Secrets of a Strong Engineering Culture

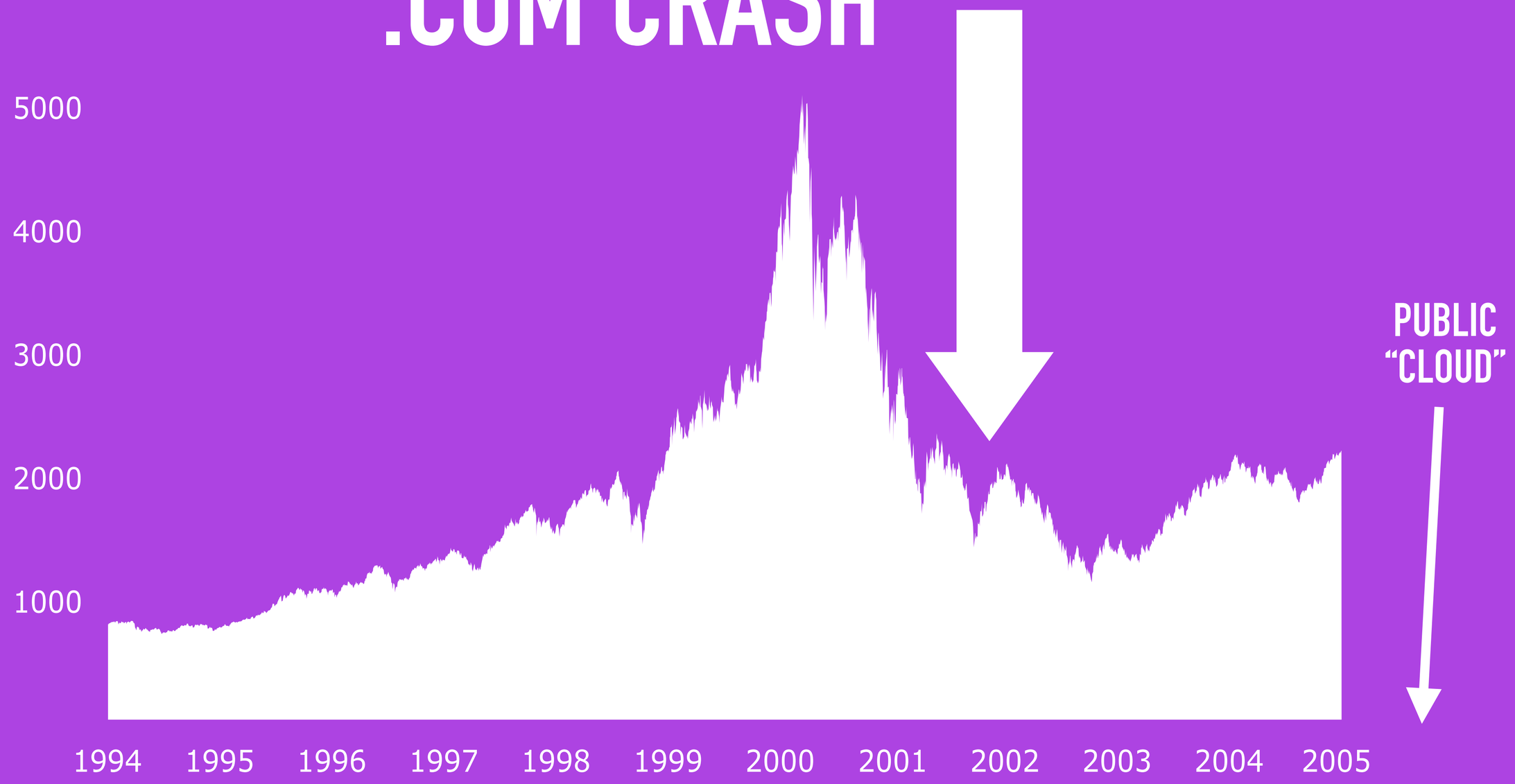
QCON London (Mar 2020)

@patkua

A TALE OF TWO COMPANIES

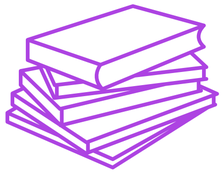
@patkua

.COM CRASH

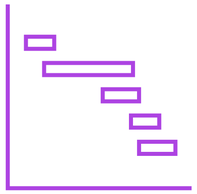


@patkua

ENTERPRISE A

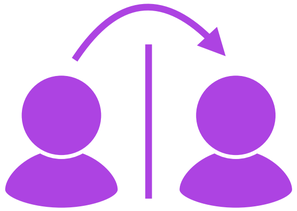


200+ PAGE DESIGN DOCUMENTS



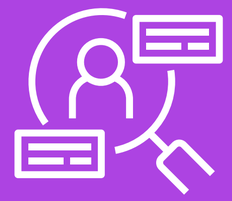
2 YEAR PROJECT PLANS

OPINIONS



HAND OFFS

STARTUP A



USER RESEARCH & TESTING

2 WEEK RELEASES

MONTHLY “IDEAS"

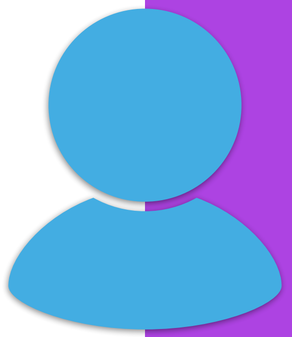
PRODUCTION SUPPORT

@patkua

ENTERPRISE A

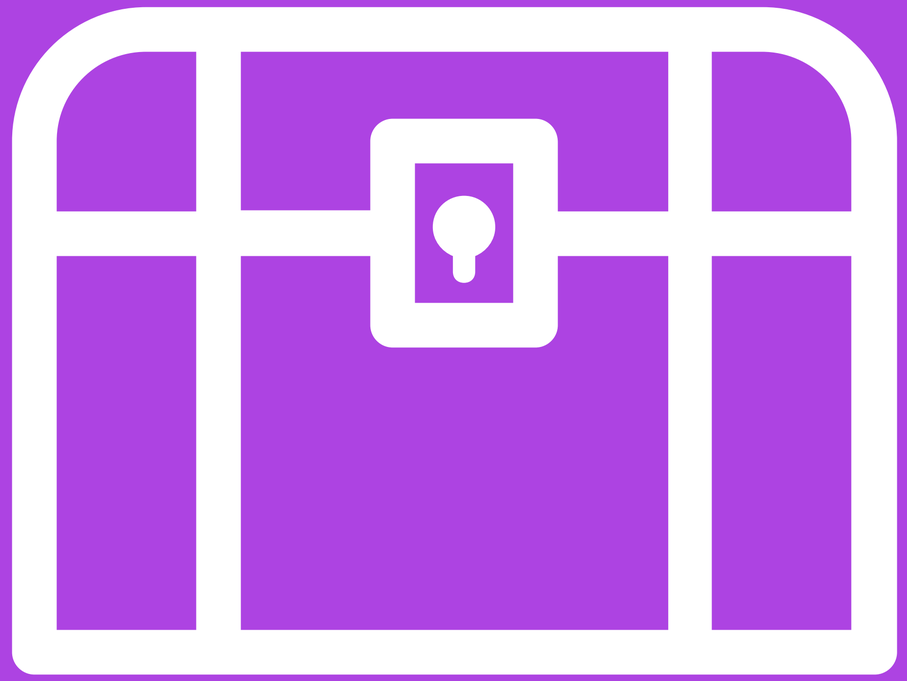
STARTUP A

???



