

PRAQMA

The logo for PRAQMA, featuring the word "PRAQMA" in a bold, black, sans-serif font. The letter "Q" is stylized with an orange circle and a black diagonal line through it.

PRAQMA

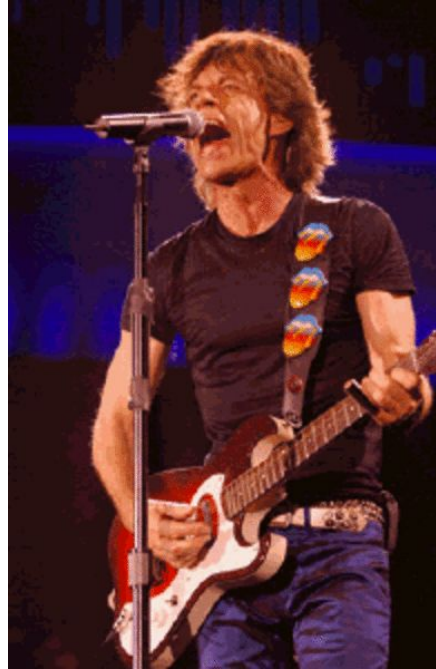
Now part of Eficode

Mastery in Software Development Teams

Emily Bache

@emilybache | coding-is-like-cooking.info

Job adverts...



The 10x Engineer

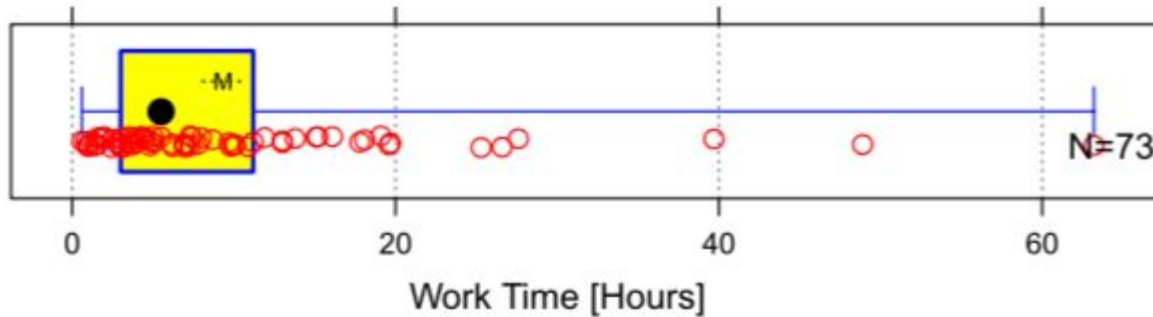
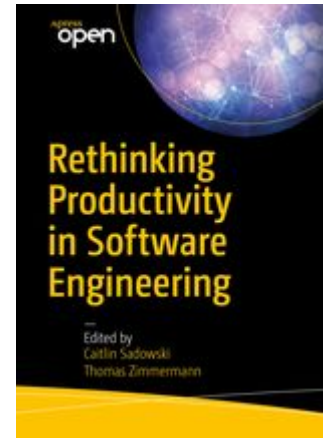
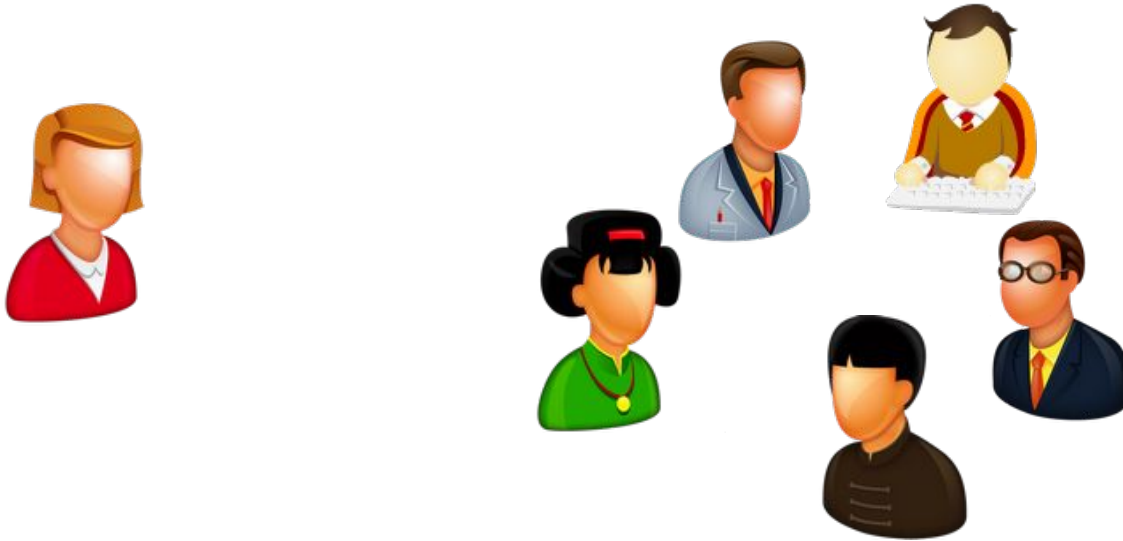


Figure 1-1. Distribution of work times for 73 developers for the same small program



Story: New to the Team





What makes a great Software Engineer?

Research done at Microsoft identified 53 attributes.

What Makes A Great Software Engineer? - Paul Luo Li, Andrew J. Ko, Jiamin Zhu

<https://faculty.washington.edu/ajko/papers/Li2015GreatEngineers.pdf>

53 Attributes of a Great Software Engineer

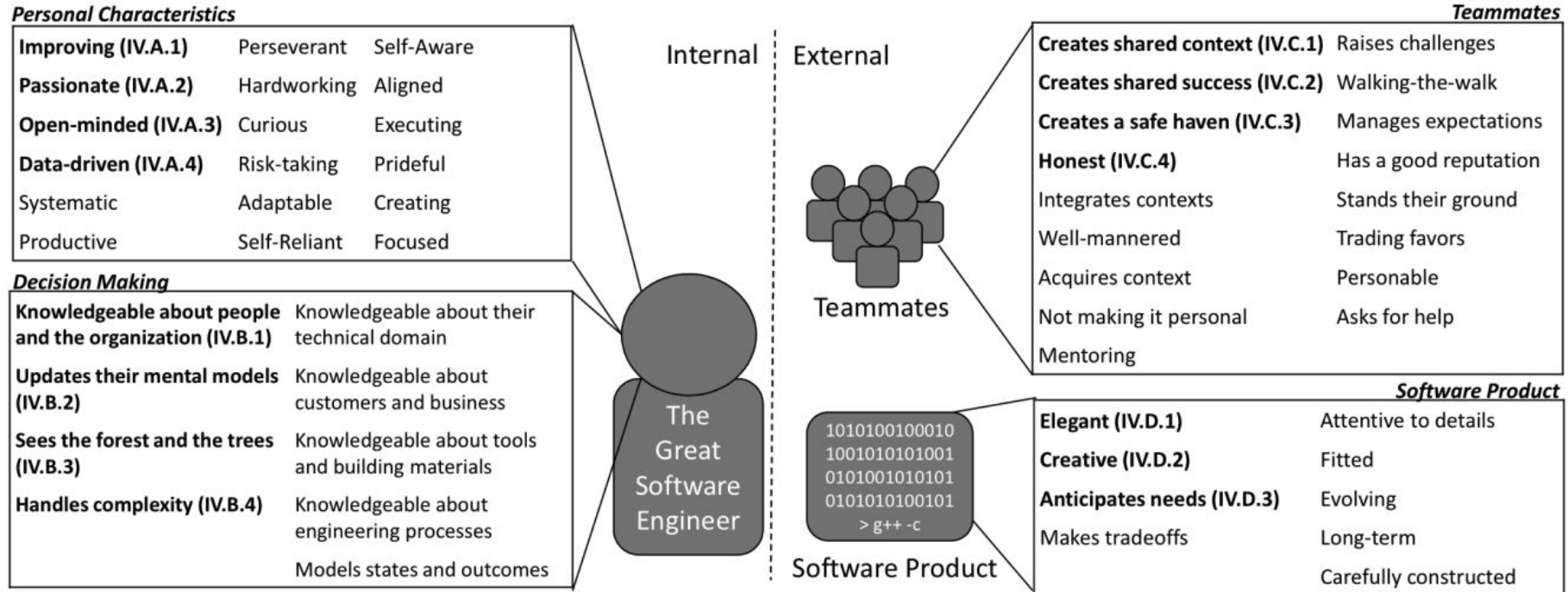


Fig. 1. Model of attributes of great software engineers, with attributes we discuss in detailed in bold.

