

Architecture in the Age of Agile

This title is grammatical nonsense because 'agile' is an adjective.

Robert Smallshire

 @robsmallshire









THE END

THE END

Architecture and Agile

Strange bedfellows or friends with benefits?

1

Sustainability and Survival

How do we keep it up for two-hundred sprints?

2

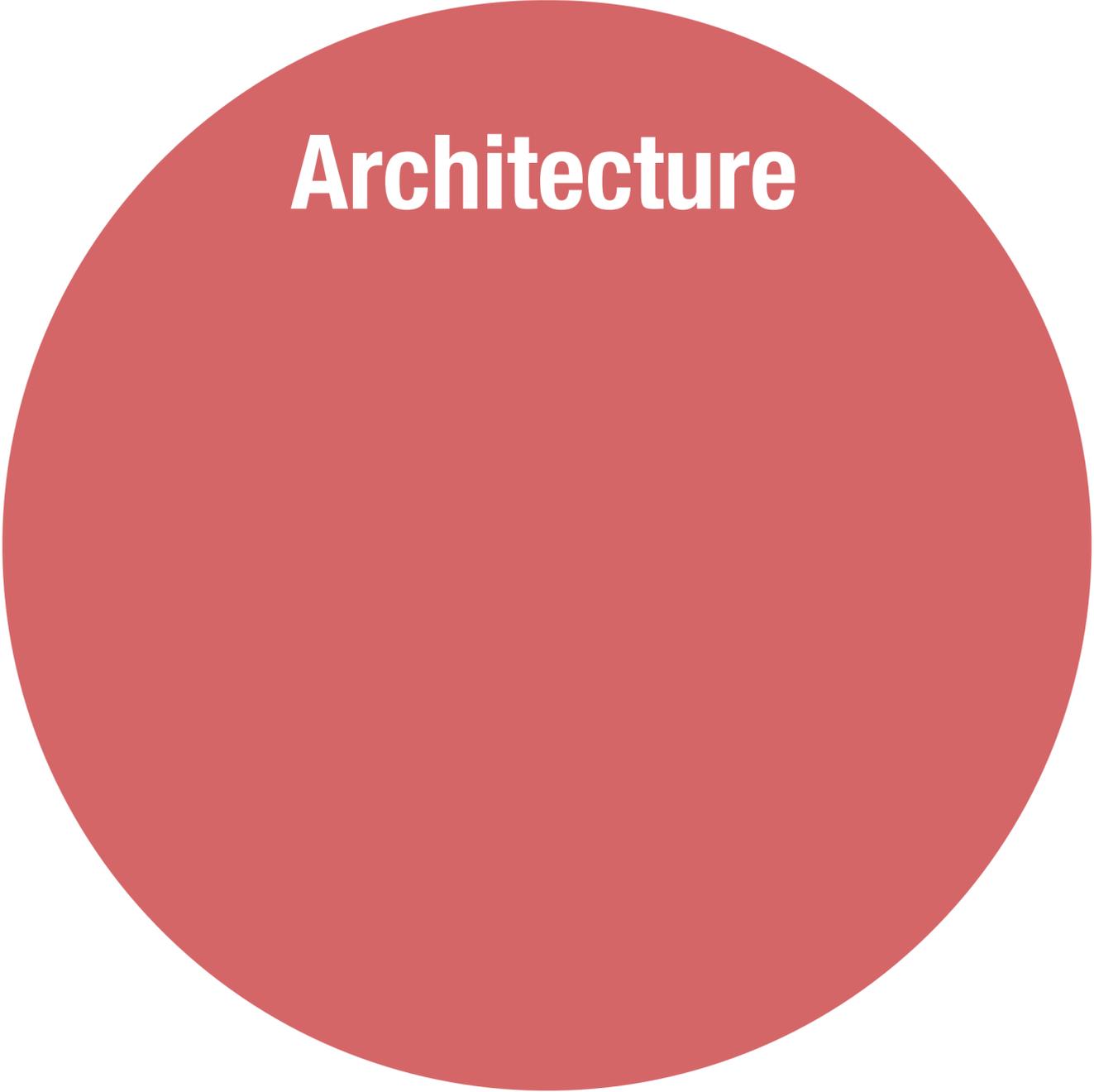
Prediction Models

Doing better than guessing with science.

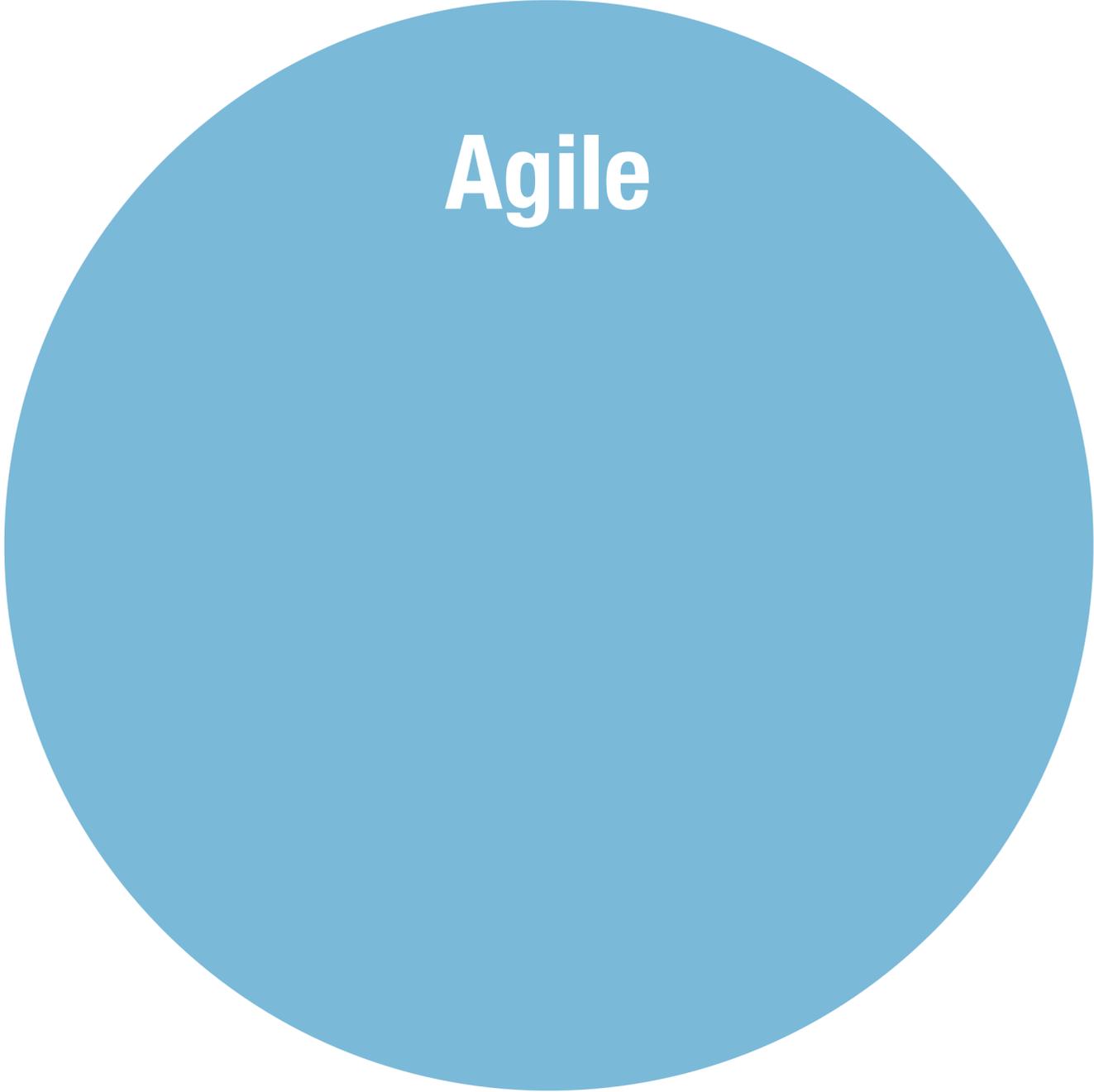
3

What do we mean?

Nouns, verbs, adjectives



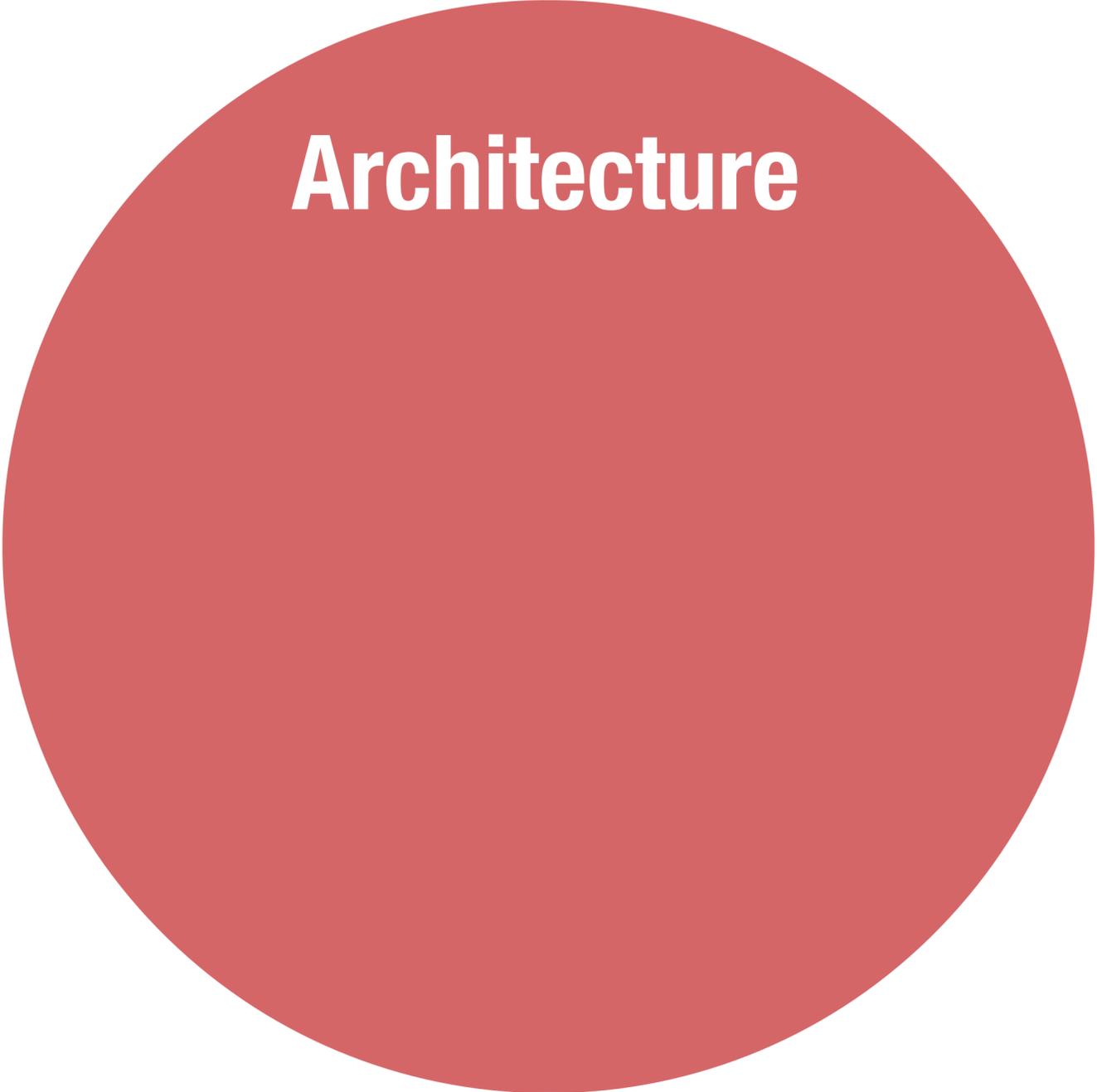
Architecture



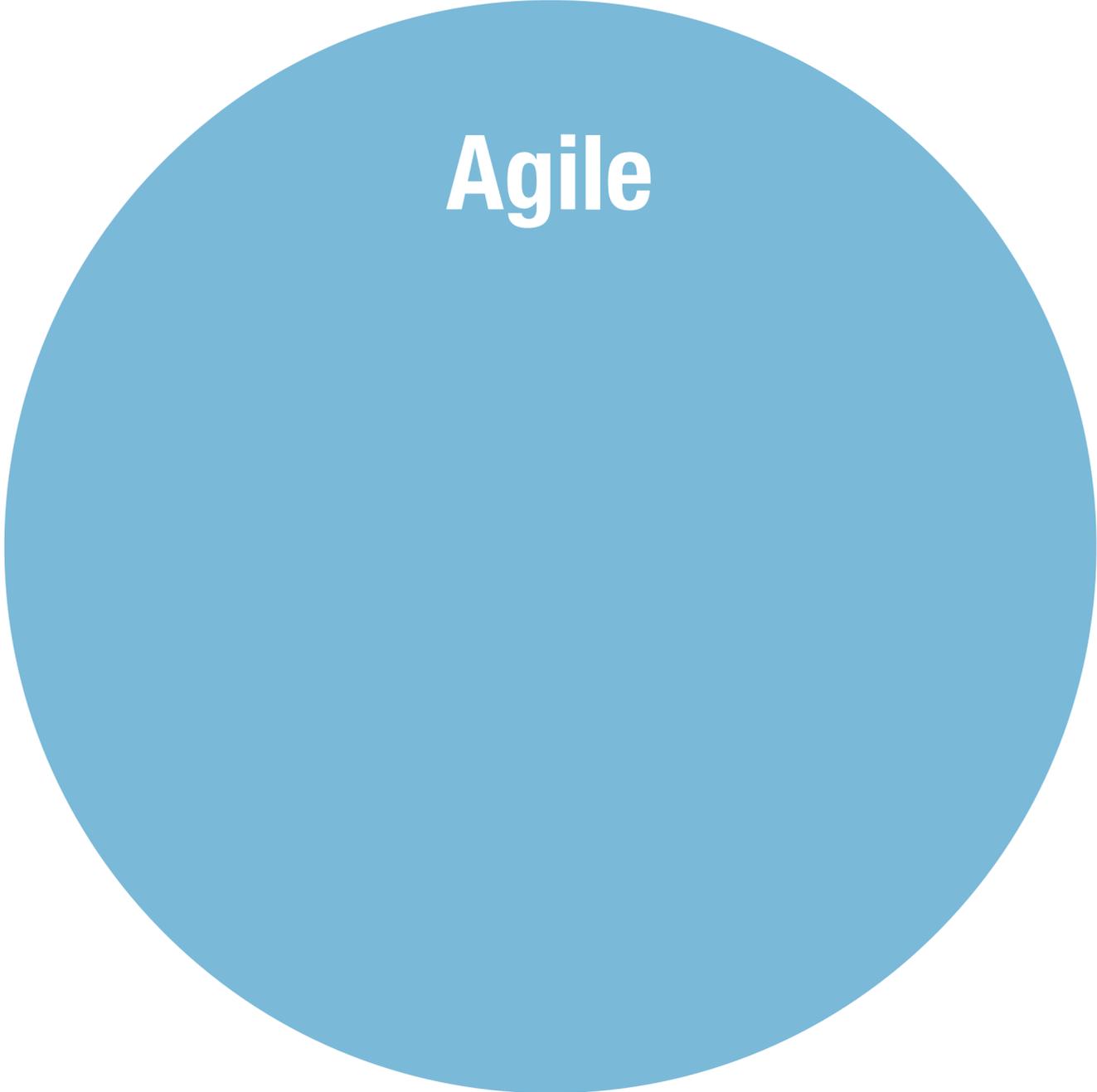
Agile

What do we mean?