@patkua

QUEST FOR SECRETS



@patkua

YOUR GUIDE FOR TODAY

Patrick Kua
~20 years experience

Agile Software Development

Organisational Change

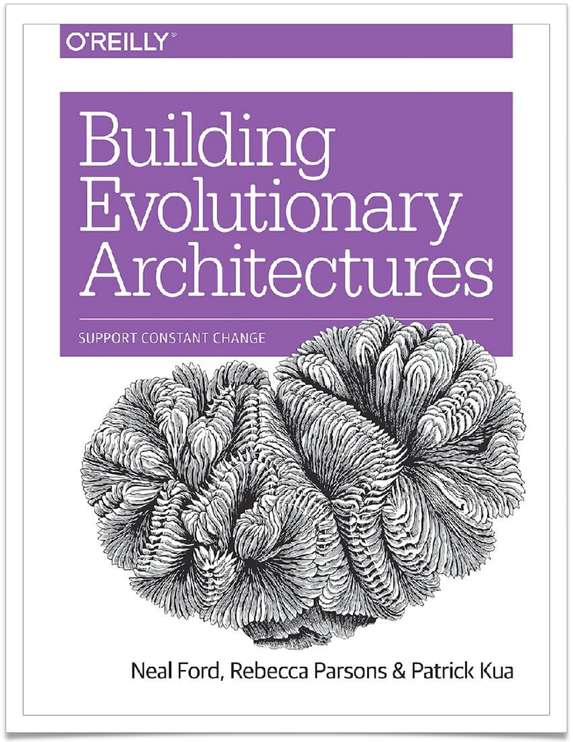
Systems Thinking

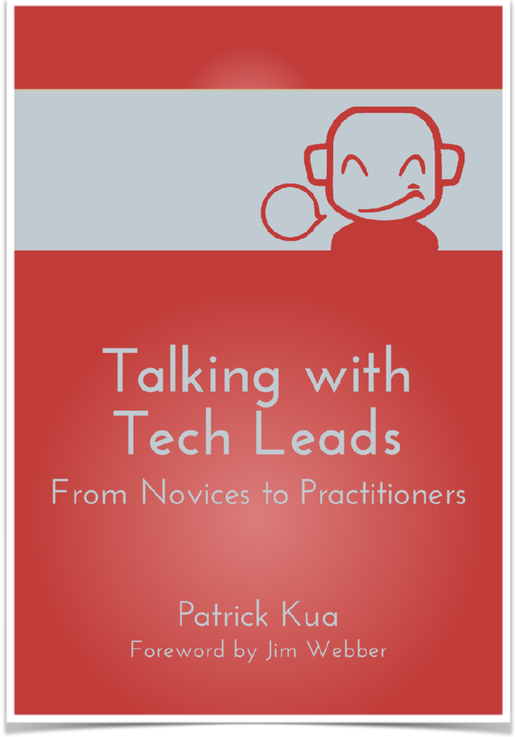
Technical Leadership

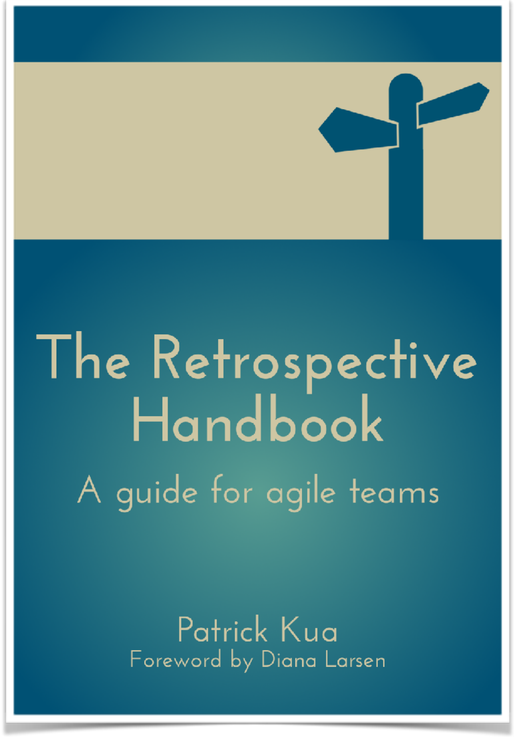
#Architect #Developer #Coach #Leader #CTO

#Life-long learner #Author #Speaker

@patkua







thekua.io/evolarch thekua.io/twtl thekua.io/retrobook

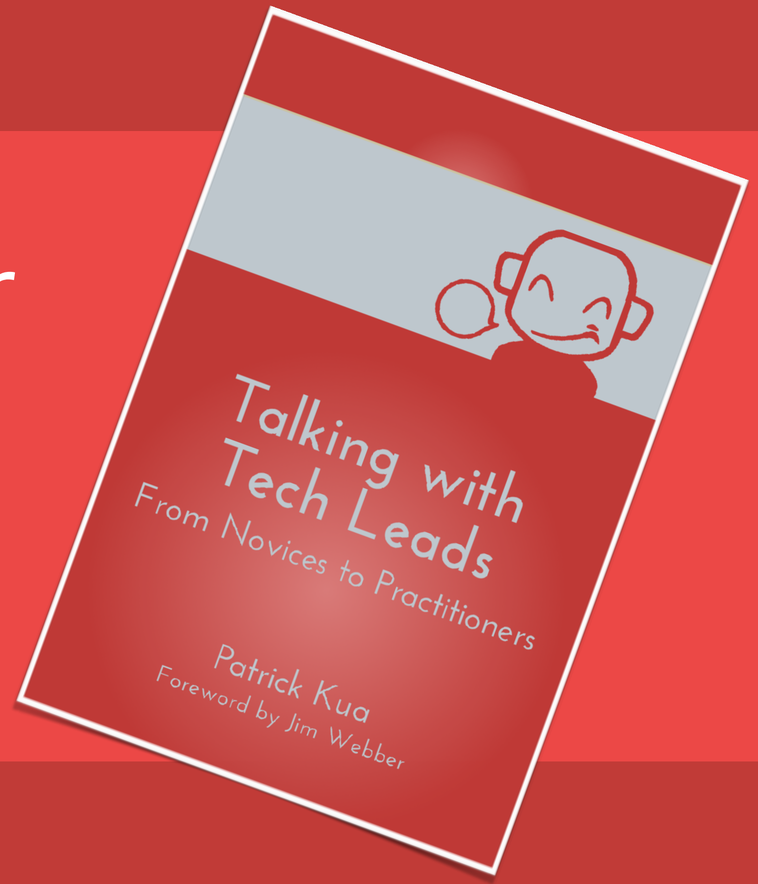
#Architect #Developer #Coach #Leader #CTO

