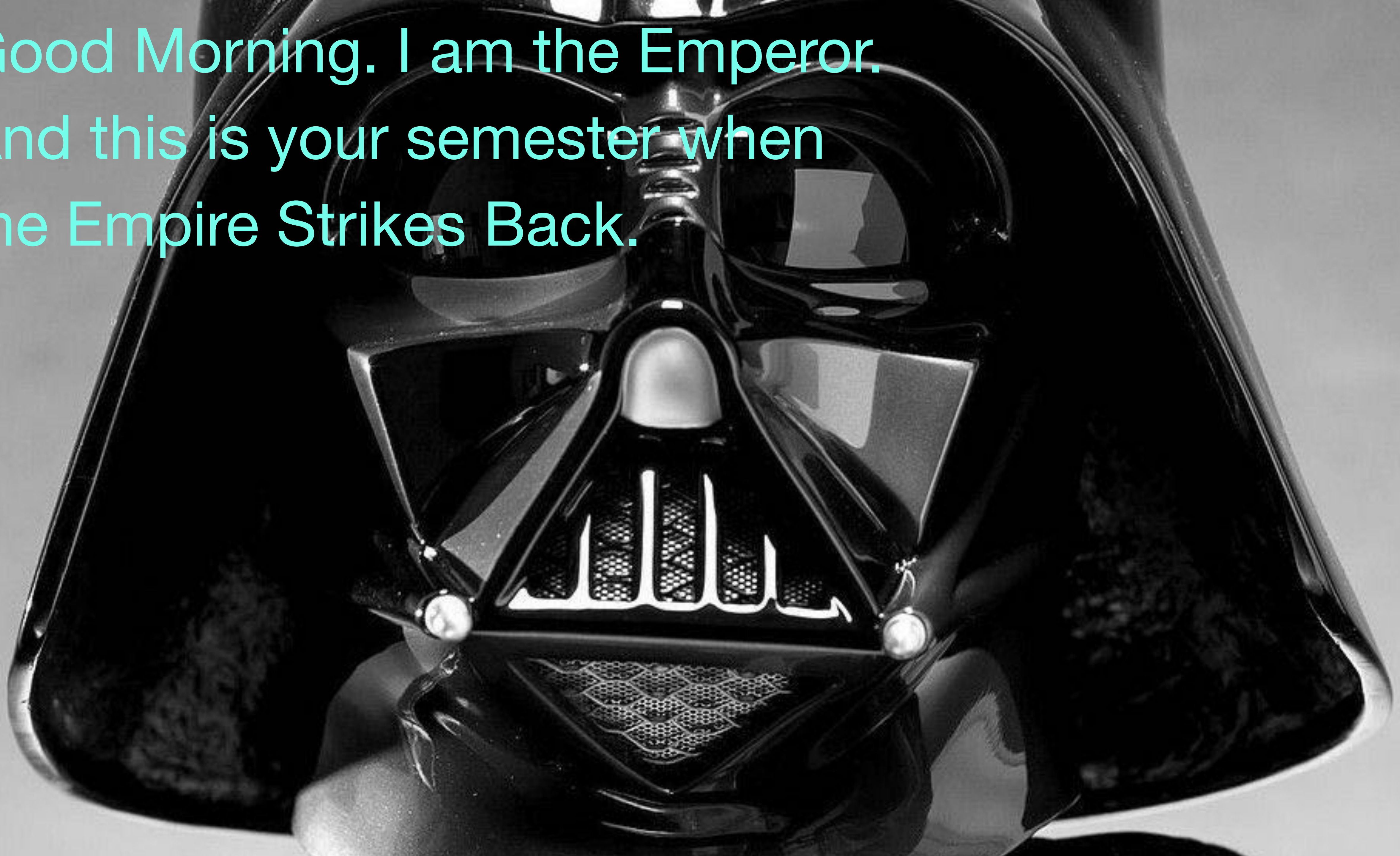






Good Morning. I am the Emperor.  
And this is your semester when  
the Empire Strikes Back.