Nouns, verbs, adjectives



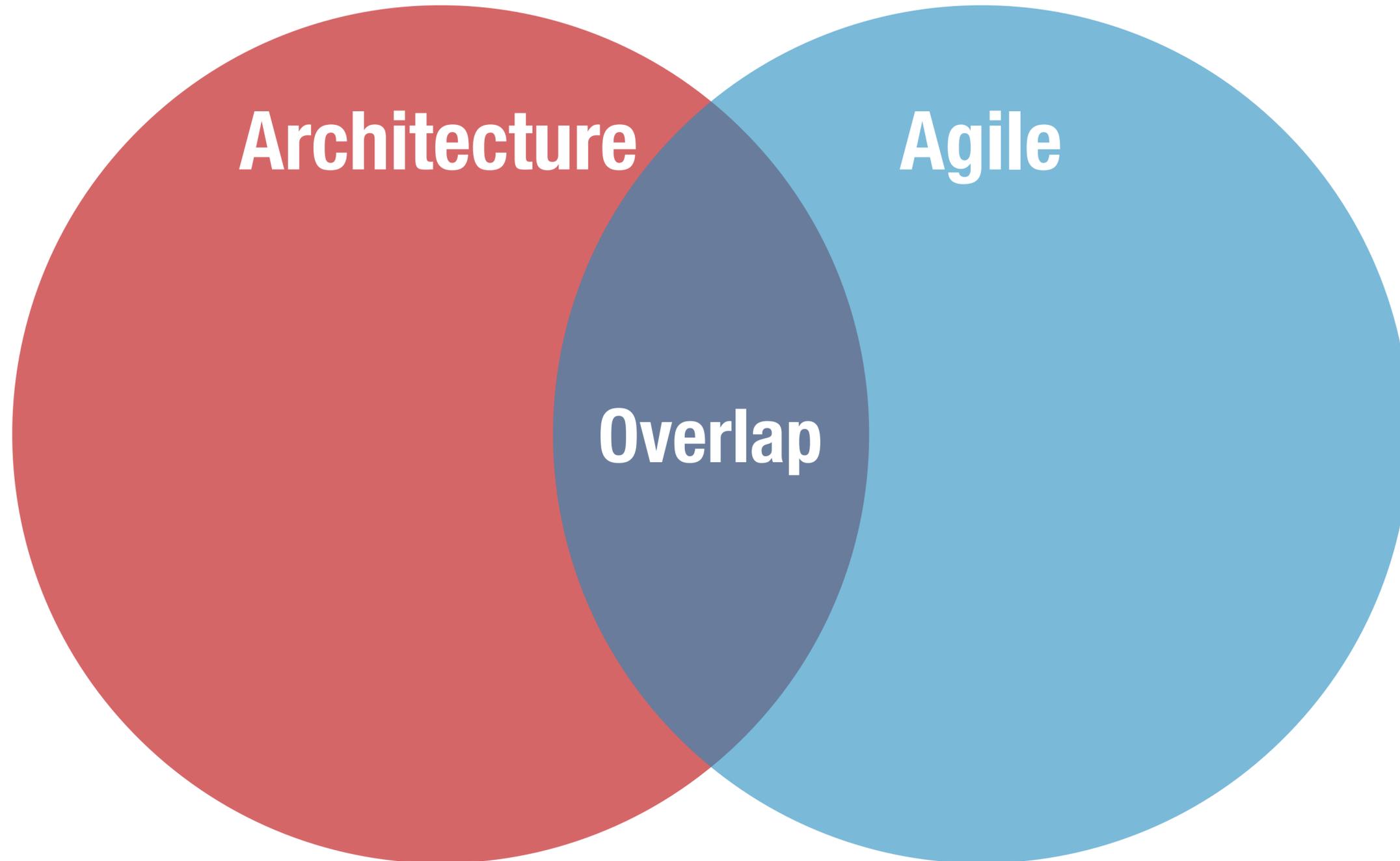
Architecture



Agile

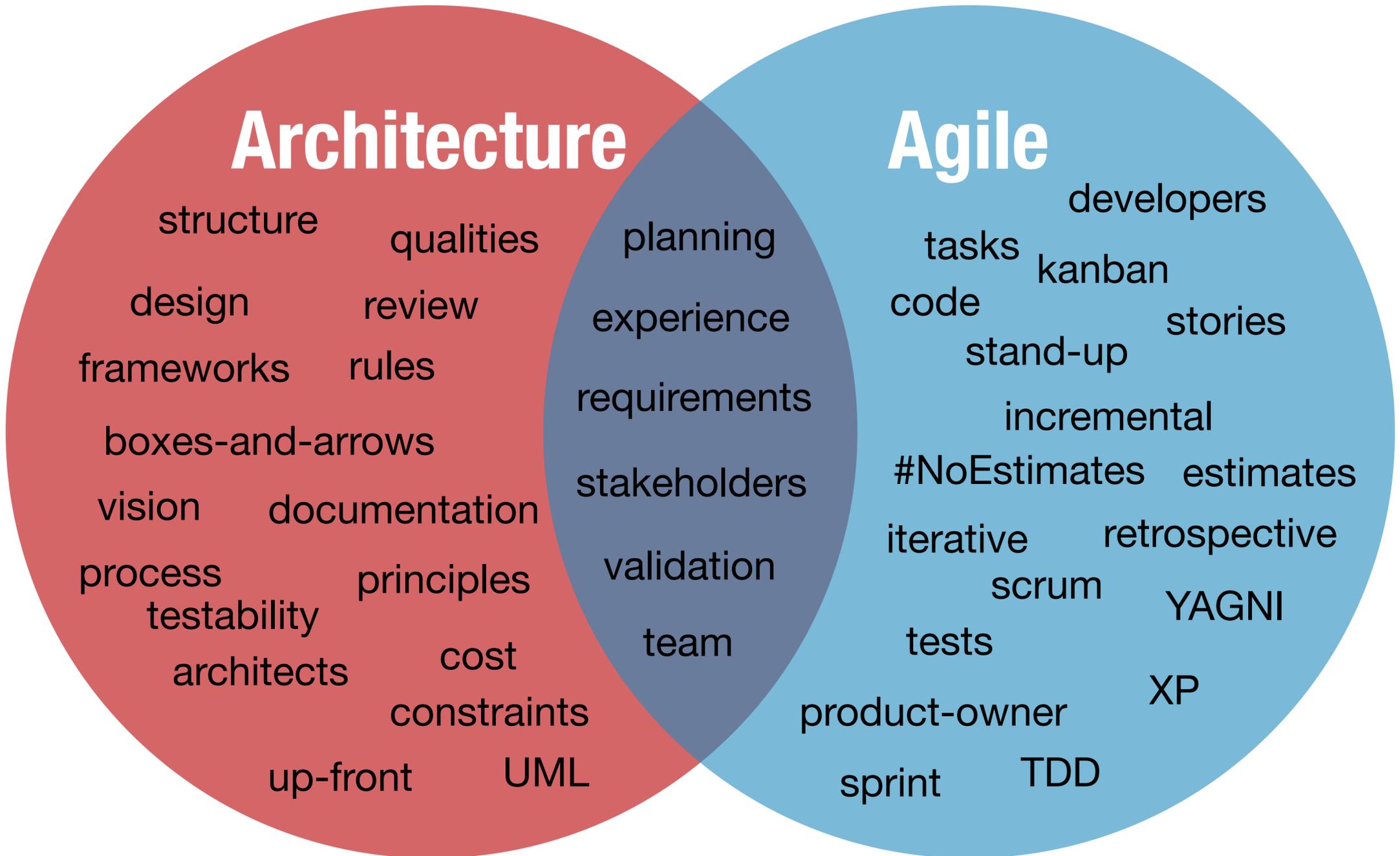
What do we mean?

Use to show two overlapping factors



What do we mean?

Nouns, verbs, adjectives



?



agile



MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS

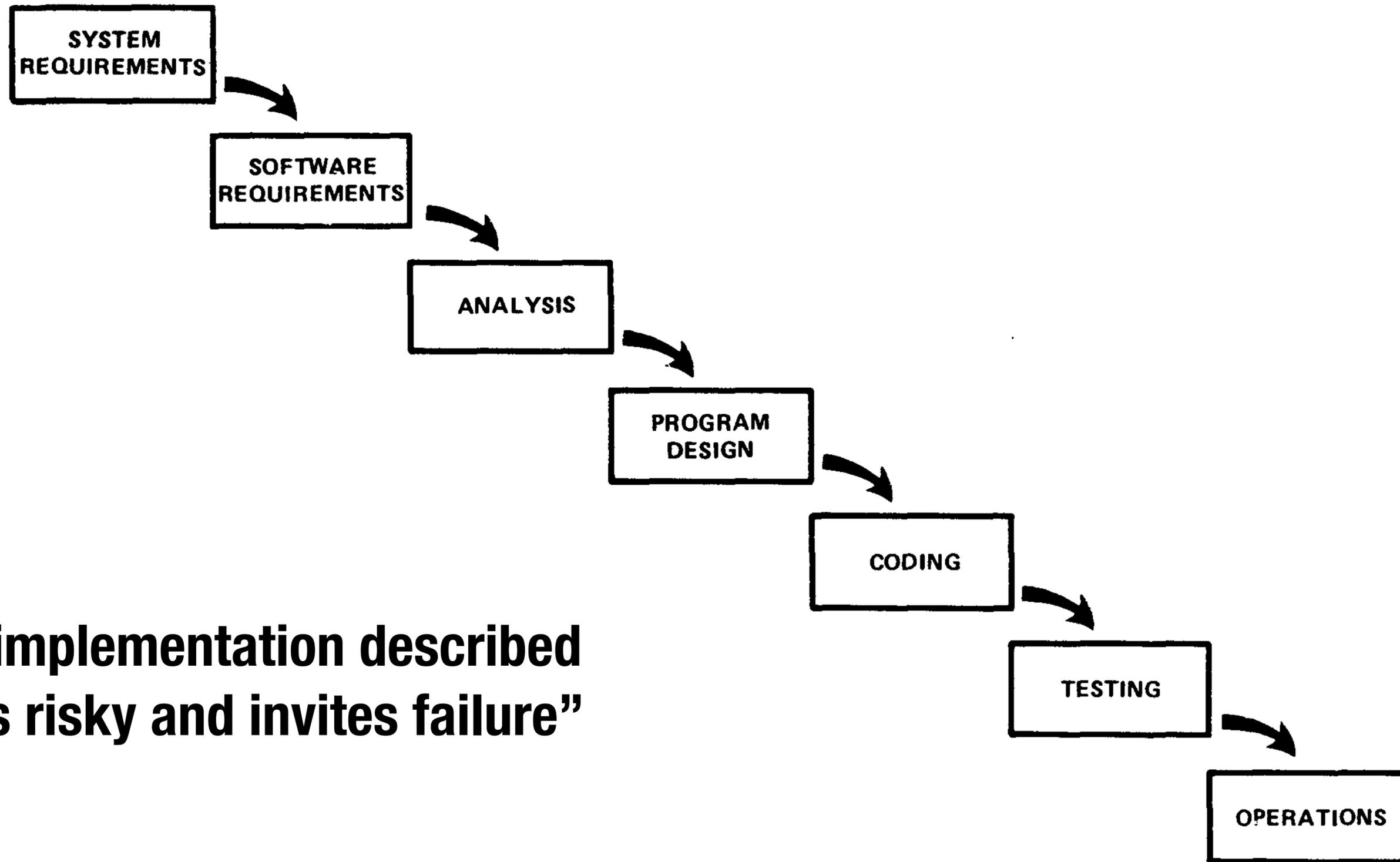
Dr. Winston W. Royce

INTRODUCTION

I am going to describe my personal views about managing large software developments. I have had various assignments during the past nine years, mostly concerned with the development of software packages for spacecraft mission planning, commanding and post-flight analysis. In these assignments I have experienced different degrees of success with respect to arriving at an operational state, on-time, and within costs. I have become prejudiced by my experiences and I am going to relate some of these prejudices in this presentation.

COMPUTER PROGRAM DEVELOPMENT FUNCTIONS

There are two essential steps common to all computer program developments, regardless of size or complexity. There is first an analysis step, followed second by a coding step as depicted in Figure 1. This sort of very simple implementation concept is in fact all that is required if the effort is sufficiently small and if the final product is to be operated by those who built it — as is typically done with computer programs for internal use. It is also the kind of development effort for which most customers are happy to pay, since both steps involve genuinely creative work which directly contributes to the usefulness of the final product. An implementation plan to manufacture larger software systems, and keyed only to these steps, however, is doomed to failure. Many additional development steps are required, none contribute as directly to the final product as analysis and coding, and all drive up the development costs. Customer personnel typically would rather not pay for them, and development personnel would rather not implement them. The prime function of management is to sell these concepts to both groups and then enforce compliance on the part of development personnel.



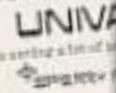
“... the implementation described above is risky and invites failure”

You installed your central computer to increase the efficiency of your business. Now it may be time to install computers to increase the efficiency of your central computer.

The UNIVAC® 9000 series is a family of small and medium-scale computers that can give you optimum use of your central computer and communications lines. You get faster throughput and a significant savings in line costs because of the flexible high speed communication capabilities of the UNIVAC 9000 series.

And, of-line, they can perform all kinds of routine on-site computing jobs such as inventory, billing, payroll, etc., so the efficiency and capacity of your total system is increased.

In addition, they can interface with other systems within the entire computer network. In short, they can do just about anything your central computer can do for your business.



Computers made your business more efficient. Here are computers to make your computers more efficient.



VERSATILE



DEPENDABLE COMPATIBLE

(MAYBE EVEN SEXY)

CALL IT WHAT YOU WANT...

We call it a PENRIL MODEM!

Penril's modems are all performers — with a family ranging from teletype (Bell 101C) modems and single card LSI 1200 BPS (Bell 202C) modems up to our adaptively equalized 4800 BPS models.

Penril
Data Communications, Inc.

5520 RANDOLPH ROAD, ROCKVILLE, MARYLAND 20852 • 301-881-8151

We'll be on display at Booth 2028 at FICC in Las Vegas.

Don't waste your life in a dead-end job!
**Become a well-paid
computer programmer—
this free McGraw-Hill
booklet tells you how.**

Improved
training under
GI Bill.



How to prepare for a bright new future
in computer programming

Now you can train at home in your spare time, for a career in this new, exciting field. More than 50,000 programmers are needed now; many more will be needed within the next few years. No college or technical background required. All you need is a logical mind and proper training. People from virtually every walk of life have found success in computer programming. This new course has been developed by experts. CREI/McGraw-Hill has trained thousands of men and women for better jobs through home study and has an unmatched reputation in educational circles. No stamp needed. This fold-over coupon forms a postpaid envelope. Cut along dotted line. Fold, seal, tape or staple and mail.



Home Study Division, McGraw-Hill Book Company
Dept. P1206A
3224 Sixteenth St., N.W., Washington, D.C. 20010

Please mail me FREE booklet with complete information on how I can learn computer programming at home. I have a high school diploma or the equivalent.

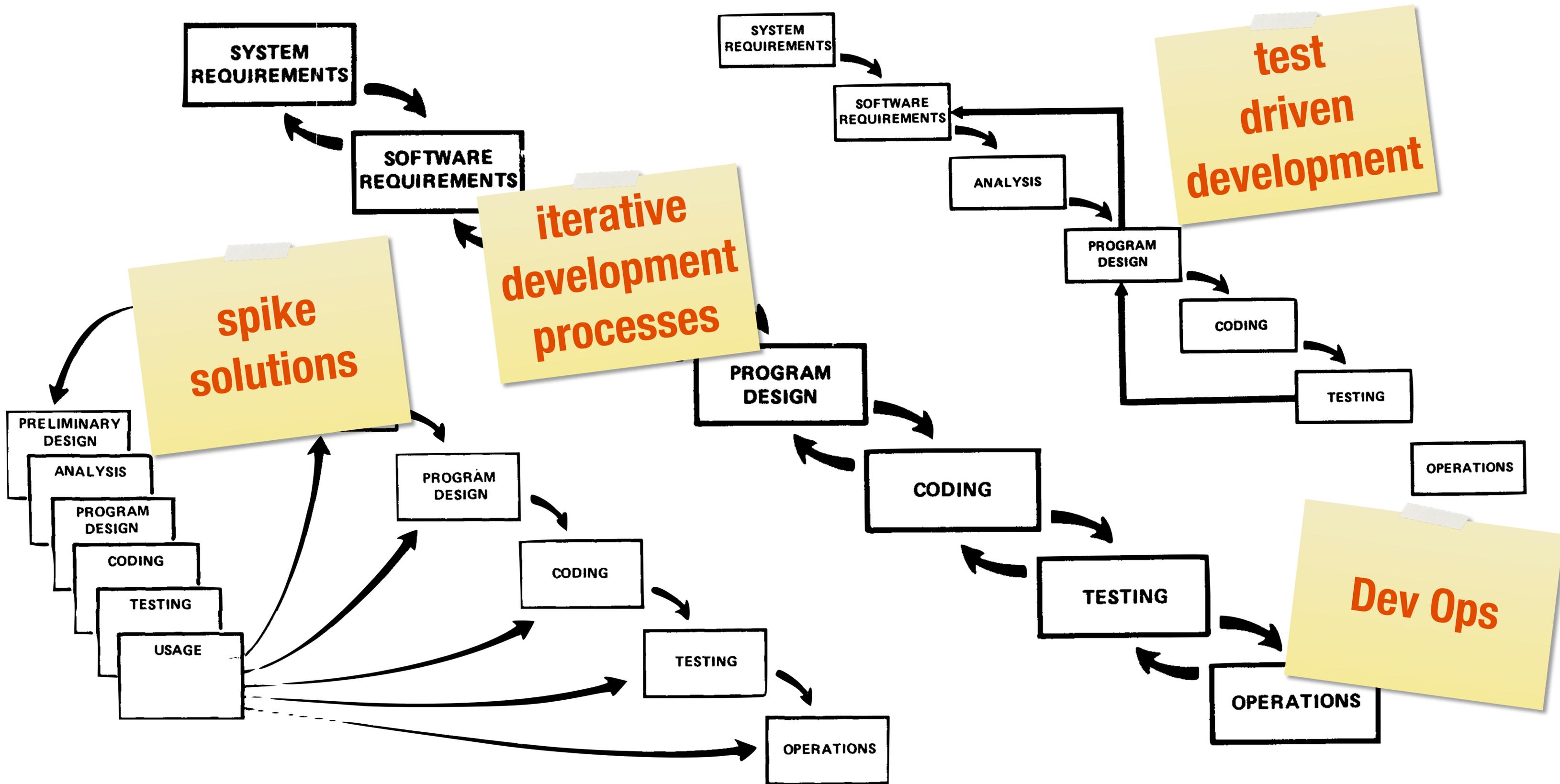
Name _____
Age _____
Address _____
City _____
State _____
Zip _____

FIRST CLASS
Permit No. 286-R
Washington, D.C.

