Metrics to measure developer experience

/thoughtworks

About Me

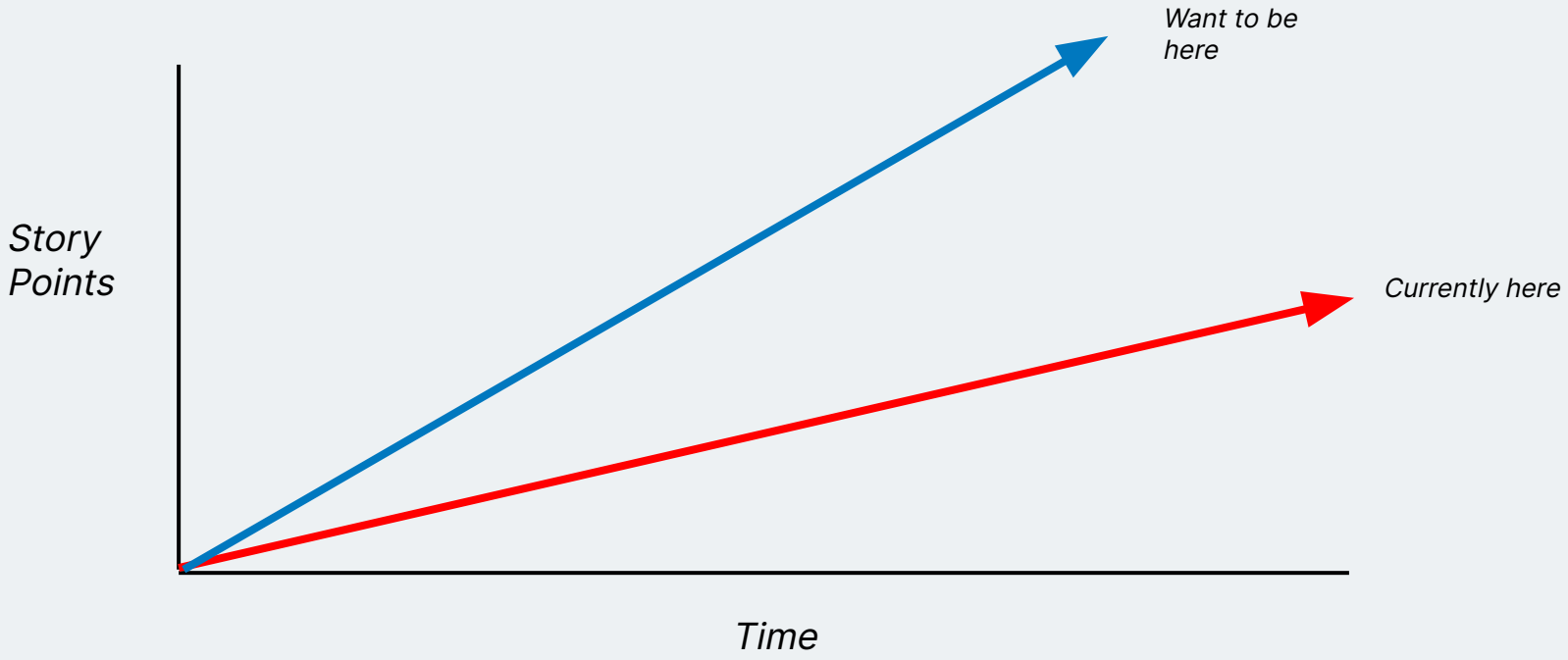
Tim Cochran, 19 years working with technology.

Technical Director @ /thoughtworks

Developer. Building products. Digital Transformation.

Enjoy Hiking, IPAs, and Travelling (less these days).

