Software architecture in a DevOps world

Bert Jan Schrijver
bertjan@openvalue.eu
@bjschrijver

Let's meet

Bert Jan Schrijver

OPENVALUE

nl. jug

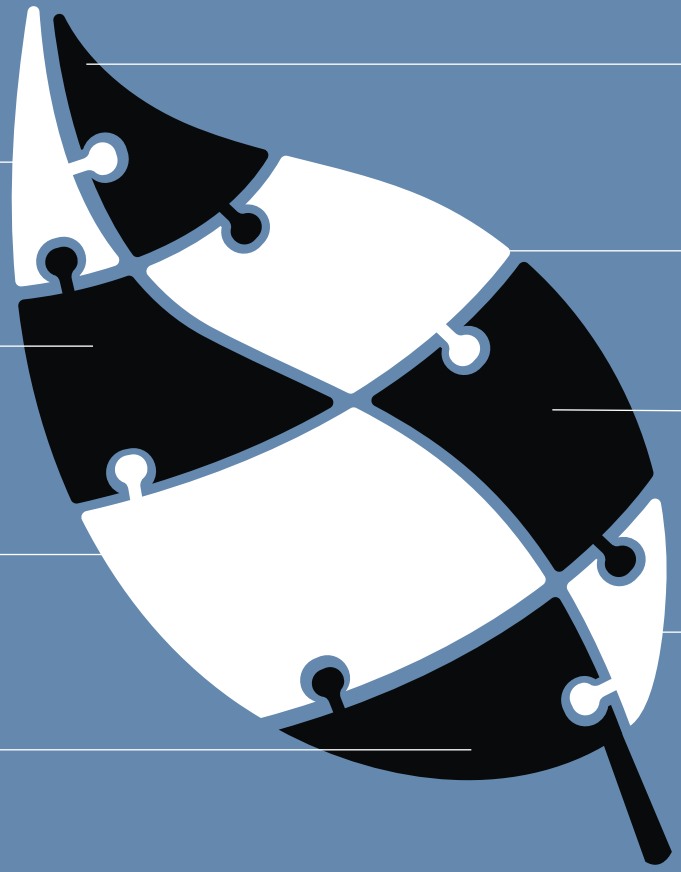
iSAQB ACCREDITED Trainer

@bjschrijver

Software architecture in a DevOps world

Outline

- Introduction
- What is DevOps?
- Why DevOps?
- Principles of DevOps



- DevOps applied to software architecture
- Flexibility in software
- Conclusion
- Q&A

@bjschrijver

What is software architecture?

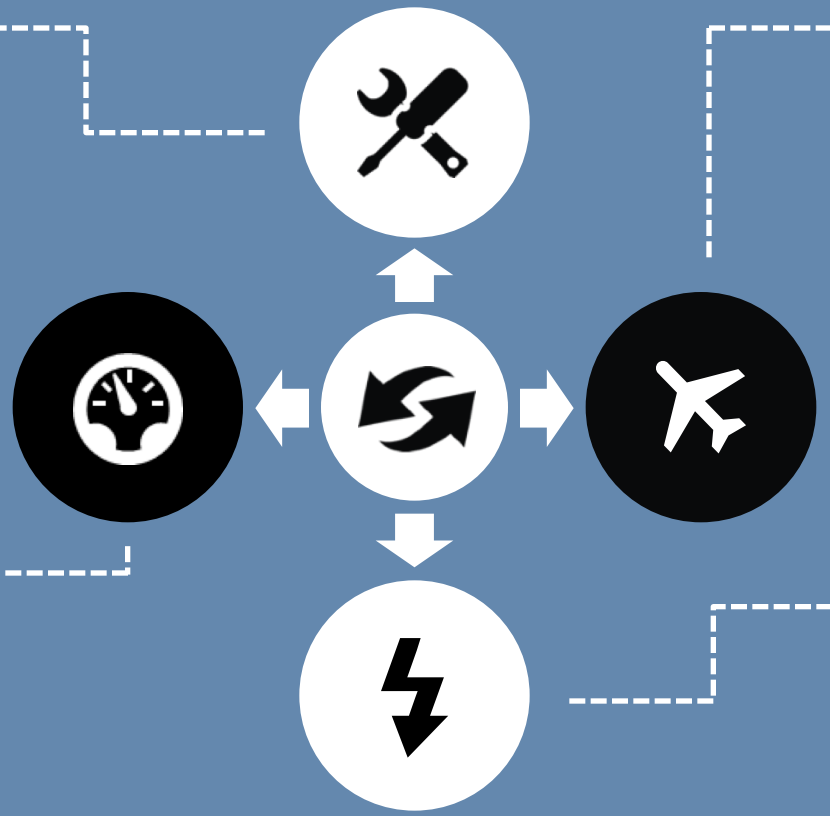
- Sum of all decisions taken during development
- Making design decisions about important stuff
- Enabling a team to build software that provides solutions for business problems
- Allows you to build systems that are beyond the scale that you can hold in your head

Definitions

Who's who in CD & DevOps

Continuous Integration

Team members integrate their work frequently. Commits are verified by automated builds and tests.



Continuous Deployment

Every change goes through the build/test pipeline and automatically gets put into production.

Continuous Delivery

Building and testing software in such a way that the software can be released to production at any time.