53 Attributes of a Great Software Engineer

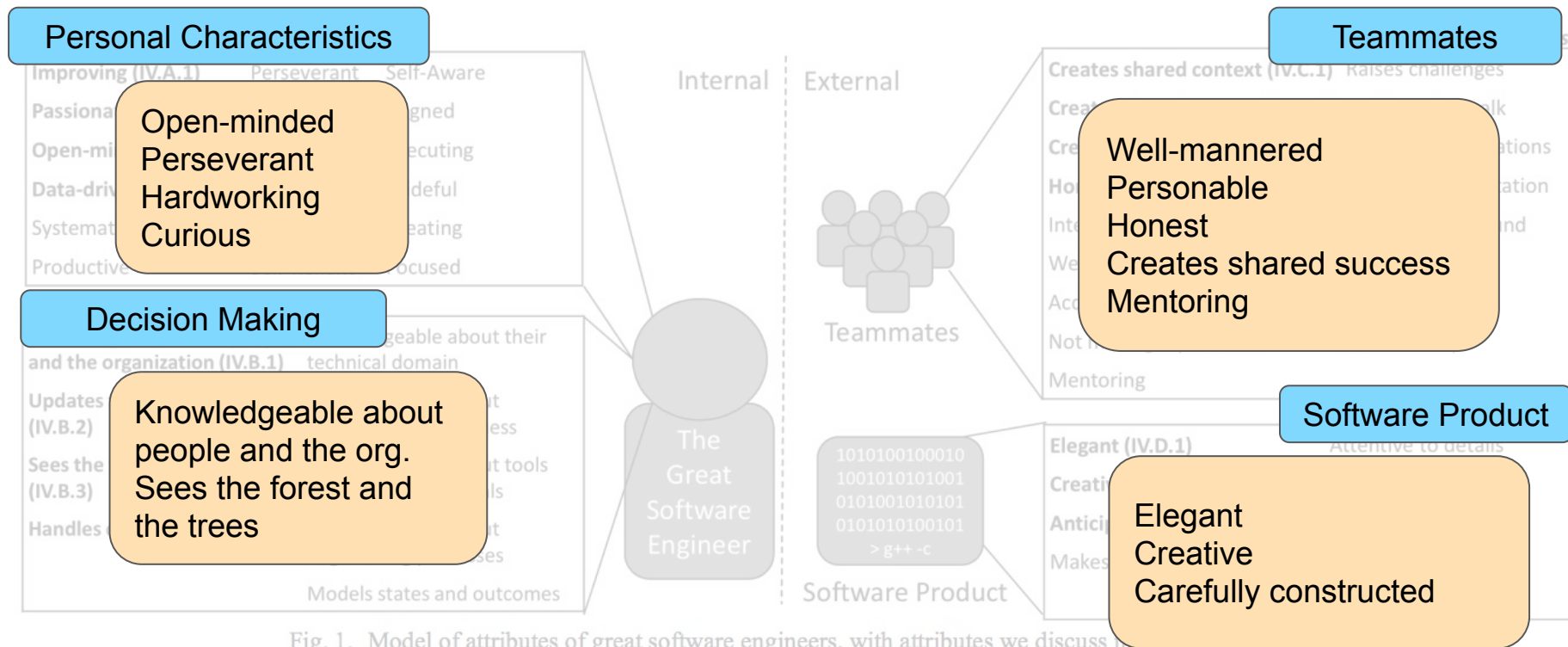


Fig. 1. Model of attributes of great software engineers, with attributes we discuss in this article



Development Teams

Attributes in teammates area revolved around four concepts:

- being a reasonable person,
- being a good leader,
- communicating effectively, and
- building trust.

Google's Project Aristotle

What factors influence team productivity?



Agile Teams



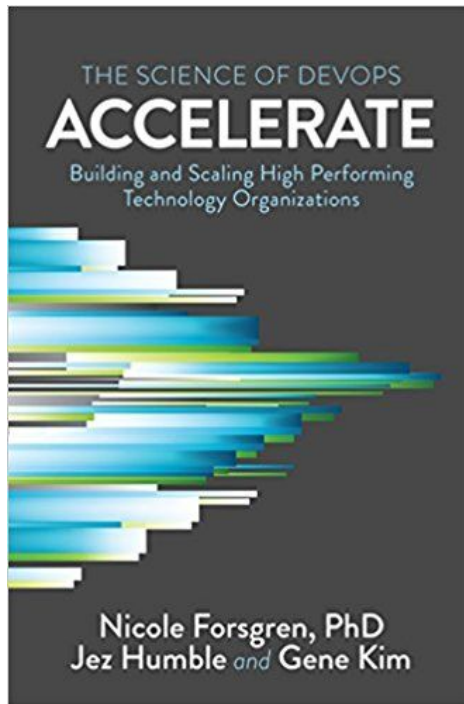
Agile Team

“Scrum Teams are self-organizing and cross-functional” - the Scrum Guide

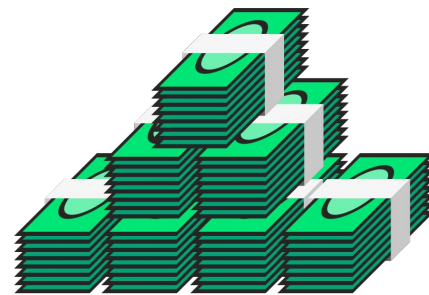
“Whole Team - A cross functional group of people with the necessary roles for a product form a single team.” - XP as described by the Agile Alliance

“The SAFe Agile Team is a cross functional group of 5-11 people” - from scaledagileframework.com

Research into the state of the art in DevOps

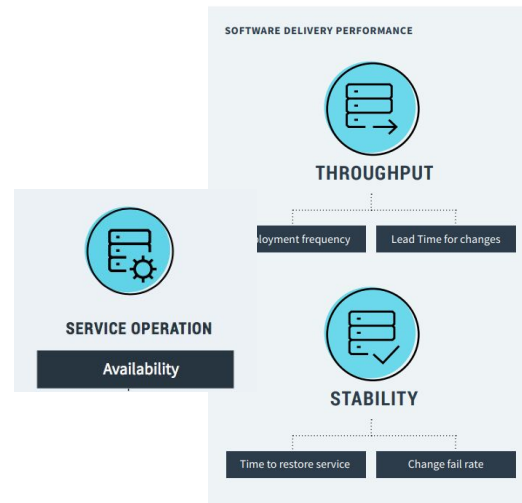


Effective development
matters for organizational
success



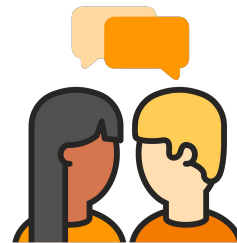
Success

You can **measure**
development organizations
effectiveness objectively



Metrics

Which behaviours drive better metrics

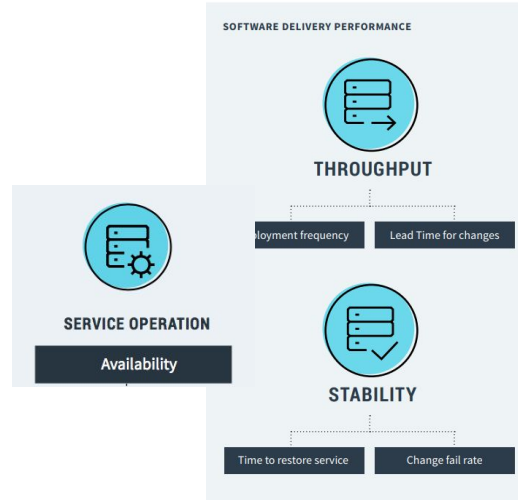
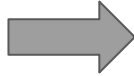