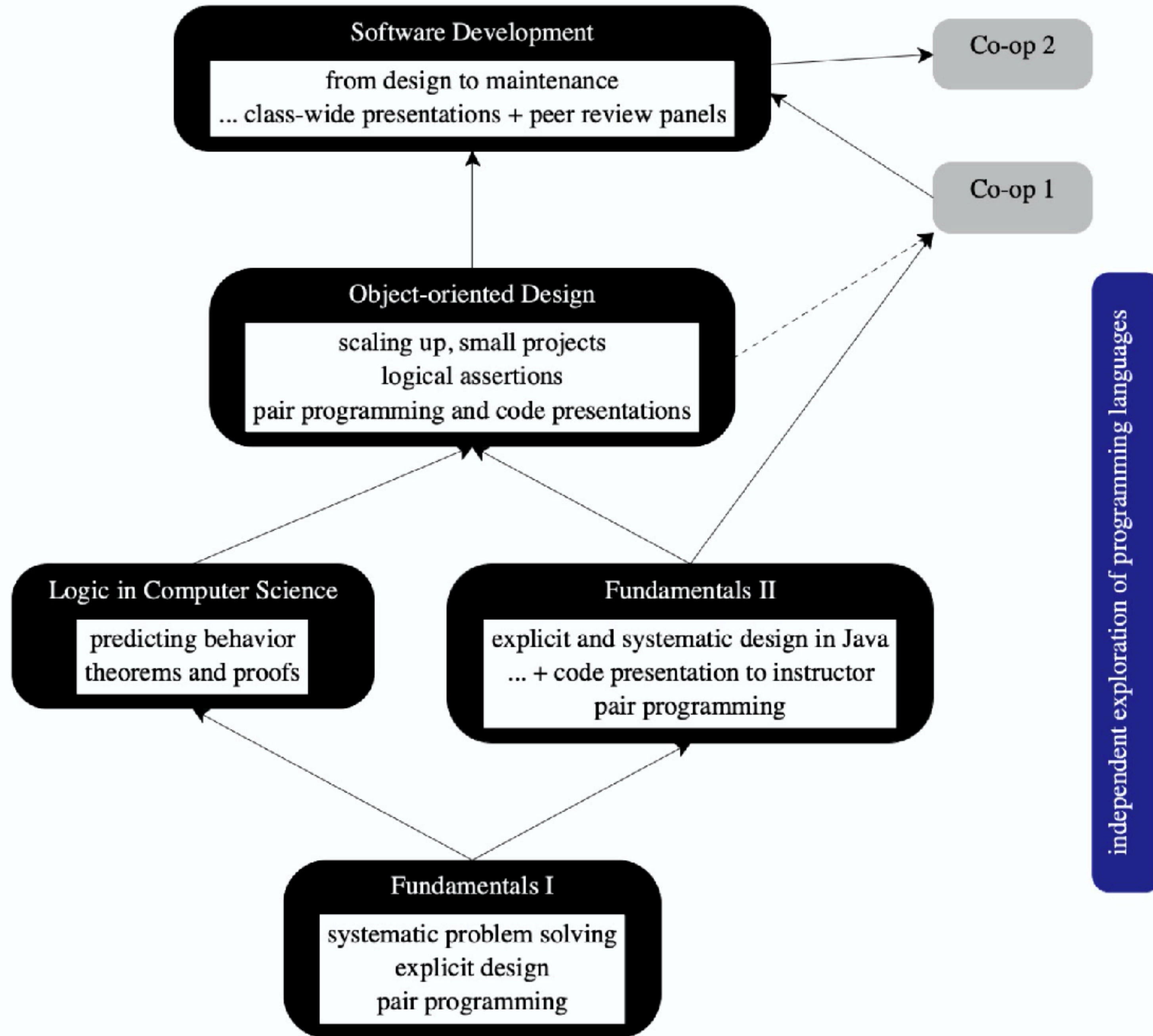
# CS 4500: Software Development

**Welcome**



Hemann/Felleisen





# Learning Outcomes

## In this course you will learn:

1. To scale the technical skills from your F1, F2, OOD, and (ideally) Logic:

- technical design skills
- human interaction skills

2. Pick up some “ephemeral skills” that otherwise fall through the cracks:

- git, IDEs, another PL, practical work with JSON, TCP, command line arguments, STDIN/OUT

# Learning Process

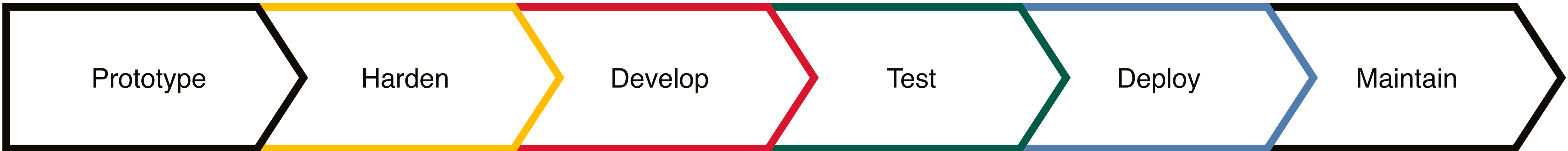
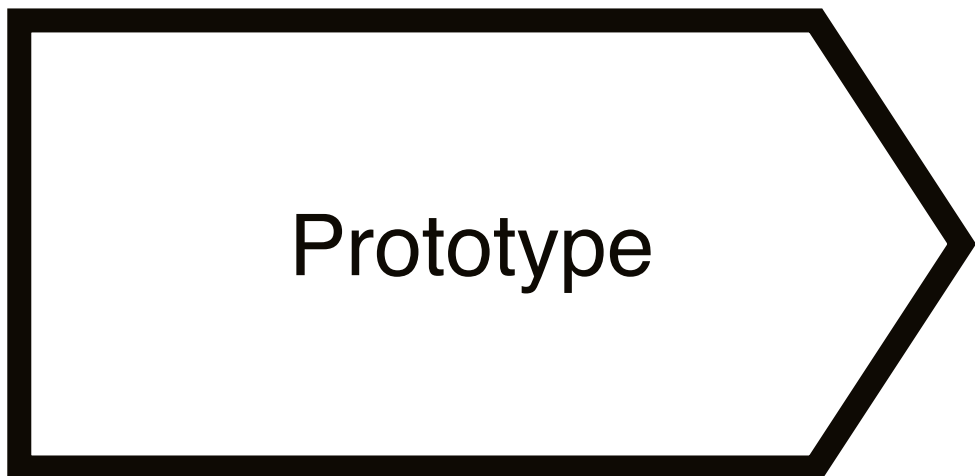
## In this course, you will

1. first explore the capabilities of your TAHBPL:
  - whether it lives up to the requirements of the anticipated project
2. second build a software system over the course of 10 weeks:
  - plan milestones
  - design interfaces, specify software components
  - implement the instructors' designs
3. while routinely presenting your code in class and reviewing the code of others

# CS 4500: Software Development

## **The Costs of Software**







**If your software survives the cradle ...**

- the monetary cost of creation and maintenance
- the people-time cost
- the cost of people's lives

**It costs time and money**

# THE FIRST

by: **Jenny List**



Interplanetary probes were a core part of the space program as it was at its height, and alongside the Mars rovers throughout the Solar System. By the time the Mars rovers were sent, the Mars rovers had moved in, and aside from the Mars rovers, the Voyager series of craft there were the only ones left.

The launch in late 1996 of the Mars rovers was indeed. Before Spirit, the exceptional sense of scale from **our recent technology** to the days, and proved the technology of the time.