"Ship early, ship often, sacrificing features, never quality" - Kyle Neath

DevOps

?!

@bjschrijver

DevOps: a definition

DevOps: another definition

Developers and other IT professionals working together on a shared goal:

building and running better-quality software more quickly and more reliably.

DevOps: another definition ;-)

A set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality.

Why DevOps?

DevOps makes your business move faster

as an enabler for Continuous Delivery, faster problem resolution and focus on value instead of problems.


What is DevOps really about?

Source: http://www.azmc.org/wp-content/uploads/2012/11/Arts-and-Culture-Large.jpg

Source: https://images.unsplash.com/photo-1449177009399-be6867ef0505

Source: http://www.scarymommy.com/wp-content/uploads/2014/06/empathy.jpg

Principles

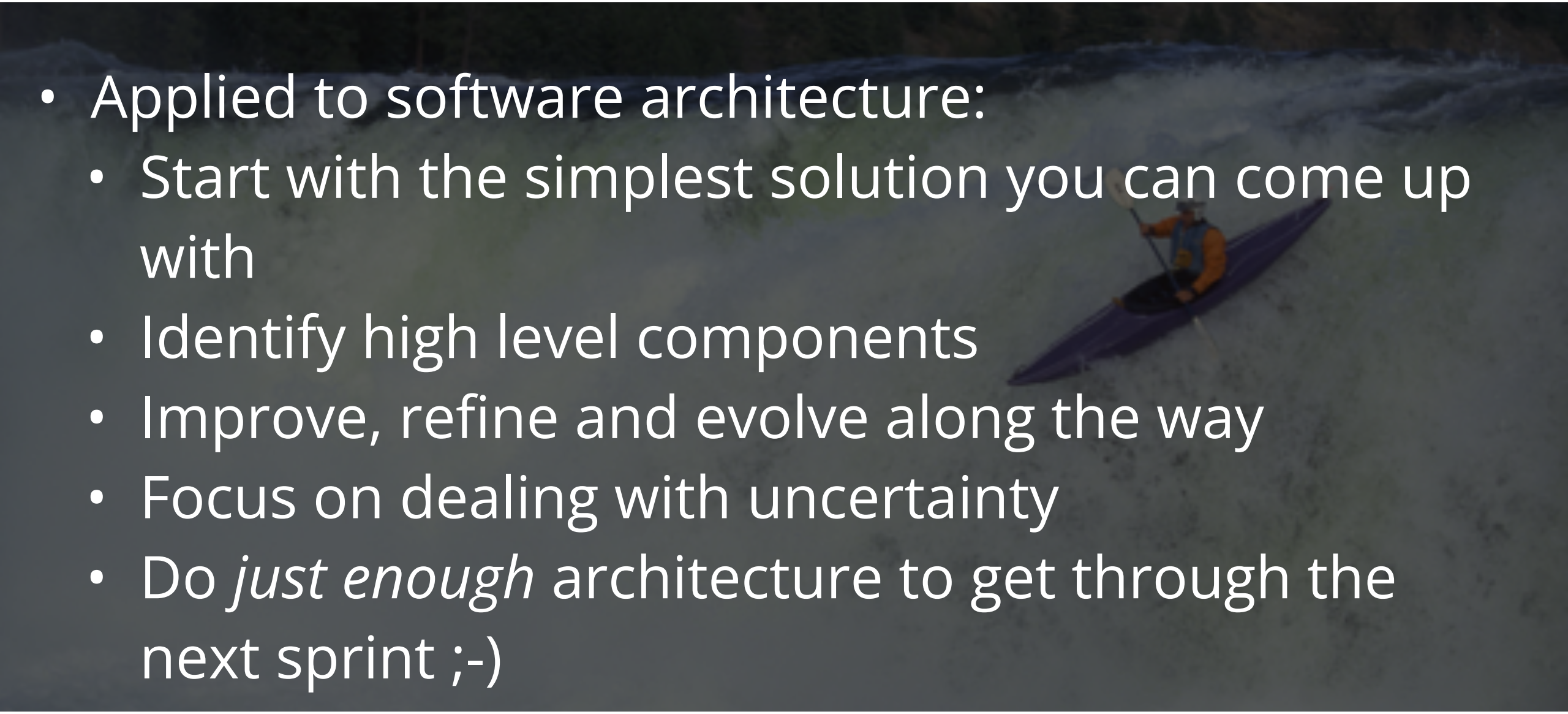
of DevOps

- Gradual changes
- Customer oriented
- Automation
- Ownership
- Collaboration
- Experimentation
- Continuous improvement

