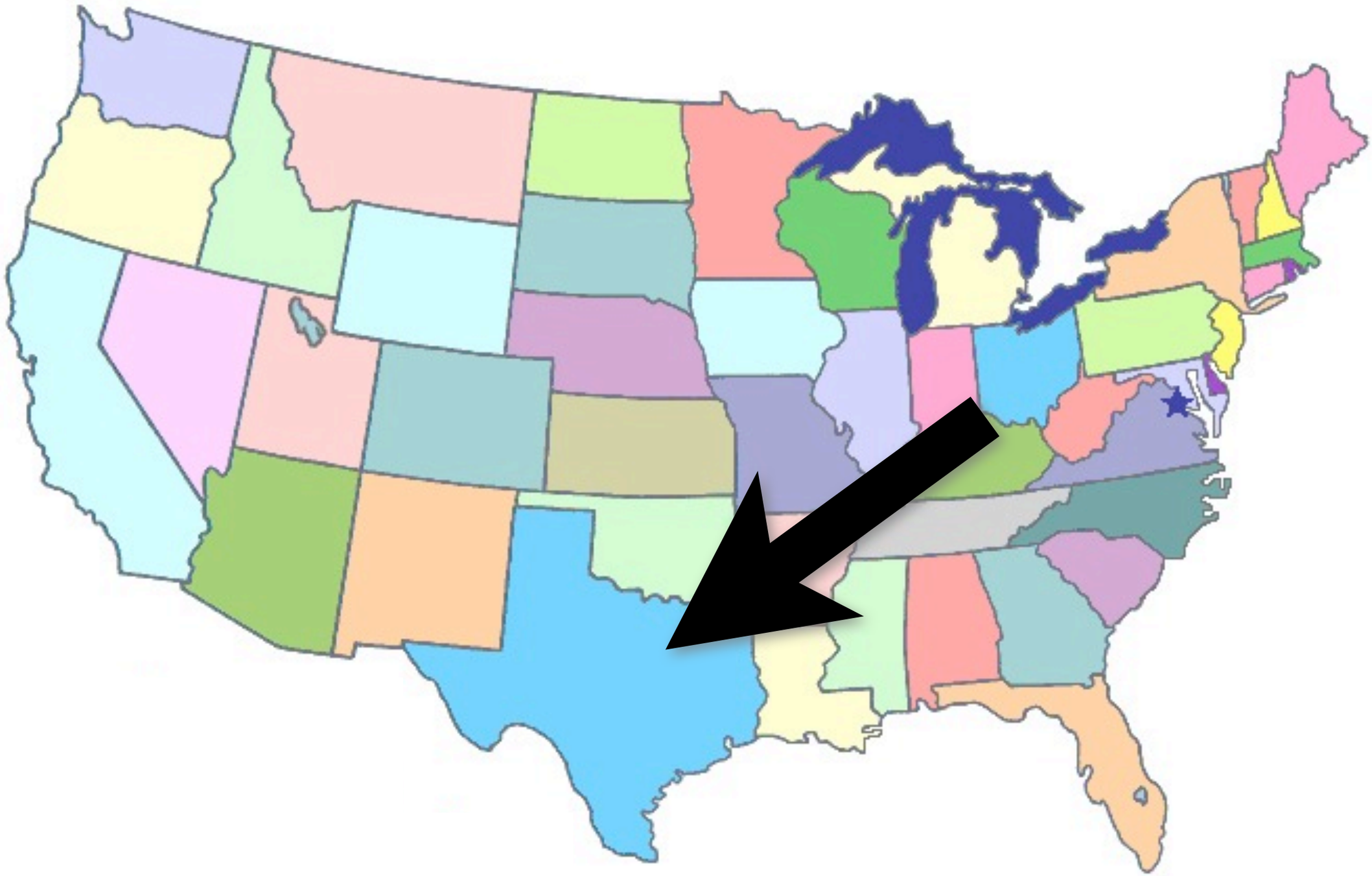
Dave Thomas

敏捷中国大会

ThoughtWorks®

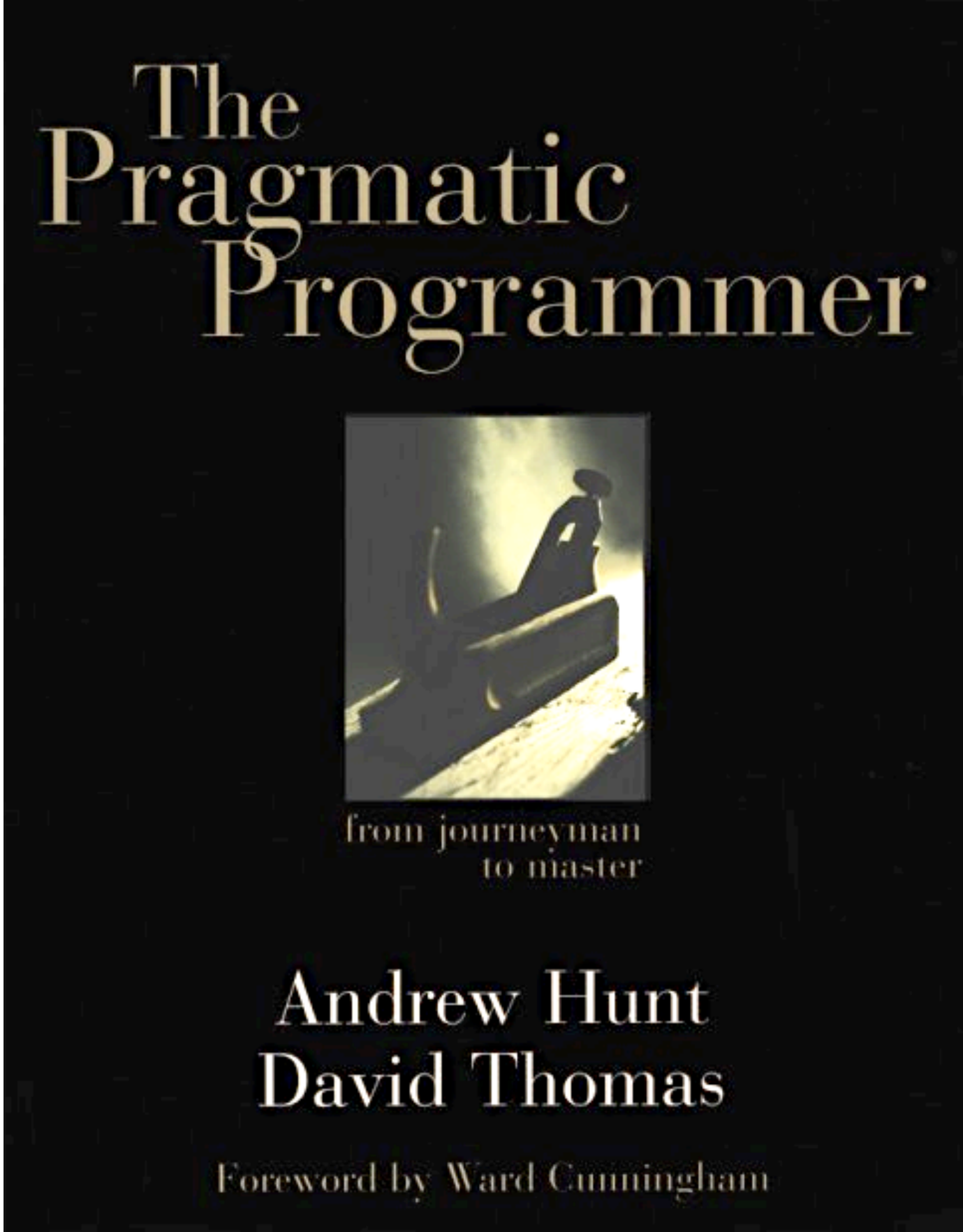
InfoQ queue
Enterprise Software Development Community