Behaviours

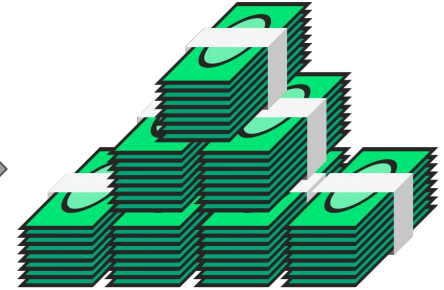
Accelerate: The Science of DevOps



Behaviours



Metrics



Success

**Great Software is built by empowered,
autonomous teams**

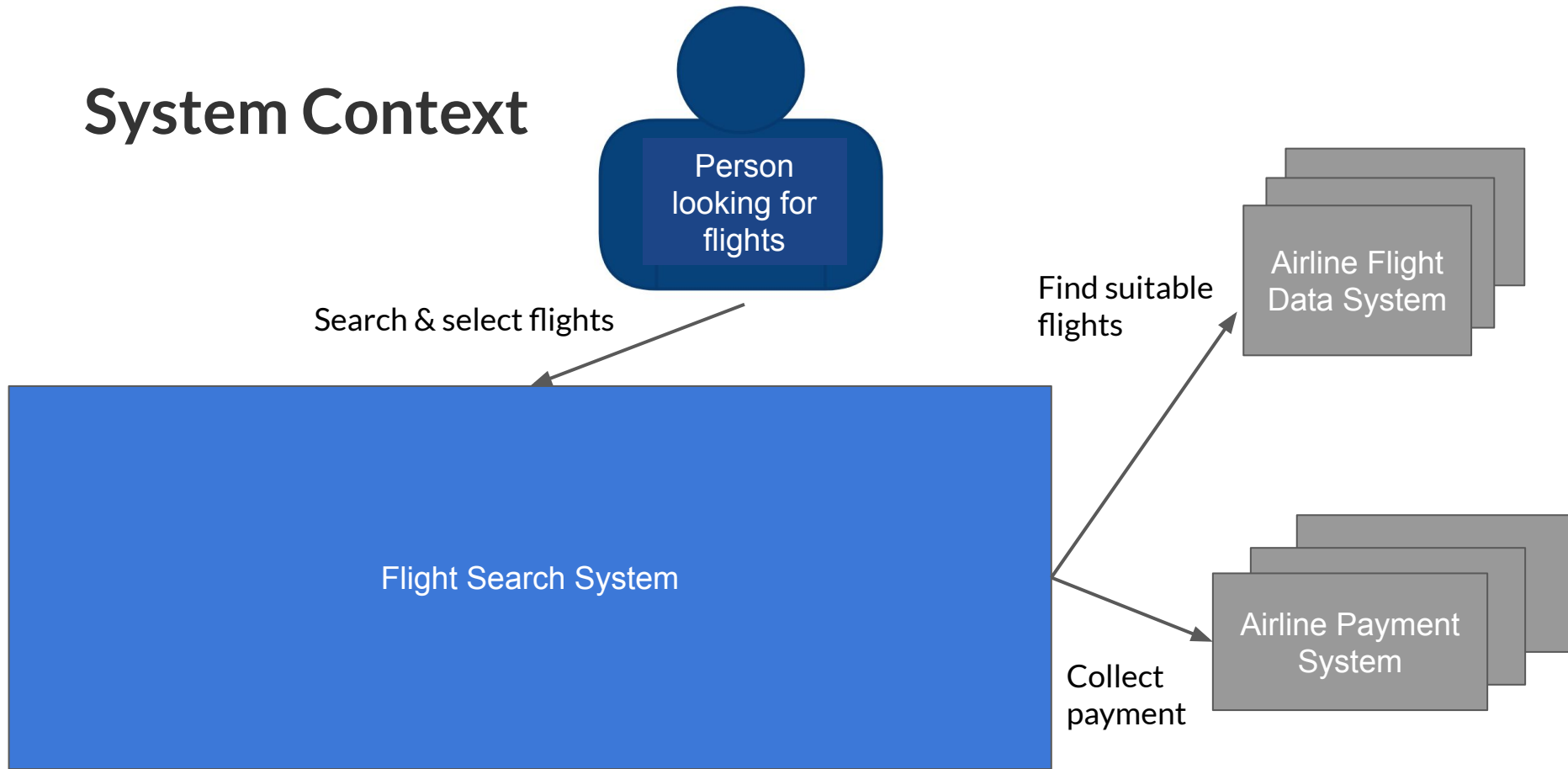


Development Teams

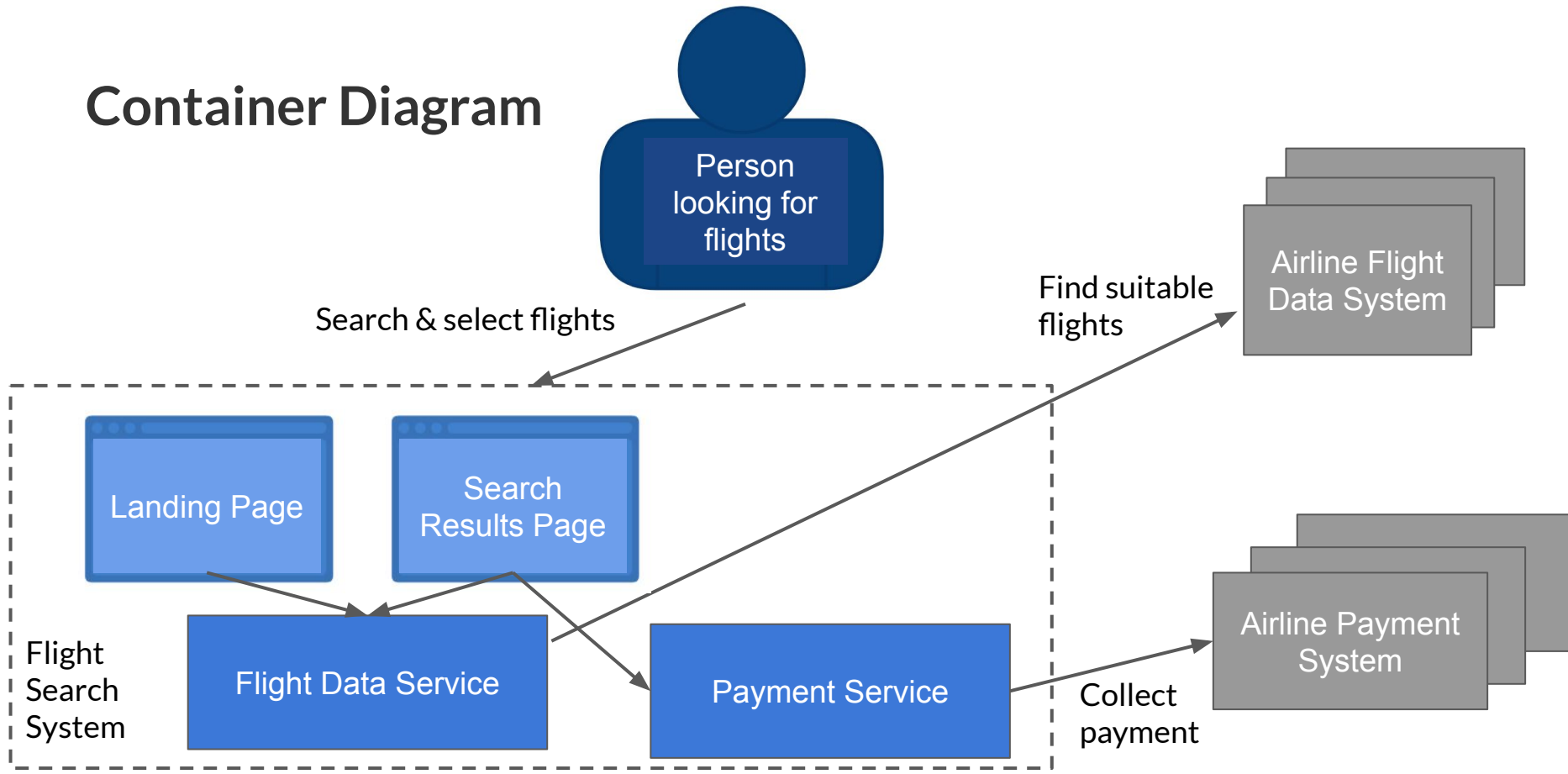
Empowered Teams:

- Large-scale design changes
- Without detailed coordination outside the team
- Deploy on-demand
- Do their own testing

System Context



Container Diagram



Empowered Teams



Customer
Acquisition



Flight Data
Presentation

Architecture &
Infrastructure



Shared Flight
Data



Payments and
Subscriptions

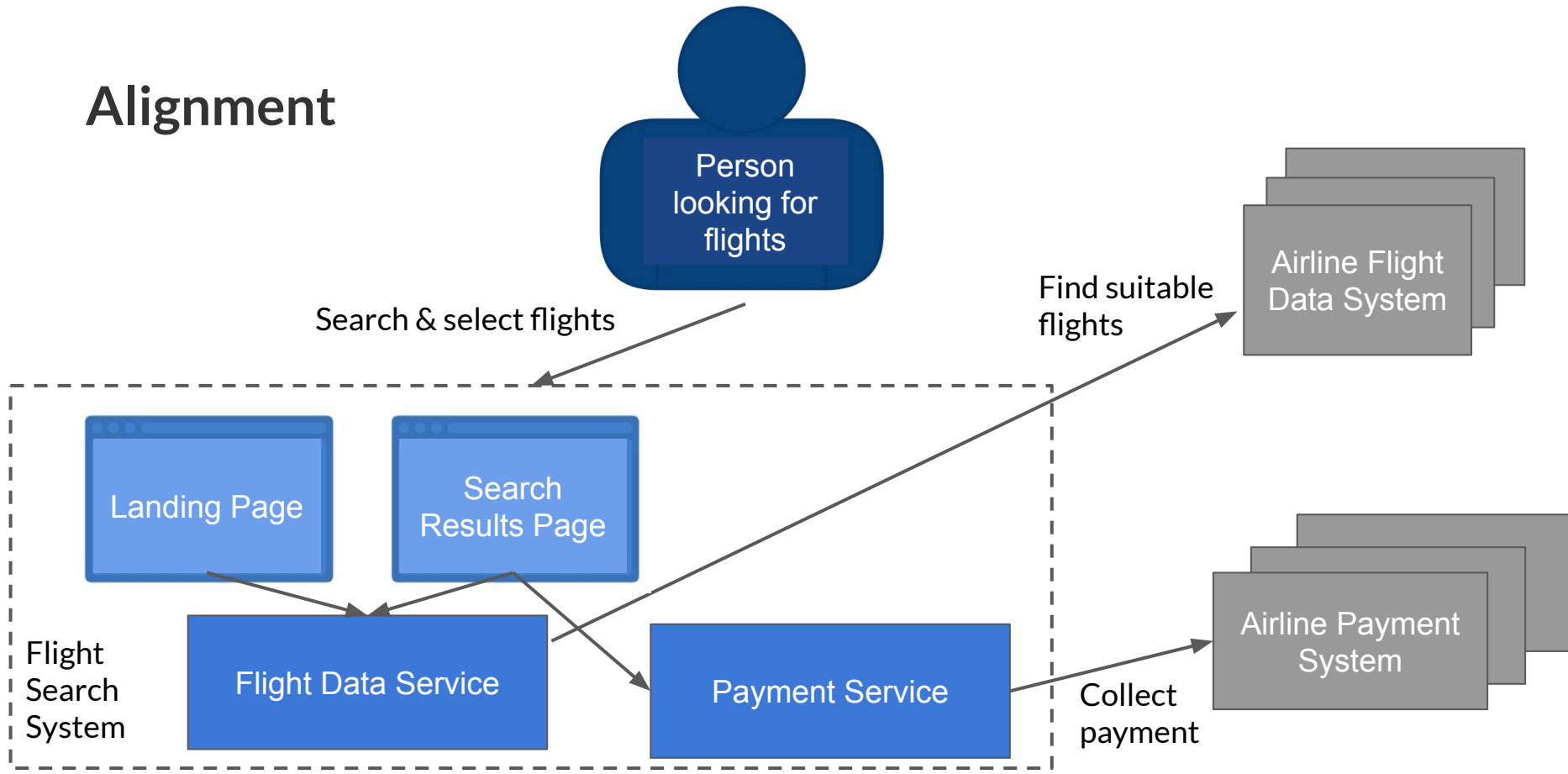


Development Teams

Empowered Teams:

- Large-scale design changes
- Without detailed coordination outside the team
- Deploy on-demand
- Do their own testing