@bjschrijver

Software architecture and DevOps

DevOps principle: gradual changes

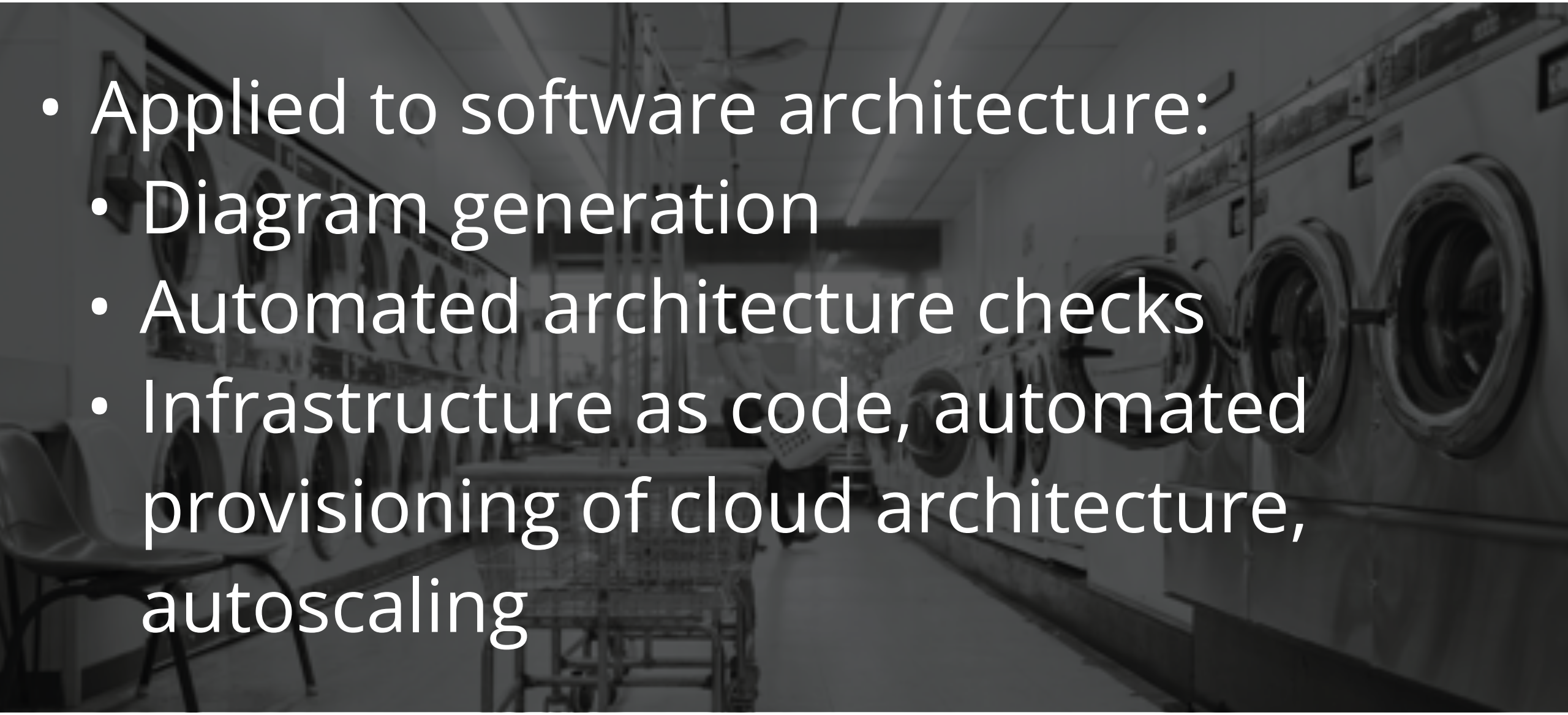


- Applied to software architecture:
  - Start with the simplest solution you can come up with
  - Identify high level components
  - Improve, refine and evolve along the way
  - Focus on dealing with uncertainty
  - Do *just enough* architecture to get through the next sprint ;-)

DevOps principle: customer oriented

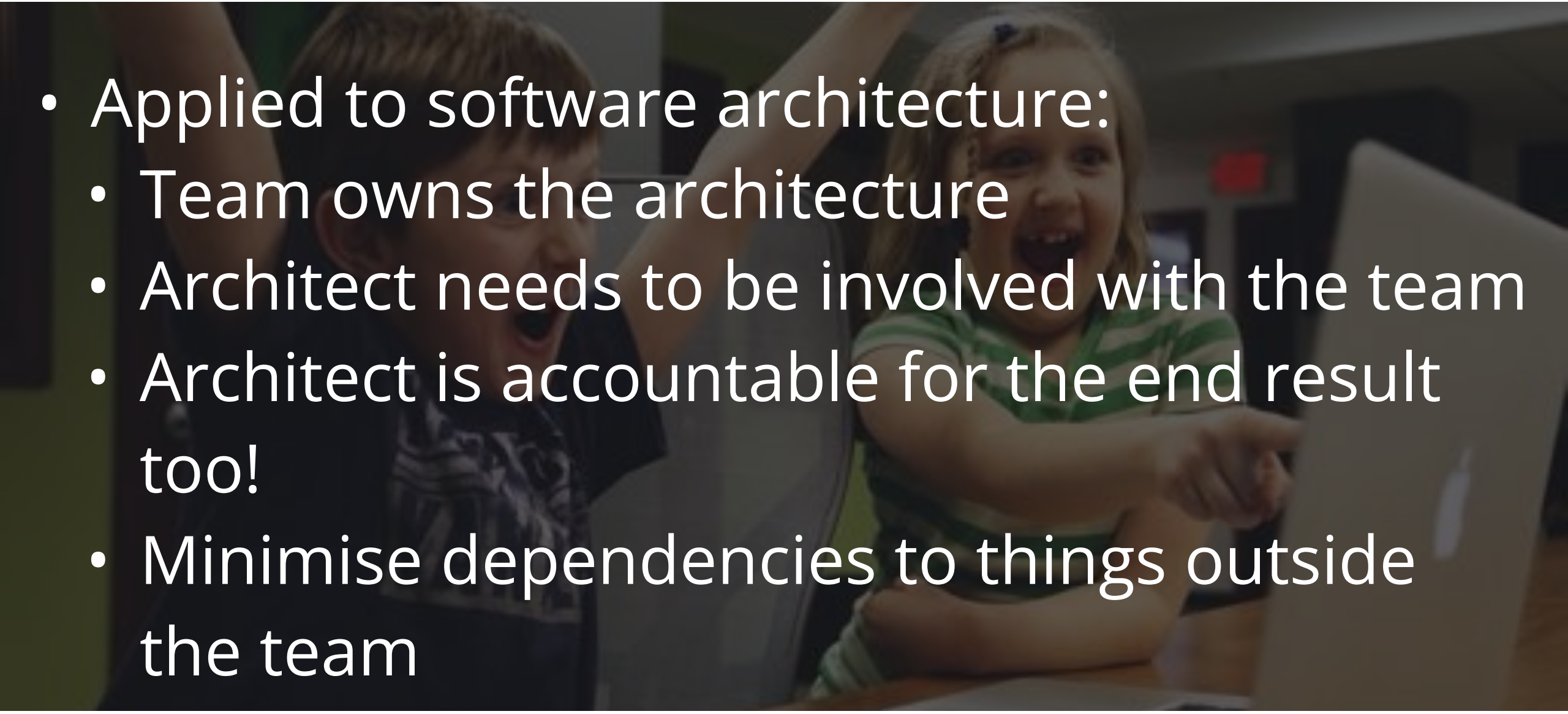
- Applied to software architecture:
  - Focus on the business' needs
  - Don't design for yourself!
  - Help out with clarifying non-functional requirements
  - Architecture is about tradeoffs