- Programmer for over 30 years

- Programmer for over 30 years
- It doesn’t get easier

Wrote a Book


The Pragmatic Programmer
from journeyman to master
Andrew Hunt
David Thomas
Foreword by Ward Cunningham

Manifesto for Agile Software Development

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

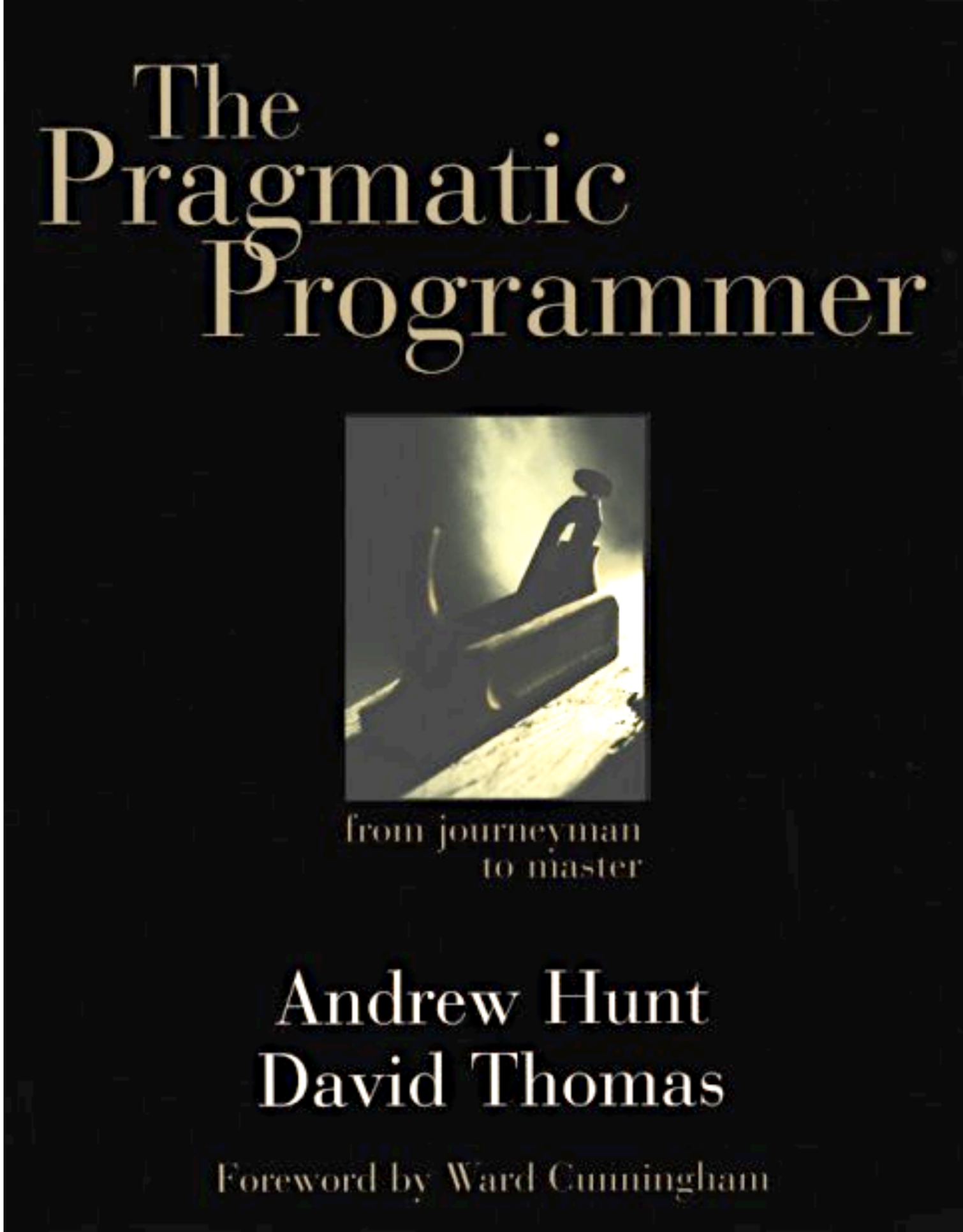
We Were Agile

- In our own way
- Just like Kent Beck and Ron Jeffries at Chrysler
- And many others were doing around the world