Alignment



Supporting Empowered Teams

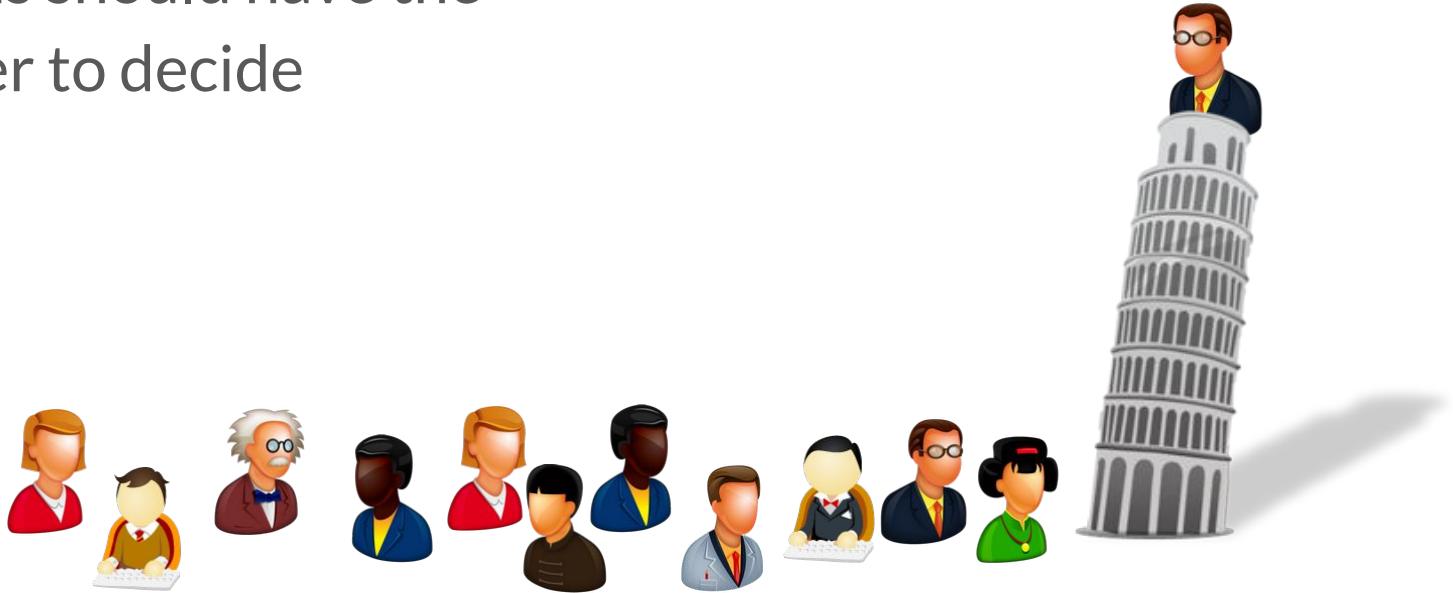
- CD Pipelines
- Architecture advice
- Standard Tools



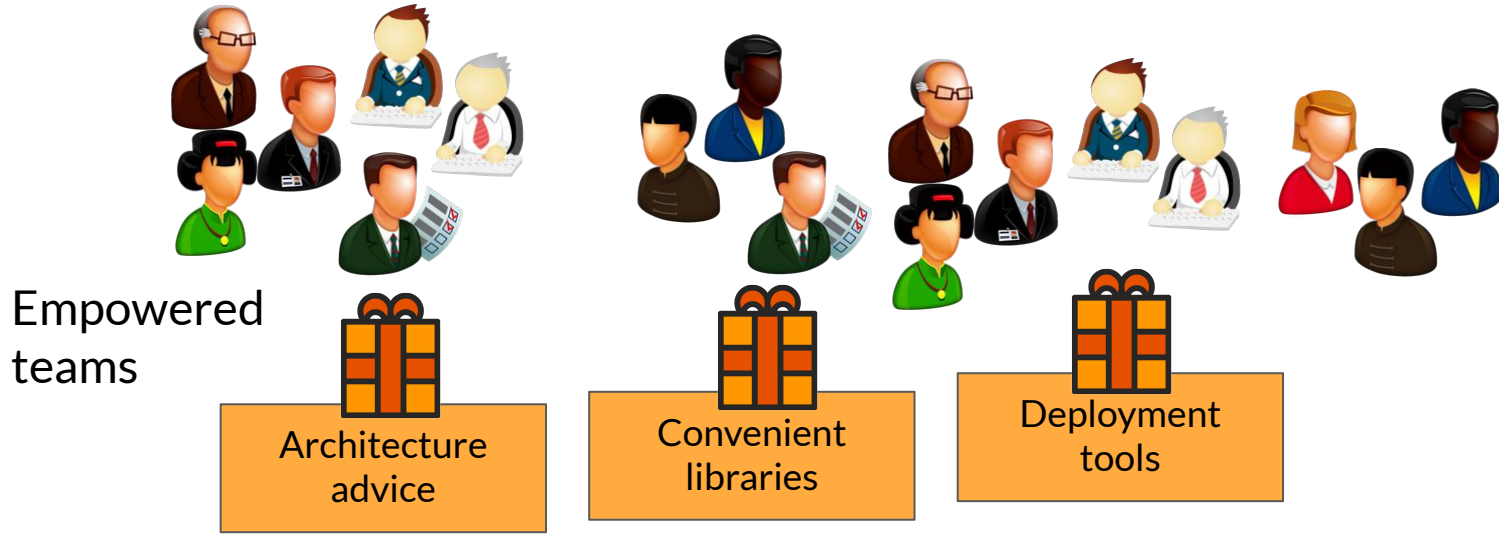
Architecture &
Infrastructure Team

Architects can't dictate from on high

- Teams should have the power to decide



Technical Leadership

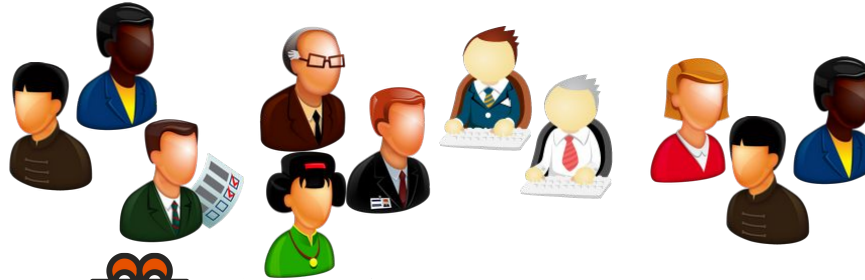


Support the teams, not slow them down

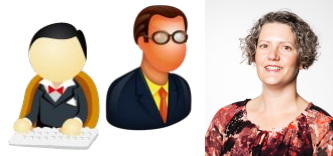
Technical Leadership - my experience



Empowered
teams



Test automation
framework



Technical Leaders

Working alone most
of the time

Mob Programming



A structured collaboration for a team of software developers coding together on one machine

“All the brilliant people working on the same thing, at the same time, in the same space, on the same computer.”
– Woody Zuill

Mob Programming

Pair Programming

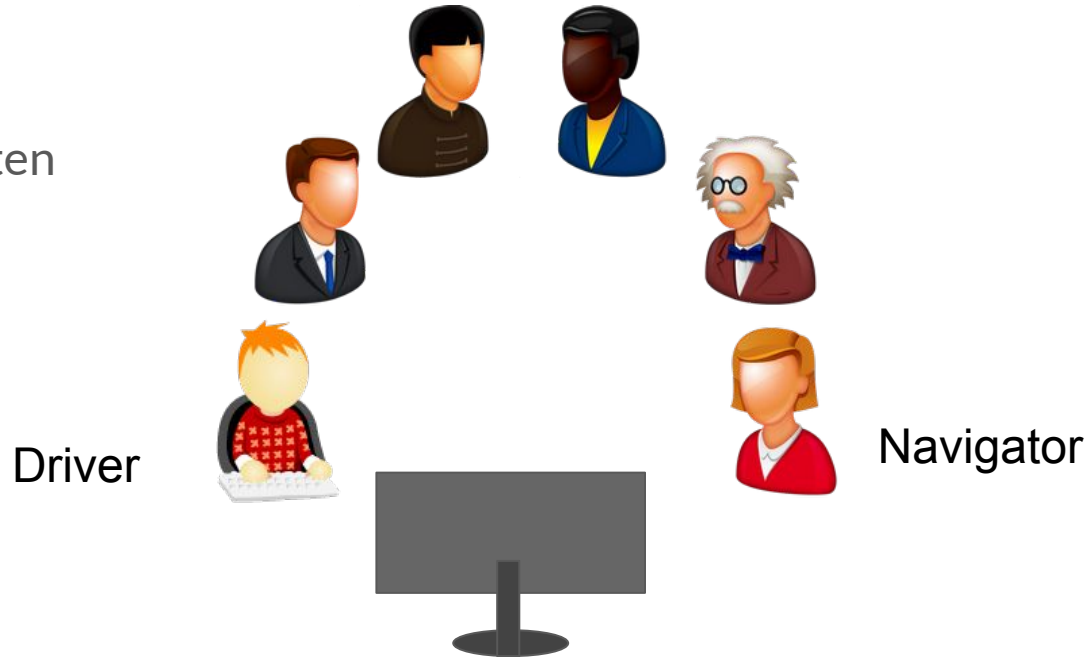
Two people working together at the same computer on production code



