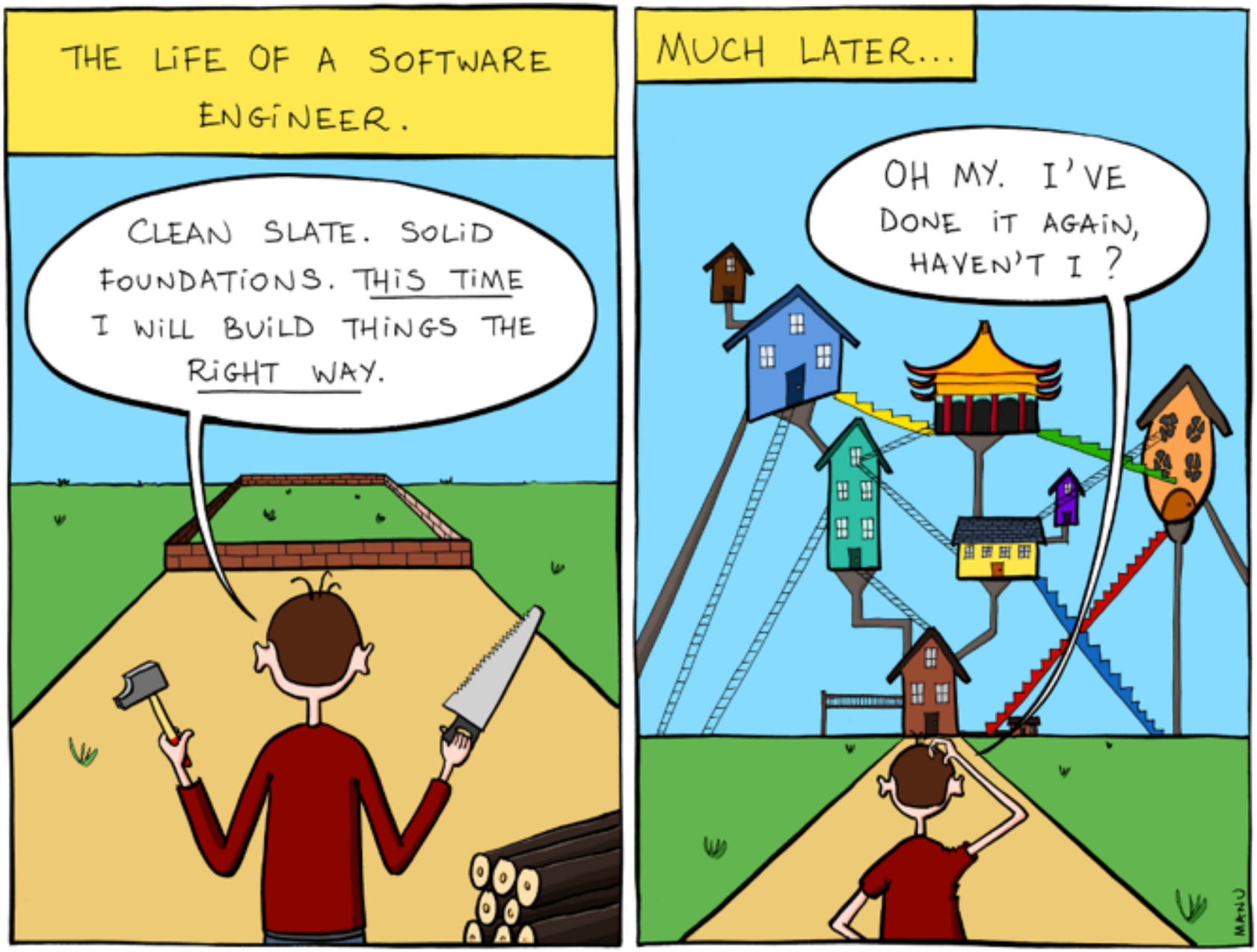
# Advanced Programming
## (with python)