It's Nothing Special

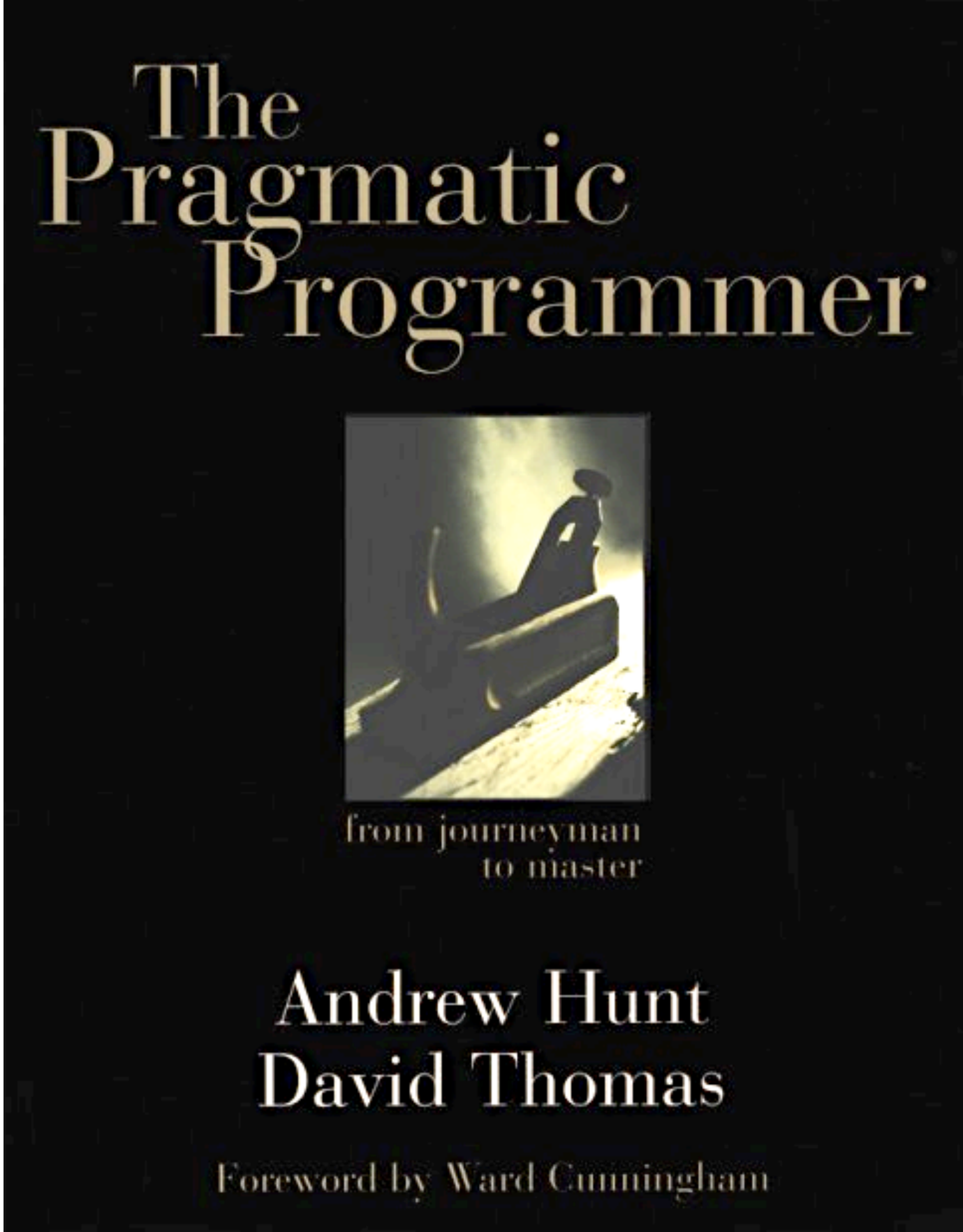
- If you've written a lot of software, and
- If you have the freedom to write it the way you know is right
- The chances are you'll end up using agile practices

Mostly Agile Advice



Mostly Agile Advice

Ten Years Old



and 10 years in

software is like 100

years in many other

disciplines

so, after ten years, is

The Pragmatic Programmer

still valid?

Some Details Have Aged

- tools have moved on
- more people use dynamic languages,
- We now deploy in the cloud,
- Starting to see more parallelism

But The Basic
Are Ideas are
Still the Same

Fix Broken Windows

Broken Windows Theory

- Fix small problems (a broken window in a building)
- Stops larger problems developing

Broken Windows Theory

- Applied in New York City

Broken Windows Theory

- Applied in New York City
- Between 1993 and 1997:
  - serious crimes dropped **44%**
  - robberies dropped **50%**
  - murders dropped **60%**

Broken Windows Theory

- Stop "code rot" and "process rot"
- Fix problems as soon as they occur.
- Fix little problems == fewer big problems

I'll do it later…

There is no later…

The DRY Principle

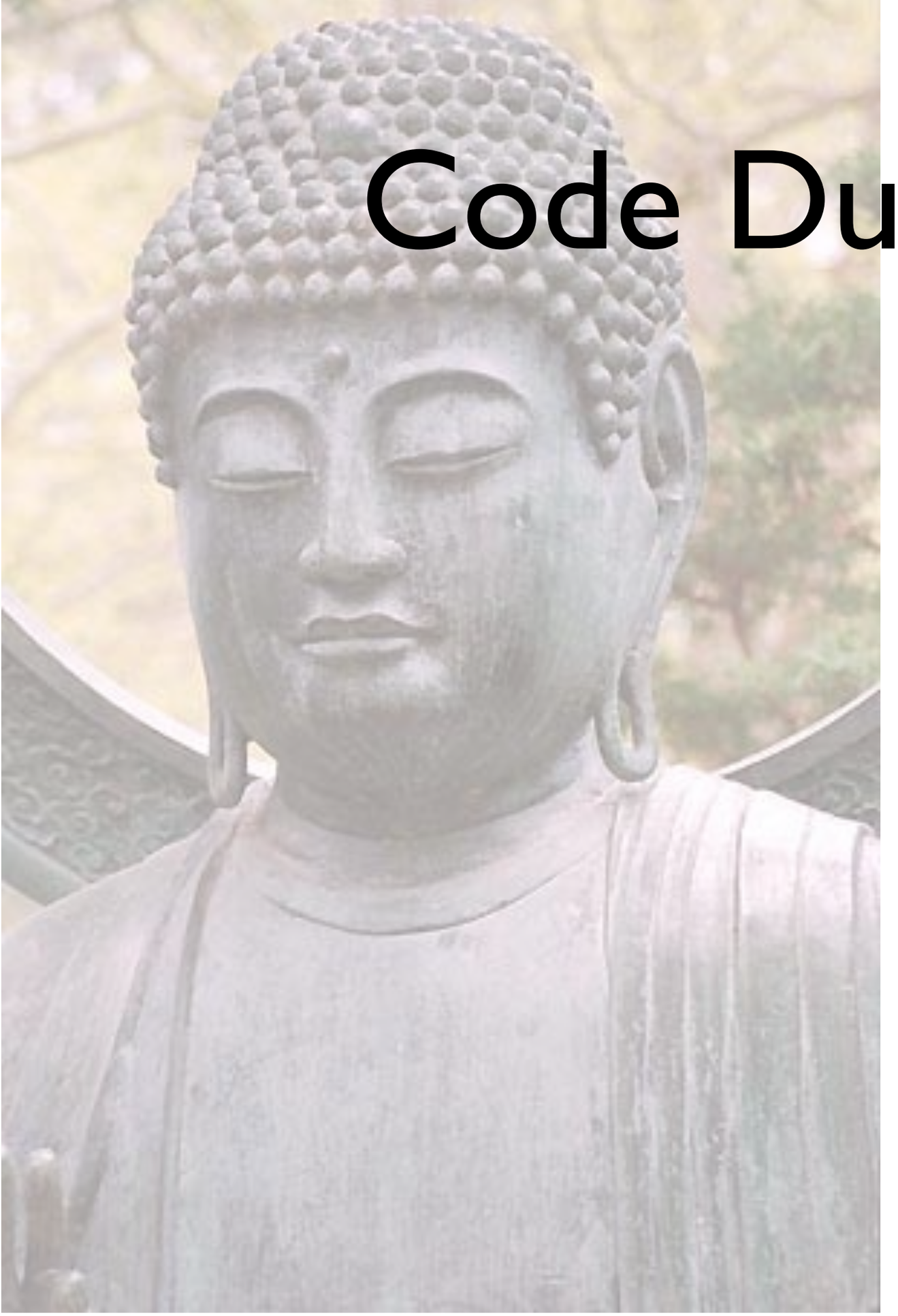
"Be serene in the oneness of things."