BUSINESS REPLY MAIL
No Postage Stamp Necessary if Mailed in The United States

Postage will be paid by
**CREI,
Home Study Division
McGraw-Hill Book Company**
3224 Sixteenth Street, N.W.
Washington, D.C. 20010





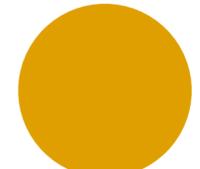
CODING

architecture

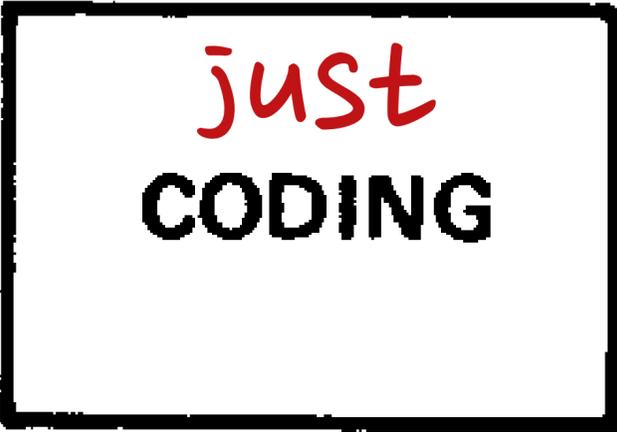


over-designed,
intricate

fragmented,
short-lived



malleable,
extensible designs



just
CODING

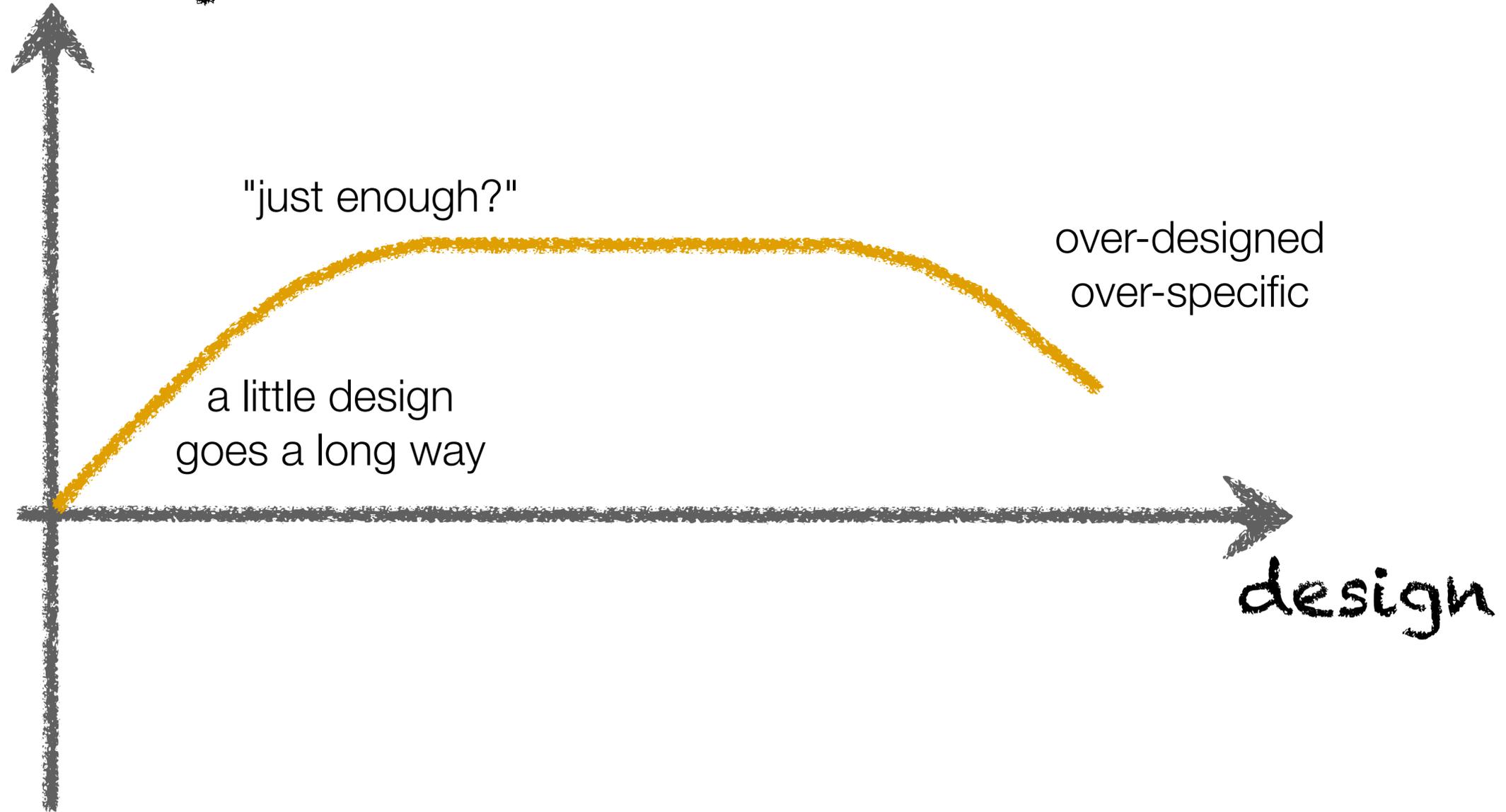


agile

Design and sustainability

What is the effect of *doing* design on long term viability?

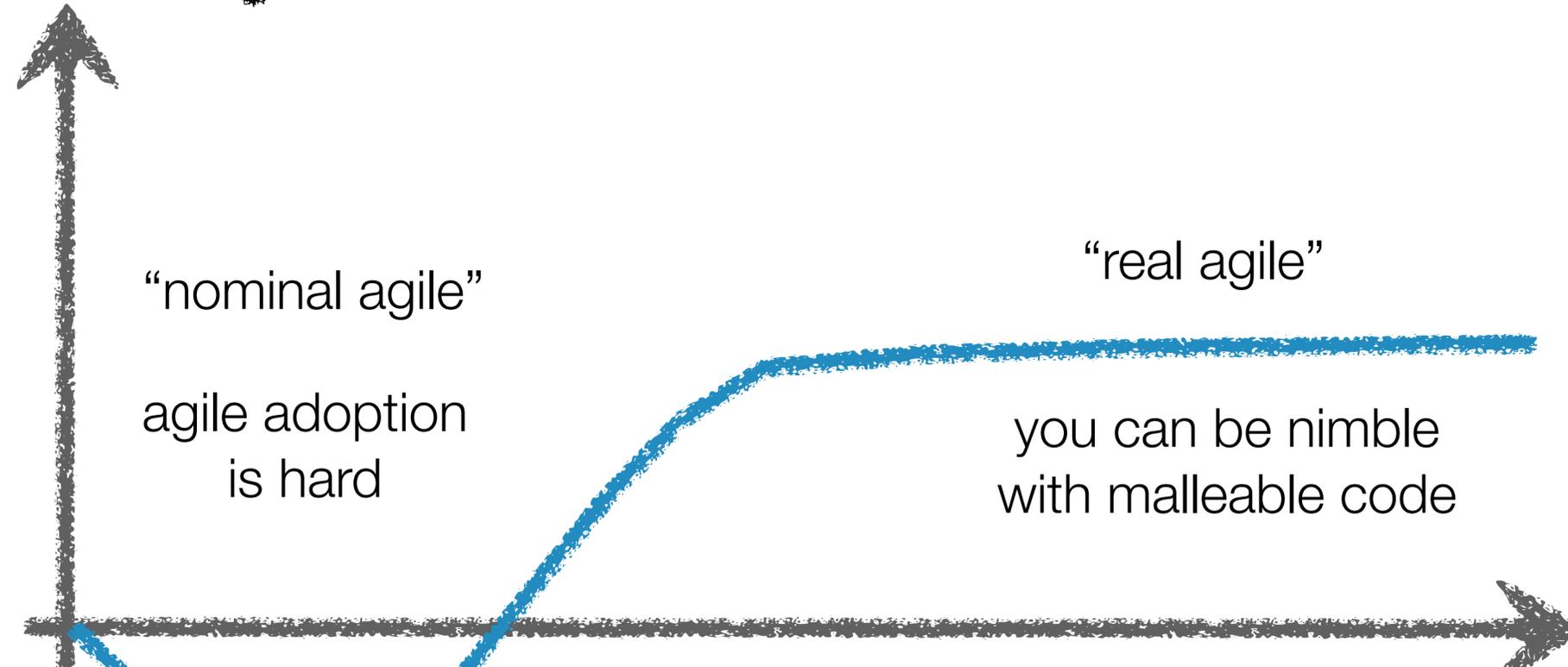
sustainability



Agile and sustainability

What is the effect of *doing* agile on long term viability?

sustainability



“nominal agile”

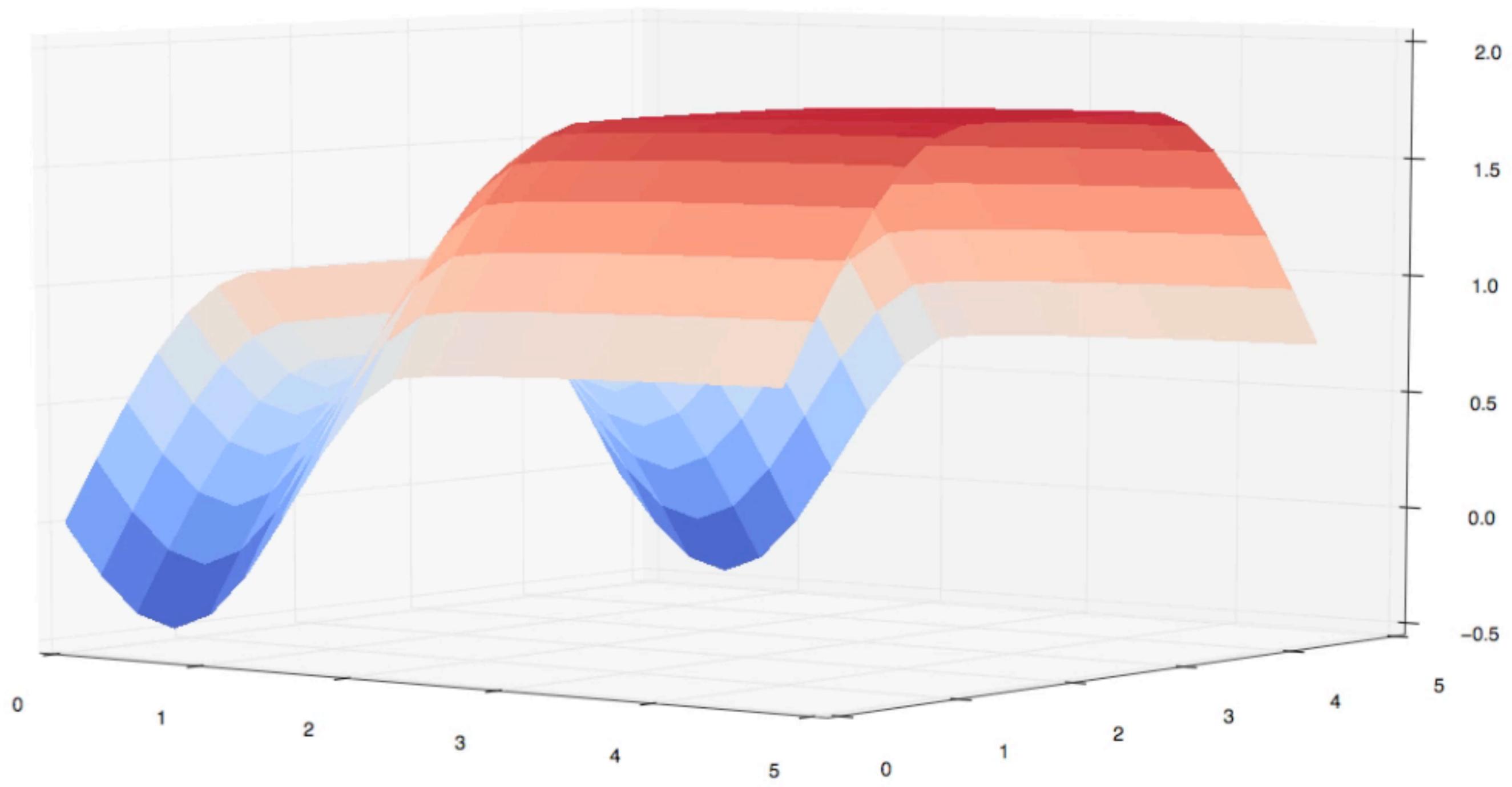
agile adoption
is hard

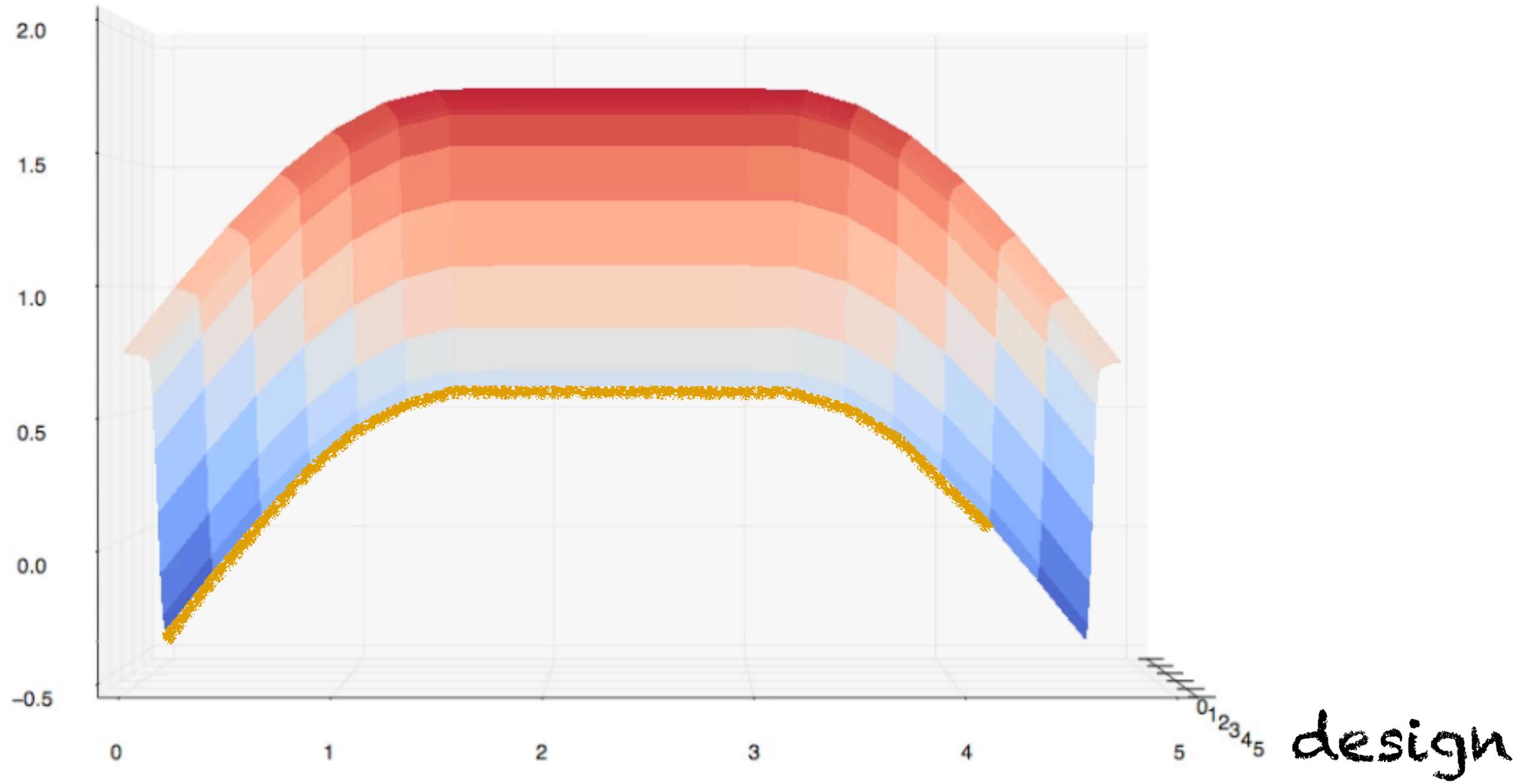
“real agile”