#Life-long learner #Author #Speaker

@patkua

Course: Tech Lead Skills for Developers

http://patkua.com/tlcourse



@patkua

Level Up

A curated newsletter for leaders in tech

http://levelup.patkua.com

@patkua

- WHY?
- 3 SECRETS
- HOW

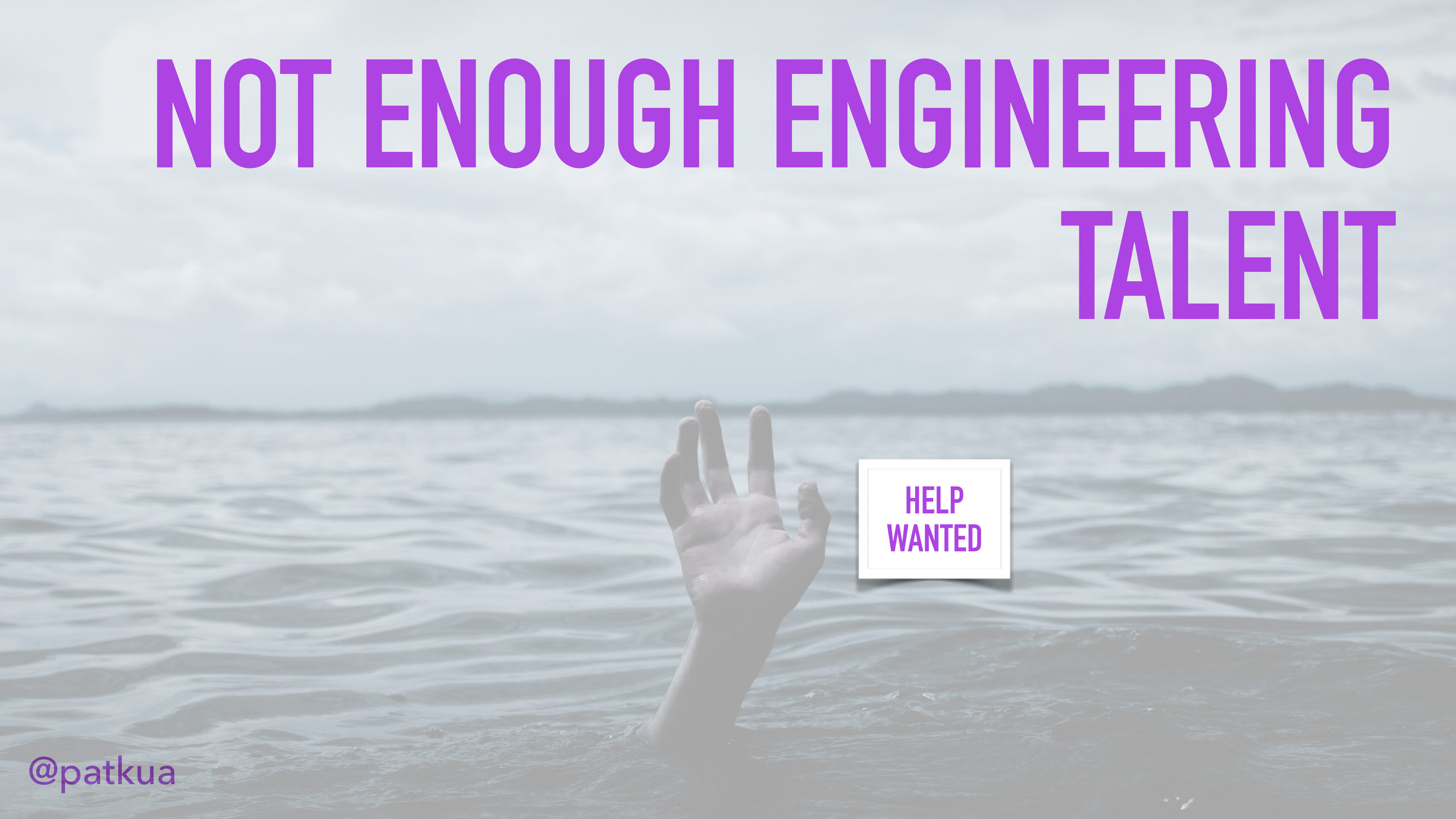
- WHY?
- 3 SECRETS
- HOW

@patkua

ENGINEERING TALENT IS LIMITED

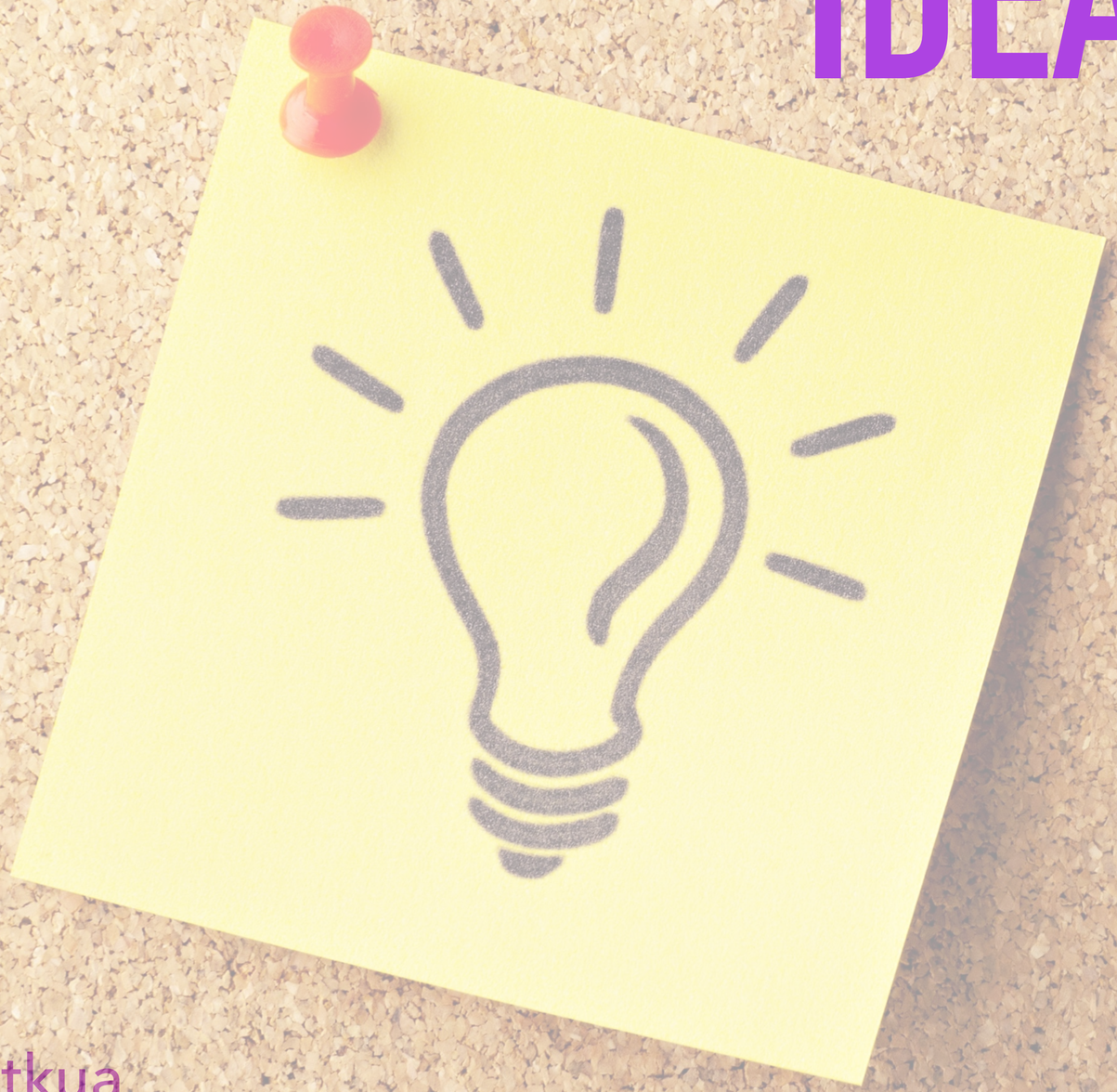
@patkua

NOT ENOUGH ENGINEERING TALENT



@patkua

IDEAS ARE CHEAP



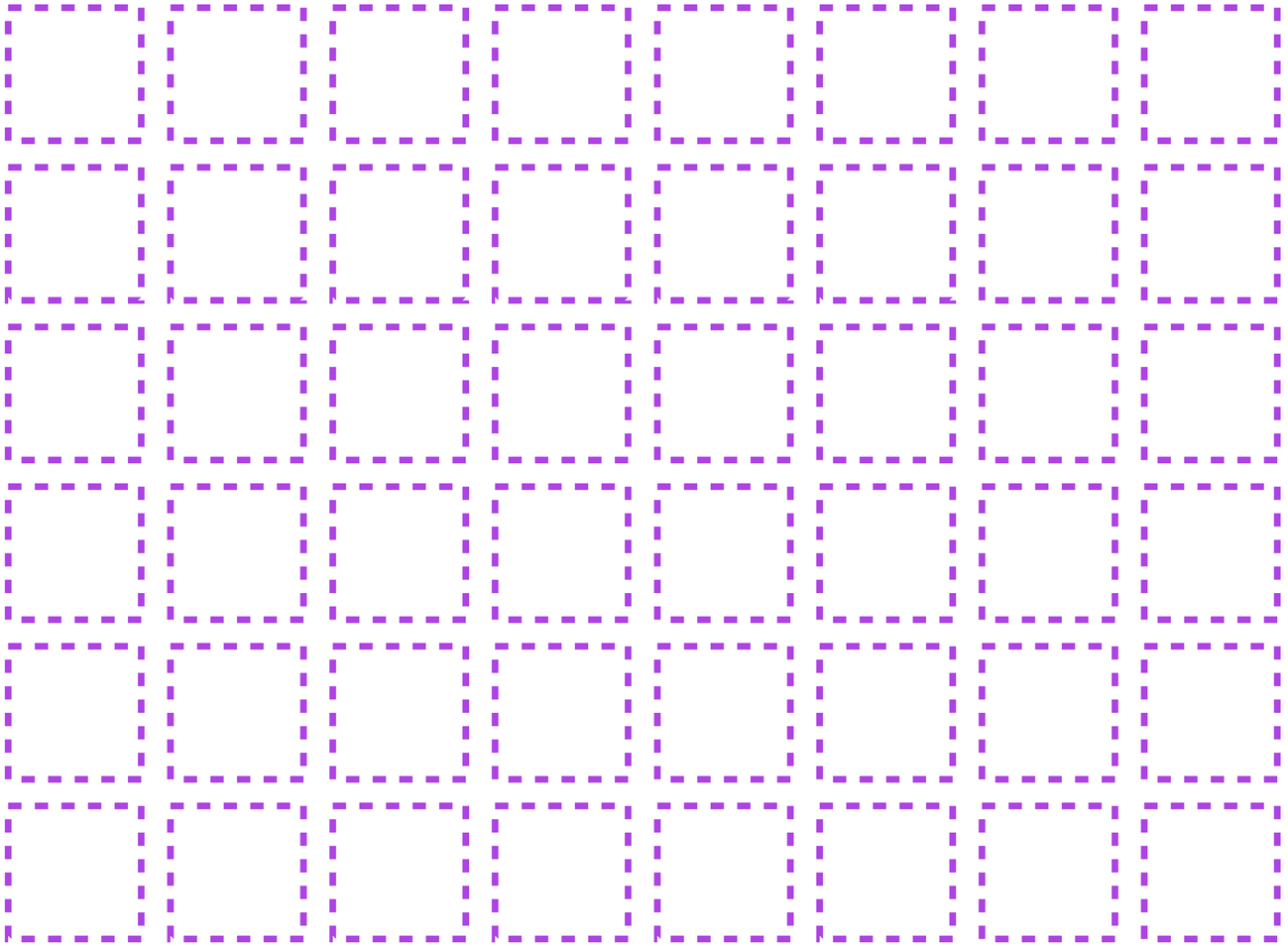