In these days of constant online information, but those of us watching with interest have suffered what is being written up as a glitch. The data, its computer would crash.

FEBRUARY 11, 2020

## NASA Engineers Fix Glitch On Voyager 2 Spacecraft From 11.5 Billion Miles Away!

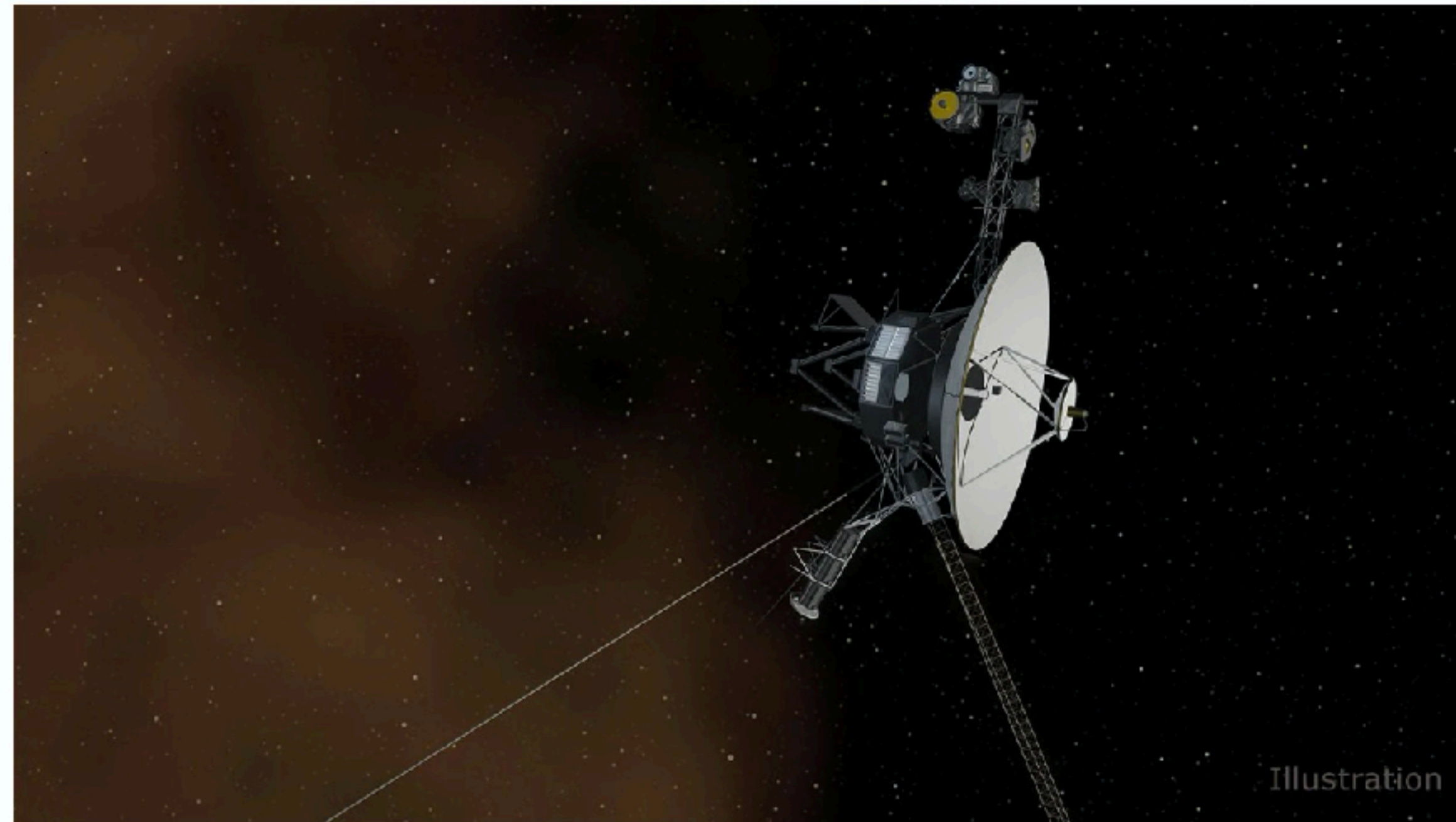
BY MEERA DOLASIA

CCSS

NAS-4

Interest Level 3-12 ▾

♥ Favorite



An artist's illustration of the Voyager 2 probe, which has been exploring space for over 40 years (Credit: NASA/JPL-Caltech)

In what can only be described as an **extraordinary** engineering **feat**, NASA scientists have remotely fixed a software **glitch** on the Voyager 2 probe, which lies 11.5 billion miles away at the edge of a **transitional** region of space. Known as the heliosphere, it is a vast, bubble-like area that surrounds the Sun and the Solar System.