DevOps principle: automation



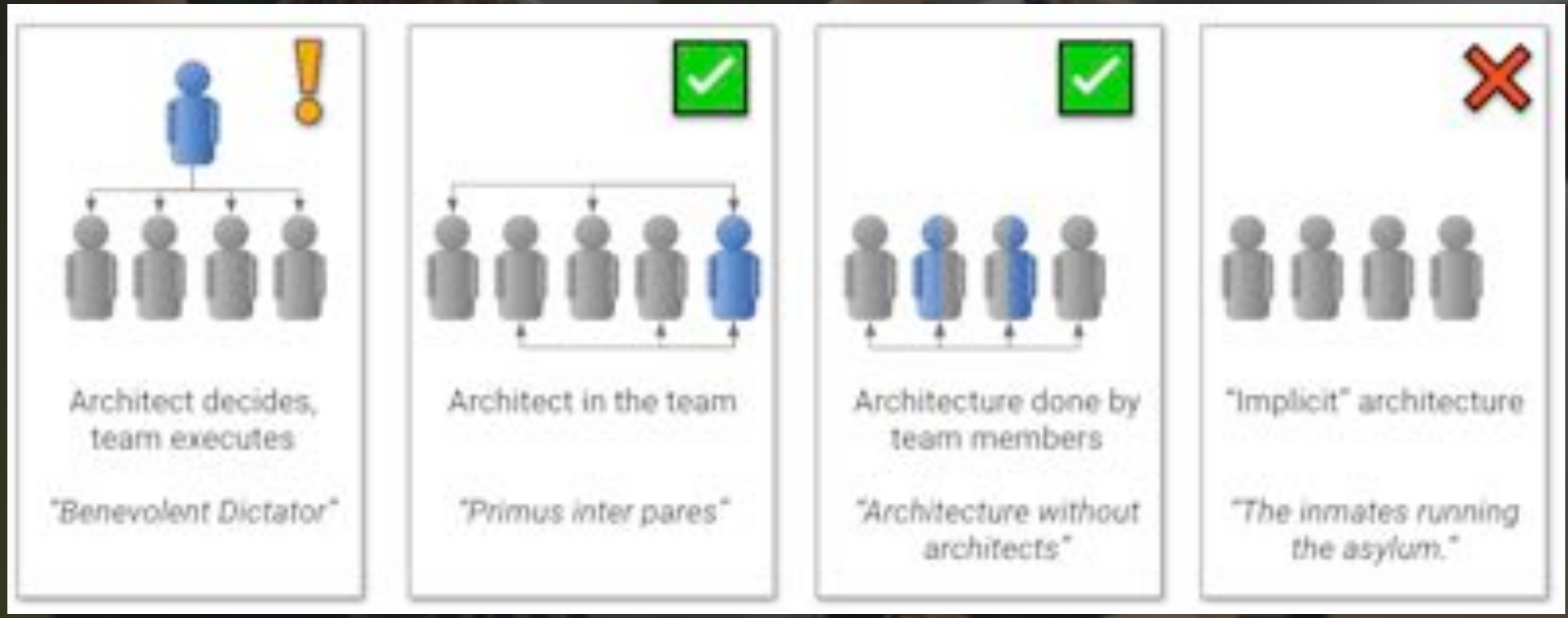
- Applied to software architecture:
  - Diagram generation
  - Automated architecture checks
  - Infrastructure as code, automated provisioning of cloud architecture, autoscaling