D.T.Suzuki,
Manual of Zen Buddhism

The DRY Principle

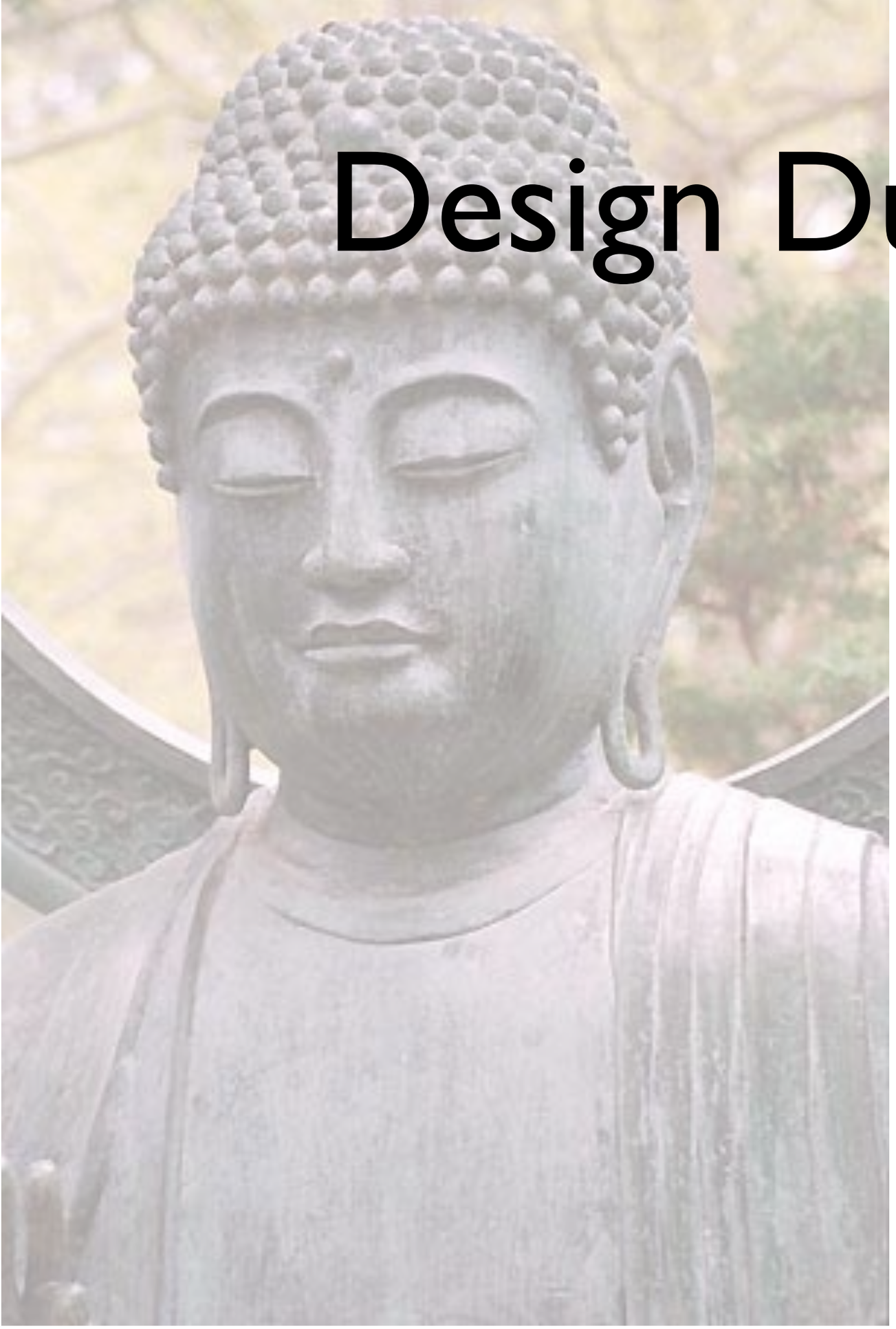
Don't Repeat Yourself:

"Every piece of knowledge must have a single, unambiguous, authoritative representation within a system"