# Wanted urgently: People a half century-old computer language so states can pay unemployment claims

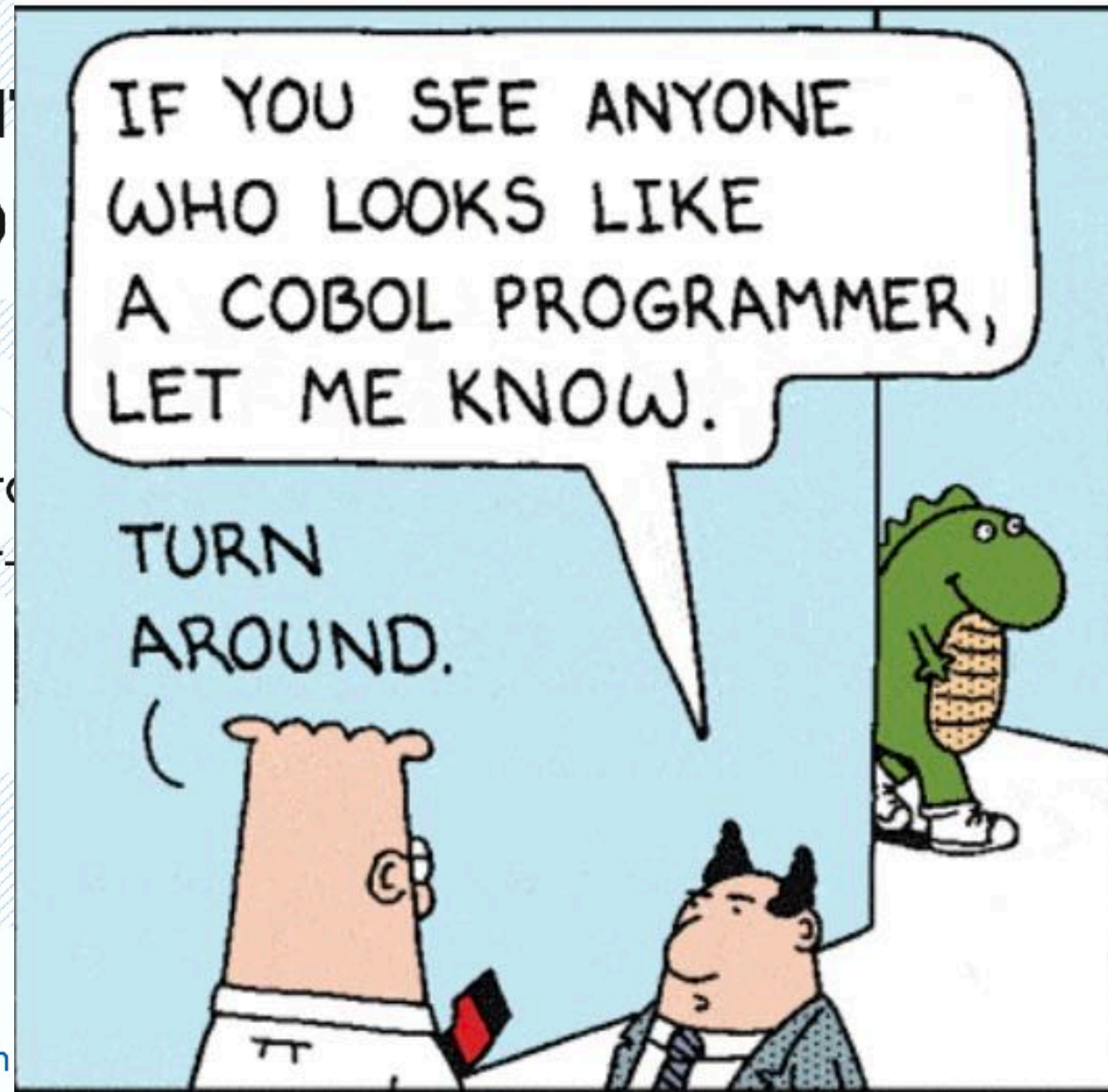
Connecticut has admitted that it's struggling to pay the volume of unemployment claims with its "40-year-

Author: CNN Wire

Published: 3:38 PM EDT April 8, 2020

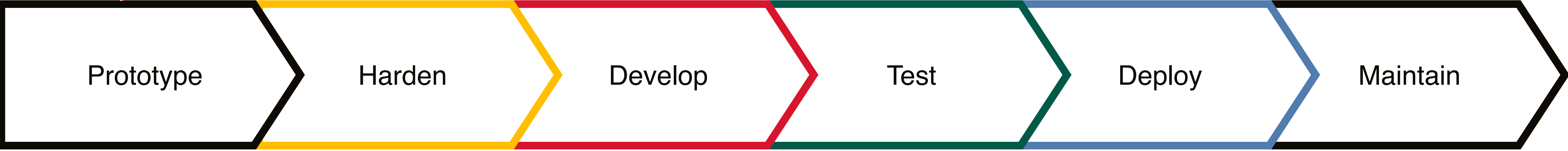
Updated: 3:38 PM EDT April 8, 2020

HARTFORD, Conn. — On top of ventilators, [face masks](#) and [health](#) add COBOL programmers to the list of what several states urgently need as they battle the [coronavirus pandemic](#).



software is a message from you to a future programmer

... and that future programmer must decode this message



**The most expensive part of  
SDLC is maintenance**



<b>Year</b>	<b>Proportion of software maintenance costs</b>	<b>Definition</b>	<b>Reference</b>
2000	>90%	Software cost devoted to system maintenance & evolution / total software costs	Erlikh (2000)
1993	75%	Software maintenance / information system budget (in Fortune 1000 companies)	Eastwood (1993)
1990	>90%	Software cost devoted to system maintenance & evolution / total software costs	Moad (1990)
1990	60-70%	Software maintenance / total management information systems (MIS) operating budgets	Huff (1990)
1988	60-70%	Software maintenance / total management information systems (MIS) operating budgets	Port (1988)
1984	65-75%	Effort spent on software maintenance / total available software engineering effort.	McKee (1984)
1981	>50%	Staff time spent on maintenance / total time (in 487 organizations)	Lientz & Swanson (1981)
1979	67%	Maintenance costs / total software costs	Zelkowitz <i>et al.</i> (1979)

**Table 1.** Proportional software maintenance costs for its supplier.

*40% to 60% of the maintenance effort is devoted to **understanding the software to be modified***

IEEE Guide to the Software Engineering Body of Knowledge, 2004

**It costs lives**

# Not to name names ...

- Therac-25
- Intel FDIV “Pentium Bug”
- Knight Capital
- Mariner 1
- Stanislav Petrov and SAD
- Mars Climate Orbiter
- Ariane 5 Flight 501
- Boeing 737 Max
- Uber Self-driving Cars
- Panama Cancer Institute
- 2003 North American Blackout
- KAL Flight 801
- ILOVEYOU virus
- Toyota Acceleration Recalls