DevOps principle: ownership



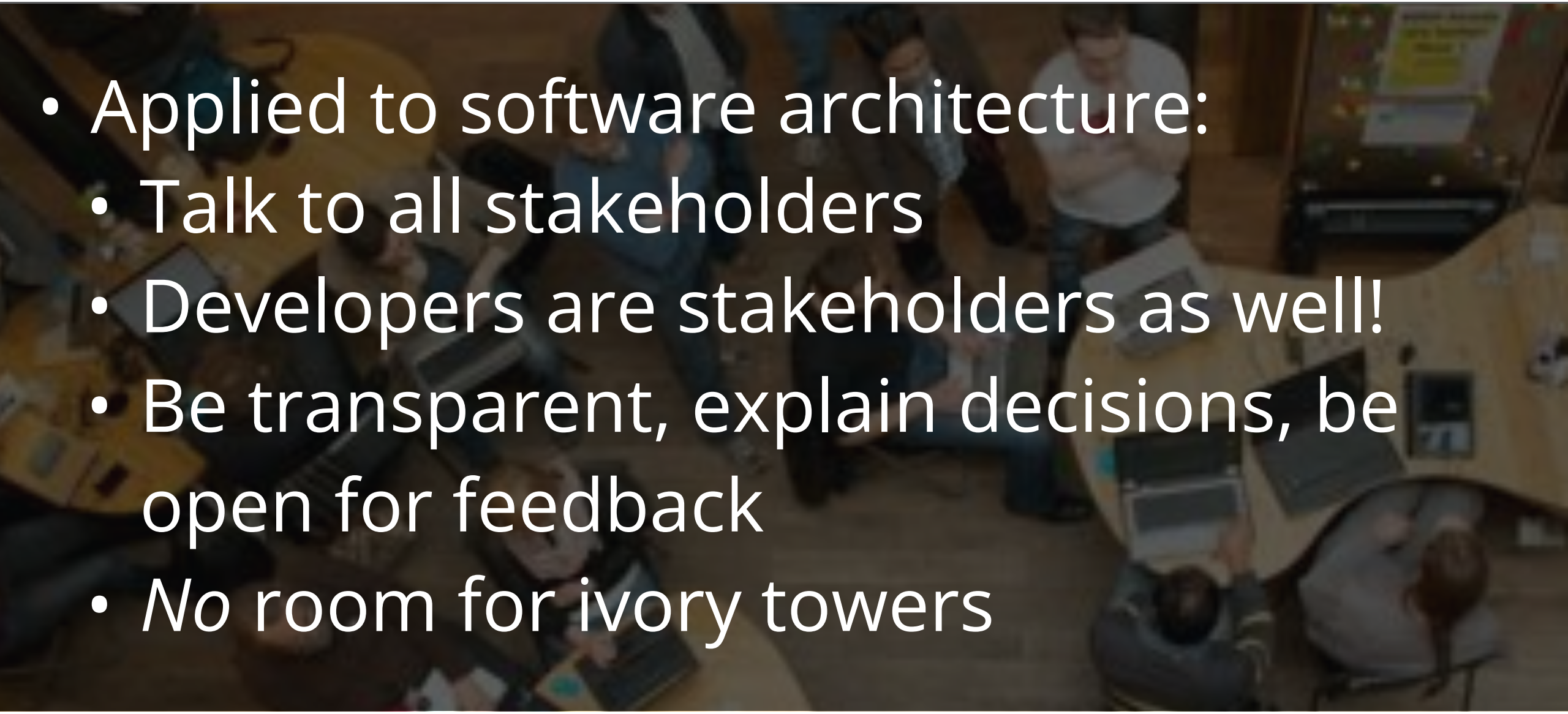
- Applied to software architecture:
  - Team owns the architecture
  - Architect needs to be involved with the team
  - Architect is accountable for the end result too!
  - Minimise dependencies to things outside the team