...AND EASY

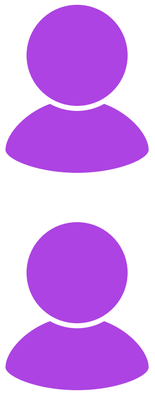
SOFTWARE IS EVERYWHERE

...AND GROWS

@patkua

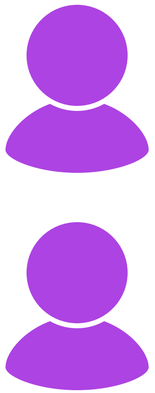
Developer Vacancies





Developers

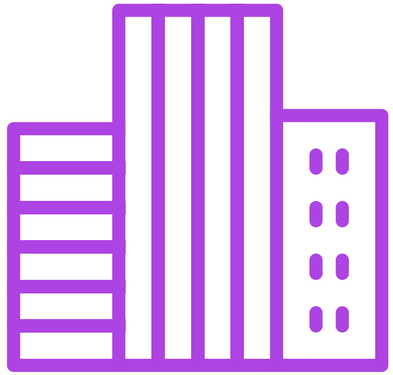
Developer Vacancies



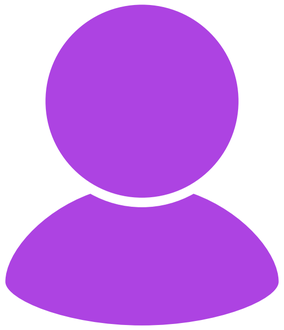
Developers

COMPANY A

COMPANY B

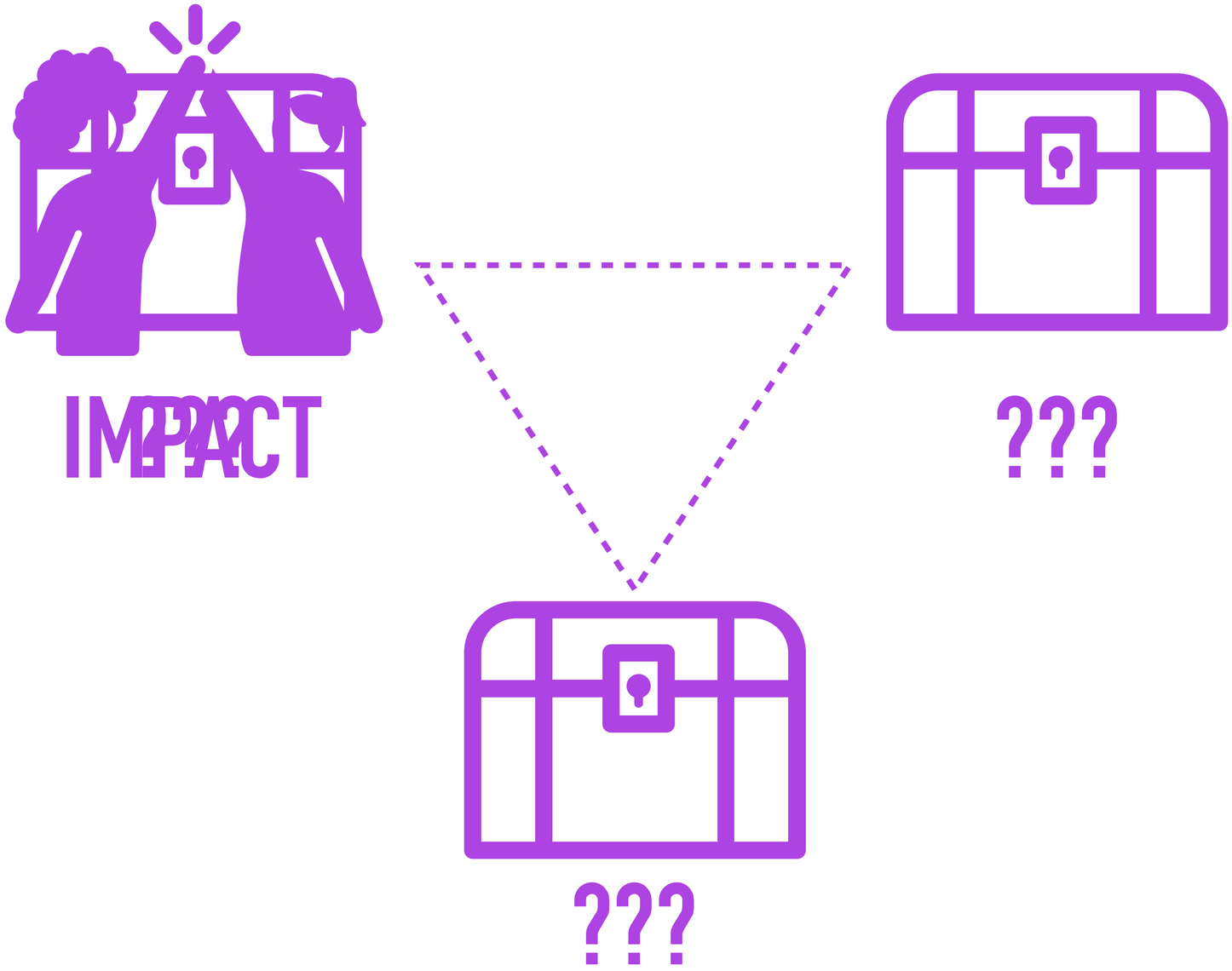


???



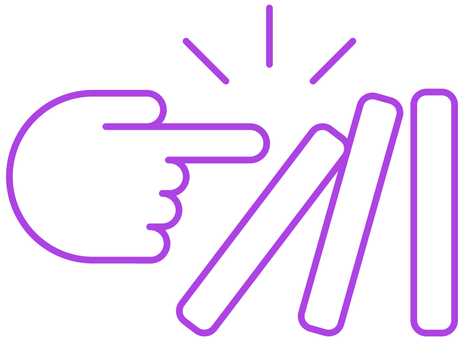
- WHY?
- 3 SECRETS
- HOW

@patkua

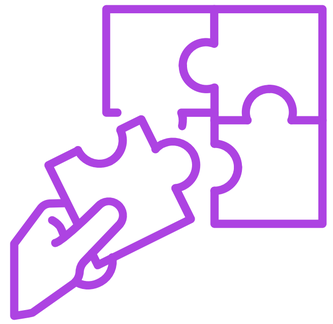

IMPACT
???
???