# Not to name names ...

- Therac-25
- Intel FDIV “Pentium Bug”
- Knight Capital
- Mariner 1
- Stanislav Petrov and SAD
- Mars Climate Orbiter
- Ariane 5 Flight 501
- Boeing 737 Max
- Uber Self-driving Cars
- Panama Cancer Institute
- 2003 North American Blackout
- KAL Flight 801
- ILOVEYOU virus
- Toyota Acceleration Recalls



Do not use  
Out of order

# Not to name names ...

- Therac-25

- Intel FDIV “Pentium Bug”

- Knight Capital

- Mariner 1

- Stanislav Petrov and SAD

- Mars Climate Orbiter

- Ariane 5 Flight 501

- Boeing 737 Max

- Uber Self-driving Cars

- Panama Cancer Institute

- 2003 North American Blackout

- KAL Flight 801

- ILOVEYOU virus

- Toyota Acceleration Recalls

Home > Mobile

## Whos Reading Your Cells Text Messages?

By: Paul F. Roberts | February 27, 2006

A college student's cell phone becomes the recipient of a blizzard of lost text messages from all over the country.

Have you ever hit "Send" on a text message on your mobile phone before addressing it? Ever wondered where all those lost SMS text messages go? If so, you might want to speak with Stan Bubrouski, whose cell phone has been channeling wayward text messages from across the country for years.

Bubrouski, a computer science major at Northeastern University in Boston, is the proud owner of Null@vtext.com, an account on the popular Verizon text messaging service that allows Internet users to send e-mail and IM messages directly to his cell phone as SMS text messages.

Bubrouski said he was just being clever when he signed up for a Verizon vText account with the user name null, after his parents bought him his first mobile phone during his freshman year at Northeastern, in 2001.

"I've been paying for it ever since," Bubrouski told eWEEK.

Bubrouski's new vText account didn't just hook him up with his friends, it also opened the door to a blizzard of unsolicited



Home > Mobile

## Whos Reading Your Cells Text Messages?

By: Paul F. Roberts | February 27, 2006

Bubrouski, a computer science major at Northeastern University in Boston, is the proud owner of Null@vtext.com, an account on the popular Verizon text messaging service that allows Internet users to send e-mail and IM messages directly to his cell phone as SMS text messages.

Bubrouski said he was just being clever when he signed up for a Verizon vText account with the user name null, after his parents bought him his first mobile phone during his freshman year at Northeastern, in 2001.

"I've been paying for it ever since," Bubrouski told eWEEK.

account on the popular Verizon text messaging service that allows Internet users to send e-mail and IM messages directly to his cell phone as SMS text messages.

Bubrouski said he was just being clever when he signed up for a Verizon vText account with the user name null, after his parents bought him his first mobile phone during his freshman year at Northeastern, in 2001.

"I've been paying for it ever since," Bubrouski told eWEEK.

Bubrouski's new vText account didn't just hook him up with his friends, it also opened the door to a blizzard of unsolicited

To recap:

- the life of a software system is likely to exceed the life span of its creator
- it thus becomes a message from now into the (near or distant) future
- .. deciphering this message takes time and thus costs money
- .. may cost lives at any point, now or far into the future

The core programming courses address this “message” problem two ways:

- systematic software construction — because slow and steady works
- collaborating with others — because compilers don't really think

# CS 4500: Software Development

## **Systematic Software Design**

# The Predictions of a Wise Man

*"There is no single development, in either technology or management technique, which by itself promises even one order-of-magnitude improvement within a decade in productivity, in reliability, in simplicity."*

**Fred Brooks, *No Silver Bullet*, 1986**

- High(er) level languages

- OOP

- AI

- Expert systems

- Program Synthesis

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations

- ~~High(er) level languages~~

- OOP

- AI

- Expert systems

- Program Synthesis

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations



- ~~High(er) level languages~~

- ~~OOP~~

- AI

- Expert systems

- Program Synthesis

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- Expert systems

- Program Synthesis

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- ~~Expert systems~~

- Program Synthesis

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- ~~Expert systems~~

- ~~Program Synthesis~~

- Graphical Programming

- Program Verification

- Environment/Tools/Workstations

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- ~~Expert systems~~

- ~~Program Synthesis~~

- ~~Graphical Programming~~

- ~~Program Verification~~

- ~~Environment/Tools/Workstations~~

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- ~~Expert systems~~

- ~~Program Synthesis~~

- ~~Graphical Programming~~

- ~~Program Verification~~

- ~~Environment/Tools/Workstations~~

- ~~High(er) level languages~~

- ~~OOP~~

- ~~AI~~

- ~~Expert systems~~

- ~~Program Synthesis~~

- ~~Graphical Programming~~

- ~~Program Verification~~

- ~~Environment/Tools/Workstations~~

What academic should not, and this course will not, teach:

- specific, currently fashionable tools (CI, bugs, tickets)
- the hottest language and IDE on Earth
- fashionable processes (agile 1, agile 2, agile 3)



# Steady, Unspectacular Progress

- Buy v. Build (COTS)
- Prototyping & Refining Requirements
- Incremental Development
- Cultivate Great Designers