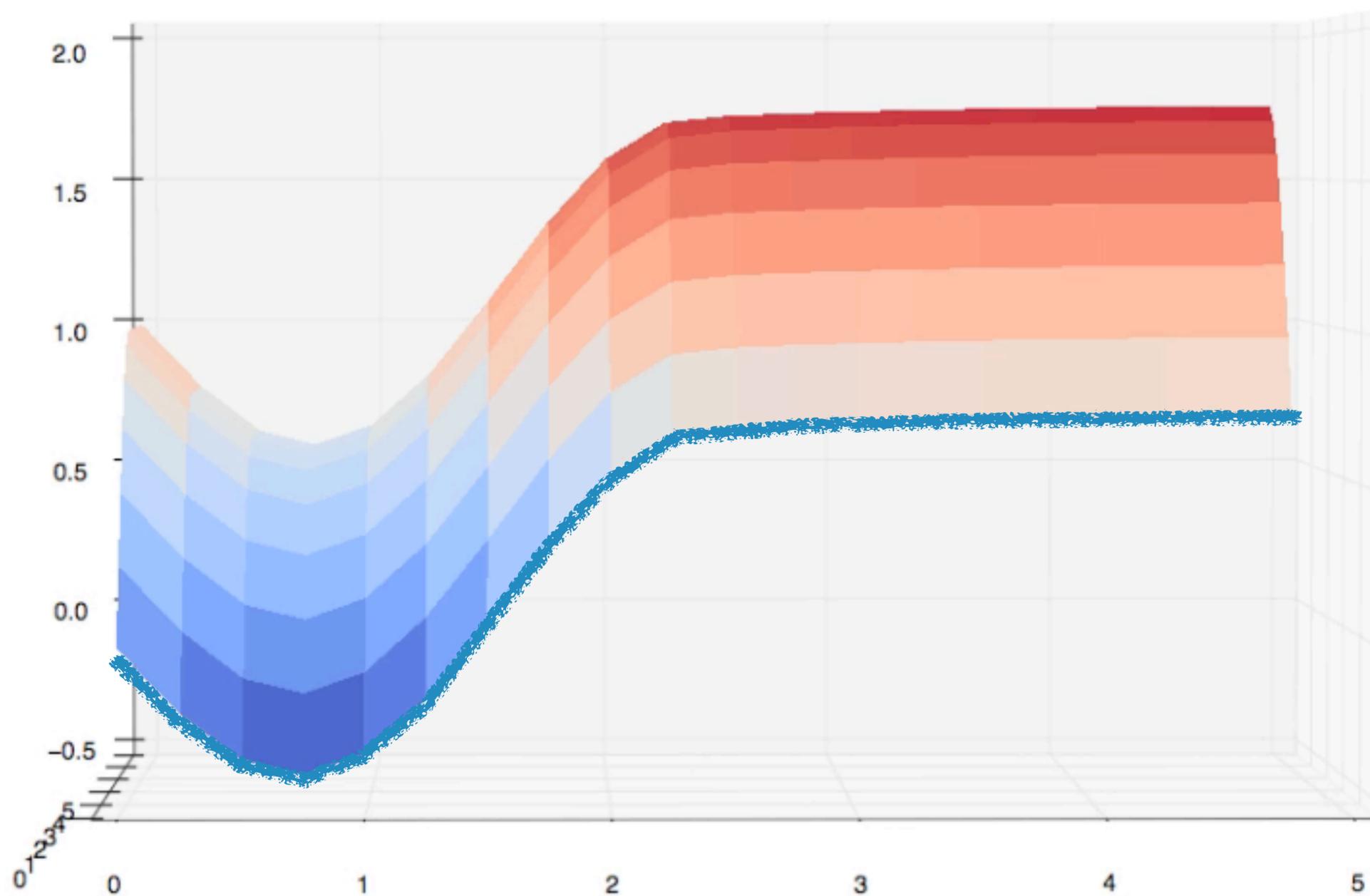
you can be nimble
with malleable code

damage
caused by
poor agile

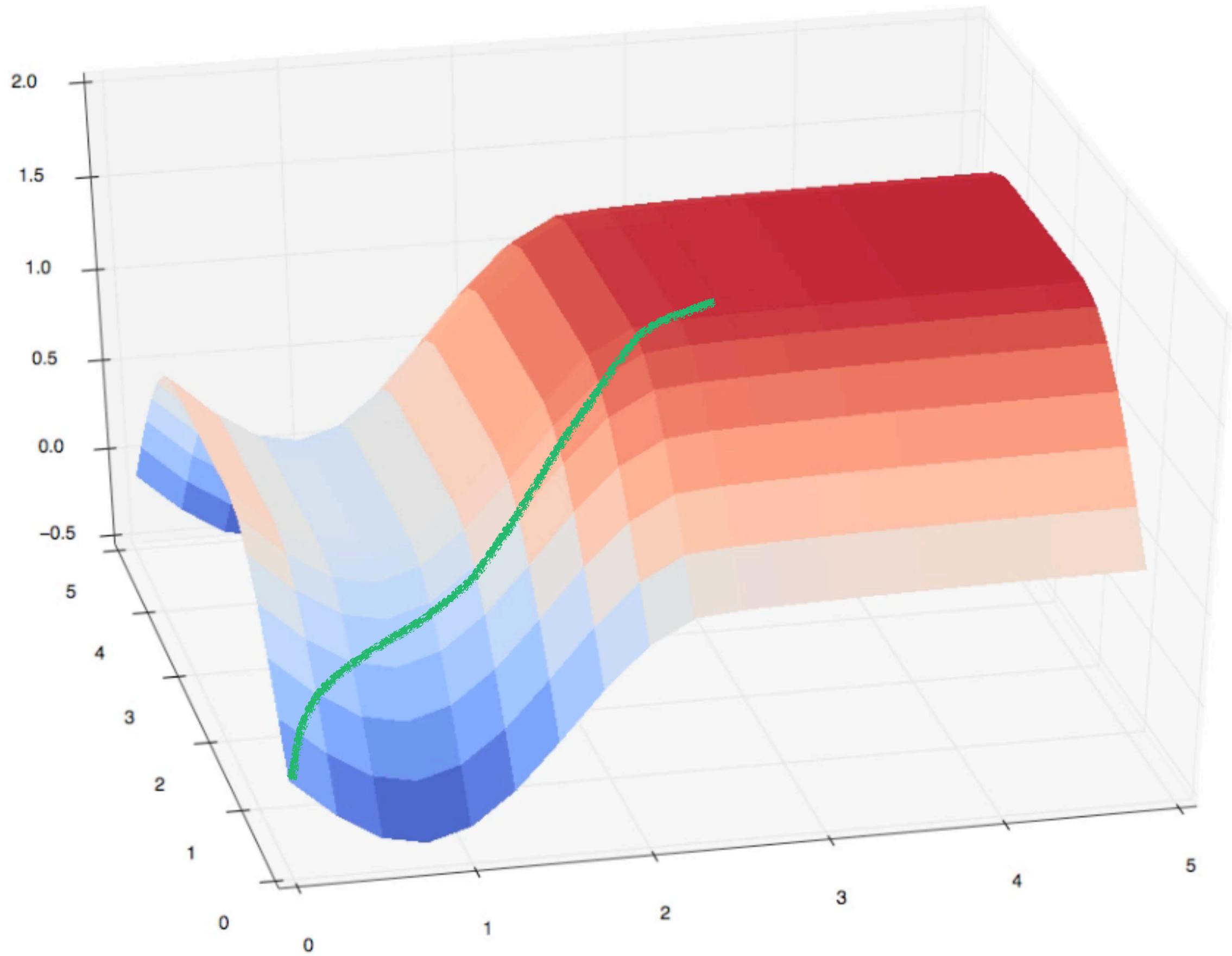
agileness



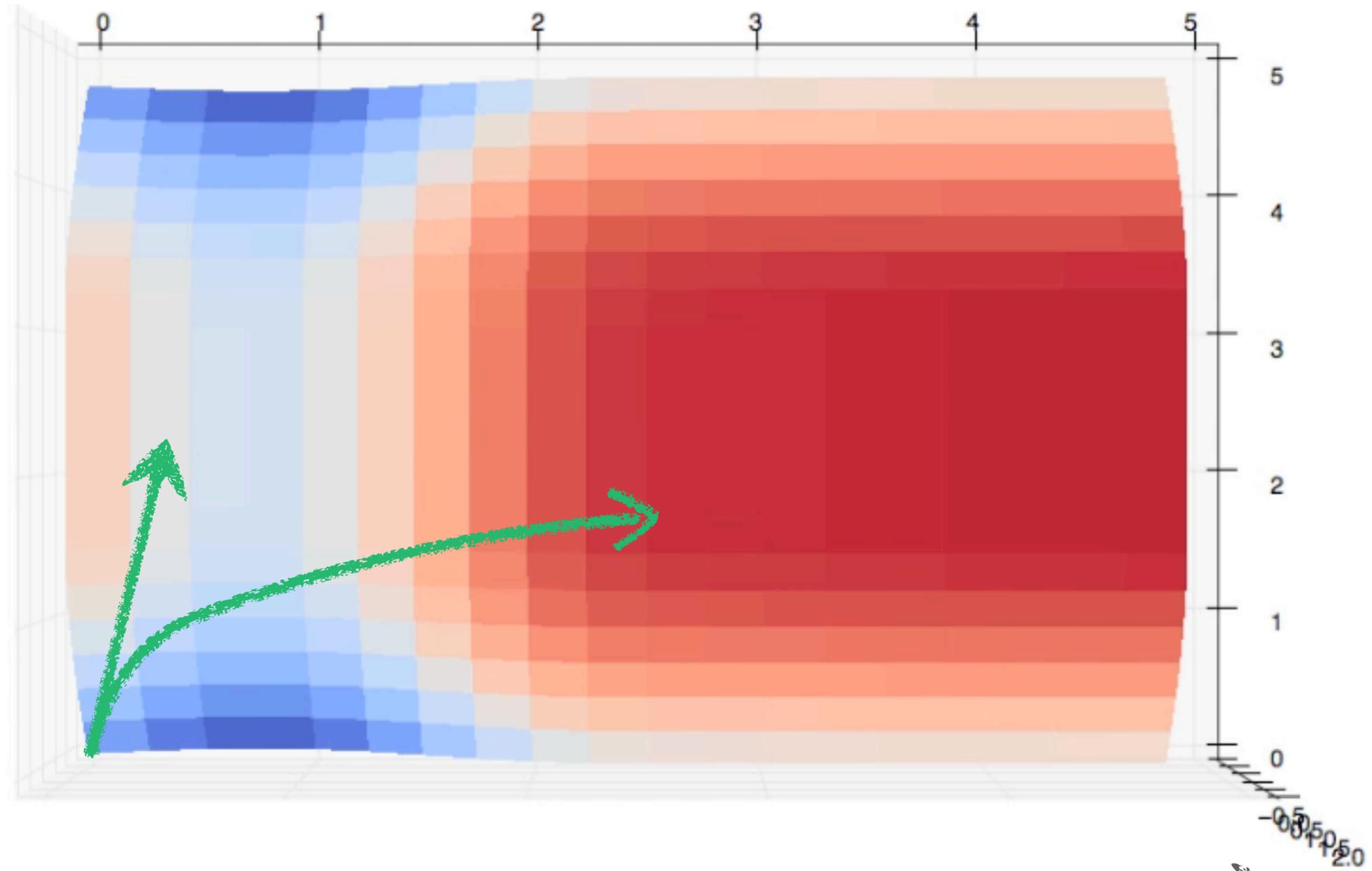




agileness



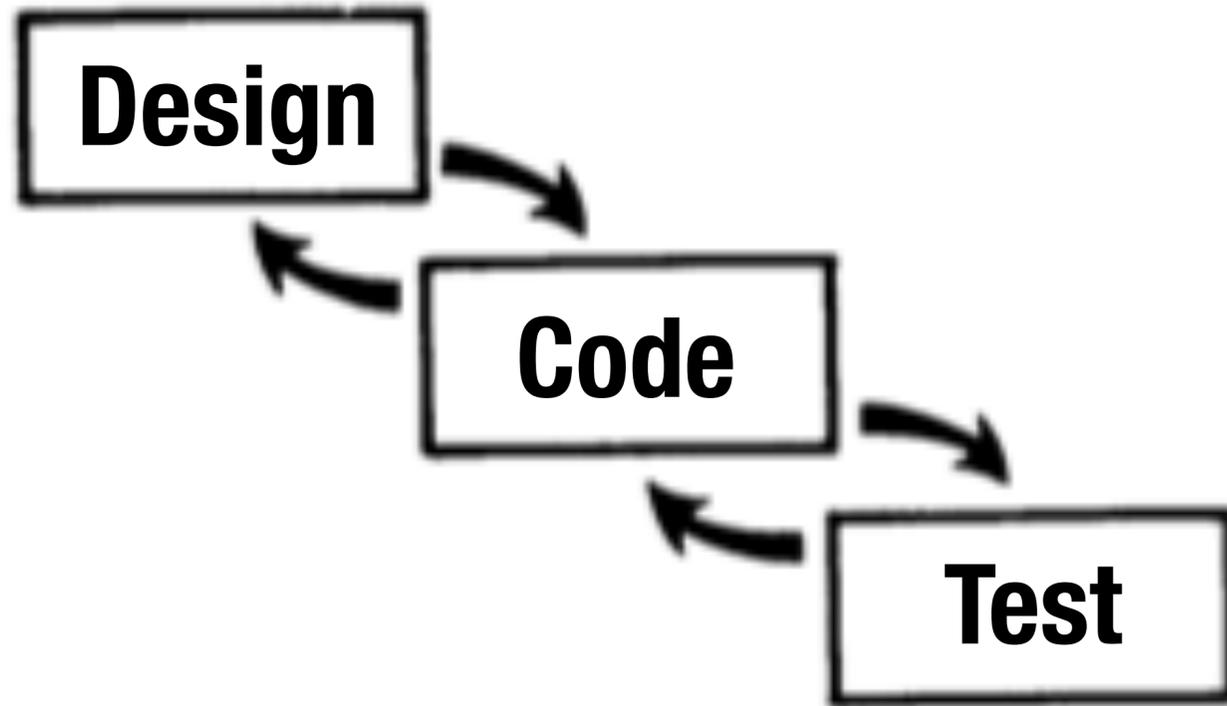
design



agileness

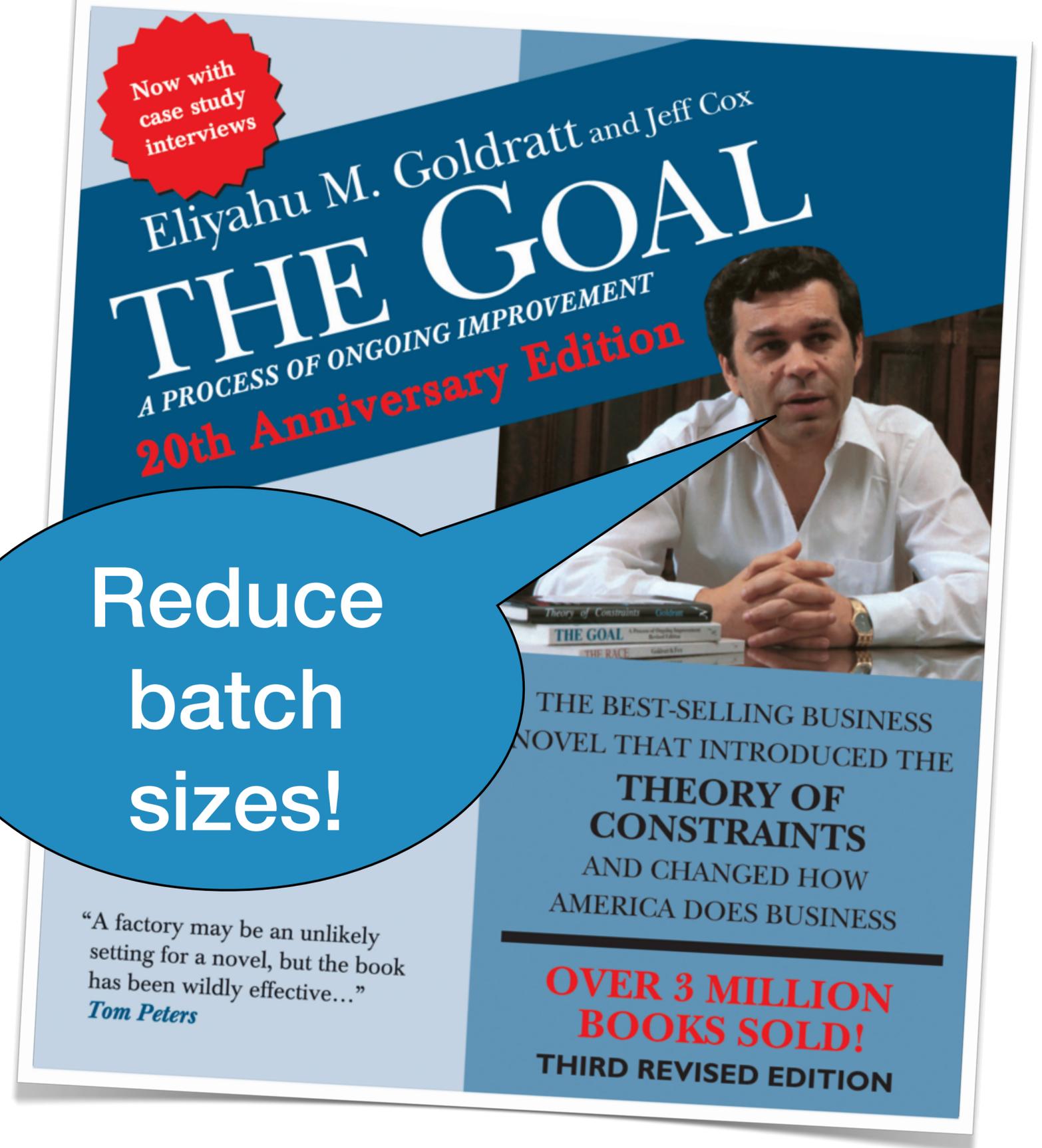
Don't throw the
baby out with
the bathwater!