DevOps principle: ownership



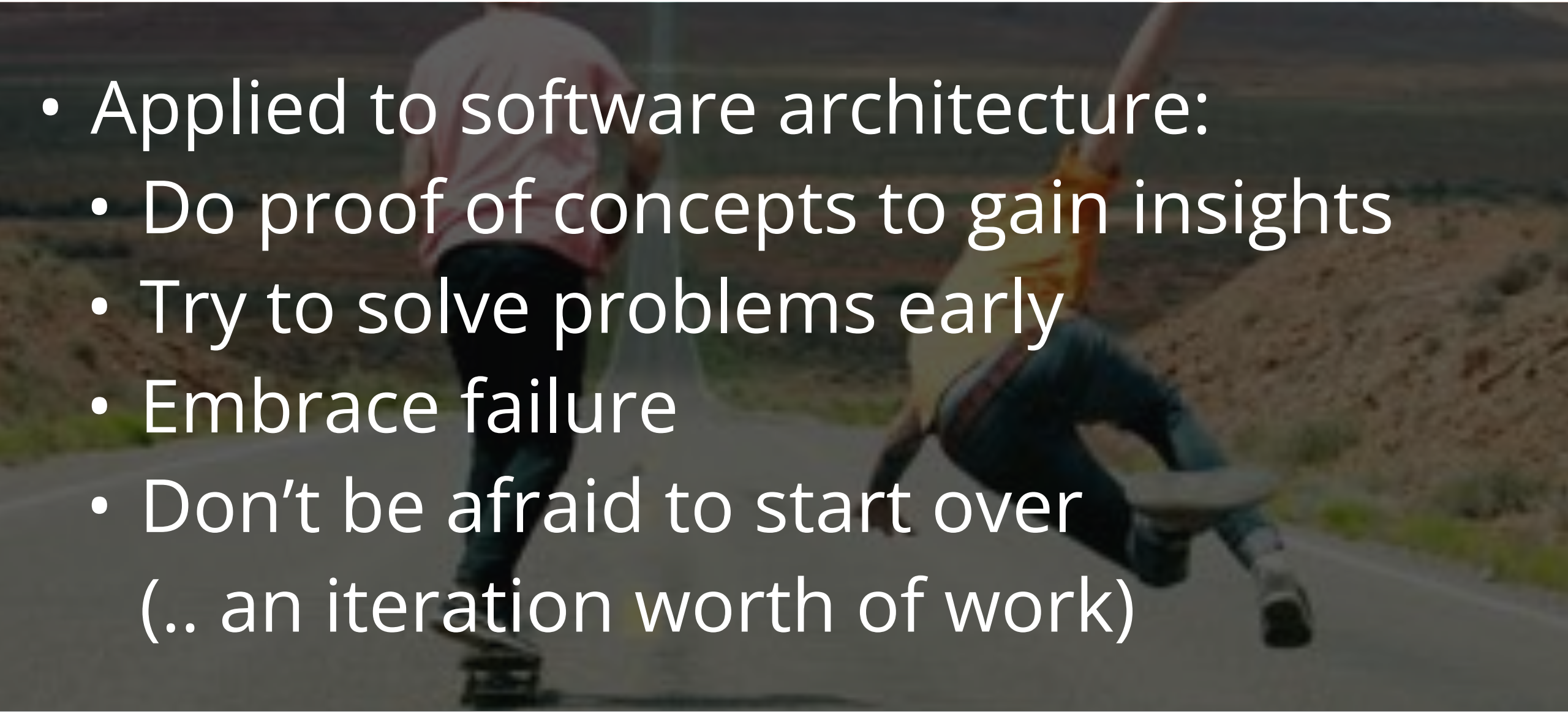
Source: https://architectelevator.com/architecture/organizing-architecture

DevOps principle: collaboration



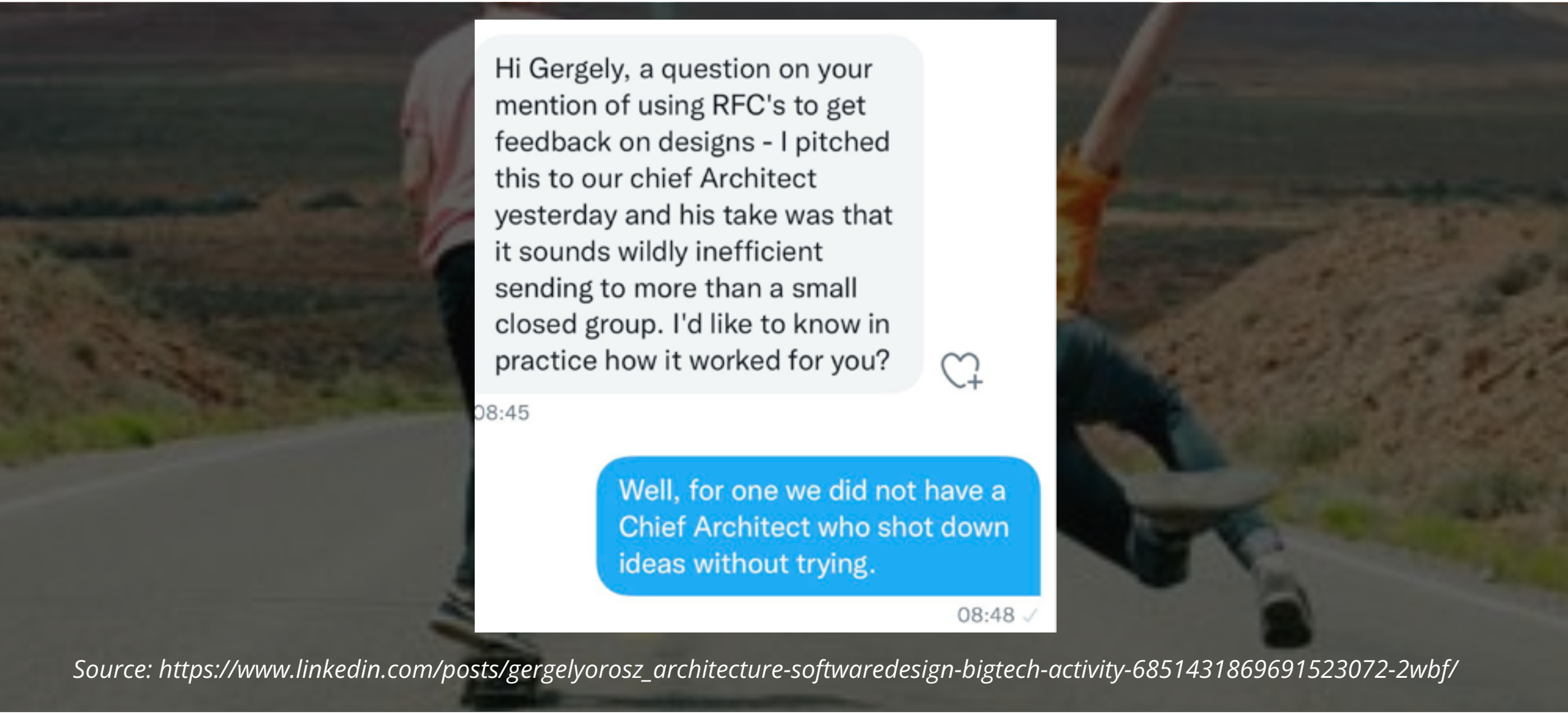
- Applied to software architecture:
  - Talk to all stakeholders
  - Developers are stakeholders as well!
  - Be transparent, explain decisions, be open for feedback
  - *No* room for ivory towers