# What are the Attributes of Systematic Program Design

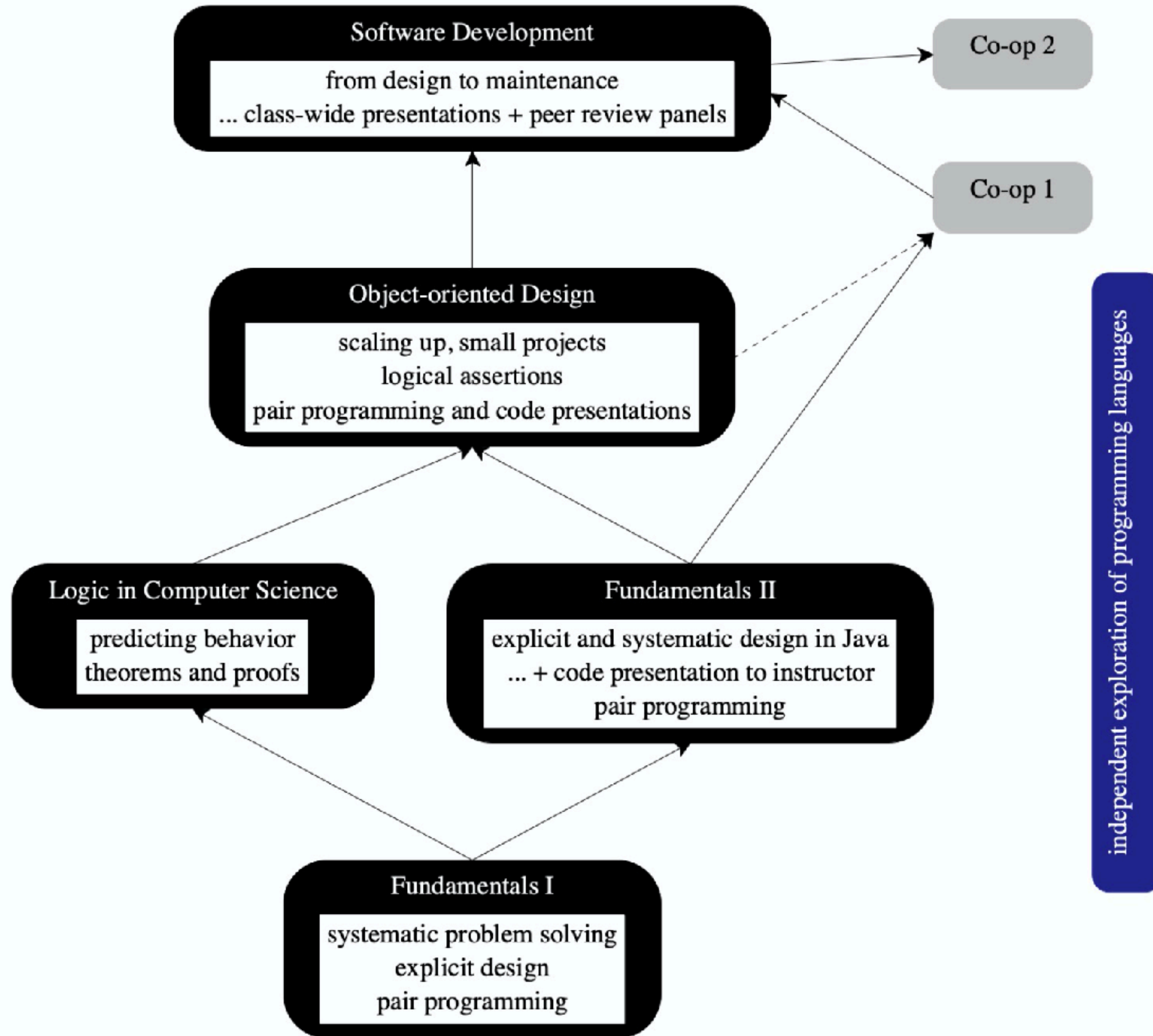
- **design strategy:** step by step, iterative refinement
- **canonical outcomes:** from problems to solutions
- **continuous process:**
  - *small changes to the problem statement result in small changes to the solution in a predictable manner*

# Structural Design: Forms of Data

Process Steps

	atomic	enumer.	structs	hier.	union	rec	mut. rec.
data def							
purpose							
examples							
template	Domain knowledge						
code!							
test							