WHAT DOES IT MEAN?

Customer Impact



Sees the outcome



Solved the puzzle

TRAPS

Customer Impact

- Too far from the customer
- “Feature factory” (what not why)
- Too solution focused

TRAPS

Sees the outcome

- Too many handoffs
- Slow feedback loops
- Multiple projects

IMPACT

@patkua

TRAPS

- Outputs over outcome
- Just delivering to plans
- Lack of celebration/feedback

Solved the puzzle

IMPACT

@patkua

PRINCIPLE

Cultivate **impact** by

reducing feedback loops

@patkua

PRINCIPLE

act by

edback

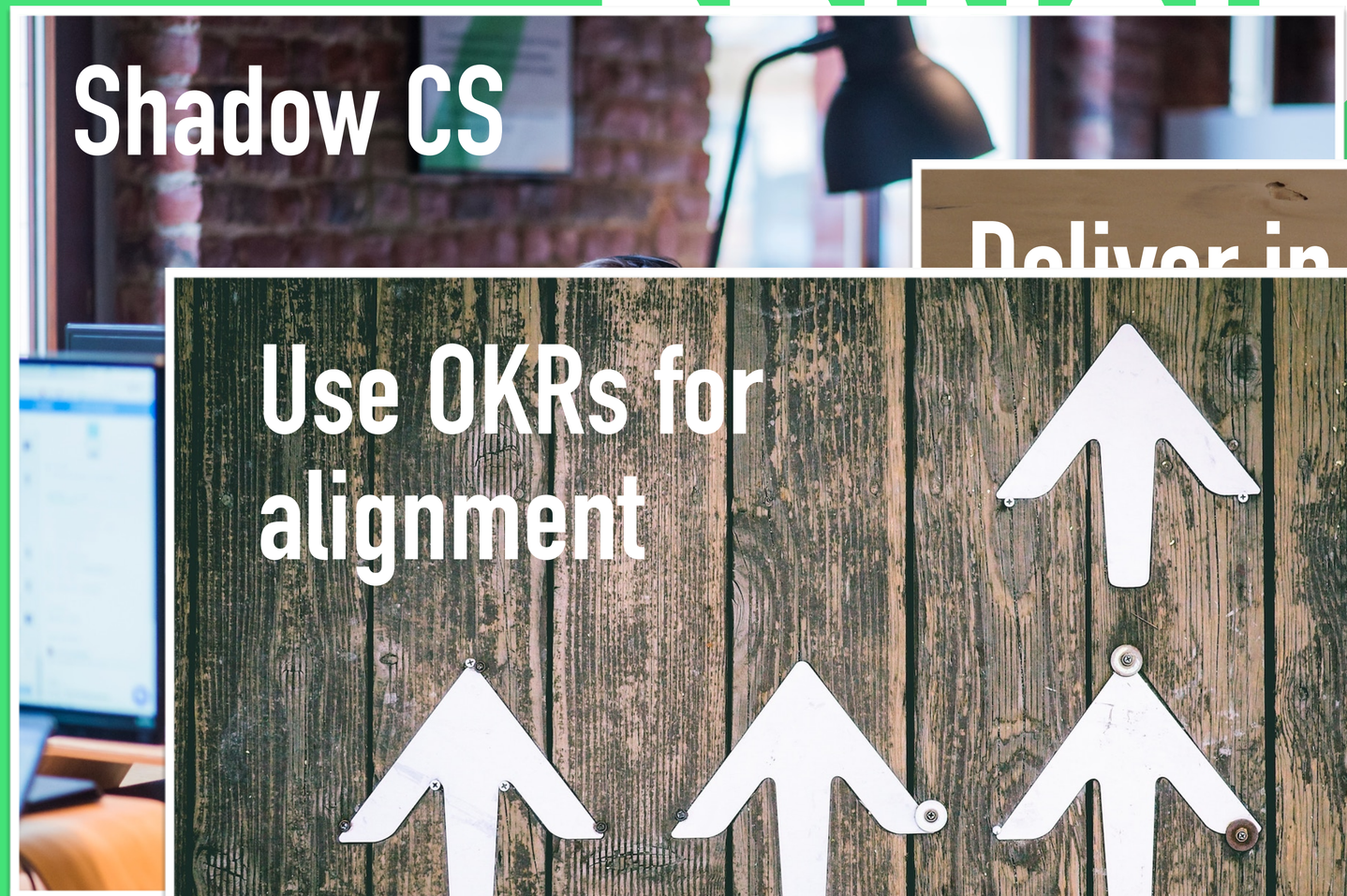
loops

@patkua

@patkua

IMPACT

PRINCIPLE

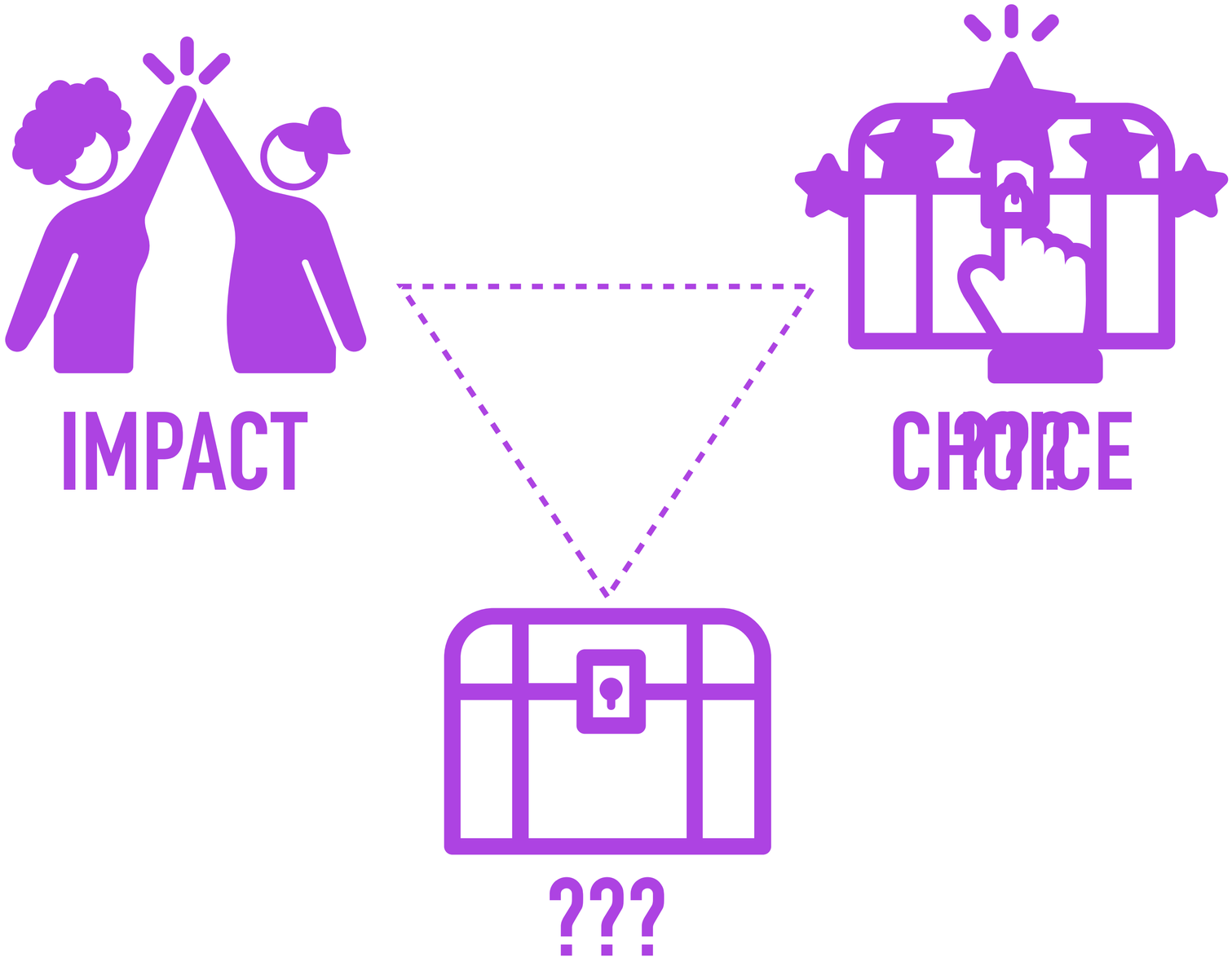


IMPACT

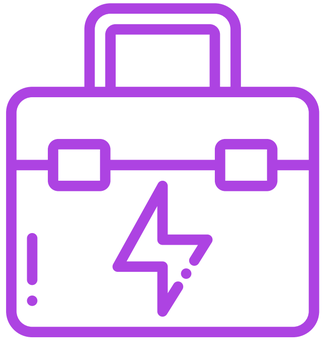
@patkua

@patkua

IMPACT


IMPACT
CHOICE
???

WHAT DOES IT MEAN?



Solution
Choice

How to
Work

Degrees of
Freedom

TRAPS

Solution Choice

- “Mandatory” frameworks
- Involved late in the process
- Output pre-determined

TRAPS

How to Work

- Single way of working
- Prescriptive processes
- Optimising for an individual

TRAPS

- Heavyweight rules
- Too much freedom
- Autonomy vs Alignment
- Unclear decision making process