DevOps principle: experimentation



- Applied to software architecture:
  - Do proof of concepts to gain insights
  - Try to solve problems early
  - Embrace failure
  - Don’t be afraid to start over
    (.. an iteration worth of work)

DevOps principle: experimentation



Source: https://www.linkedin.com/posts/gergelyorosz_architecture-softwaredesign-bigtech-activity-6851431869691523072-2wbf/

DevOps principle: continuous improvement

- Applied to software architecture:
  - Focus on feedback loops
  - Ask developers for feedback!
  - Measure & monitor
  - Evolving architecture instead of up-front
  - Software architecture is never finished

Photo: Dave Lehl, text source: Jez Humble

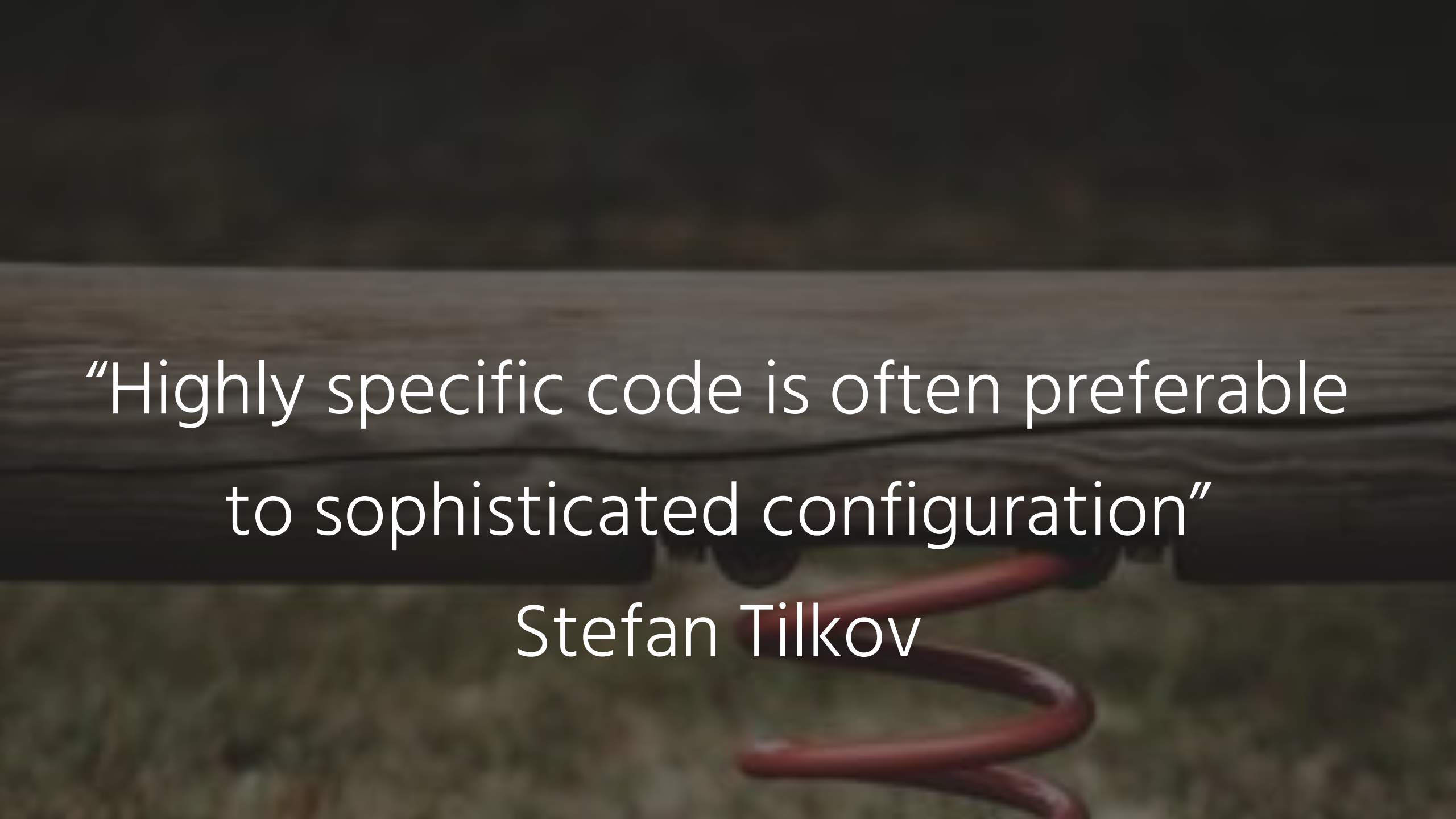
“CD & DevOps won’t work here”