Where CS Works

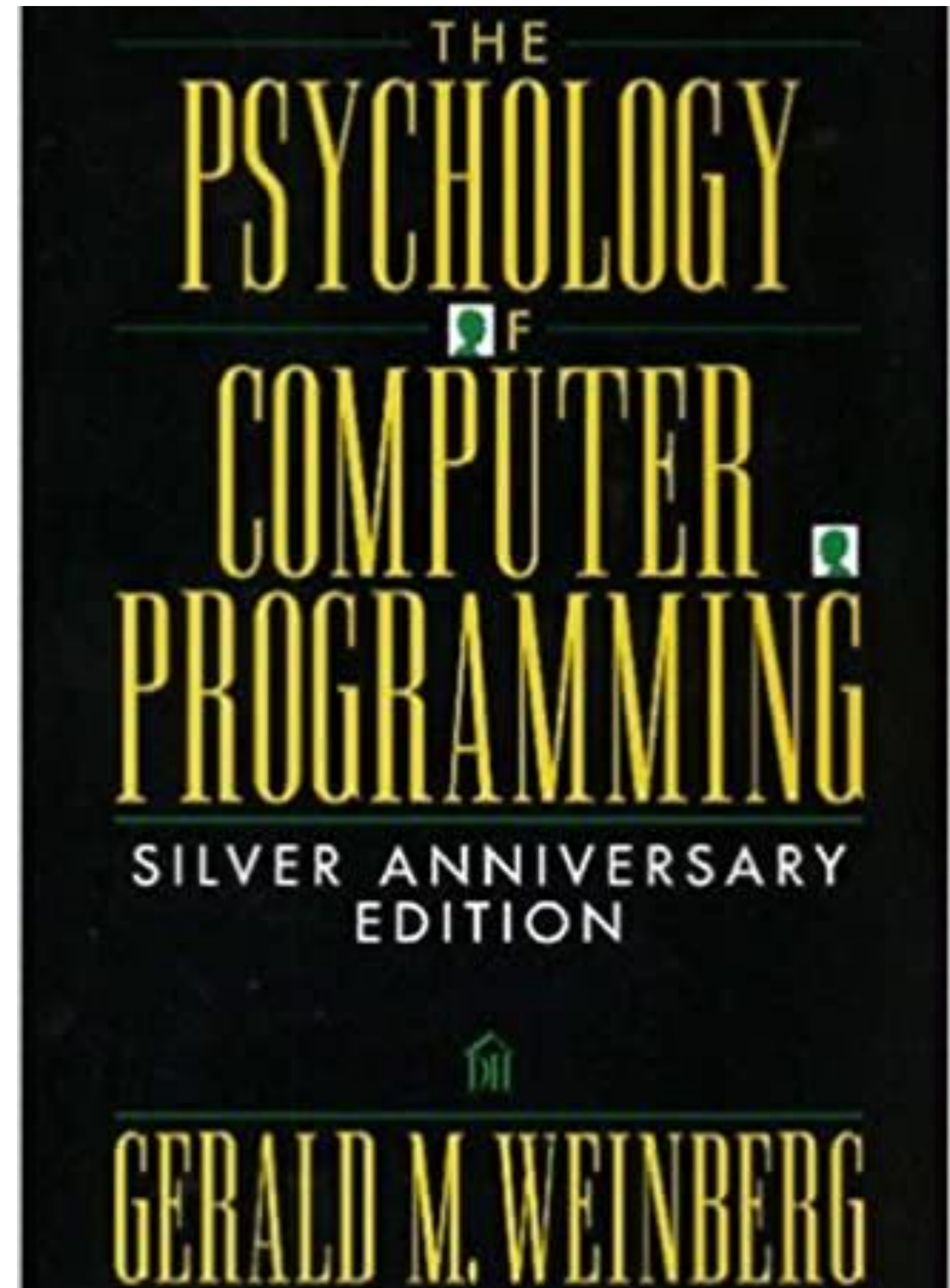


# CS 4500: Software Development

**The Human Element: Management 101**

**“Soft Skills” are hard!**

*“Egoless Programming”*



*See Readings:*

*The Psychology of Computer Programming, Weinberg 1971, one chapter*



KEEP IN MIND THAT I'M SELF-TAUGHT, SO MY CODE MAY BE A LITTLE MESSY.

LEMME SEE- I'M SURE IT'S FINE.

...WOW. THIS IS LIKE BEING IN A HOUSE BUILT BY A CHILD USING NOTHING BUT A HATCHET AND A PICTURE OF A HOUSE.

IT'S LIKE A SALAD RECIPE WRITTEN BY A CORPORATE LAWYER USING A PHONE AUTOCORRECT THAT ONLY KNEW EXCEL FORMULAS.

IT'S LIKE SOMEONE TOOK A TRANSCRIPT OF A COUPLE ARGUING AT IKEA AND MADE RANDOM EDITS UNTIL IT COMPILED WITHOUT ERRORS.

OKAY, I'LL READ A STYLE GUIDE.

