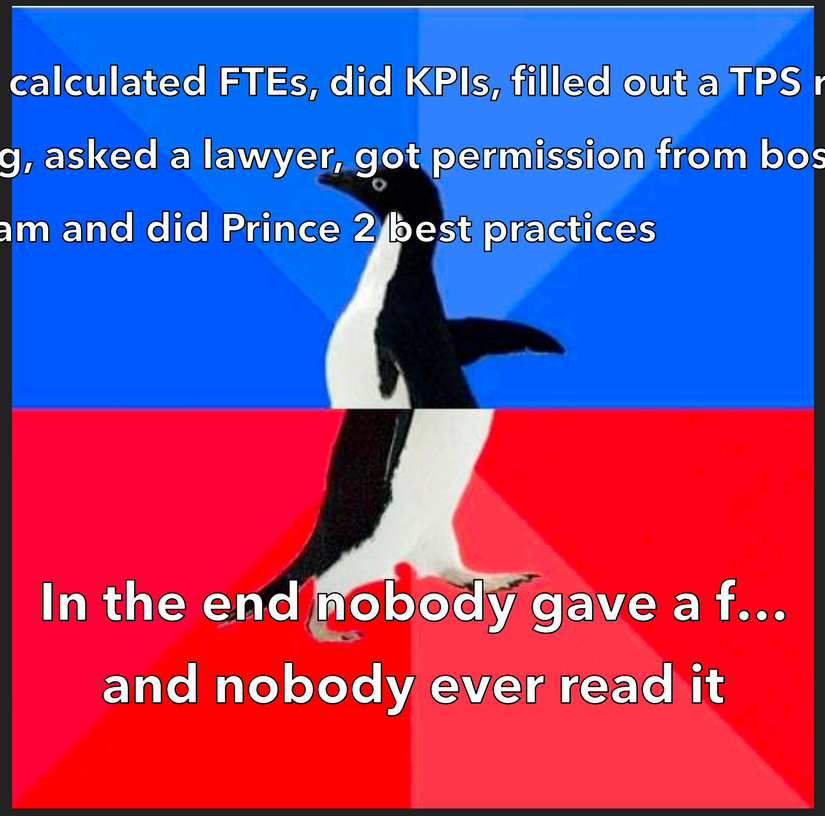
BE BRAVE.. IT WILL ALL BE OK..

- Get to testing fast
- Everyone has a test environment
- Some have a separate live environment
- Assume that users have no time or interest in testing
- Assume some users have A LOT of time for testing
- The best tests often happens at the users desk
- If it seems to work evaluate risk and deploy it
- No.. Seriously… If it is not that risky just deploy it



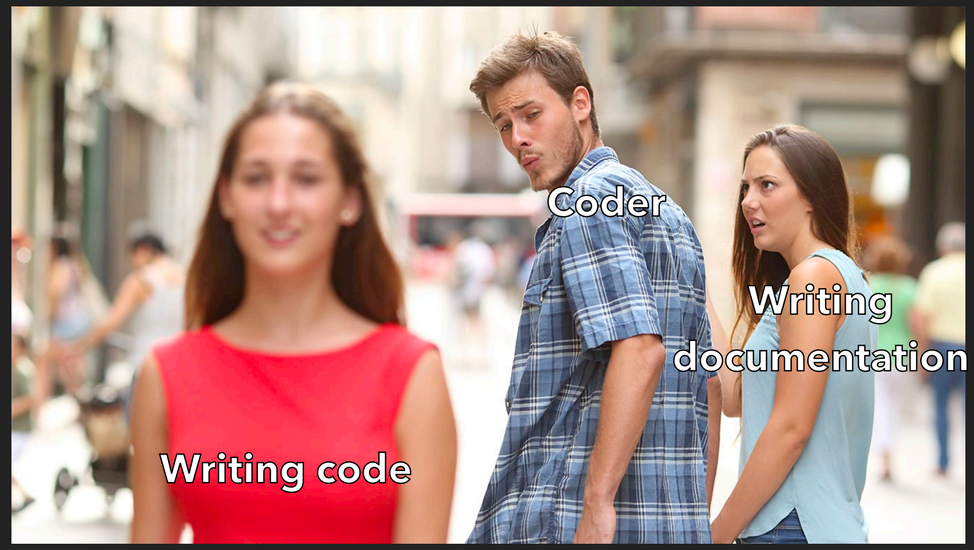
ON WASTING TIME

I wrote a businesscase, calculated FTEs, did KPIs, filled out a TPS report, had a meeting, documented everything, asked a lawyer, got permission from boss, did user stories, prepared a gantt diagram and did Prince 2 best practices



ON DOCUMENTATION

- Accept that the code is the truth
- Only do a very high level documentation
- 1 diagram and 1 A4 page max.
- Heavily standardize your code
- Build for reusability
- The best code is no code



THE LAST GUY THAT TRIED TO DOCUMENT EVERYTHING AND KEEP IT UP TO DATE.