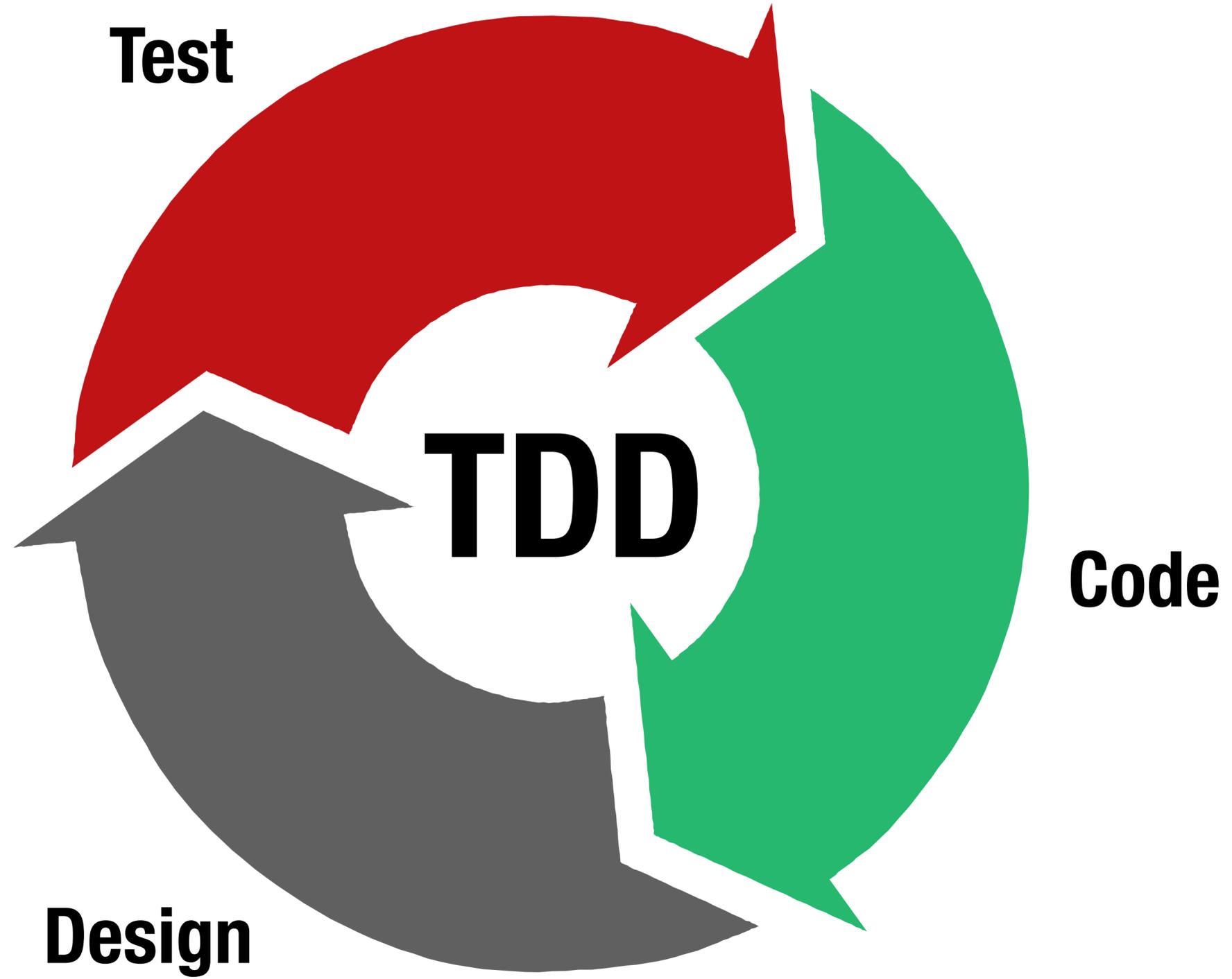




1 year
1 fortnight
1 minute

Reduce batch sizes!

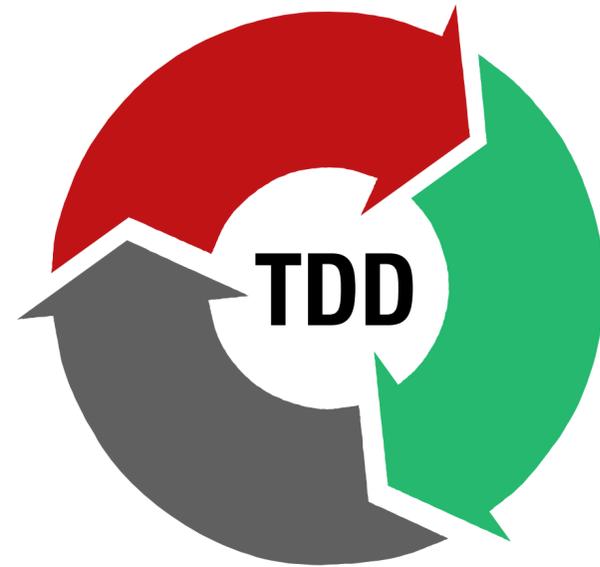




Tactical and Strategic Design

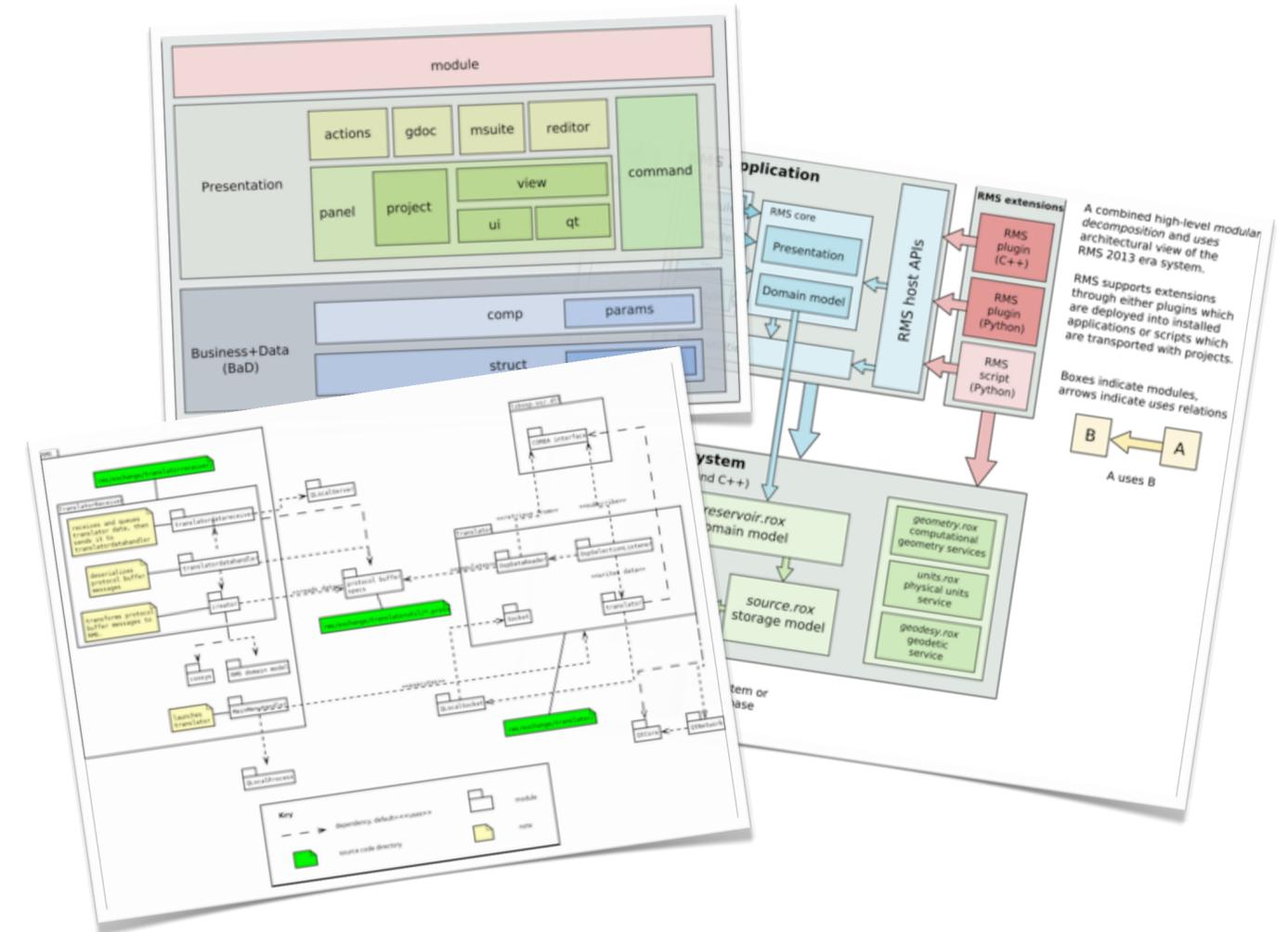
Responding to events and shaping the future

Agile Development Practices (Tactics)



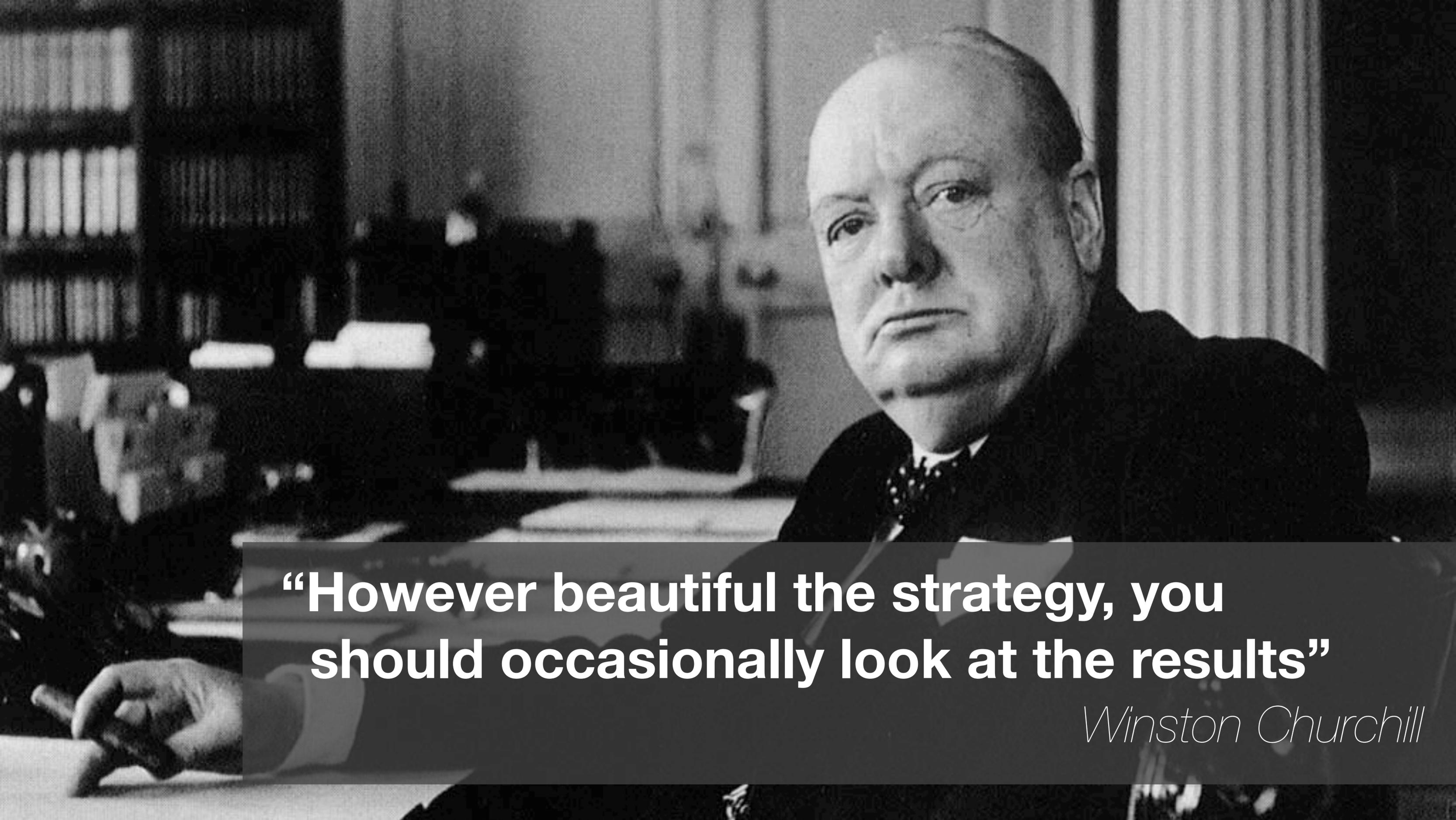
Tactical response to actual conditions

Architecture (Strategic Design)



High level plan to achieve one or more goals under conditions of uncertainty.



A black and white photograph of Winston Churchill. He is seated at a desk, looking off to the side with a thoughtful expression. The background shows a bookshelf filled with books and a window with curtains. A semi-transparent dark grey box is overlaid on the lower half of the image, containing a quote in white text.

“However beautiful the strategy, you should occasionally look at the results”

Winston Churchill

Architecture and Agile

Strange bedfellows or friends with benefits?

1

Sustainability and Survival

How do we keep it up for two-hundred sprints?

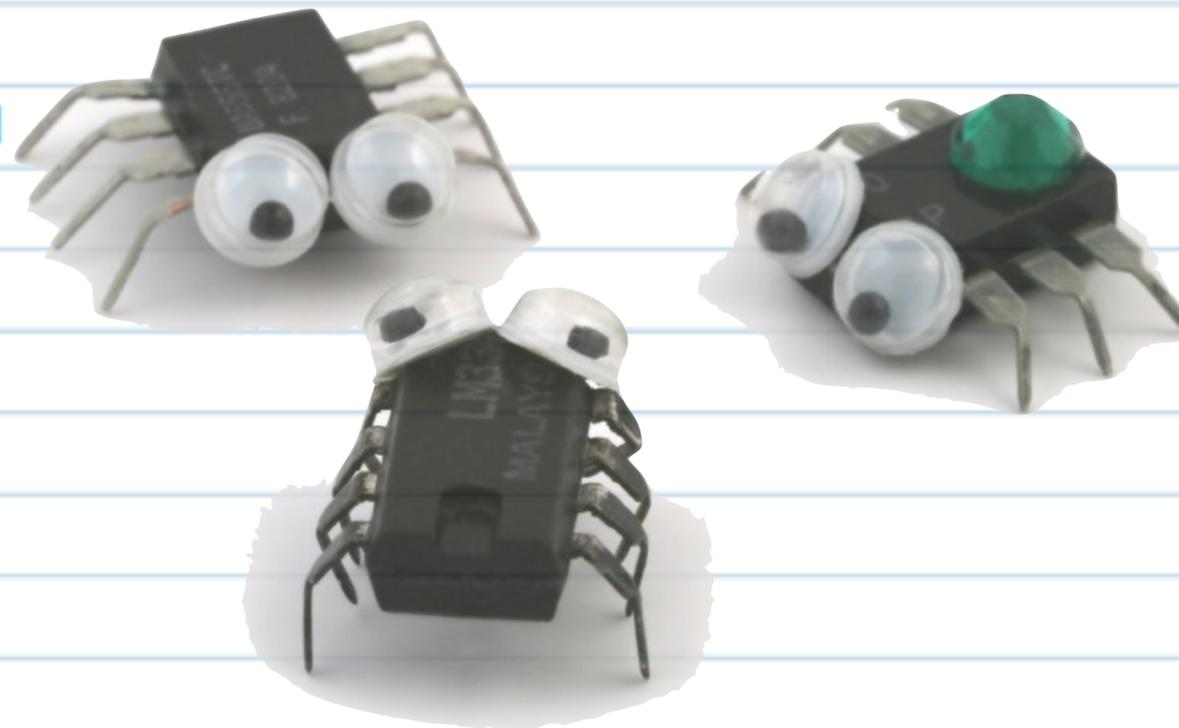
2

Prediction Models

Doing better than guessing with science.