image: flickr user Lisamarie Babik

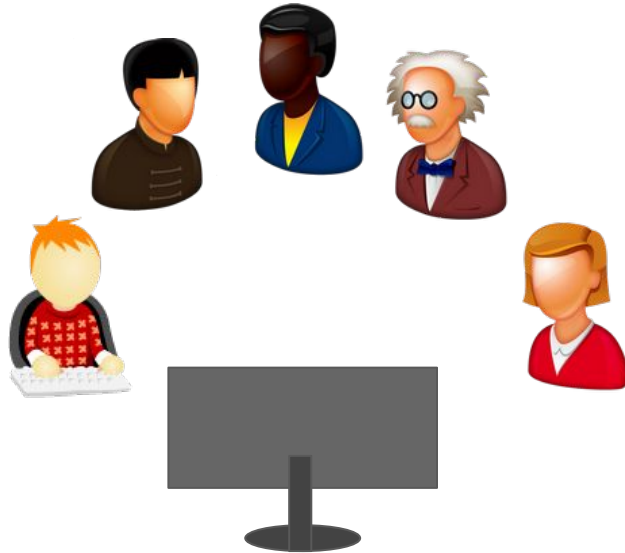
Mob Roles

- Driver
- Navigator
- Mob member
- Rotate roles often

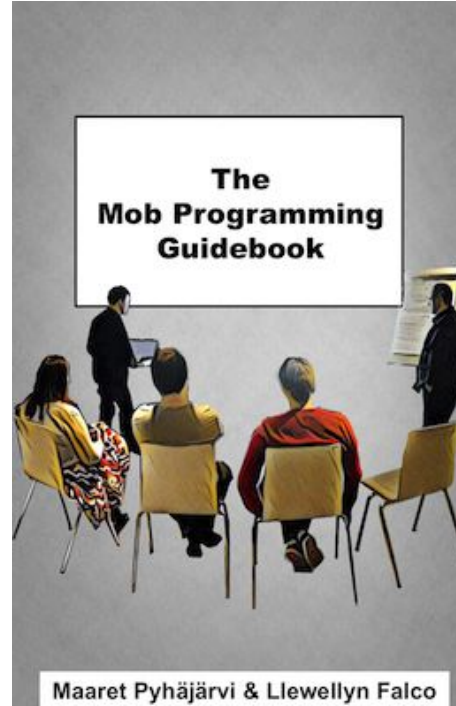
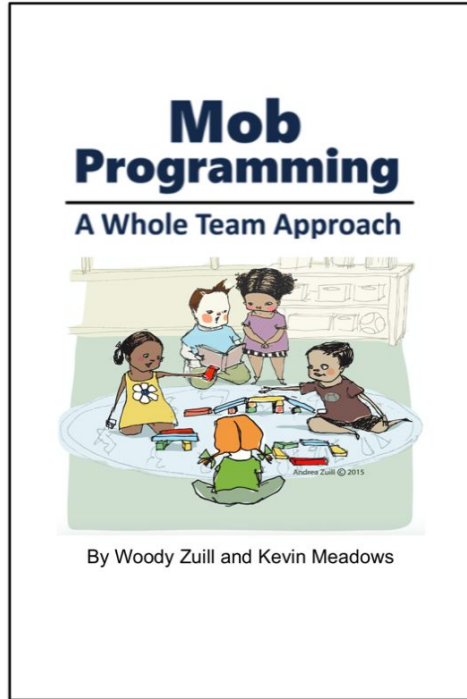


Working Agreement

- We treat everyone with kindness, consideration and respect



Recommendations:



MOB PROGRAMMING GUIDEBOOK

MAARET PYHÄJÄRVI

Page 1 of 51

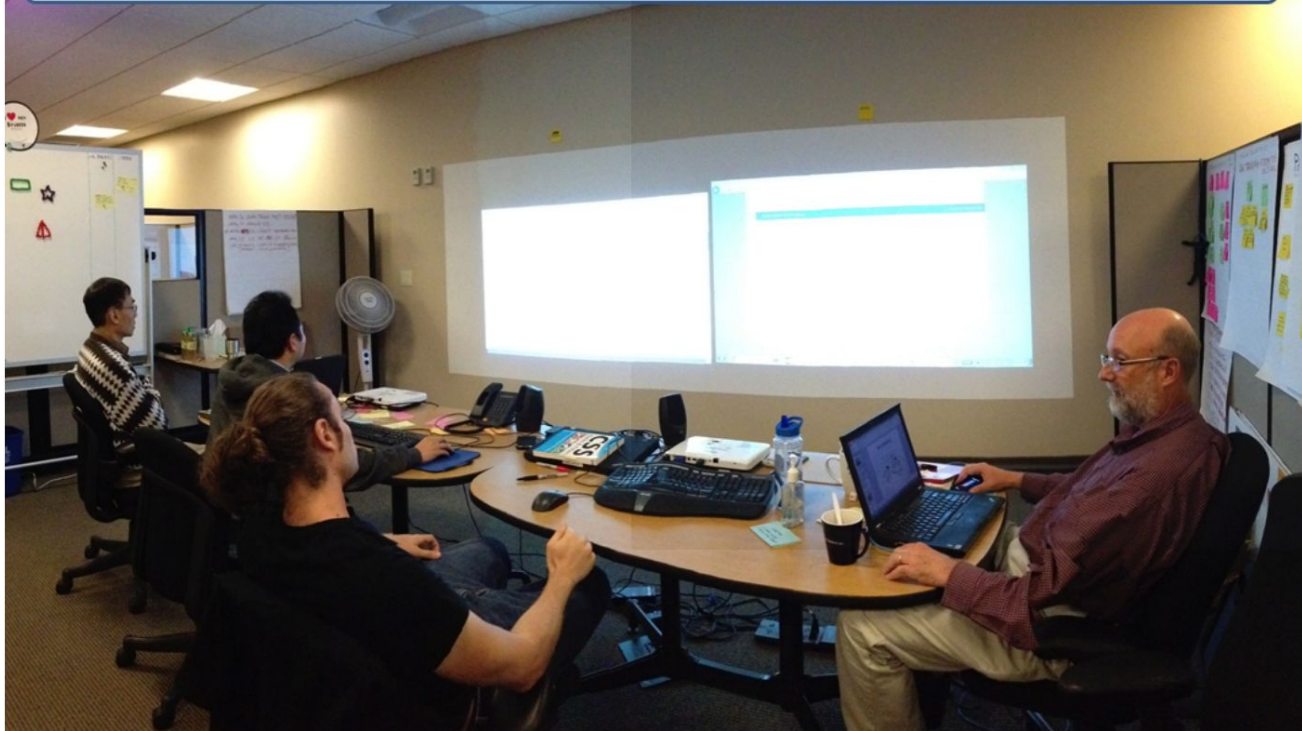
Mobbing on Production Code

This is what it looks like from the front



Mobbing in Production Code

... And what it looks like from the back.



Isn't it really unproductive?

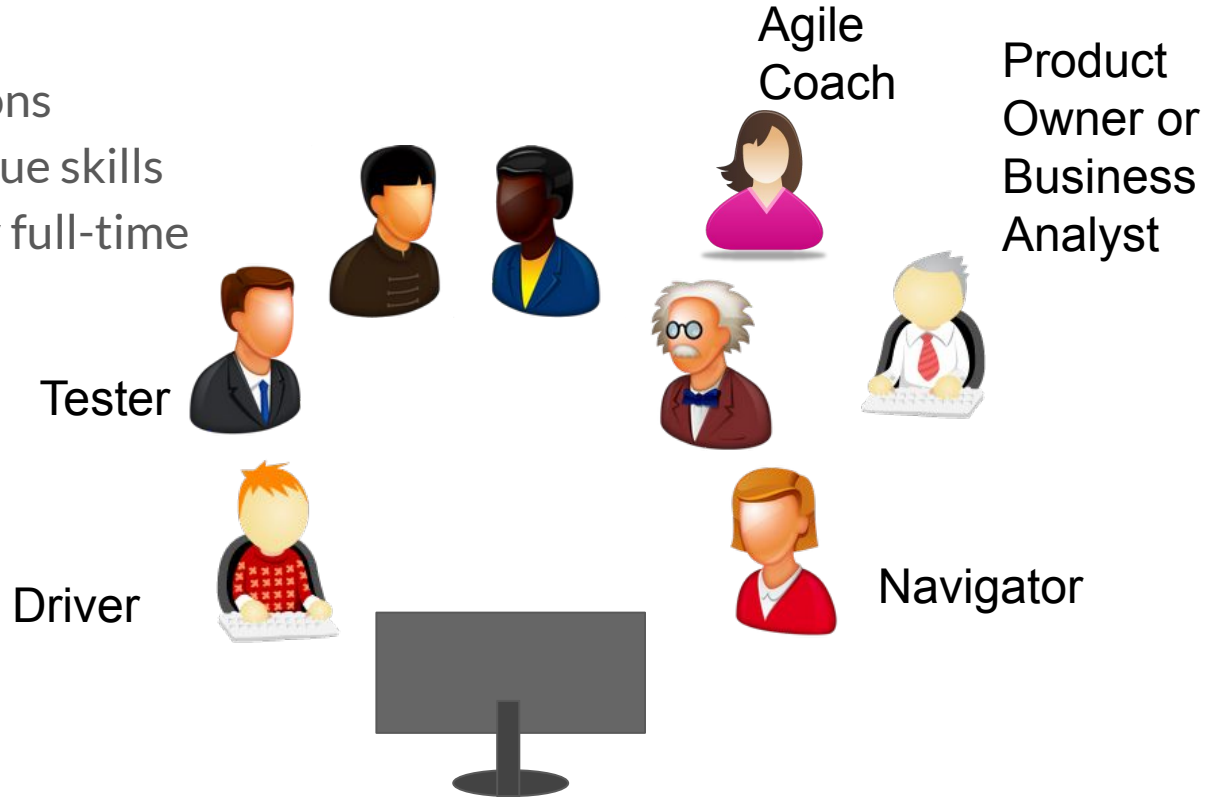
That hasn't been measured
(yet).

What destroys productivity?

- Context switching & Interruptions
- Unclear or missing requirements
- Bad code
- Team member off sick
- ...