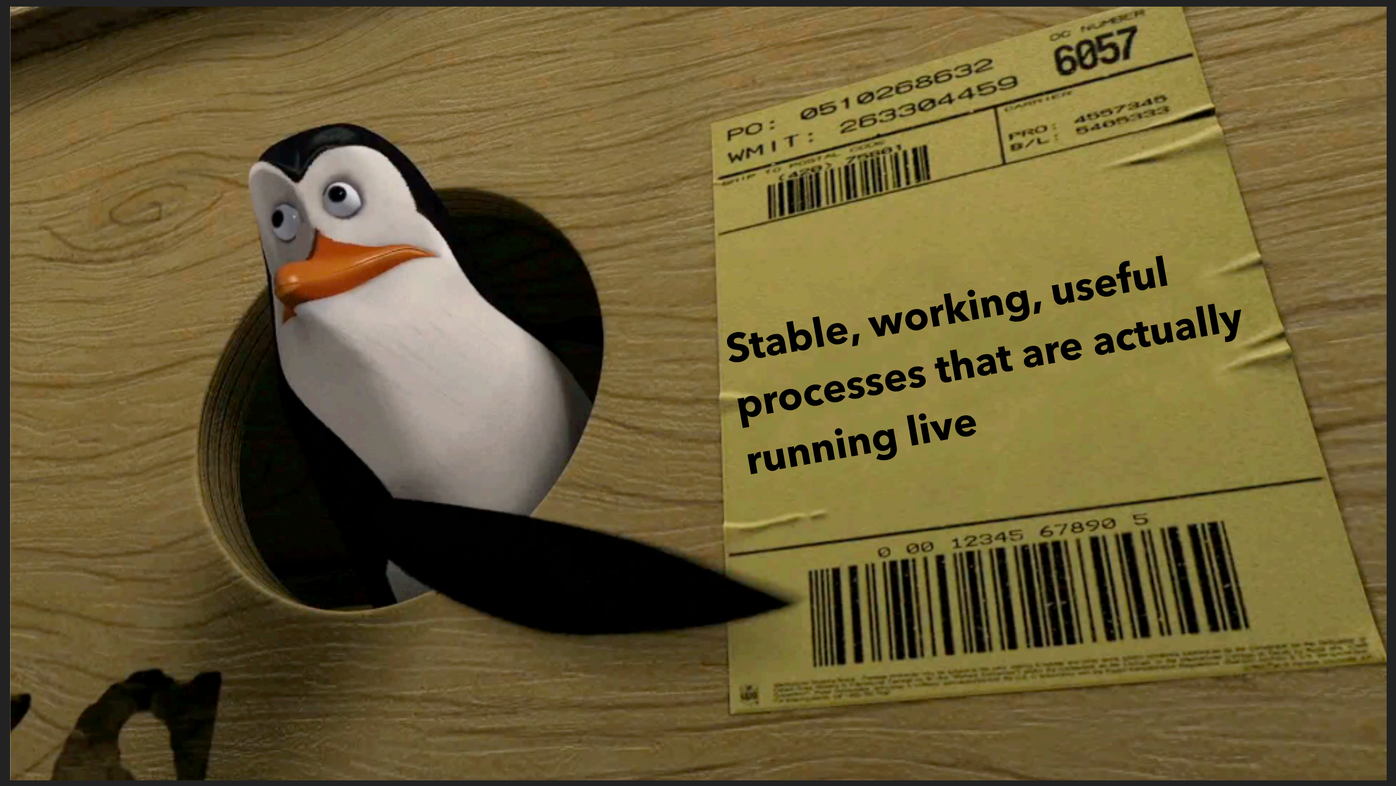
ON OPERATIONS

- Dogfooding is paramount
- If you can code it, you can support it
- Incentivizes stability over speed
- Accept that sometimes stuff just breaks
- It’s often not that important
- Operations should stay flat over time
- Release notes are worthless
- Everyone is responsible

EMPLOYEE TURNOVER IS A ....

- No code monkey likes timesheets
- They like pointless timesheets even less
- Company wide meetings can and should be ignored
- Keep the problems interesting
- Reserve time so they can go back and pay technical debt
- Accept that butt-in-chair or l.o.c is a terrible measure of anything
- Empower them to make decisions

ON WHAT COUNTS IN THE END



QUESTIONS?