3

Multi repository support - Master-Slave infrastructure	In Progress
Improve usability of the Create New Project wizard for demoseverers	Implemented
Provide support and doc for using custom images in enums	In Progress
It should not be possible to create "wrong" relationships	In Progress
Find solution for lack of disk space: Rotate or delete old logs	In Progress
Improve usability for new users by showing additional information or help in tooltips everywhere	Done
Multi repository support - Unified login and Slave switching	Accepted
Training for support : Build Management	Accepted
Native Linux packaging (rpm or deb packages)	Open
I need to insert a table in the WI description (HTML formatting)	In Progress
Unacceptable performance of some wiki usecases	Accepted
Linked Work Items should be sorted also by creation time on WI form	Done
LDAP : support groups (object groupOfNames)	Accepted
The "duplicate" functionality needs to be reviewed and fixed	Accepted
Rework the topic concept for Modules and Livedocs	Open
I want to have standard fields to be mandatory (required)	Open
Automated generation of install guides	Open
Document the Support process	Accepted
HTTPS access - improve docs and examples	Accepted
Automated tests for detecting UI memory leaks	Accepted
Define and setup infrastructure for load/stress tests	In Progress
Simplify and automate the installation and upgrade process and its management	Open
Unify log files location	Implemented



Required

Actual

Agile Process

delivers

Features



facilitates *constrains 'how'*

Constraints



deliberate

design for

Qualities



Defects

×1000



cross-cutting concerns



The System

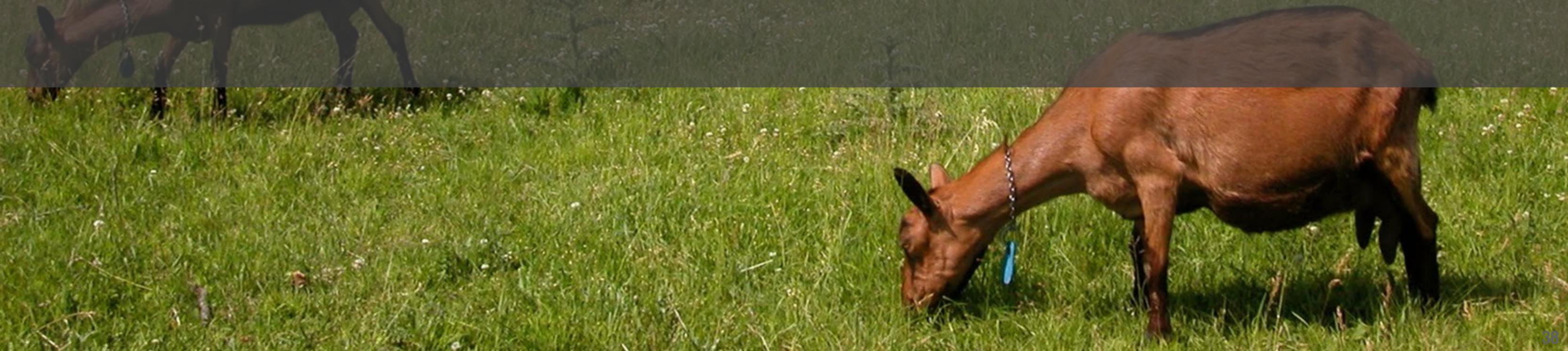
emergent qualities
Usability
Maintainability
Scalability





Sustainable Development

Provision meets ongoing needs whilst preserving the supporting environment.



As an “architect”...

**You look after the quality attributes.
The features will look after themselves.**

Architecture and Agile

Strange bedfellows or friends with benefits?

1

Sustainability and Survival

How do we keep it up for two-hundred sprints?

2

Prediction Models

Doing better than guessing with science.

3

Experimental Science

Randomised controlled trials

- ▶ **Developers don't like to be watched**
- ▶ **Eliminating extraneous factors**
- ▶ **Toy problems aren't realistic**
- ▶ **No two projects are the same**
- ▶ **Can't do double-blind**
- ▶ **Students have little experience**
- ▶ **Time and money**





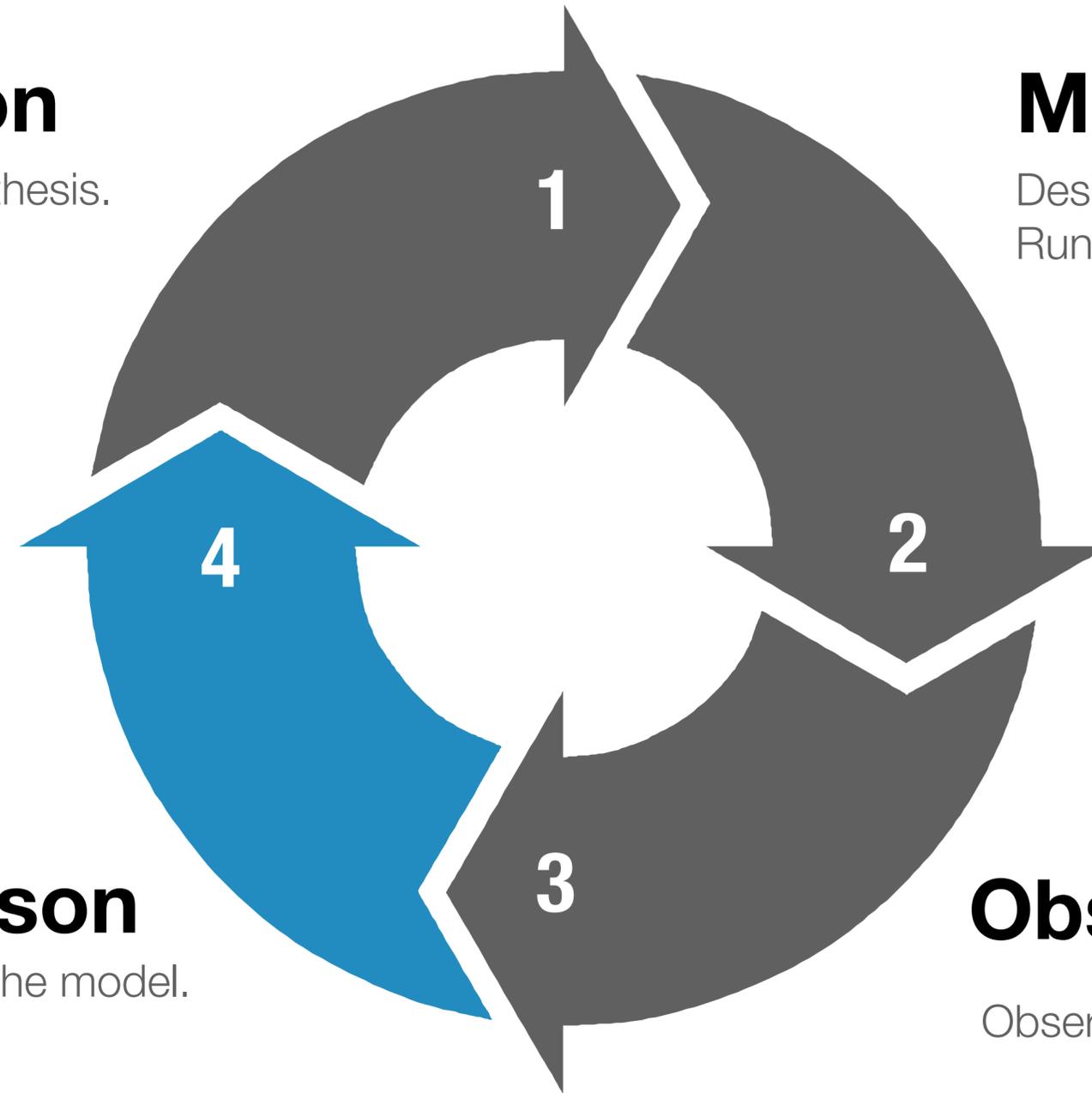
How can we know?

Prediction

Formulate a hypothesis.

Modelling

Design a conceptual model.
Run simulations.



Comparison

Validate or refute the model.

Observation

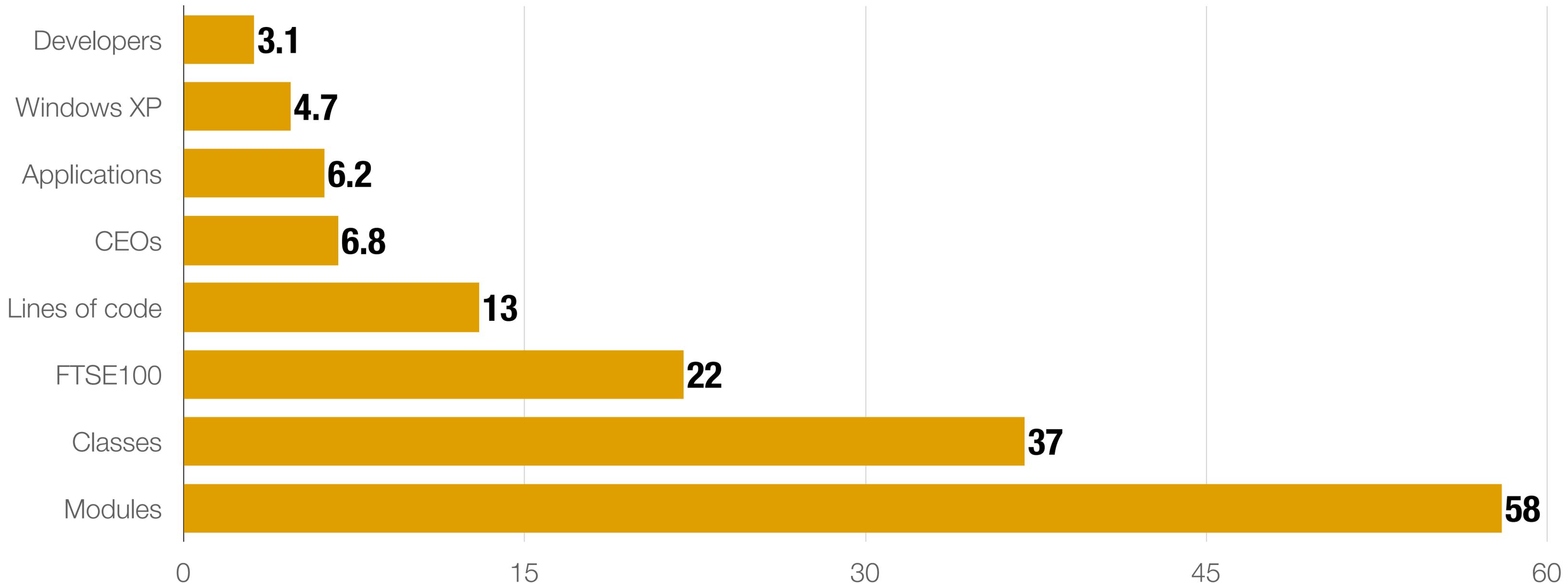
Observe and record reality.

Lifetimes in the software industry

Systems and their architectures are long lived

Half-lives of software related entities

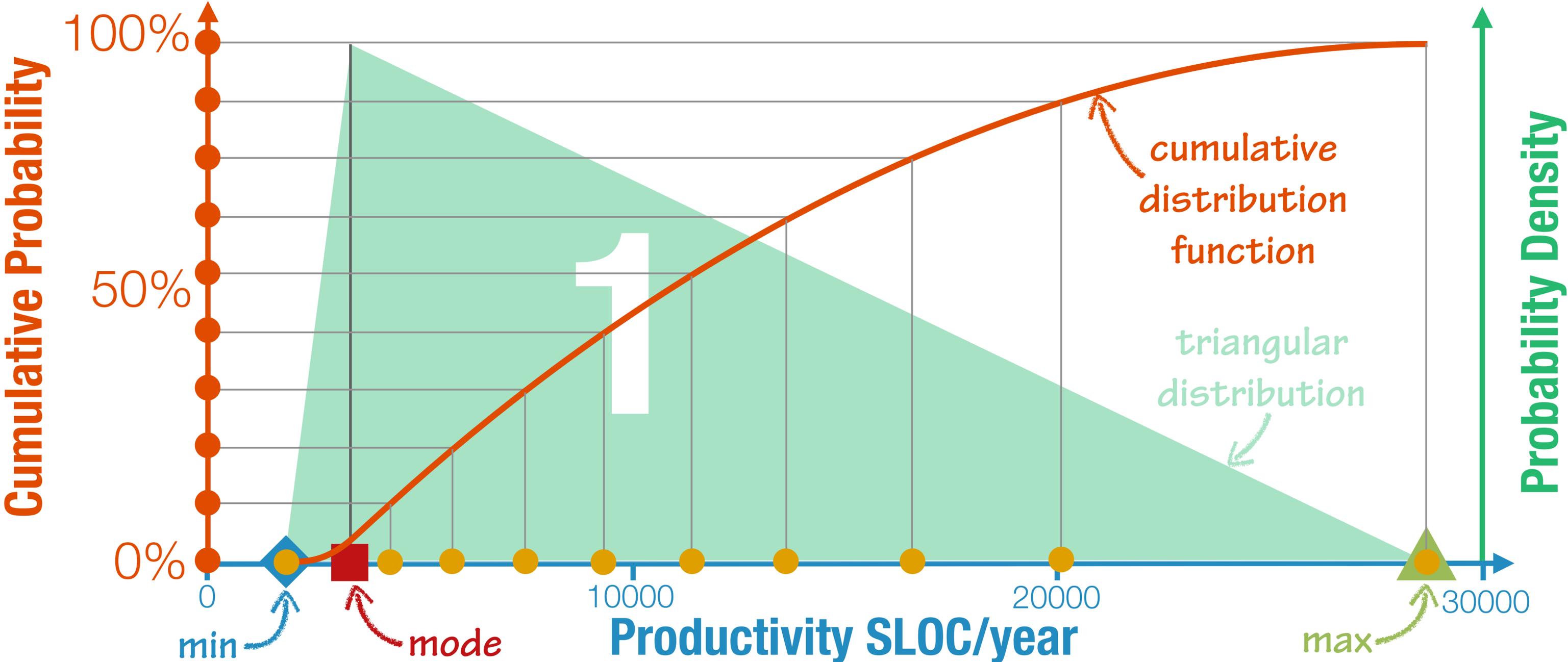
The number of years over which half the entities are replaced