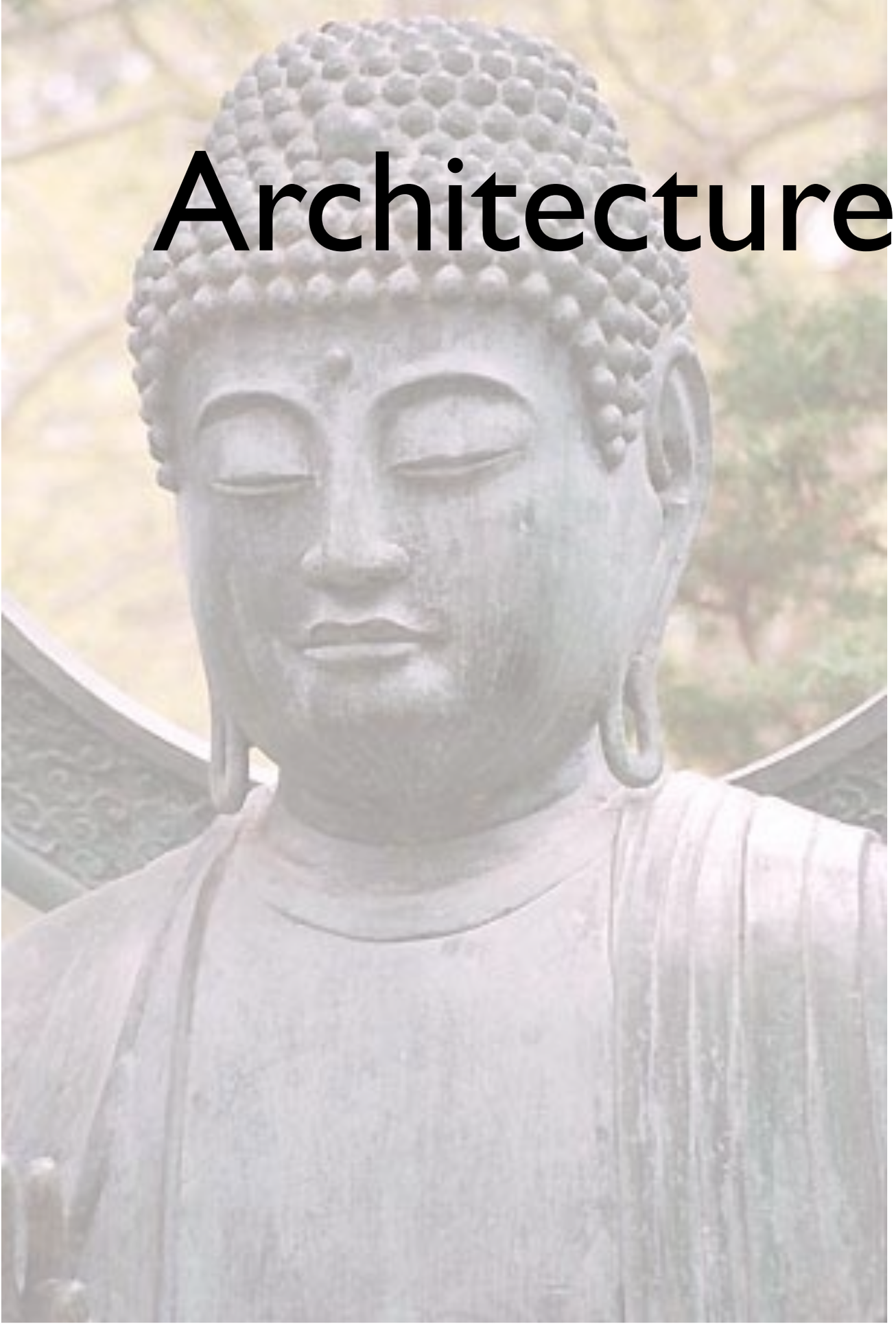
Code Duplication

- Fix with methods, modules, classes…
- Fix with code generators
- Fix with metaprogramming