David Beazley
@dabeaz
http://www.dabeaz.com




http://www.bonkersworld.net

# What is Programming?

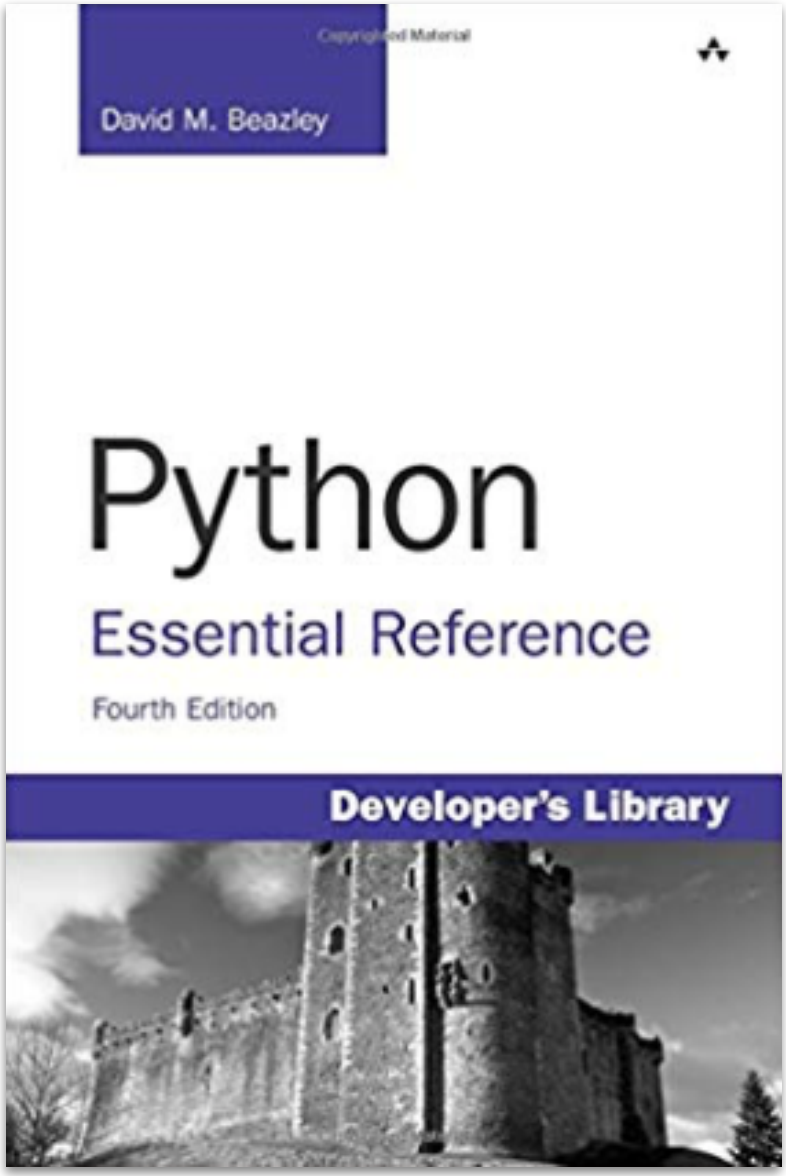
- Problem solving
- Software design/architecture
- Composition of components
- Abstraction
- Decision making
- ???? Probably many other things...