Productivity

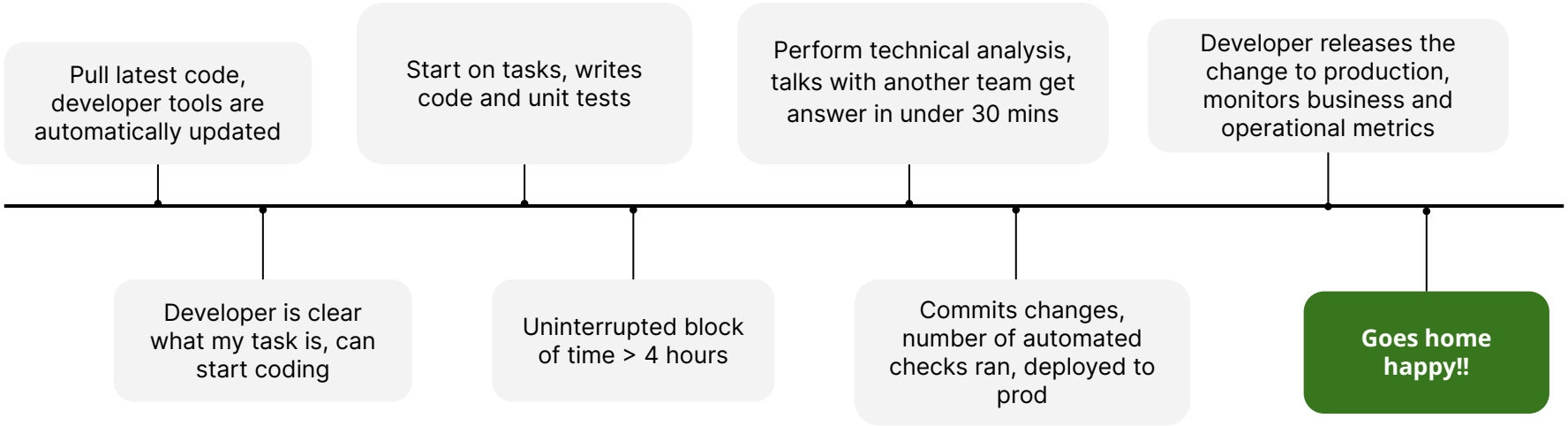


Developer Effectiveness

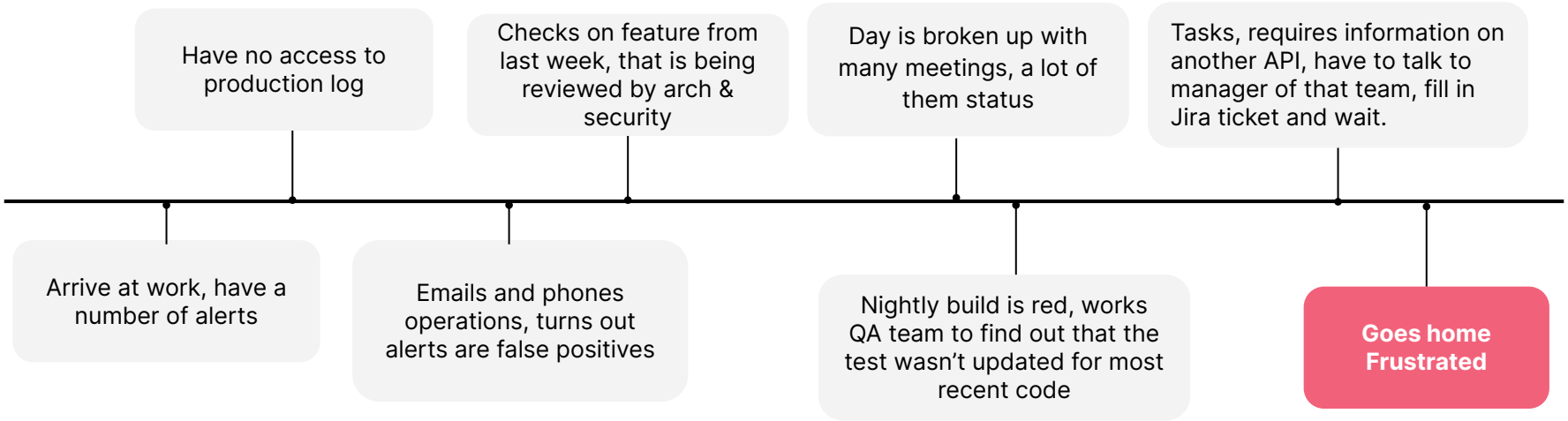
“What does being effective mean? As a developer, it is delivering the maximum value to your customers. It is being able to apply your energy and innovation in the best ways towards the company’s goals.”