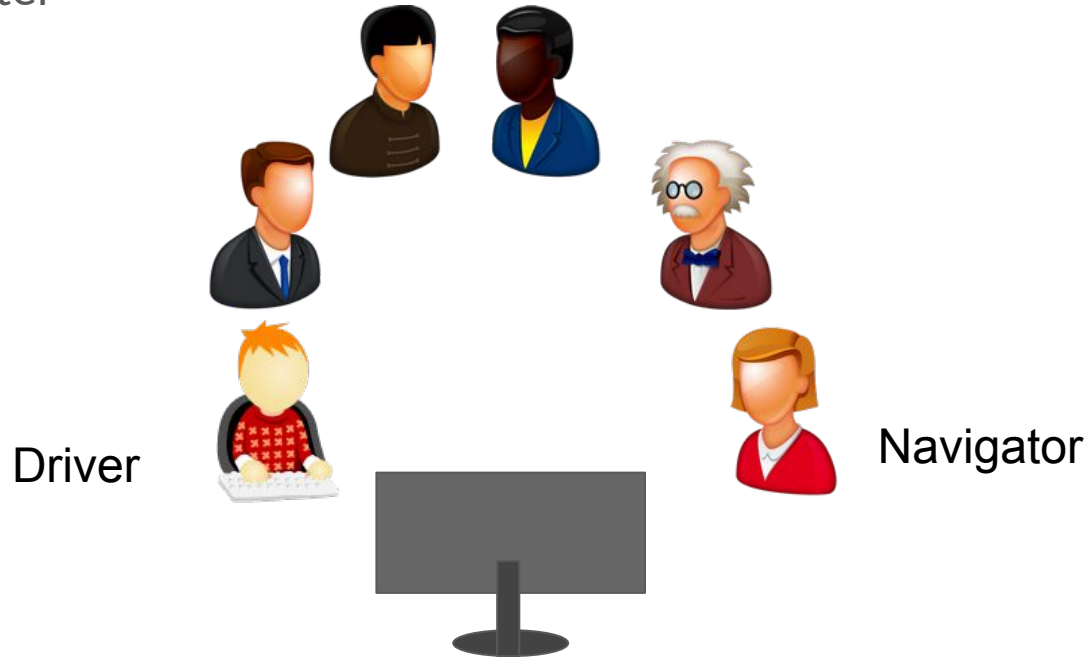
Mob Roles for non-programmers

- Answer questions
- Bring your unique skills
- Not necessarily full-time



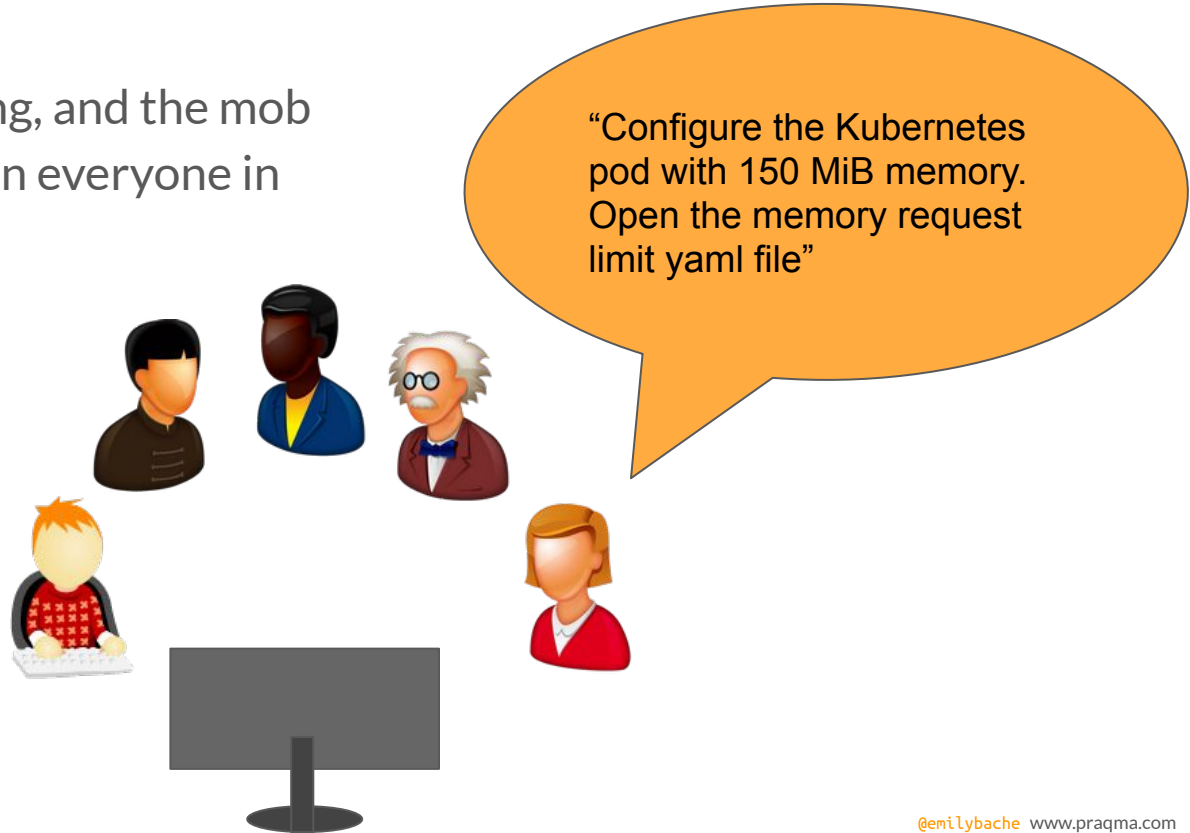
Everyone understands the code you mob on

- Team learns to collaborate better



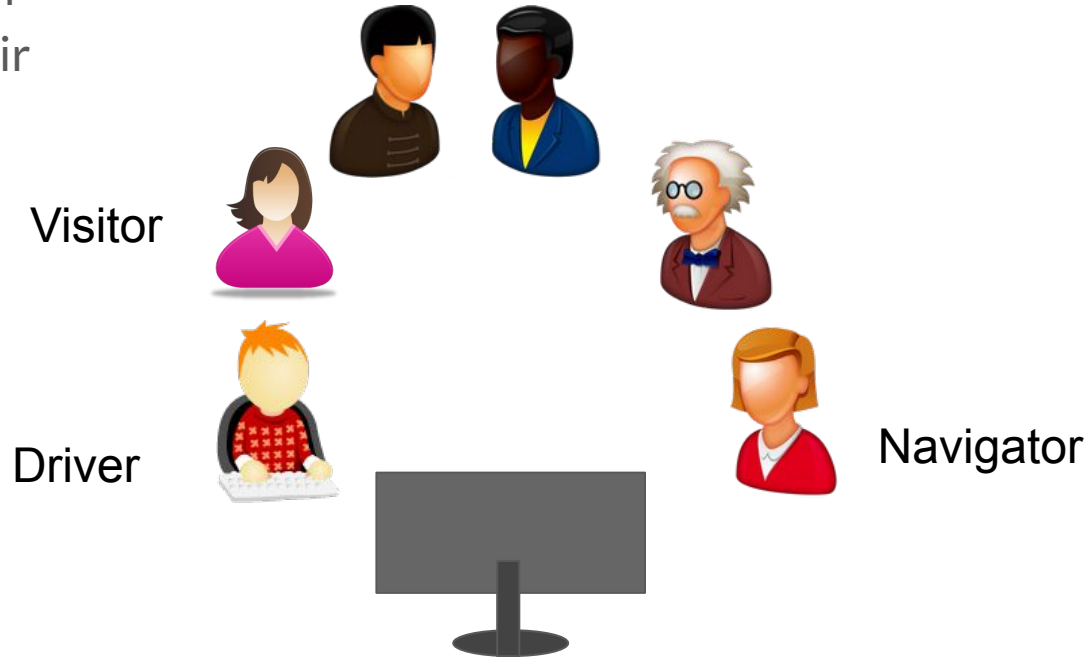
Skill multiplier

- If one person knows a thing, and the mob needs that thing, then soon everyone in the mob knows it.



Visitors quickly become an asset

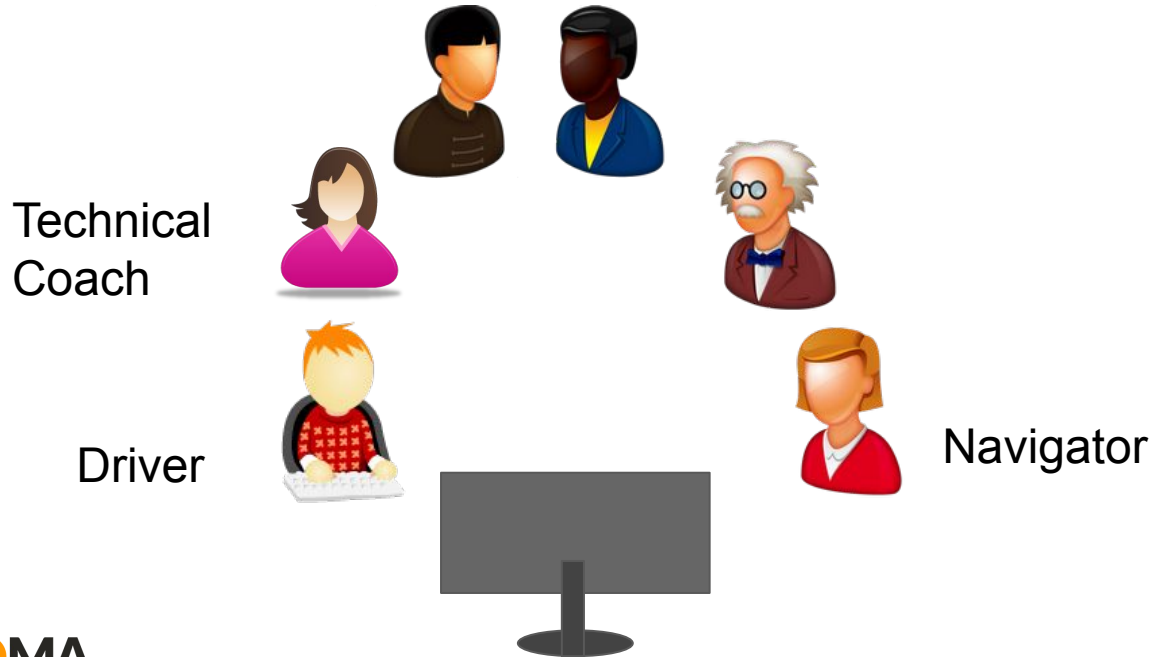
- Experienced developer, new to the team - contributes their expertise