# This Course

- It's mostly about programming
- We'll be using Python
- And to be sure, we'll say a lot about Python
- But it's meant to be more general

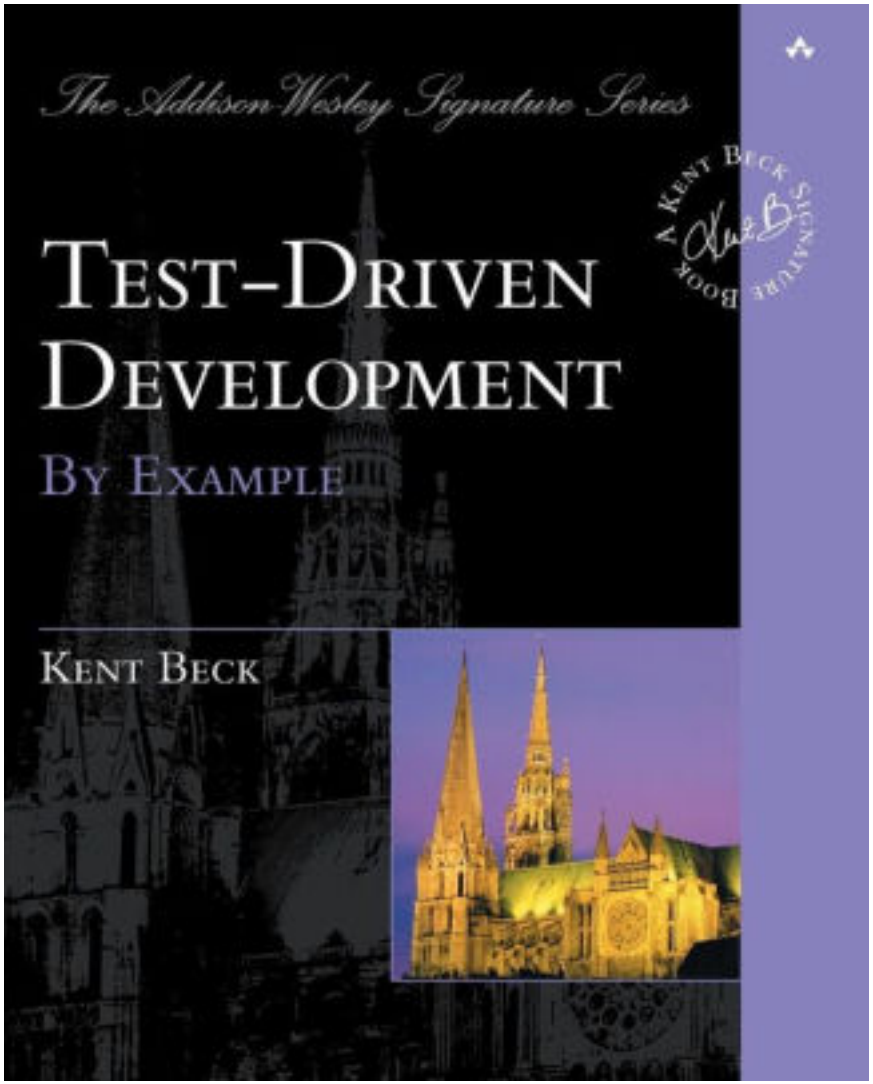
# Not a Python Reference

- Not focused on "features"
- Not "Advanced Python"
- Not about the library
- Or third-party modules
- Or project organization