- Stated reasons:
  - we’re regulated
  - we’re not building websites
  - too much legacy
  - our people are too stupid
- Actual reasons:
  - our culture stinks
  - our architecture stinks

Source: talk by Jez Humble
https://www.youtube.com/watch?v=837Z_oehhRg

Photo: Dave Lehl, text source: Jez Humble


About flexibility in software…

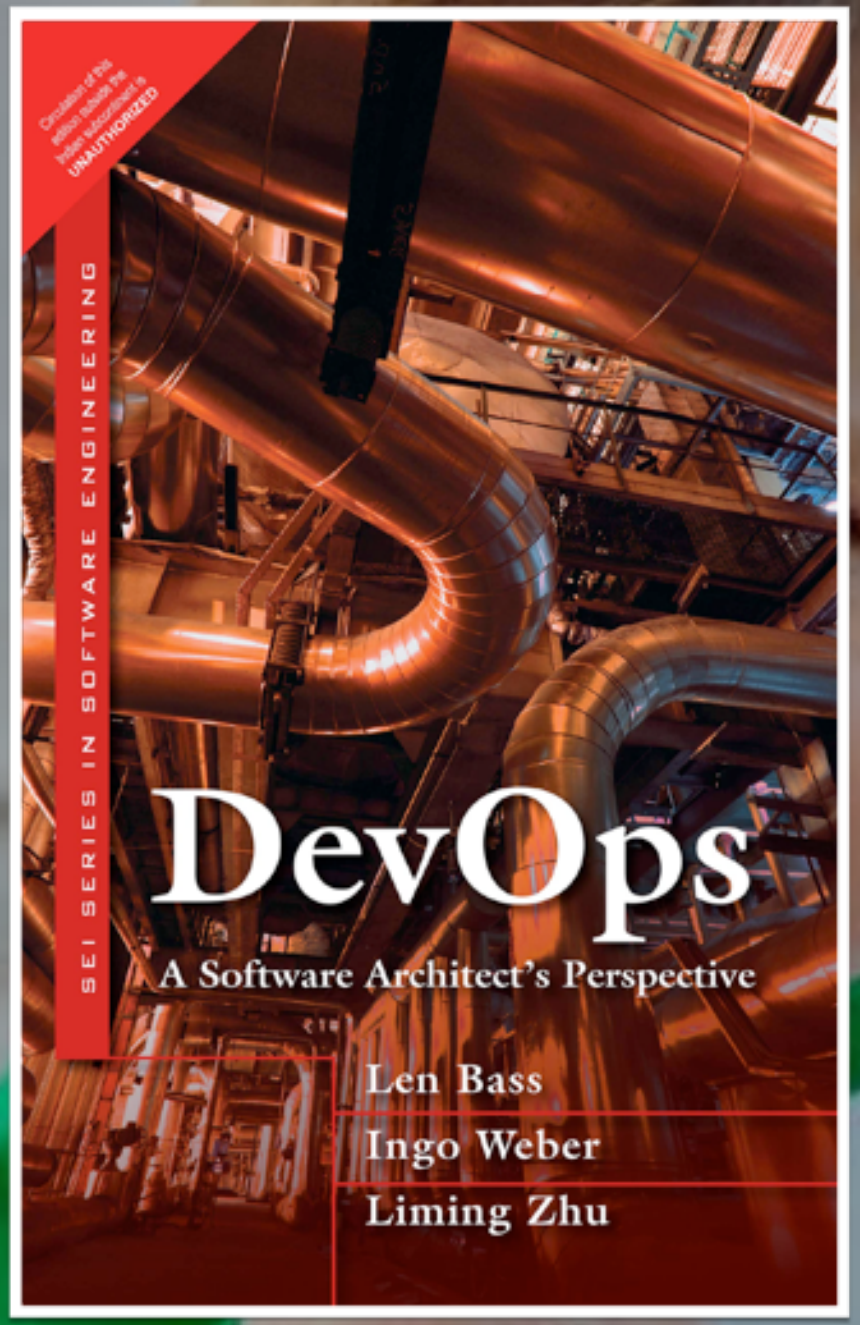

“Highly specific code is often preferable to sophisticated configuration”
Stefan Tilkov

Summary

How do you work as an architect with(in) a DevOps team?

By applying the same DevOps principles to the domain of software architecture.

Source: https://cdn2.vox-cdn.com/thumbor/J9OqPYS7FgI9fjGhnF7AFh8foVY=/148x0:1768x1080/1280x854/cdn0.vox-cdn.com/uploads/chorus_image/image/46147742/cute-success-kid-1920x1080.0.0.jpg



THAT'S IT.

NOW GO KICK SOME ASS!

Source: https://cdn2.vox-cdn.com/thumbor/J9OqPYS7FgI9fjGhnF7AFh8foVY=/148x0:1768x1080/1280x854/cdn0.vox-cdn.com/uploads/chorus_image/image/46147742/cute-success-kid-1920x1080.0.0.jpg


Questions?
@bjschrijver

Thanks for your time.

Got feedback? Tweet it!

@bjschrijver