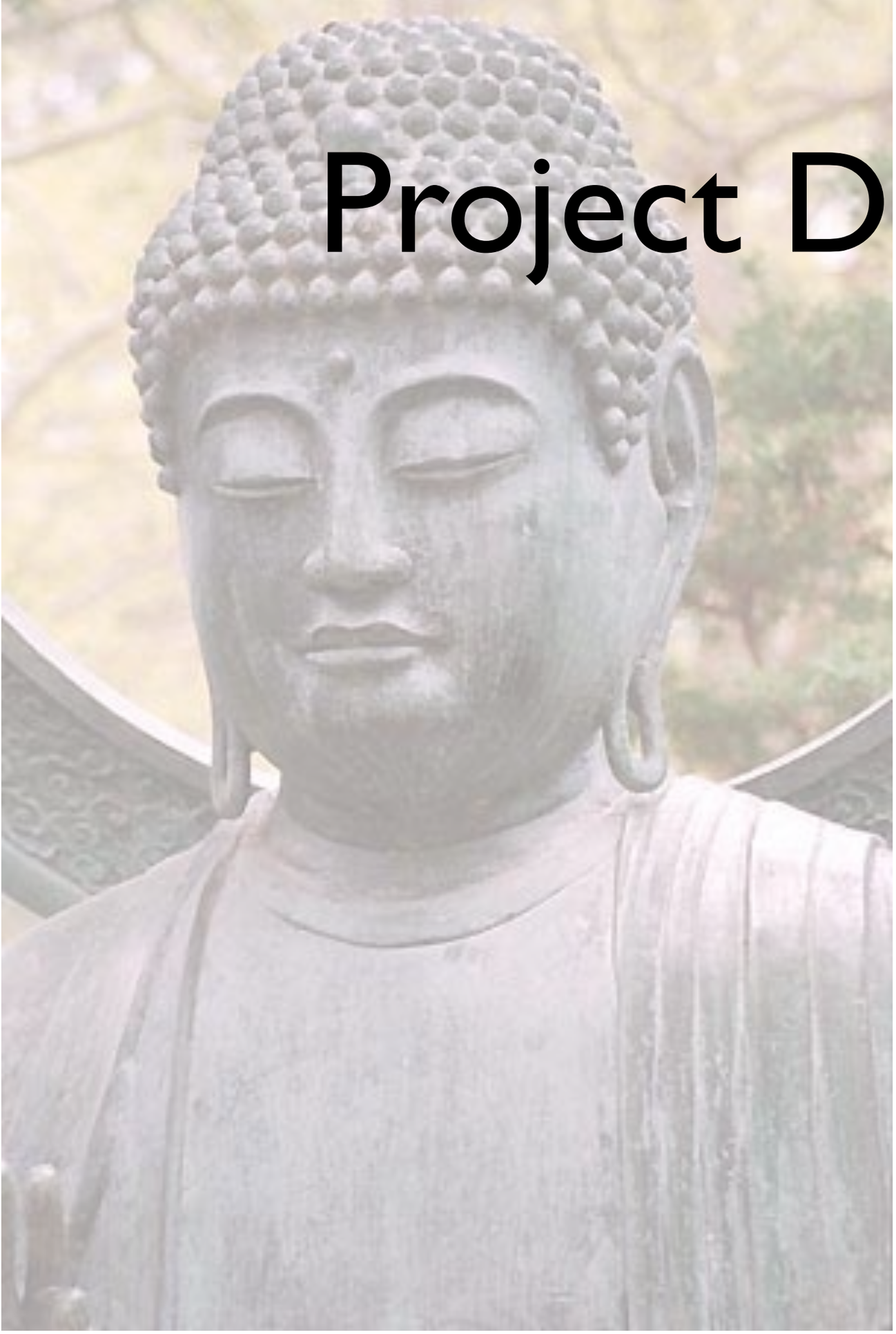
Design Duplication

- Fix with patterns
- Fix with conventions
- Fix with metaphors



Architecture Duplication

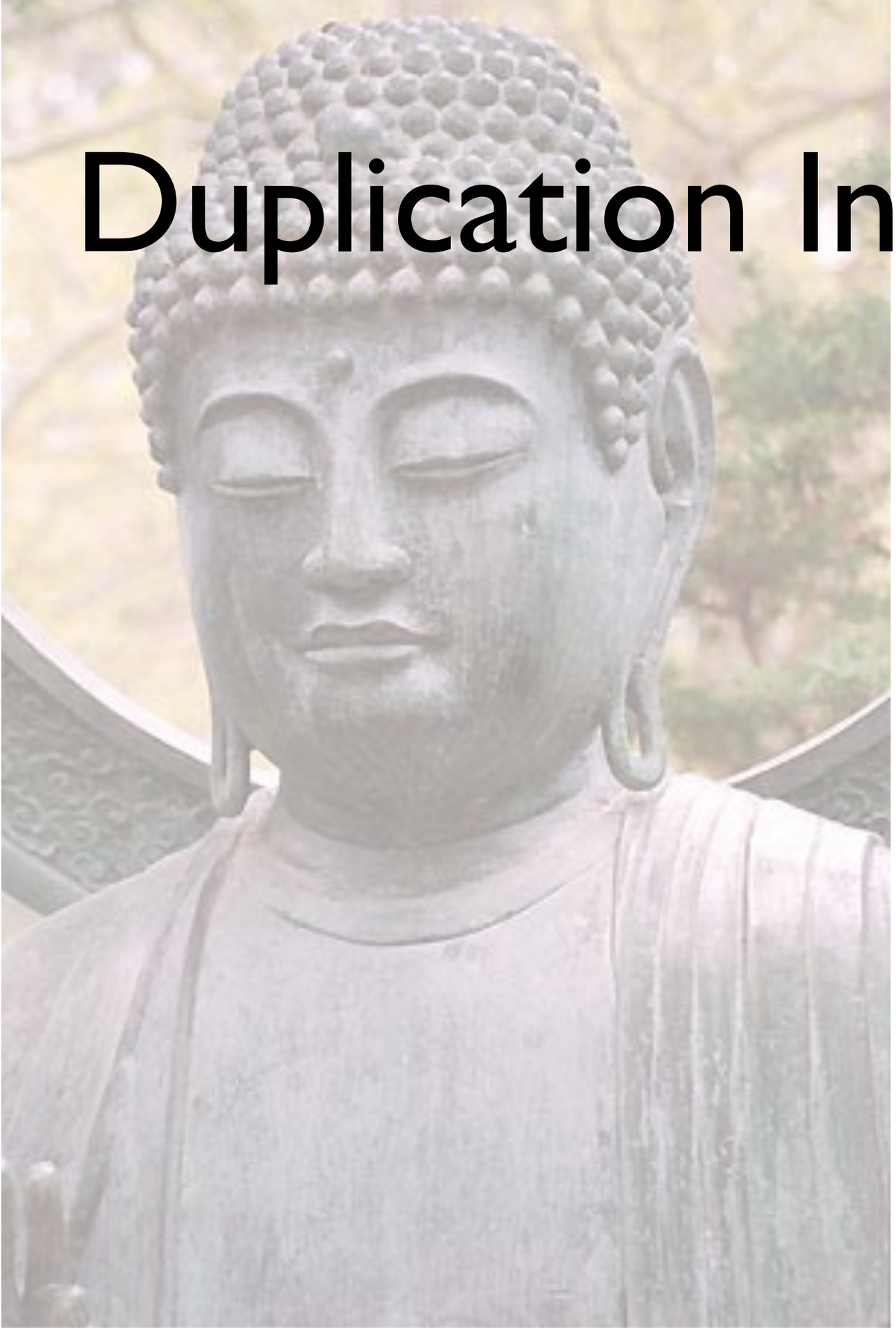
- Fix with frameworks
- Fix with service-based architectures



Project Duplication

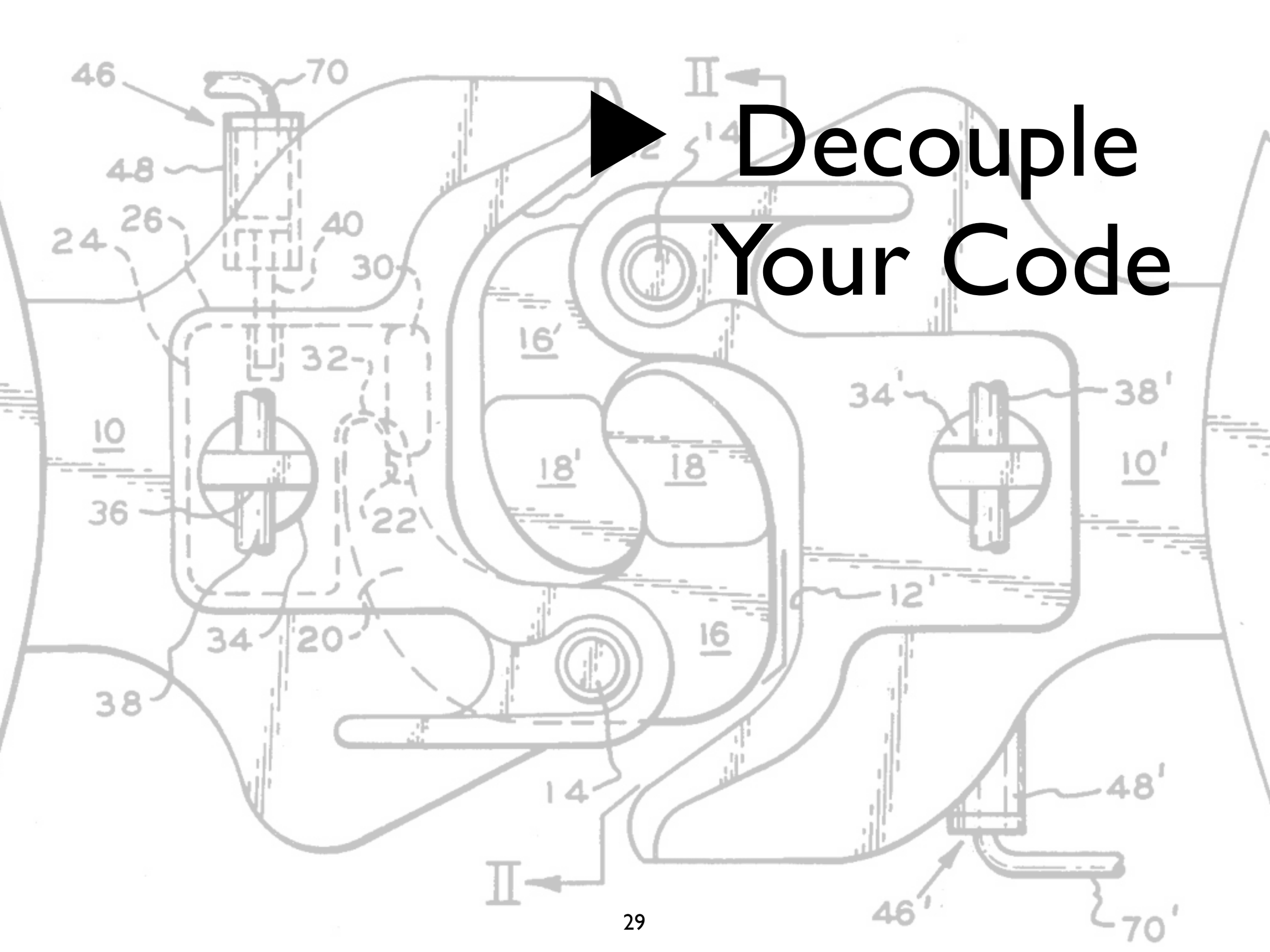
- Fix by producing products
- Fix with data-driven designs