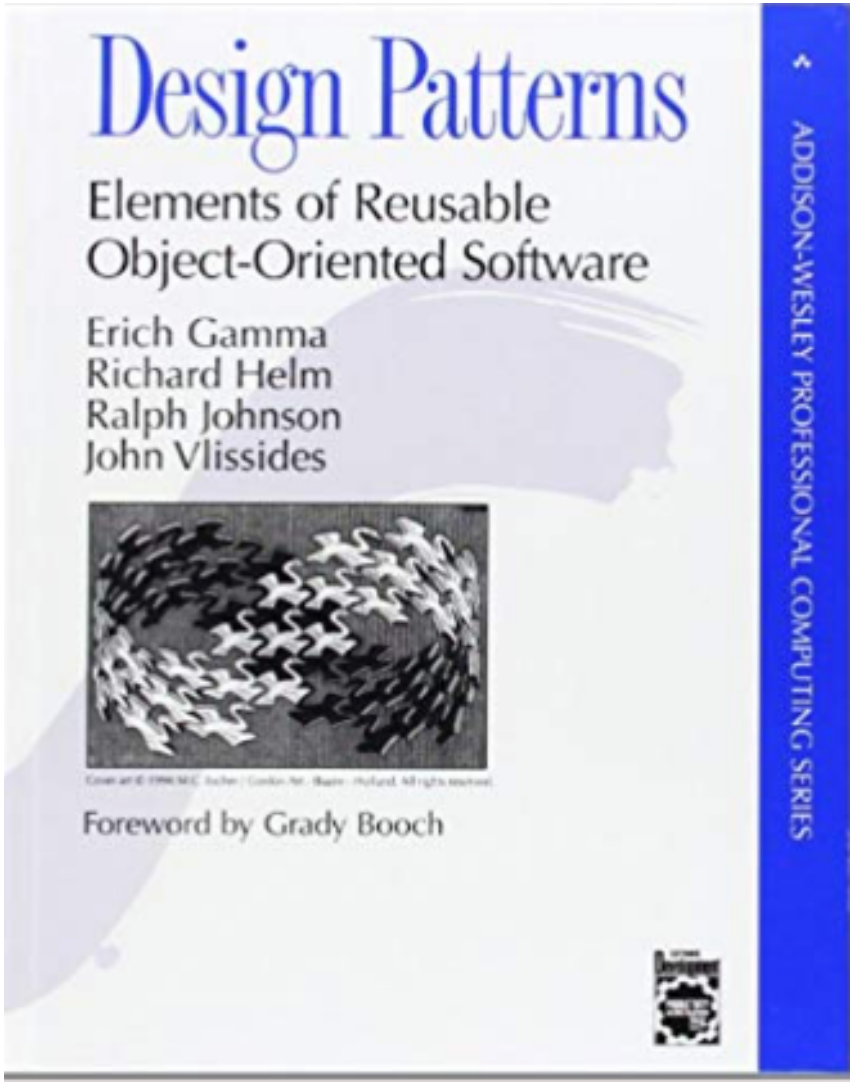
# Not Software Process

- Not about TDD
- Or Agile
- Or Waterfall
- Or project management...



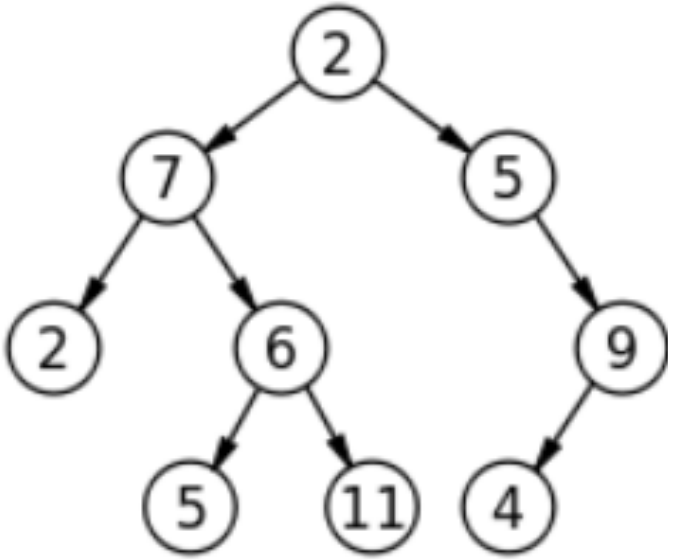
# Not Recipes

- This is a fine book...
- But most Python devs don't think like this
- Very C++/Java focused