Degrees of Freedom

PRINCIPLE

Cultivate **choice** by

making the right thing

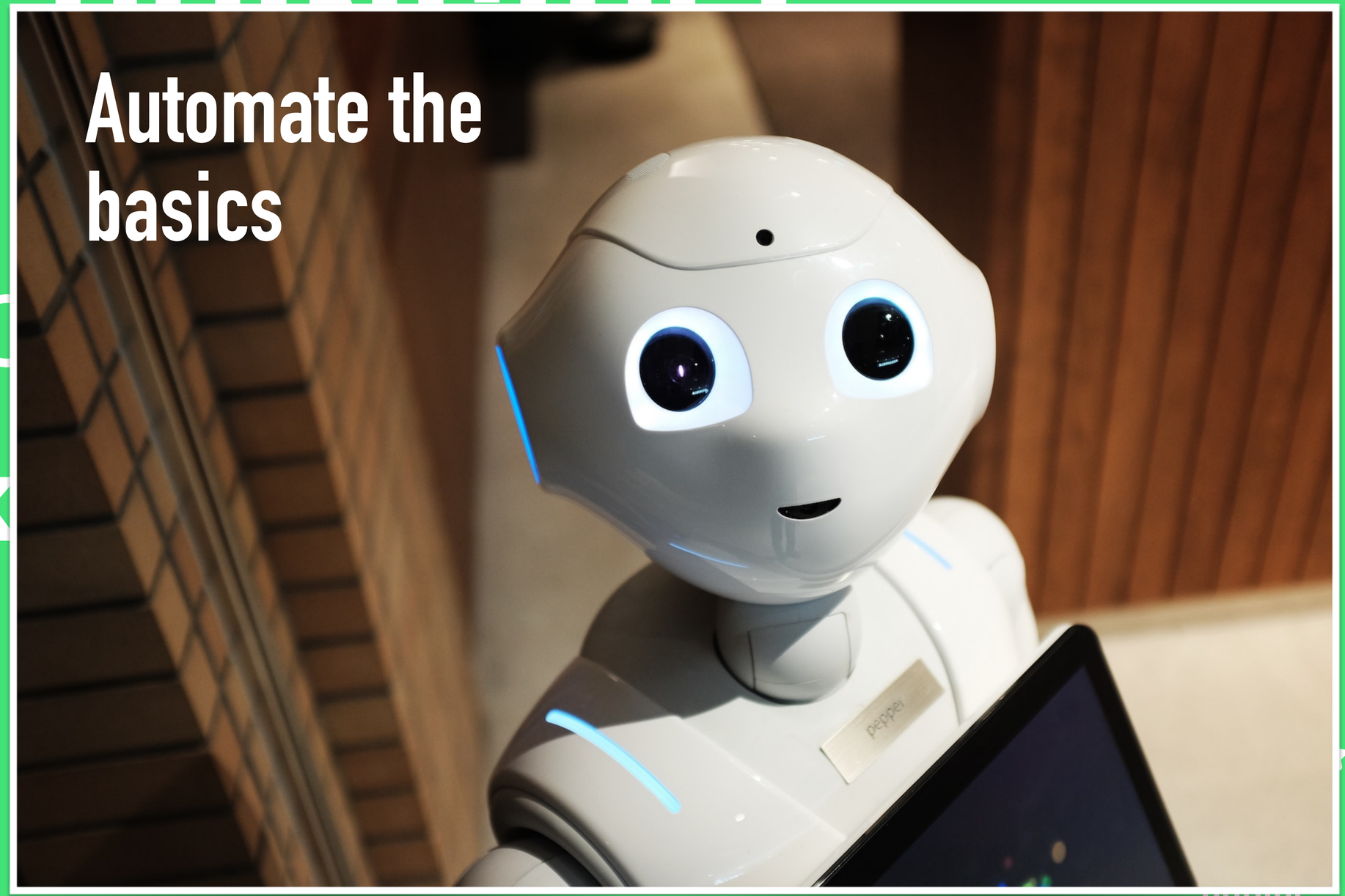
easy

@patkua

PRINCIPLE

C

mak



choice

@patkua

Prefer principles over rules

The Twelve Factors

I. Codebase
One codebase tracked in revision control, many deploys

II. Dependencies
Explicitly declare and isolate dependencies

III. Config
Store config in the environment

IV. Backing services
Treat backing services as attached resources

V. Build, release, run
Strictly separate build and run stages

VI. Processes
Execute the app as one or more stateless processes

VII. Port binding
Export services via port binding

VIII. Concurrency
Scale out via the process model

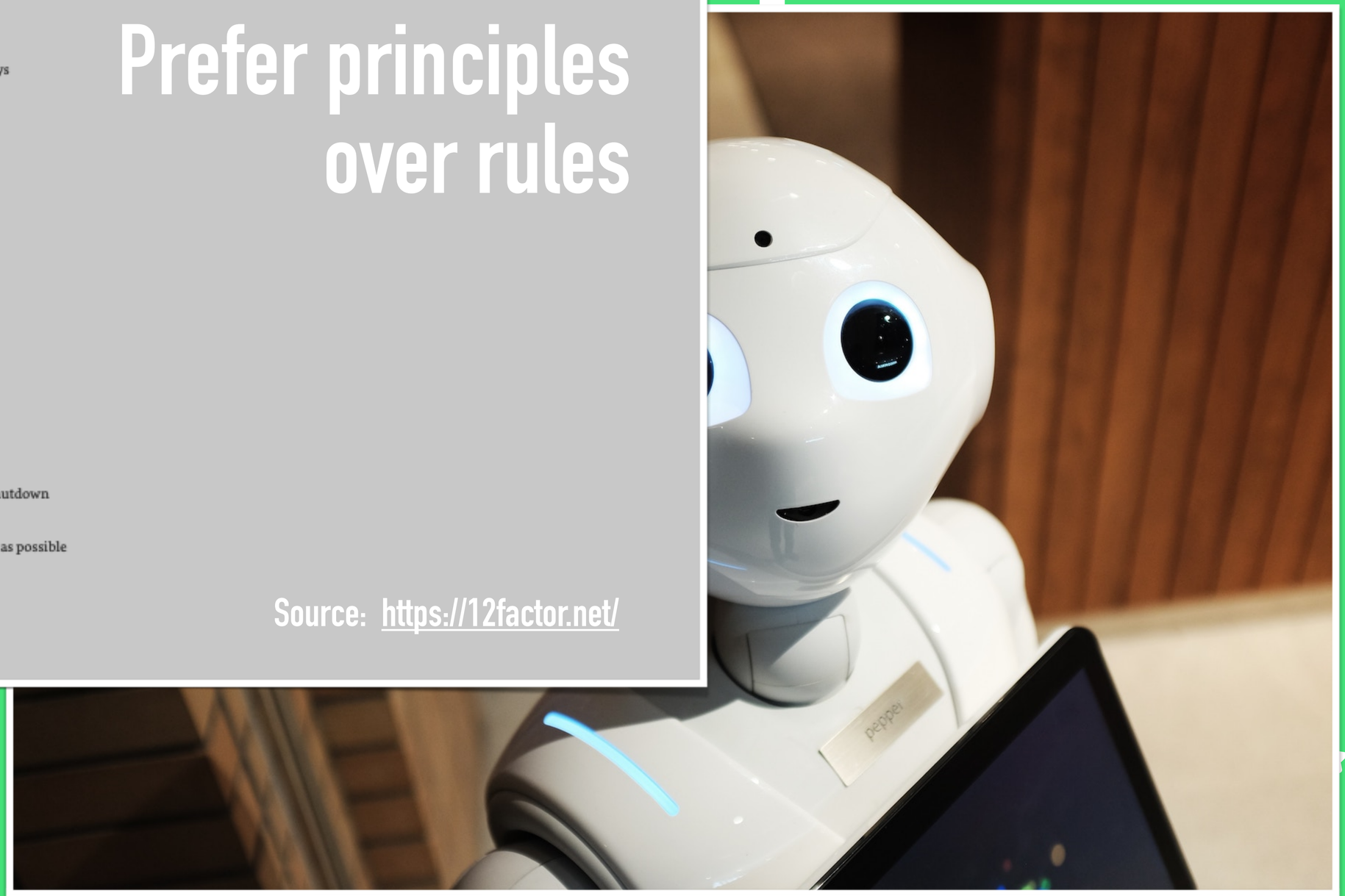
IX. Disposability
Maximize robustness with fast startup and graceful shutdown