Duplication Inhibits Change



- One business decision—multiple changes
  - potentially at many levels
- Maintenance becomes difficult
  - easy to forget to change something

Decouple Your Code

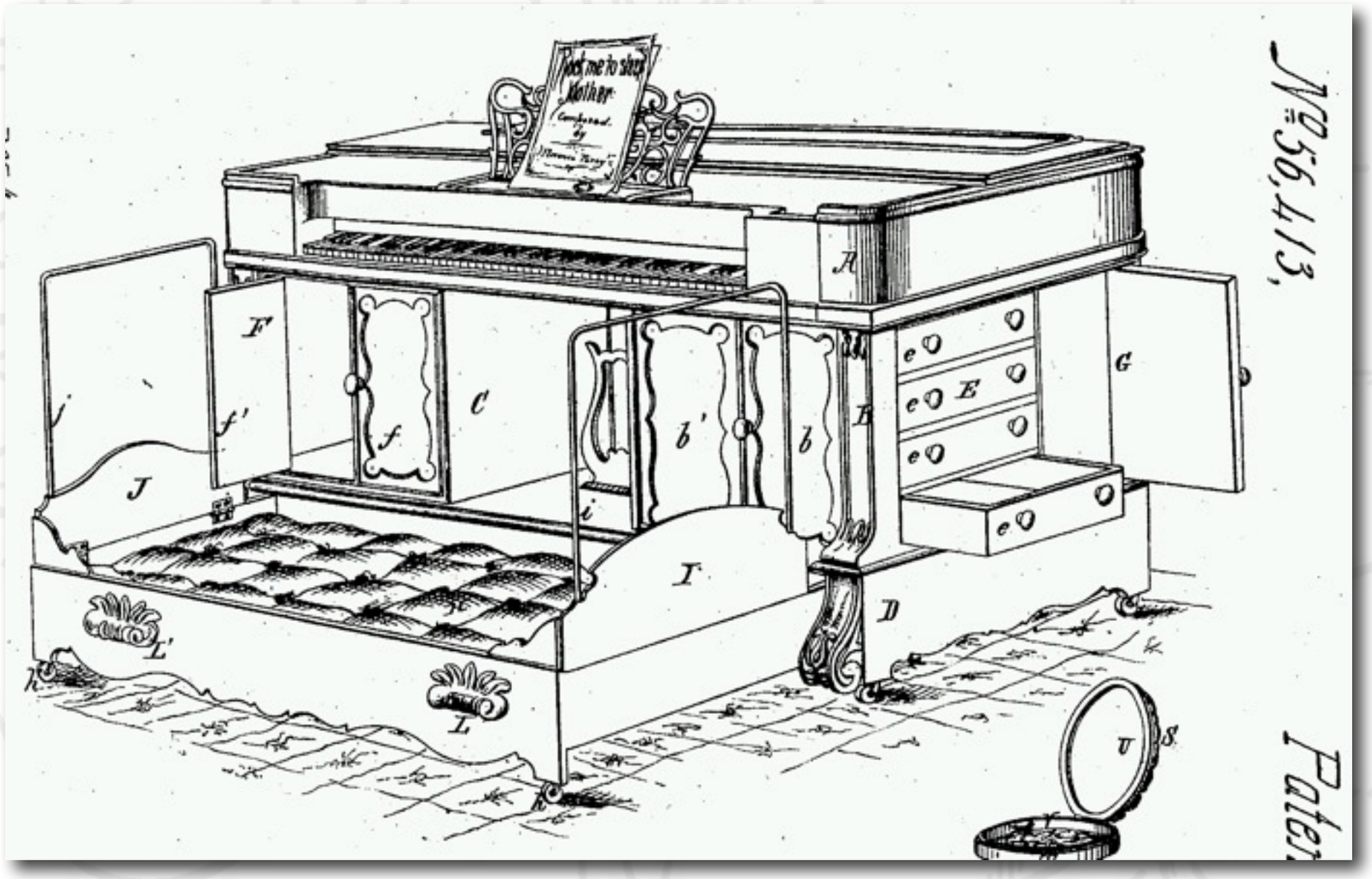


Keep Things Unrelated

- Why?
  - Easier to write
  - Easier to understand
  - Easier to change
- Techniques
  - increase cohesion
  - reduce coupling

Cohesion:

Do One Thing Well...



U.S. Patent No. 56,413, July 17, 1866, Charles Hess, “Convertible Sofa Bedroom Piano”

Reduce Coupling

- Overly-coupled code is:
  - Hard to understand
  - Hard to maintain
  - Difficult to reuse

▶ Source Control

▶ Unit Test

"The easiest person to deceive is oneself"