Maximizing Developer Effectiveness | martinfowler.com

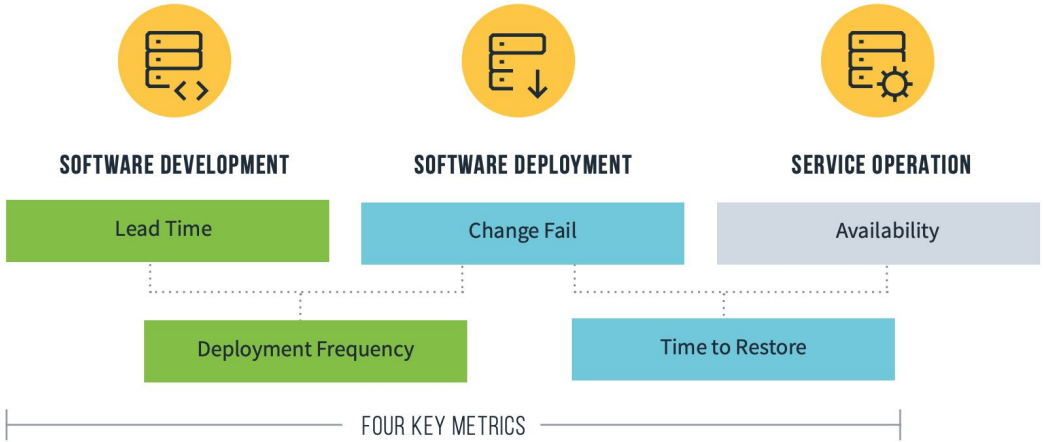
Day In a life - High effectiveness

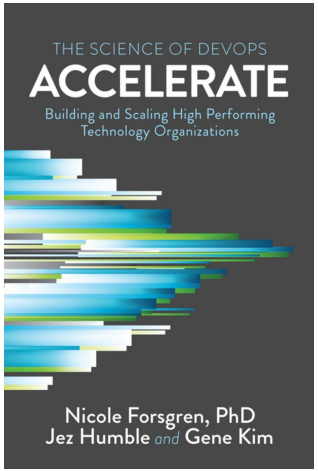


Day In a life - Low effectiveness



Developer Effectiveness

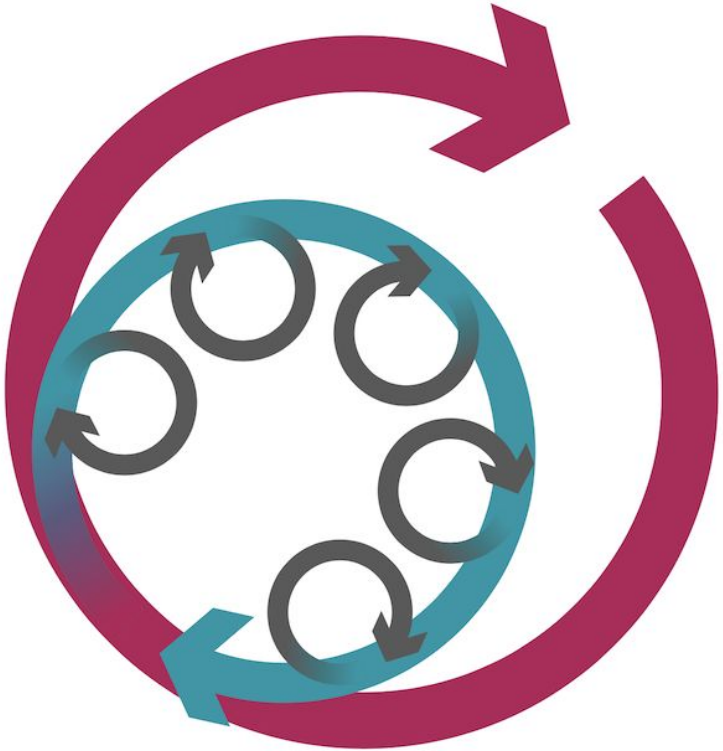




Whats does a developer do all day?

What are the outcomes they are trying to achieve?