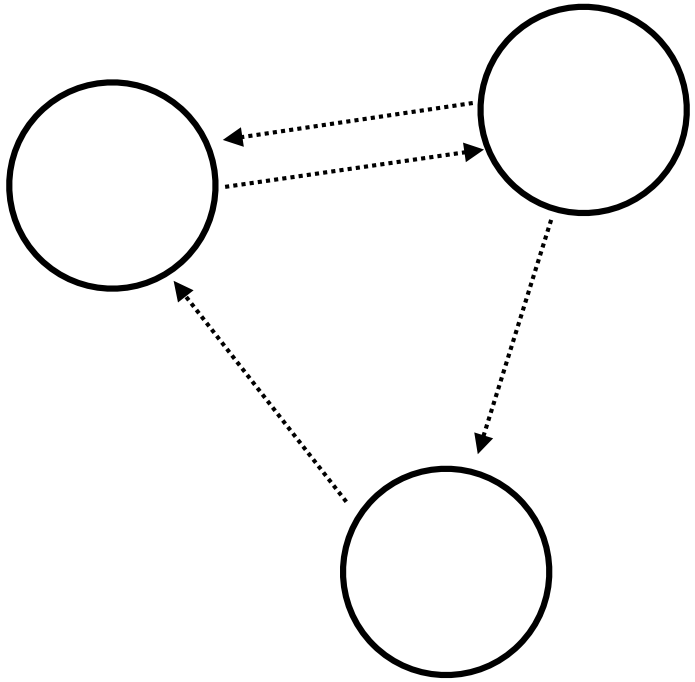
# Not Algorithms

- When was the last time you had to implement a classic "algorithm"?
- No, you'd use a library



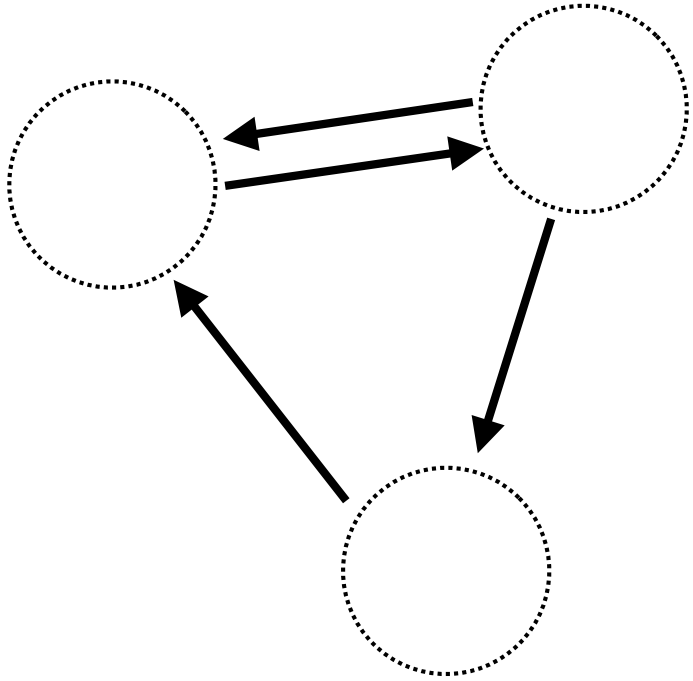
# Not About Making Things

- Modern programming is often focused on using existing things
- Libraries, components, packages, etc.
- Don't reinvent the wheel



# It's About the Arrows!

- Modern programming is more about gluing components together
- Components interact with other components
- It's about relationships.



# Overview

- This is a course about relations
- Breaking problems down into parts
- Thinking about how the parts fit together
- Thinking about complexity (avoiding it)

# Core Topics

- Data Abstraction
- Programming with objects
- Functional programming
- Concurrency
- Linguistic abstraction

# Format

- Entire course is organized around coding
- There are no formal lecture slides
- There are no given "solutions"
- There is often no "right" answer
- Driven by group discussion
- You will get more out of the course by participation (questions, discussion, etc.)

# Approach

- I think important ideas emerge naturally from problem solving and thinking
- Instead of presenting a topic and then working on an exercise, the course introduces a project and explores topics as they emerge during work/discussion.
- It's a bit flipped around, but the idea is to present programming topics in "context."

# Let's Begin