X. Dev/prod parity
Keep development, staging, and production as similar as possible

XI. Logs
Treat logs as event streams

XII. Admin processes
Run admin/management tasks as one-off processes

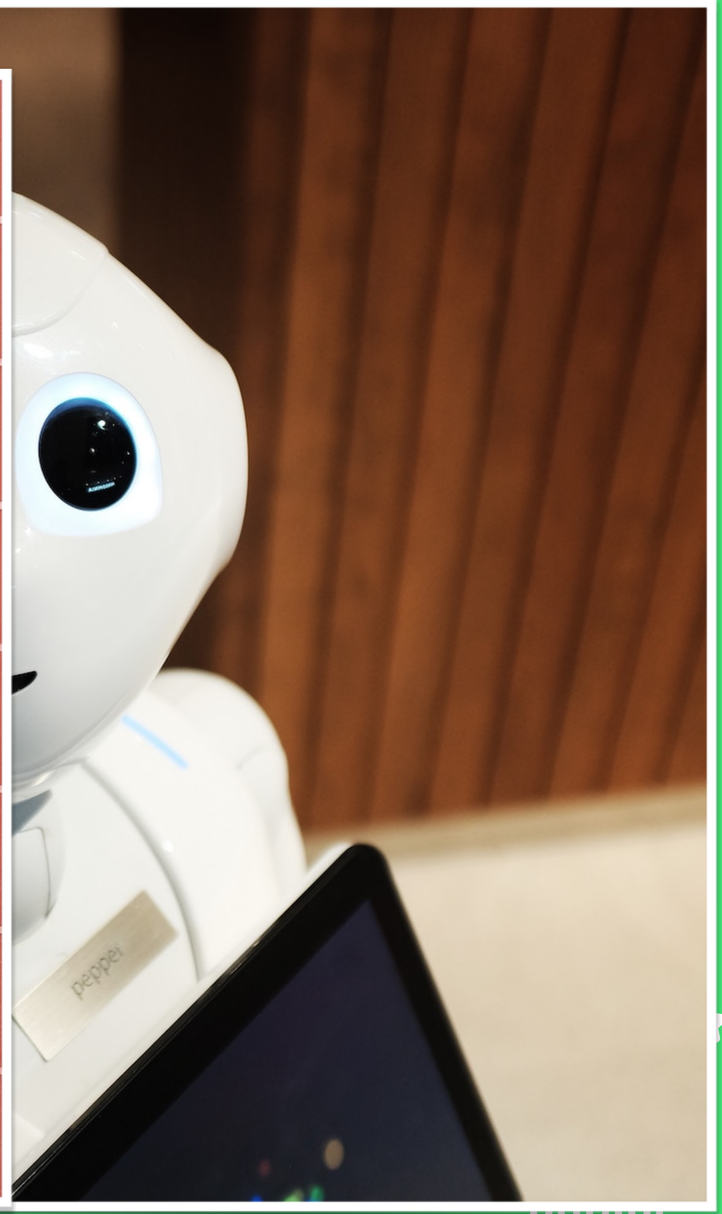
Source: https://12factor.net/



THE TWELVE FACTORS

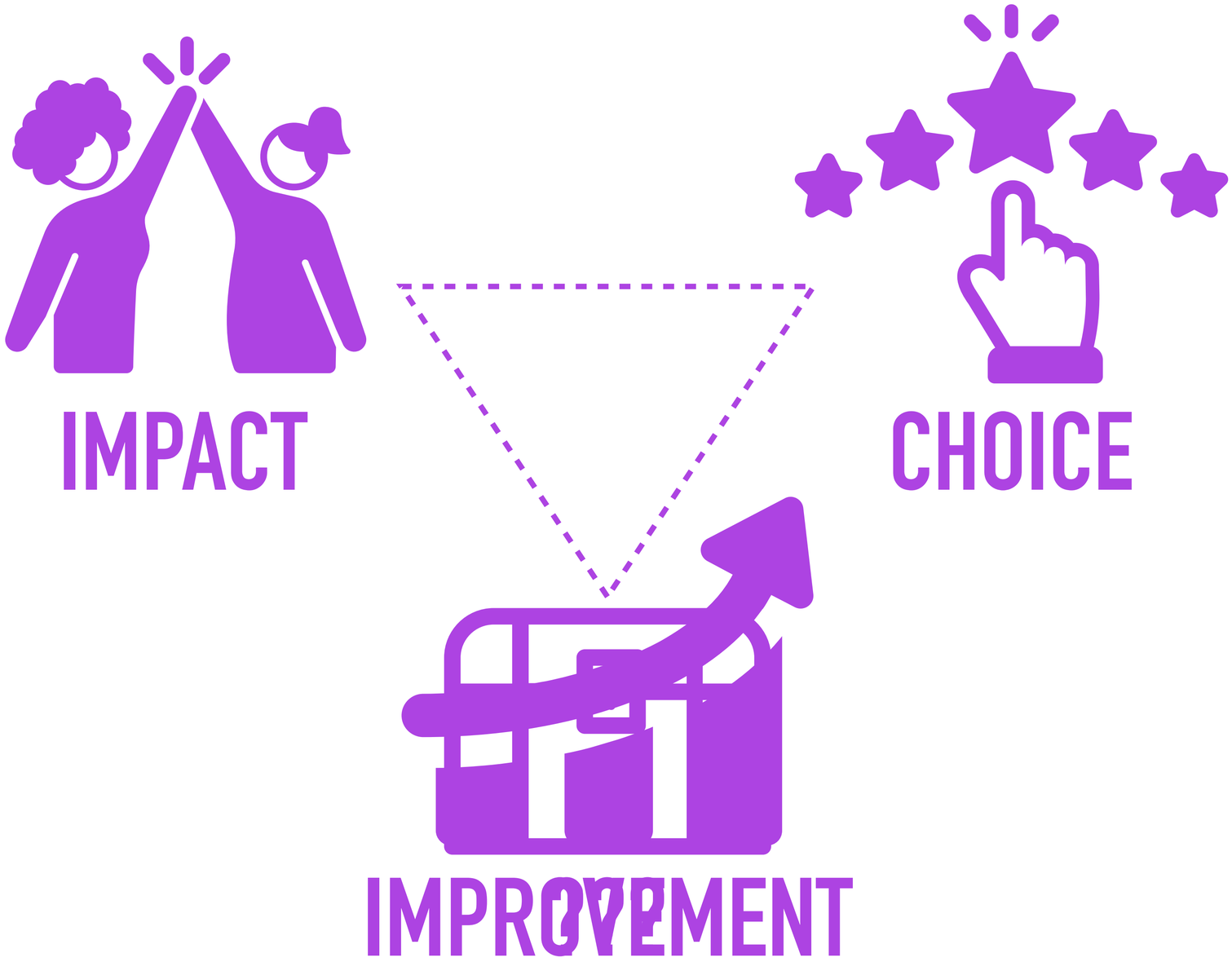
I. Codebase

Set the goal and step out of the way




THE TW
I. Codebase
Define decision
making boundaries
Set the goa
step out of

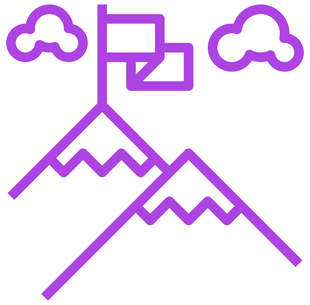
@patkua



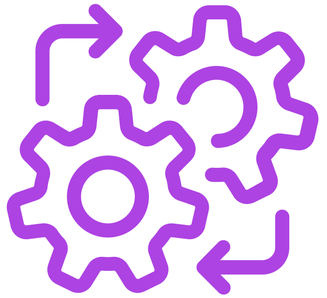
@patkua

WHAT DOES IT MEAN?

Personal Growth



Steps to Mission



Good Work Environment

IMPROVEMENT

TRAPS

Personal Growth

- No growth opportunities
- Repetitive work
- Lack of feedback/support
- Fear of failure

IMPROVEMENT

BORED PEOPLE QUIT

@patkua

TRAPS

Steps to Mission

- Unclear priorities
- Mission-task gap
- Intransparent information

IMPROVEMENT

TRAPS

- That won't work here
- "Ask for permission" atttiude
- Lack of feedback

Good Work Environment

IMPROVEMENT

PRINCIPLE

Cultivate **improvement** by

making small changes easier

IMPROVEMENT

@patkua