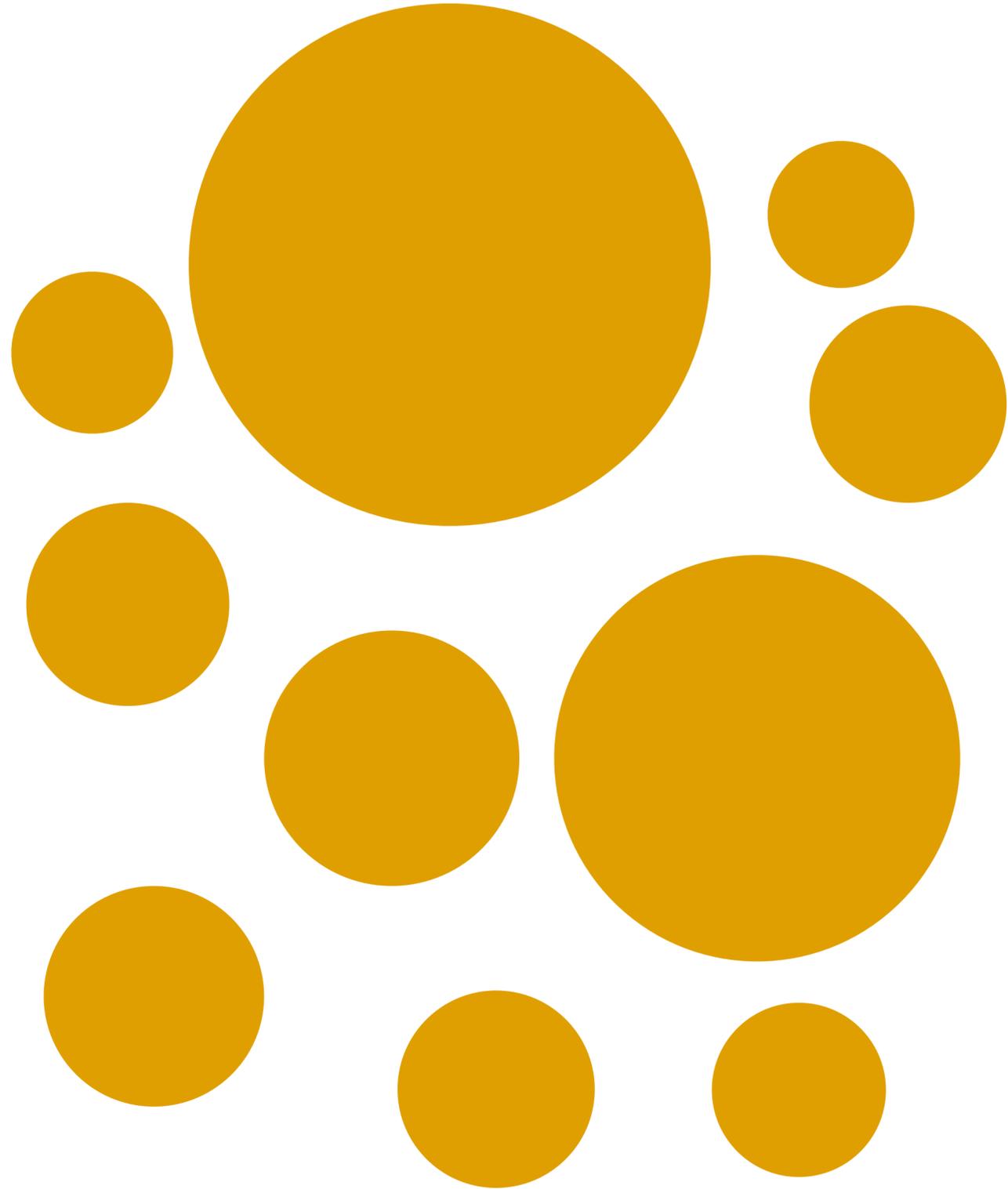


Simulating Developer Productivity

Draw teams at random from a productivity distribution

Productivity on 10000 SLOC codebase

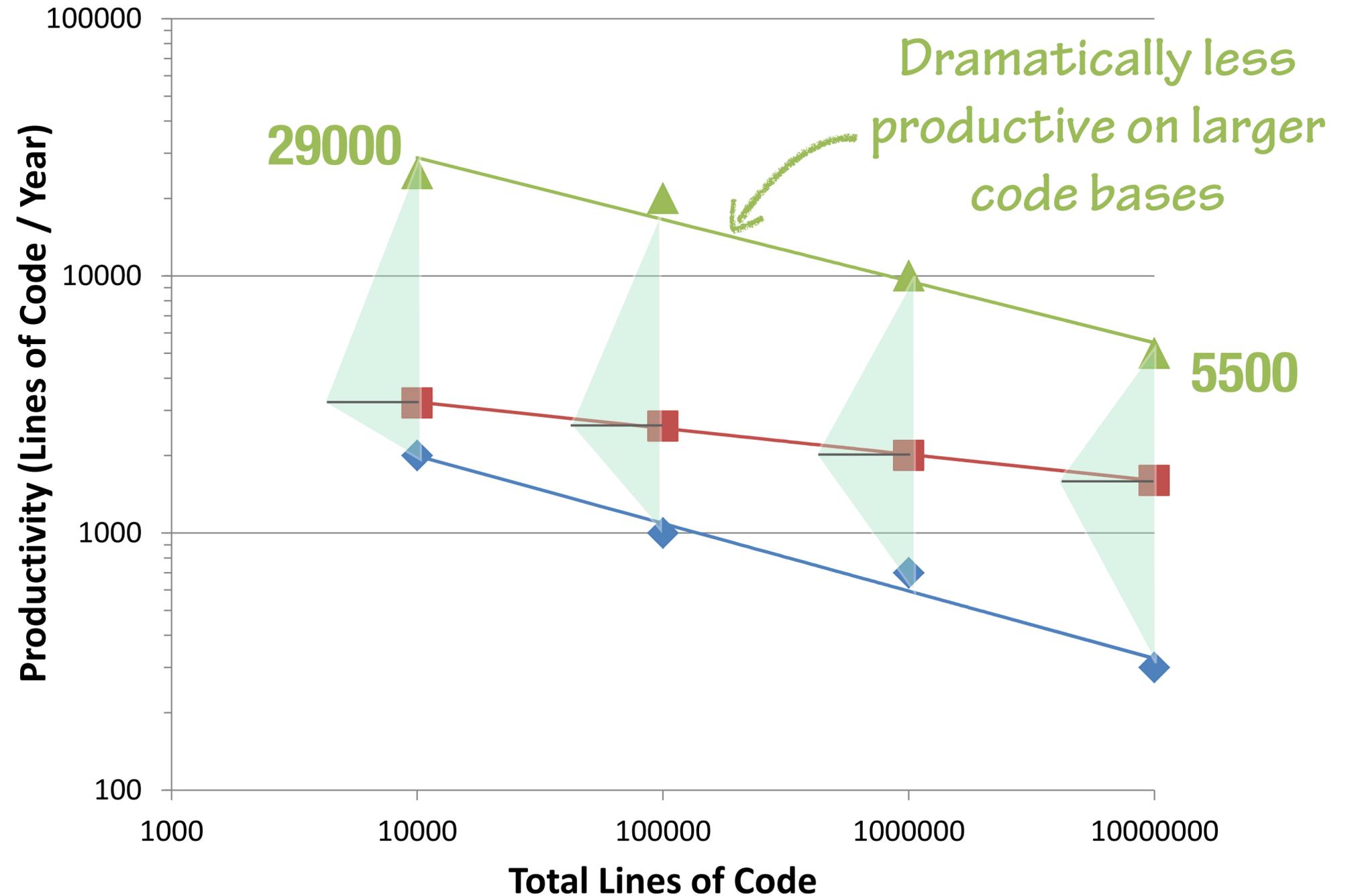




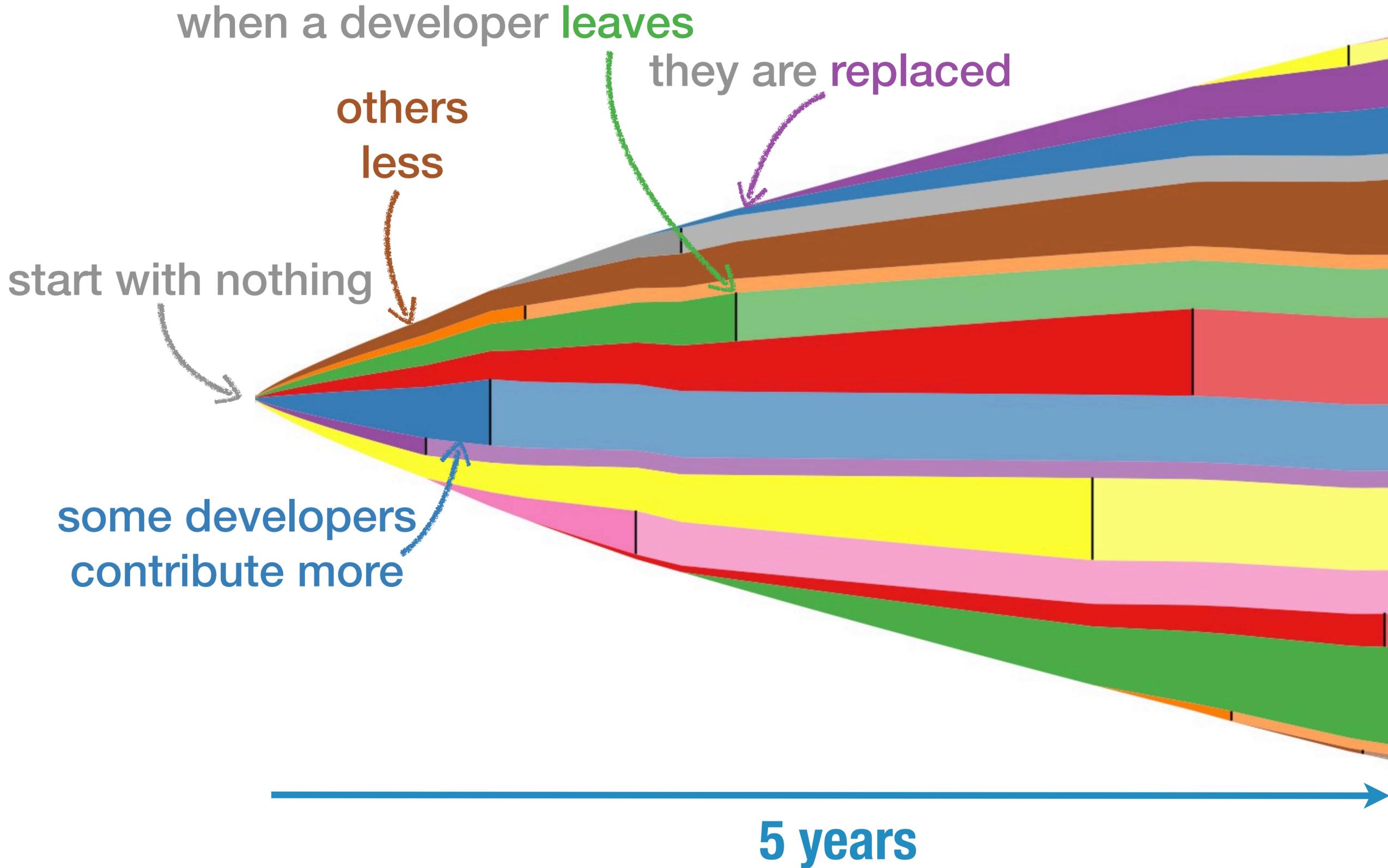
Modelling team and code evolution

Use published productivity data to forward model code size.

At any given system size we can predict a distribution for developer productivity.



Simulating a team of seven over five years



After 5 years we have **235 k** lines of code written by a total of **19** people.

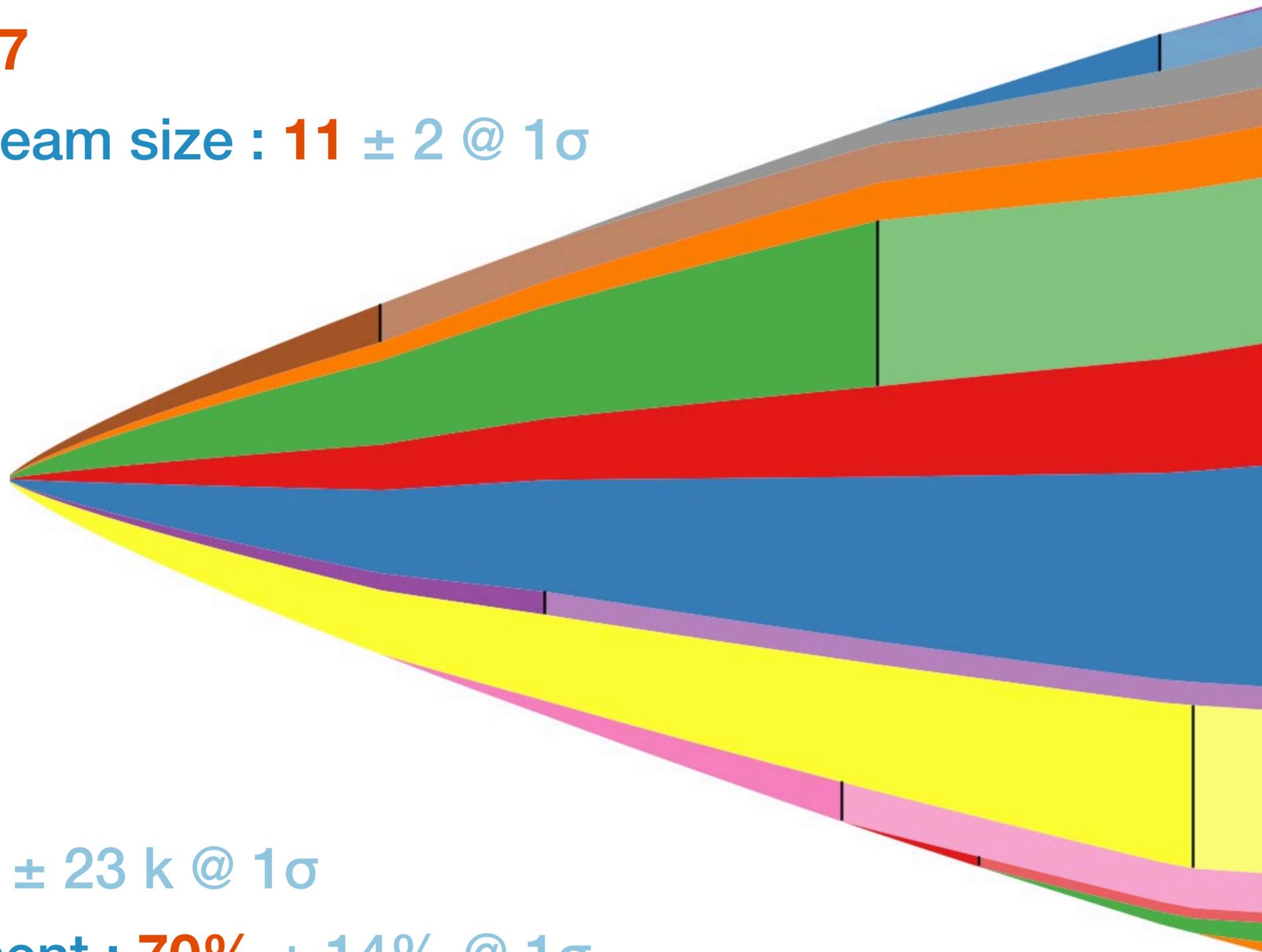
Only **37%** of the code is by current team



3 years

Team Size : 7

Cumulative team size : $11 \pm 2 @ 1\sigma$



157 kLoC

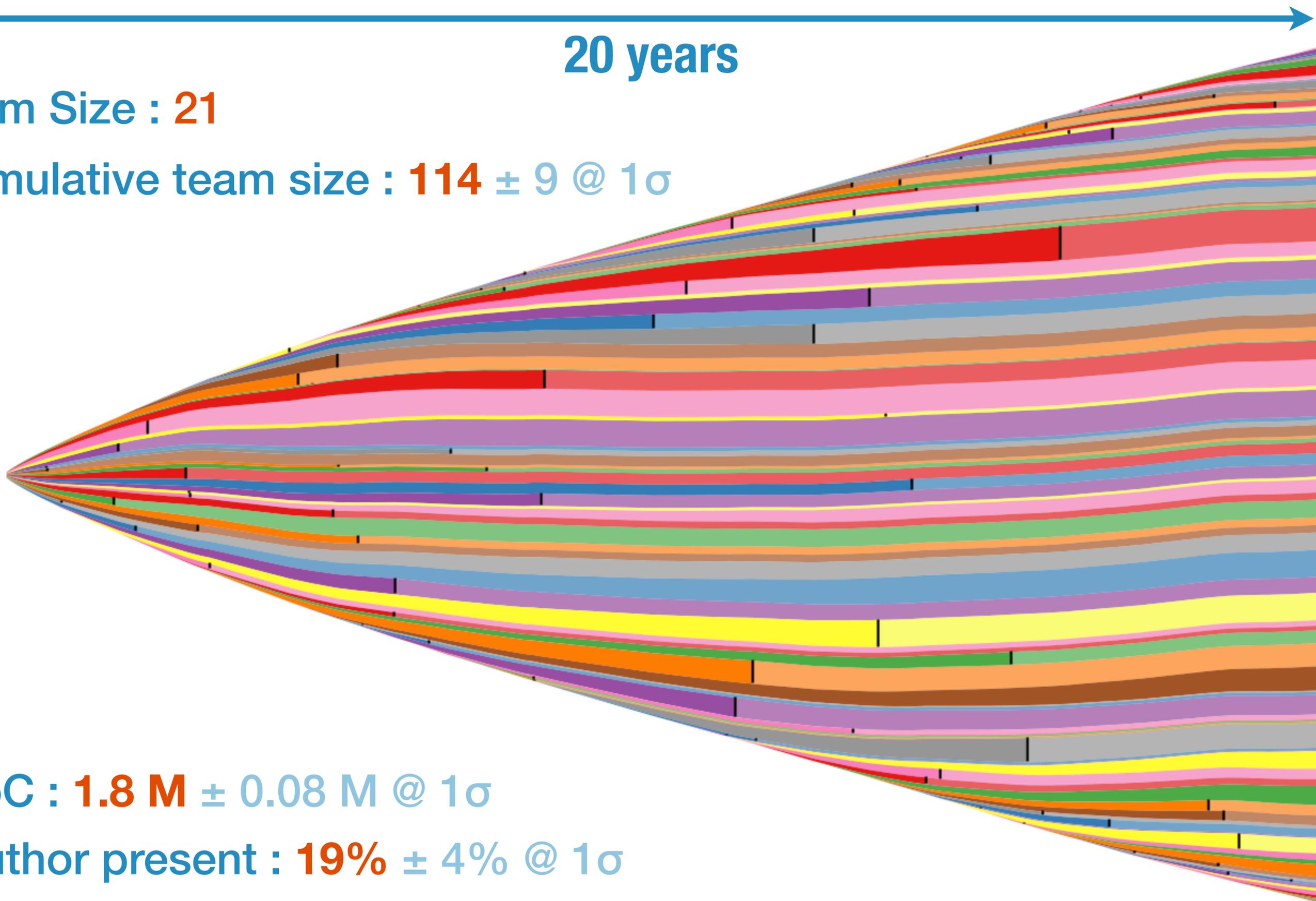
LoC : $157 \text{ k} \pm 23 \text{ k} @ 1\sigma$