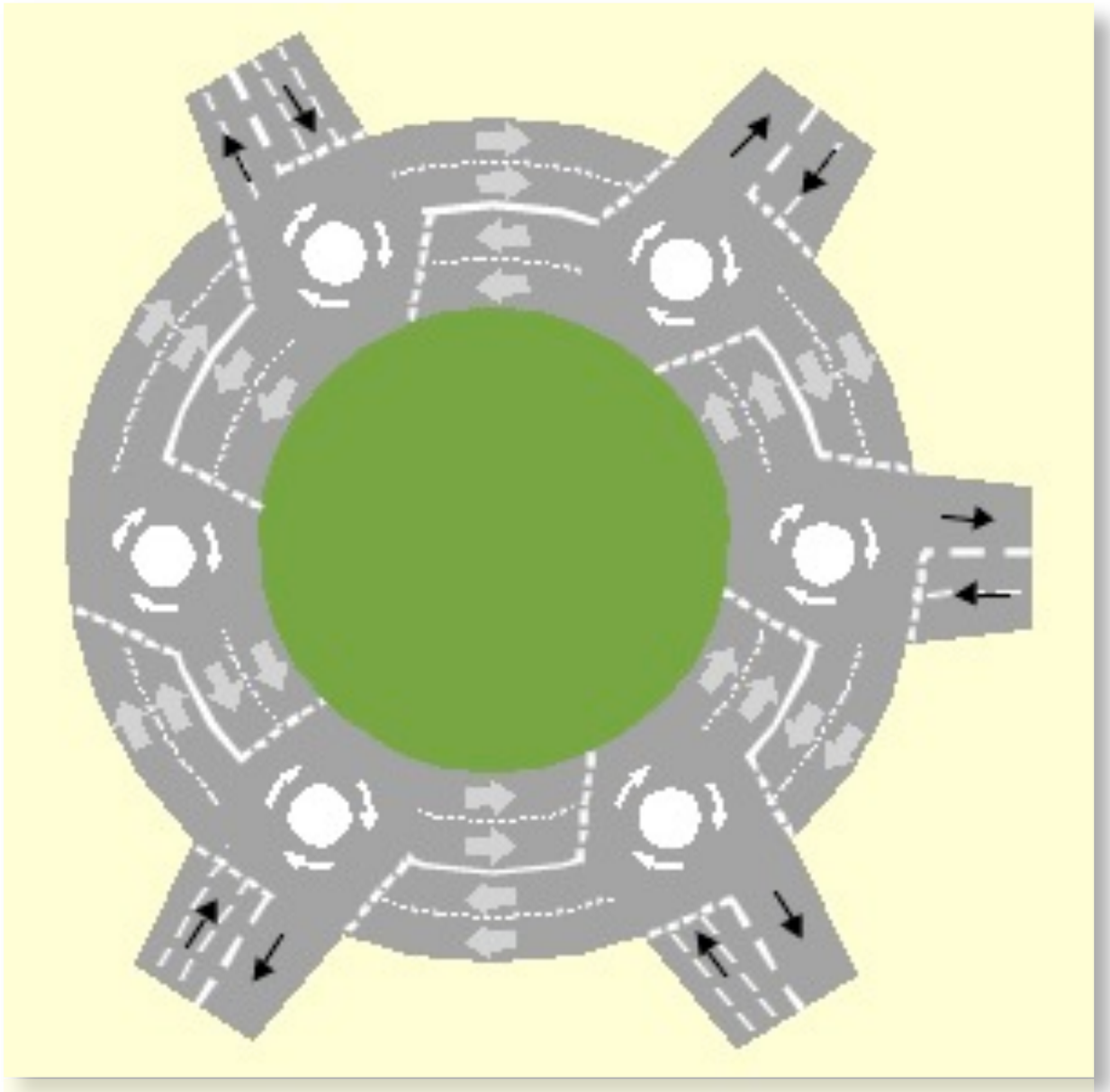
Edward Bulwer-Lytton, The Disowned

"Civilization advances by extending the number of important operations we can perform without thinking"

Alfred North Whitehead

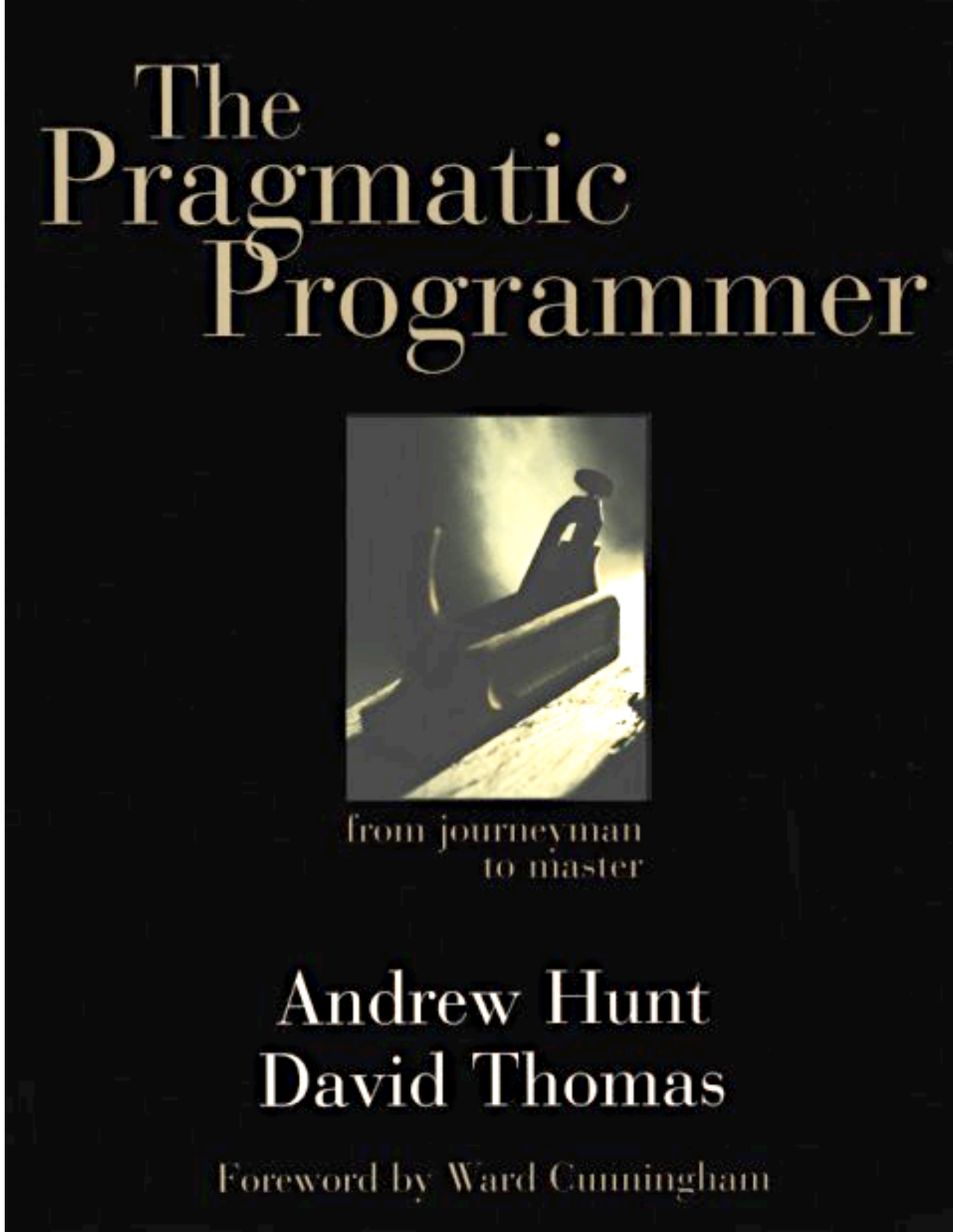
Automate Everything

Do Nothing Twice



Develop in Iterations

What should change?



- Dynamic languages
- Distributed version control
- Decoupling in the cloud
- …

Pragmatic Programmer

Agile Manifesto

Pragmatic Agile

Seems natural

Except...

agile |ˈajəl|

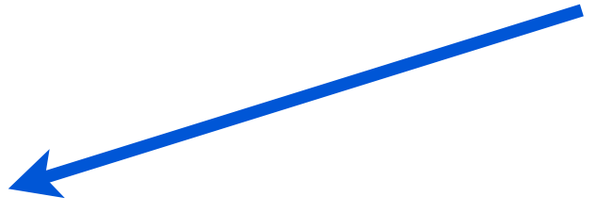
adjective

able to move quickly and easily

agile |ˈajəl|

adjective

Agile is not a thing



able to move quickly and easily

Agile is not a thing

Pragmatic Red

Pragmatic Dangerous

Pragmatic Ugly

Agile is an Adjective

- applies to something else:
  - what you do
  - how you do it
  - what you think about

Agile Needs An Actor

you