- *Everyone understands the code*
- *Skill Transfer*
- *Visitors are quickly contributors*

Mob Programming

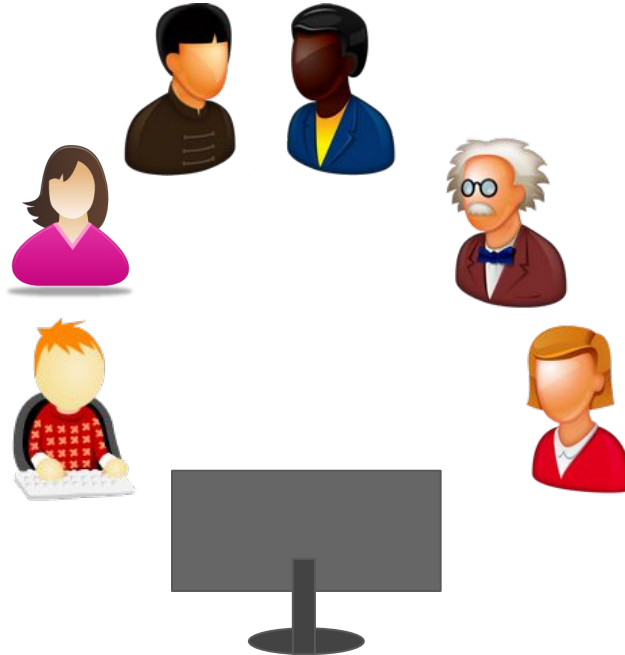
Architect joining the mob as Technical Coach



Coaching Empowered Teams

- Coach encourages & assists
- Spends time with many teams

Technical
Coach

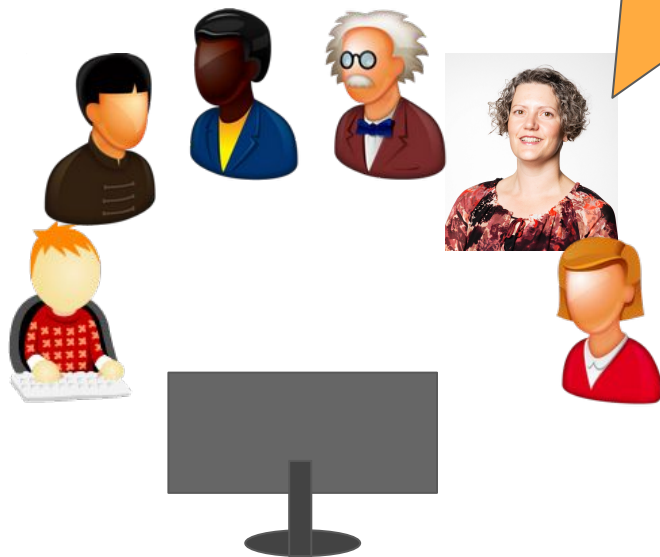


Driver

Navigator

I coach TDD

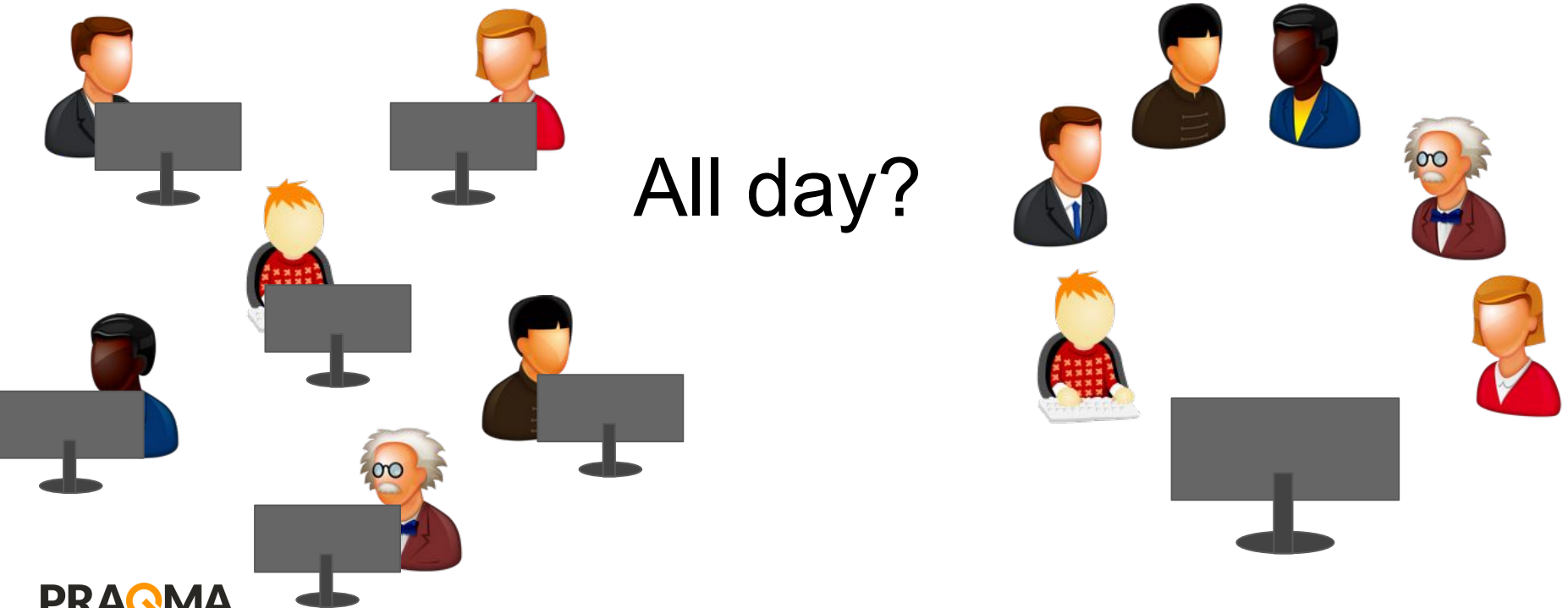
- Encourage the mob to follow red-green-refactor
- Navigate writing the tests
- Doesn't usually drive



“Start with the assert: the length of the duplicates list should be two. Then write the act step: call the ‘detectDuplicates’ method...”

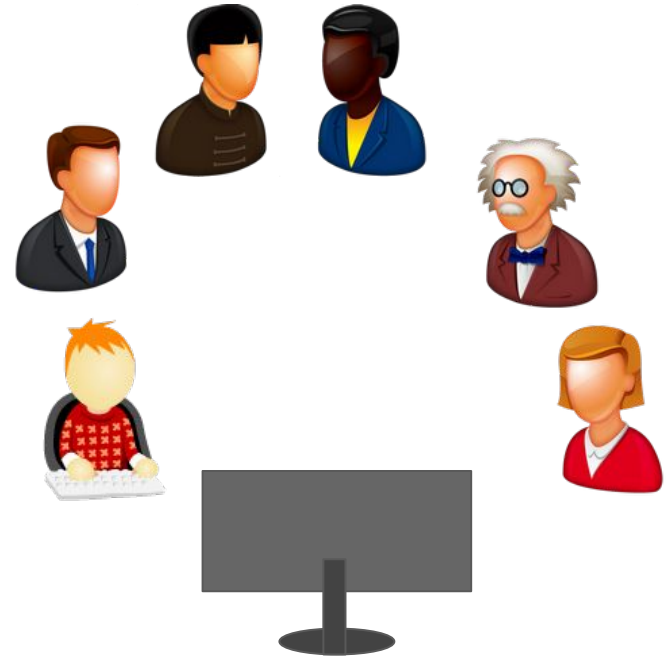
How much should teams mob program?

All day?