PRINCIPLE

ment by

anges

IMPROVEMENT

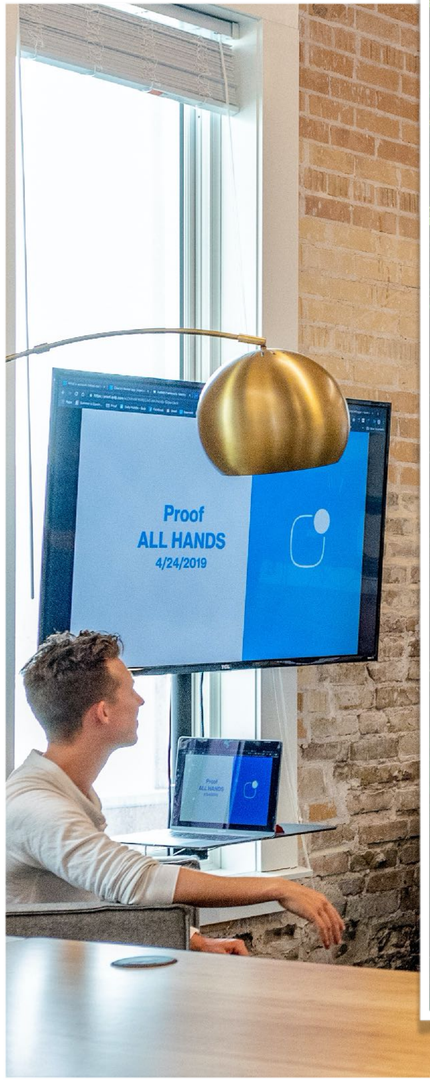
Improve the System from Incident Post-Mortems

@patkua

Take Action from

Create Forums for sharing Lessons Learned

Define Growth Paths



- WHY?
- 3 SECRETS
- HOW

@patkua

5 STEP RECIPE

1. Gather input
2. Publish your “tech culture”
3. Prioritise key improvement areas
4. Decide on actions (and do them!)
5. Repeat

@patkua

3 SIMPLE QUESTIONS

1. How many hand-offs do you have between Software Engineers and customers?

2. Are Software Engineers involved in **HOW** they solve a business problem?

3. What opportunities do Software Engineers have to grow?

1. How many hand-offs do you have between Software Engineers and customers?

2. Are Software Engineers involved in **HOW** they solve a business problem?

IMPACT

CHOICE

IMPROVEMENT

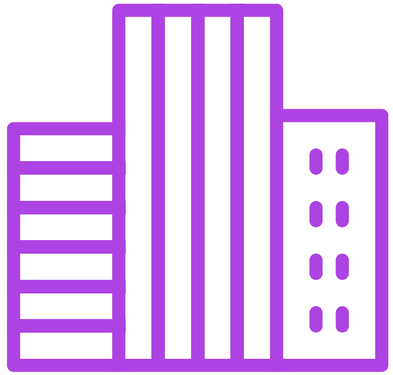
3. What opportunities do Software Engineers have to grow?

@patkua

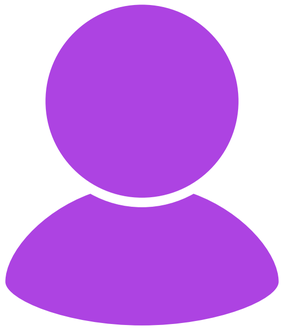
CONCLUSION

COMPANY A

COMPANY B

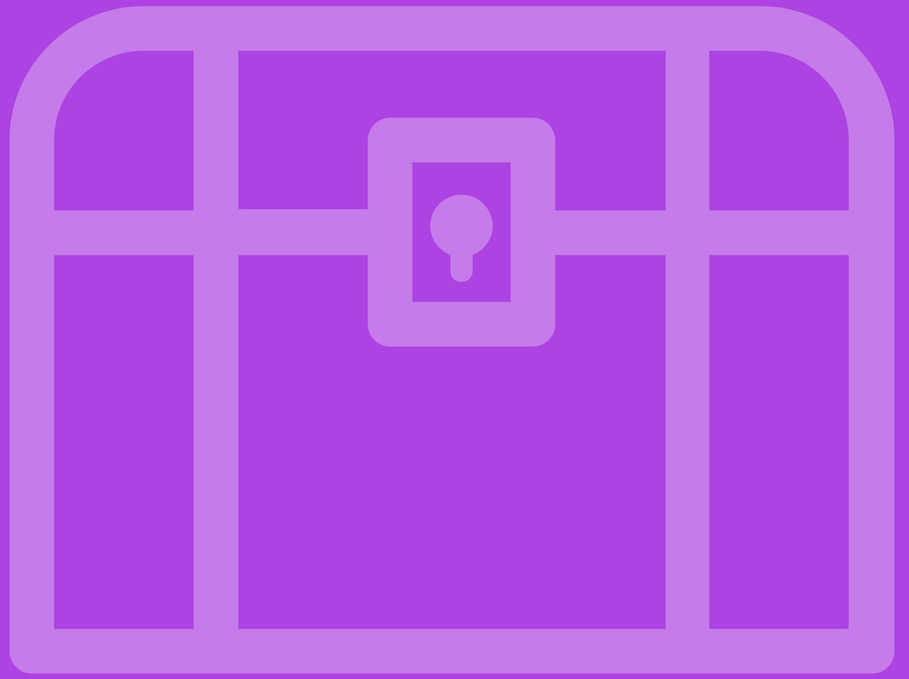


???



Questions?

Secrets of a Strong Engineering Culture



@patkua