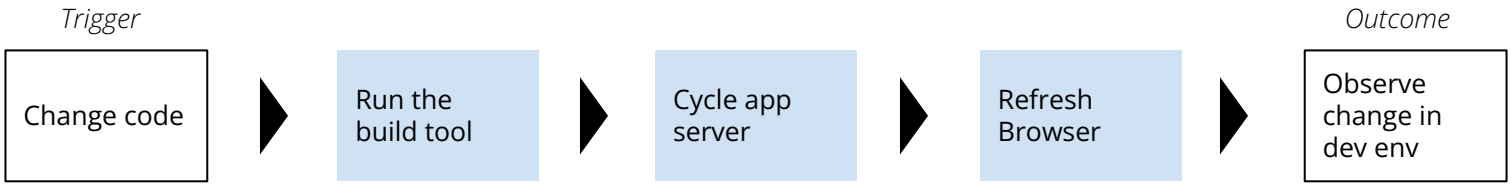
Feedback Loops



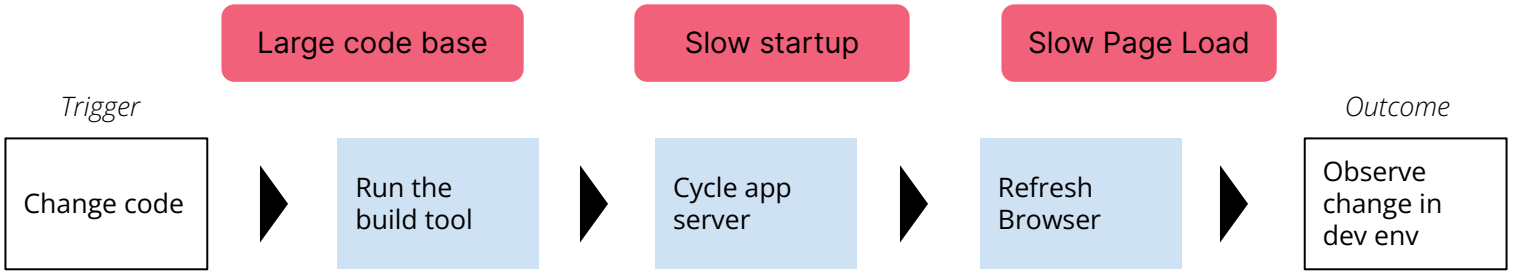
Core Principal: Optimized Feedback Loops

Validate a local code change works	2 mins	→	5-15 seconds (depending on tech choice)
Find root cause for defect	4-7 days	→	1 day
Get answers to a technical query	1-2 weeks	→	30 mins

Micro Feedback Loops



Micro Feedback Loops



2 mins – 3 manual steps

Micro Feedback Loops

2 mins x 20-50 times a day

Micro Feedback Loops

2 mins x 20-50 times a day

+Context Loss

Core Principal: Optimized Feedback Loops

Validate a local code change works	2 mins	→	5-15 seconds (depending on tech choice)
Find root cause for defect	4-7 days	→	1 day
Get answers to a technical query	1-2 weeks	→	30 mins

Core Principal: Leading Metrics

- Compile time
- Time to reload dev app server / UI hot reload
- Test run - Single test
- Test run - For the component I changed
- Test run - For all regression unit tests
- Integration tests - Connected to database / third party
- Unit tests coverage
- Linting - code style, tech debt rules
- Search time for knowledge base
- Search relevance for knowledge base
- API docs completeness
- Tech docs quality - via surveys, pull requests

Listen to your developers!

Core Principal: Optimized Feedback Loops

Validate a local code change works	2 mins	→	5-15 seconds (depending on tech choice)
Find root cause for defect	4-7 days	→	1 day
Get answers to a technical query	1-2 weeks	→	30 mins
Validate component integrates with other components	3 days - 2 weeks	→	2 hours
Validate a change meets non-functional requirements	3 months	→	1 day - 1 week (depending on scope of change)
Becomes productive on new team	2 months	→	4 weeks
Launch a new service in production	2-4 months	→	3 days
Validate a change was useful to a customer	6 months or never	→	1 - 4 weeks (depending on scope of change)

Thank you

Tim Cochran
https://www.linkedin.com/in/timcochran/