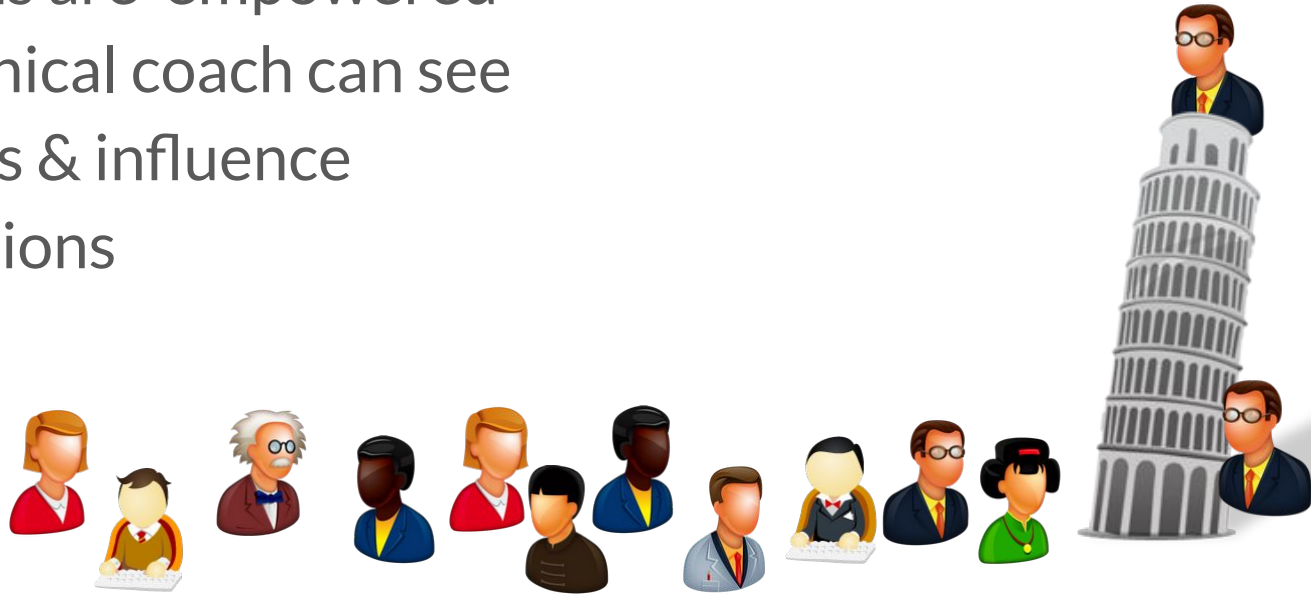
First learn to mob program

- 2 hours per day, with coach, for about 2 weeks
- After that, the team decides



Technical Coaches mob with all the teams

- Teams are empowered
- Technical coach can see needs & influence decisions



53 Attributes of a Great Software Engineer

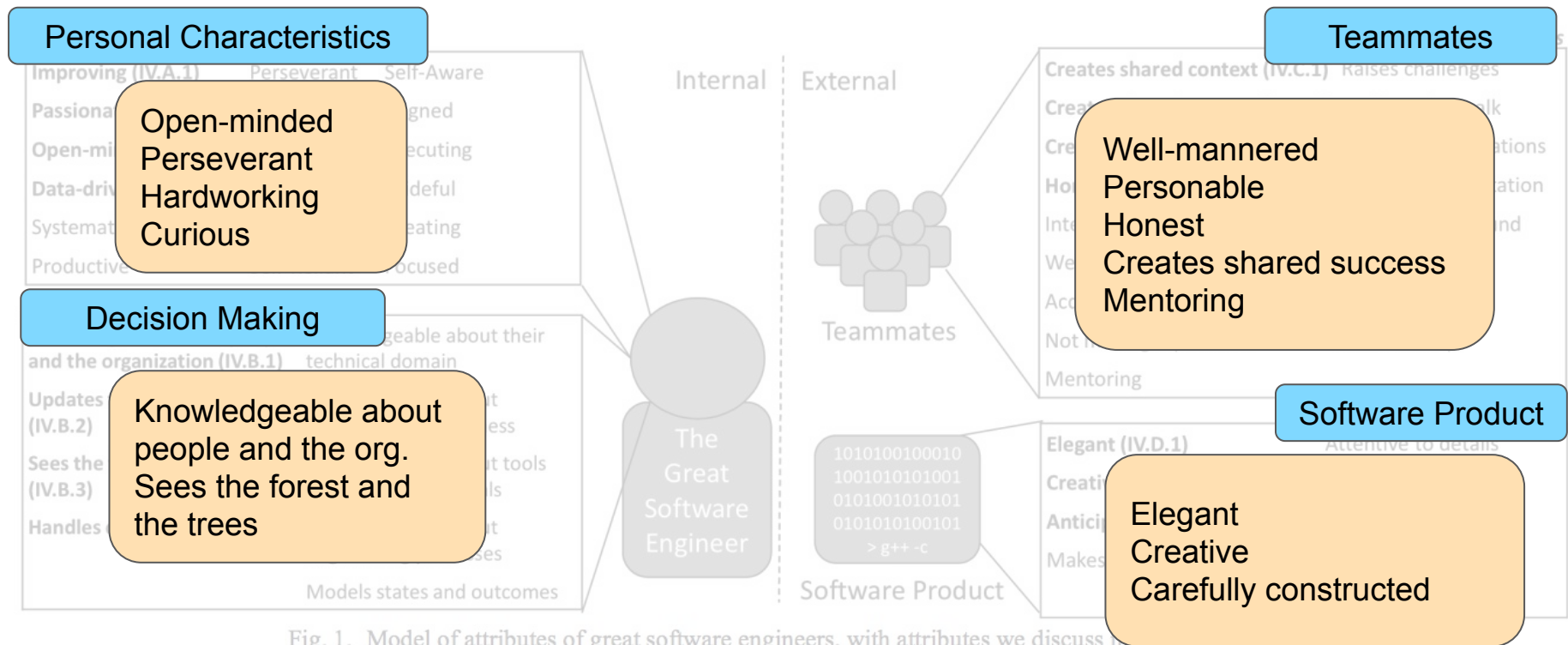
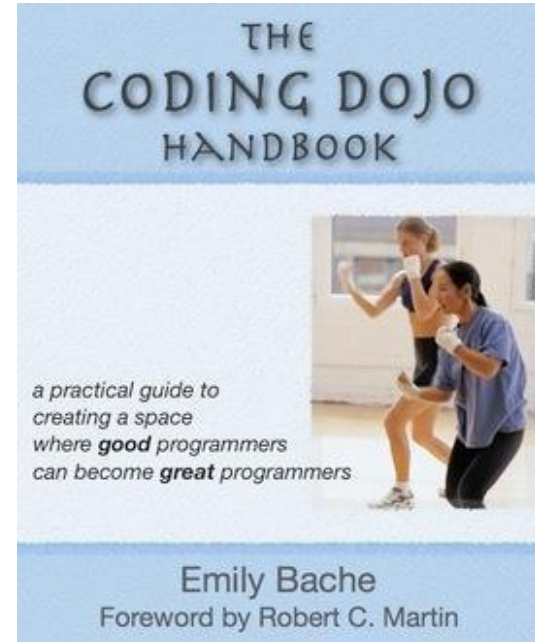


Fig. 1. Model of attributes of great software engineers, with attributes we discuss in this article

Coding Dojos

Dojo == the place you go to learn



Learning Hour

- Invite all the teams
- 1 hour daily
- Practice skills & learn theory





Mob Programming



Architects

Mastery in
Software Teams

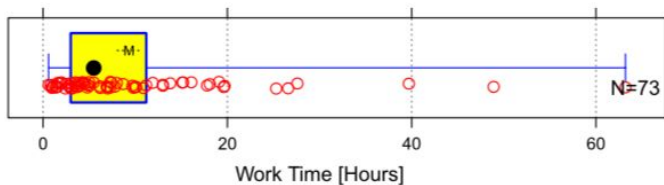


Figure 1-1. Distribution of work times for 73 developers for the same small program

10x Engineers



Gifts not Rules



@emilybache



Empowered
Teams

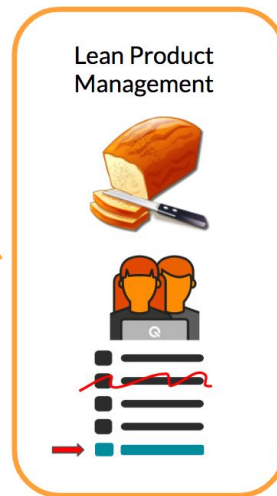
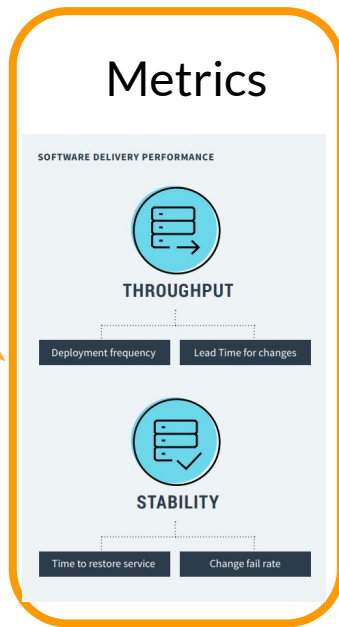
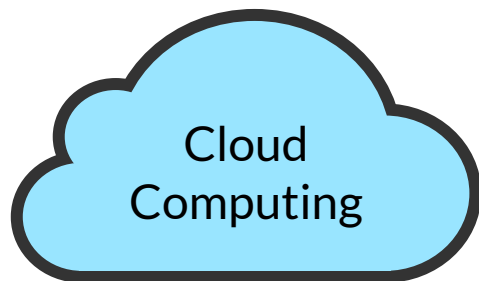


Thank you

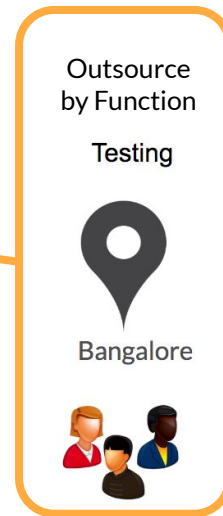
Expand Conference 2019

@emilybache | coding-is-like-cooking.info

Accelerate: The science of DevOps



negative



How Developers Spend Time & How Productive they Feel

PersonalAnalytics: Participant Survey

Please fill out the following survey:

Hint: The term session refers to the time period since the beginning of your workday or the time you last answered this survey today.

Compared to your normal level of productivity,
how productive do you consider the previous session?

not at all 1 2 3 4 5 6 7 very much

Please specify the activities (tasks, meetings, breaks, etc.)
you performed on in the previous session:

Quick insert: [Planned meeting](#) [Unplanned meeting \(helped co-worker\)](#) [Lunch](#) [Break](#)

Remove

Thank you!

Finish Survey

André N. Meyer, Laura E Barton, Gail C Murphy, Thomas Zimmermann, and Thomas Fritz. 2017. The Work Life of Developers: Activities, Switches and Perceived Productivity. Transactions of Software Engineering (2017), 1–15.



Development
Productivity

Study of 20 developers at 4 companies in 3 countries

- Spent only half their time actively clicking mouse or using keyboard. Otherwise thinking, talking, resting
- 25% of time actively coding
- 25% of time collaborating
- 25% hard to classify
- 25% browsing & writing documentation
- Spend less than 2 minutes on an activity before switching



Development
Productivity

Productivity seems very individual.

- They tried to find out if people feel more productive when doing certain activities - no consistent answer, very individual
- They tried to find out if people feel more productive in the afternoon - some people do, others in the morning, others in both but not lunchtime.
- They tried to find out if a lot of mouseclicks/keyboard work made people feel more productive - no consistent answer

Software Development Teams





Thank you

Expand Conference 2019

@emilybache | coding-is-like-cooking.info

PRAQMA