Author present : $70\% \pm 14\% @ 1\sigma$

20 years

Team Size : 21

Cumulative team size : $114 \pm 9 @ 1\sigma$



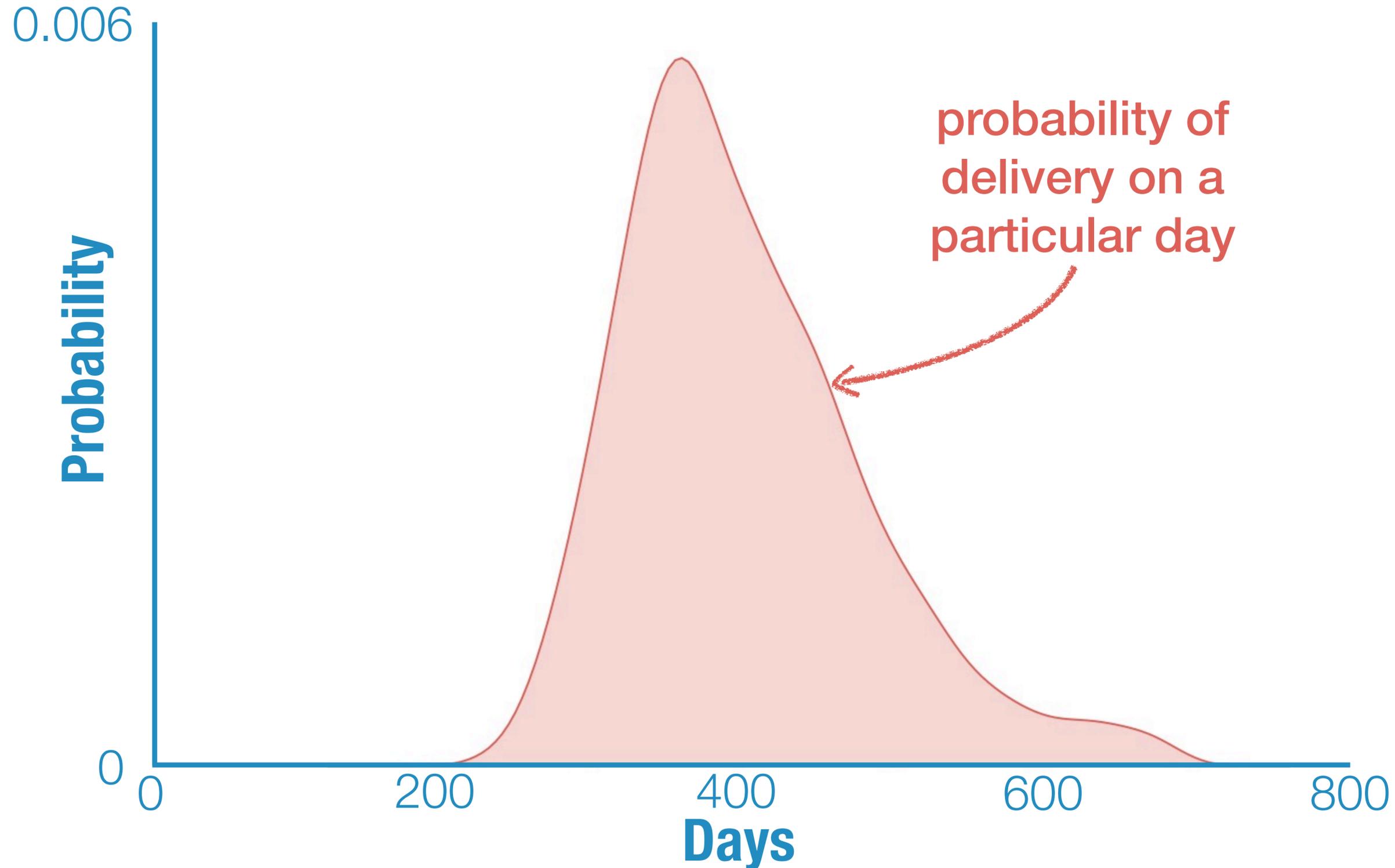
1.8 MLoC

LoC : $1.8 M \pm 0.08 M @ 1\sigma$

Author present : $19\% \pm 4\% @ 1\sigma$

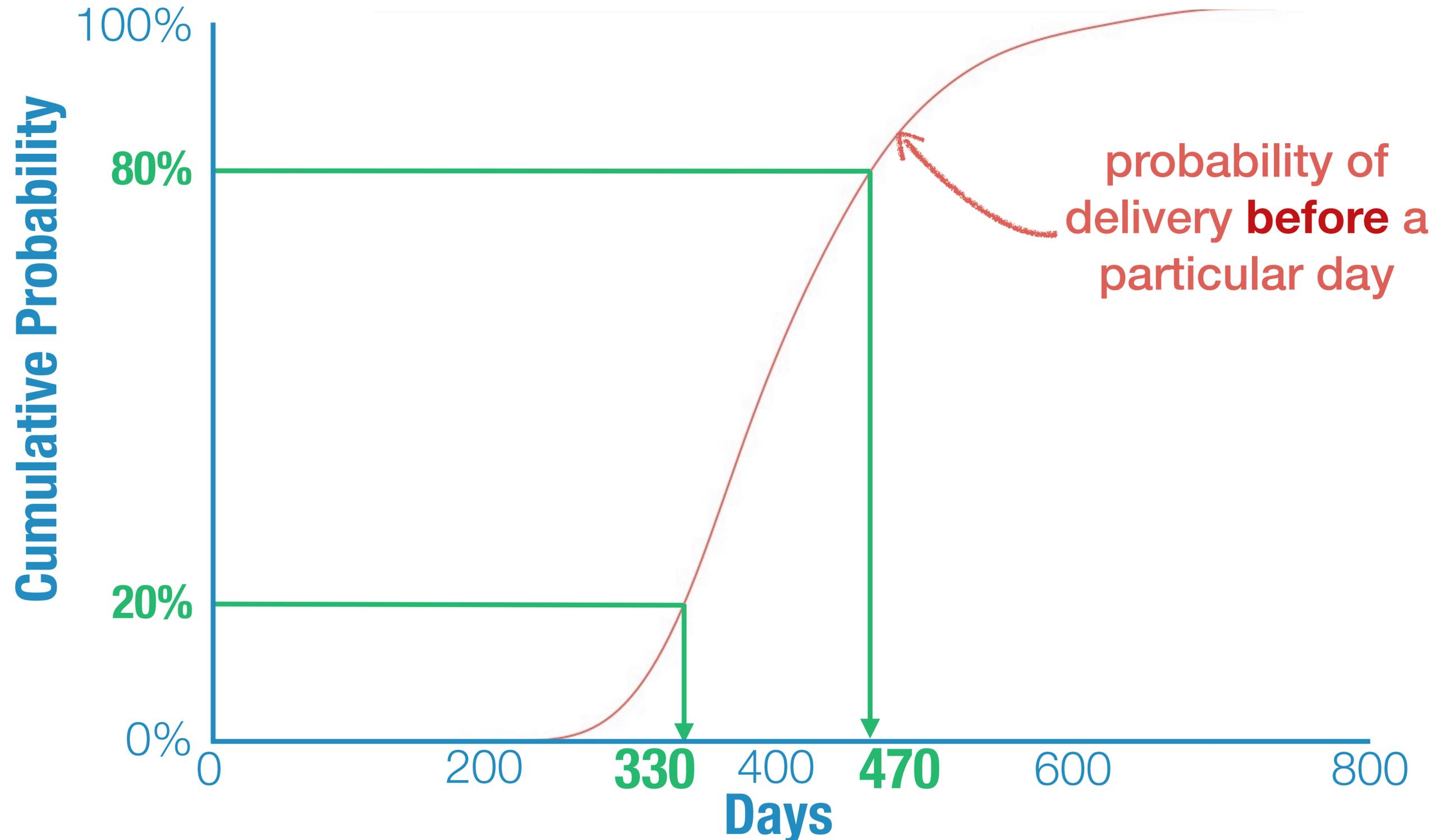
How long for seven to produce 100 000 lines of code?

Probability density from 1000 simulations



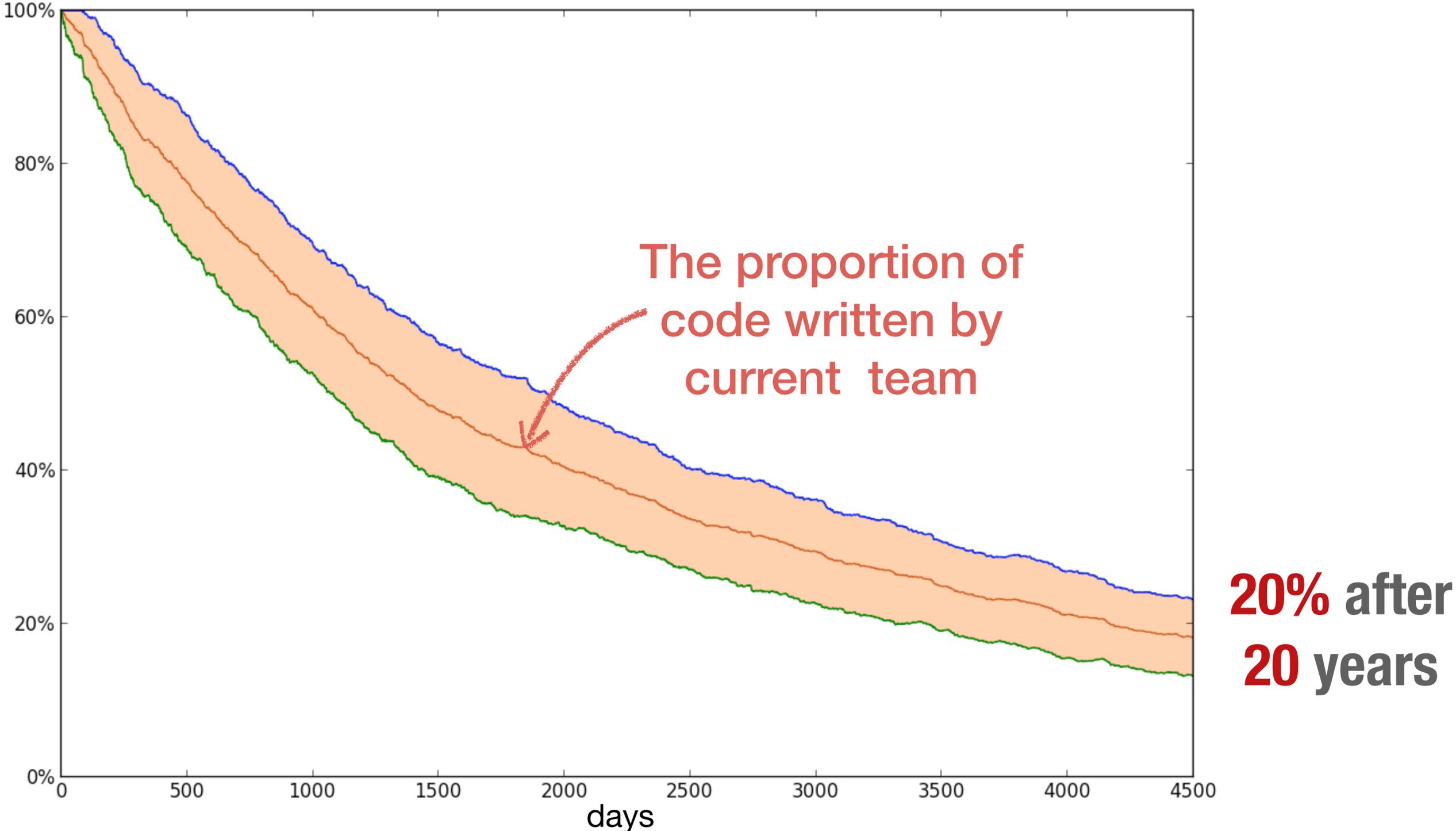
How long for 7 to produce 100 000 lines of code?

Cumulative probability from 1000 simulations



Who can you still talk to?

Most authors of your product quit way back when.





Stefan Tilkov

@stilkov



Following

If you think there's a conflict between "agile development" and "software architecture", you don't understand at least one of them

Reply Retweet Favorited More

36
RETWEETS

9
FAVORITES



1:28 PM - 3 Oct 13

Thank you!

Questions?

Robert Smallshire

 [@robsmallshire](https://twitter.com/robsmallshire)

SixtyNORTH

 [@sixty_north](https://twitter.com/sixty_north)



Thank you!

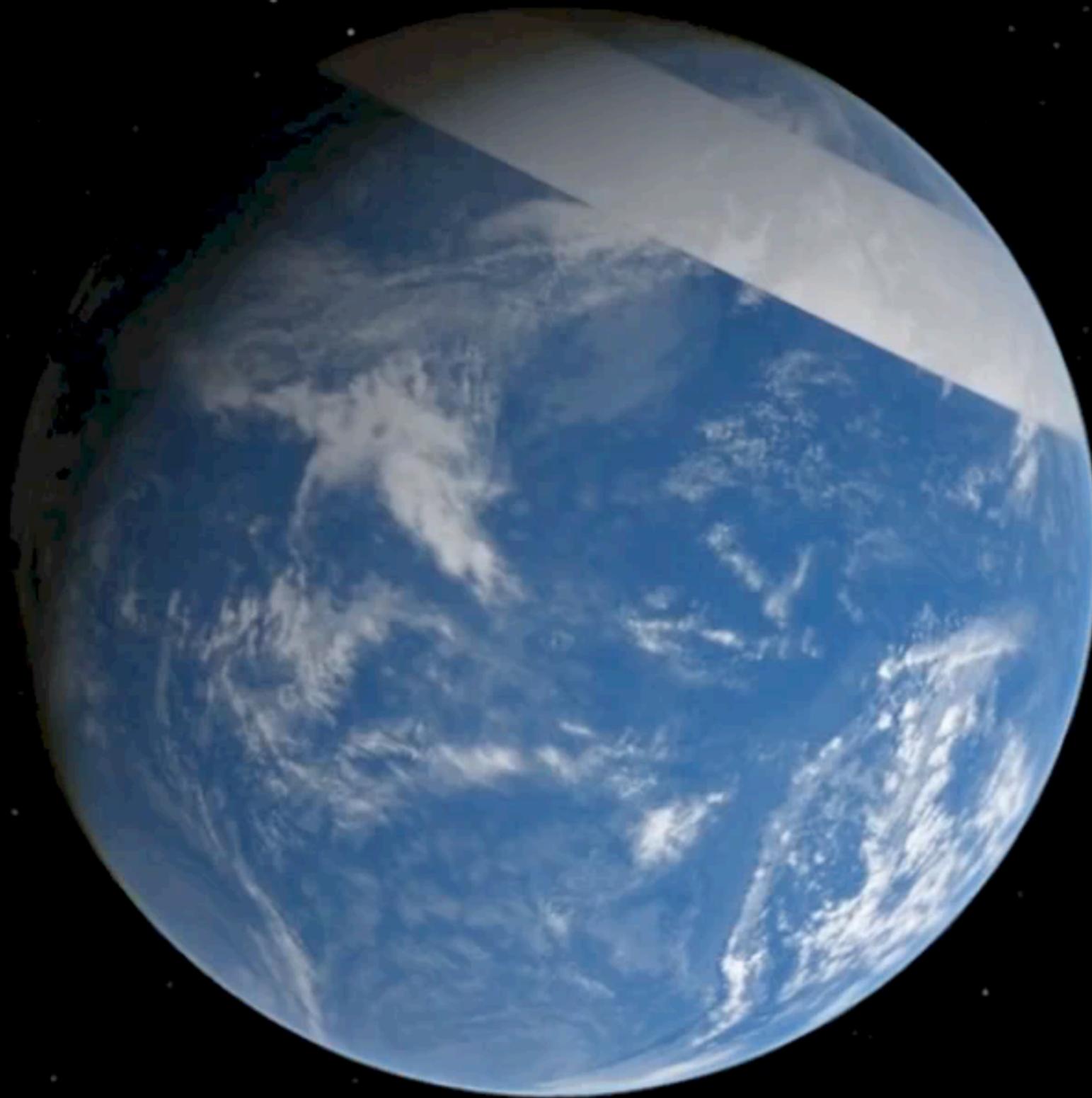
Questions?

Robert Smallshire

 [@robsmallshire](https://twitter.com/robsmallshire)

SixtyNORTH

 [@sixty_north](https://twitter.com/sixty_north)





SixtyNORTH