**How to be a good manager  
(employee, friend ...)**

Care Personally

RUINOUS  
EMPATHY

RADICAL  
CANDOR

Challenge Directly

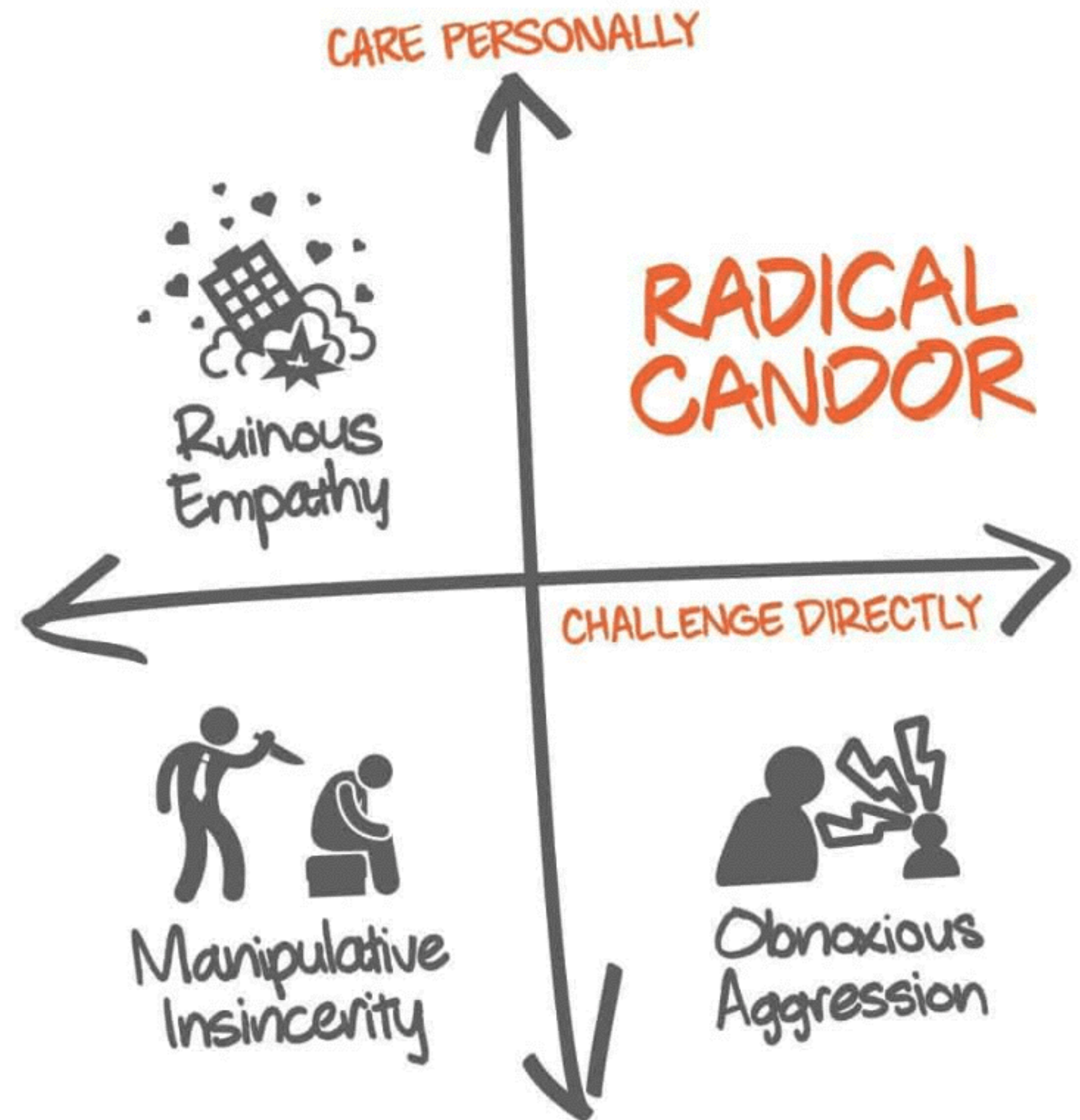
MANIPULATIVE  
INSINCERITY

OBNOXIOUS  
AGGRESSION



# Our approach:

- Be frank, be honest, be direct
- Be warm, kind and compassionate
- The book, the podcast, the blogs, the talks



# Radical Candor: Why brutal honesty is tech's hottest management trend

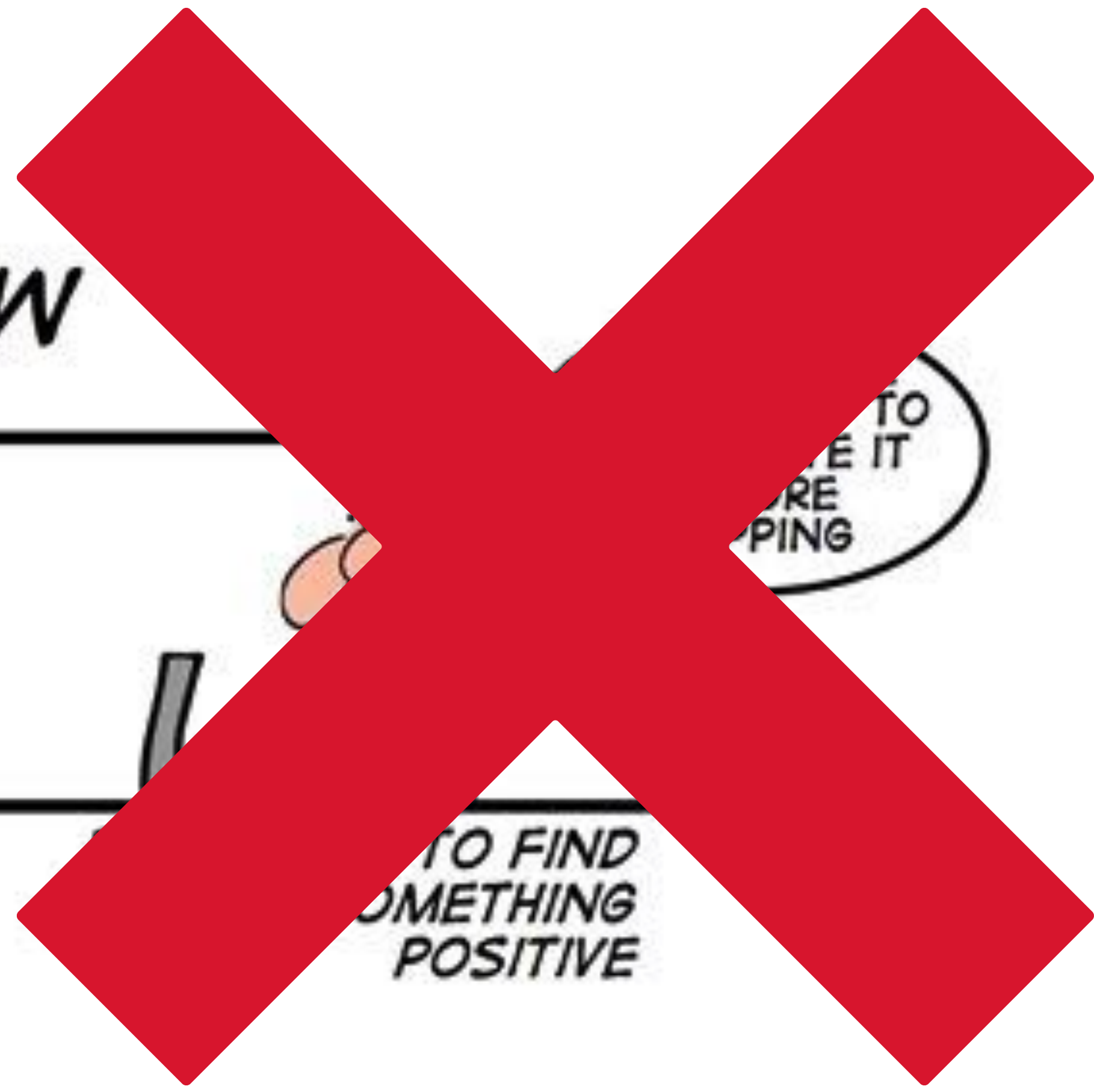


— "Radical Candor"

Feb. 13, 2018, 12:44 PM EST / Updated Feb. 13, 2018, 12:44 PM EST

By Kim Bainbridge and Lisa Everson

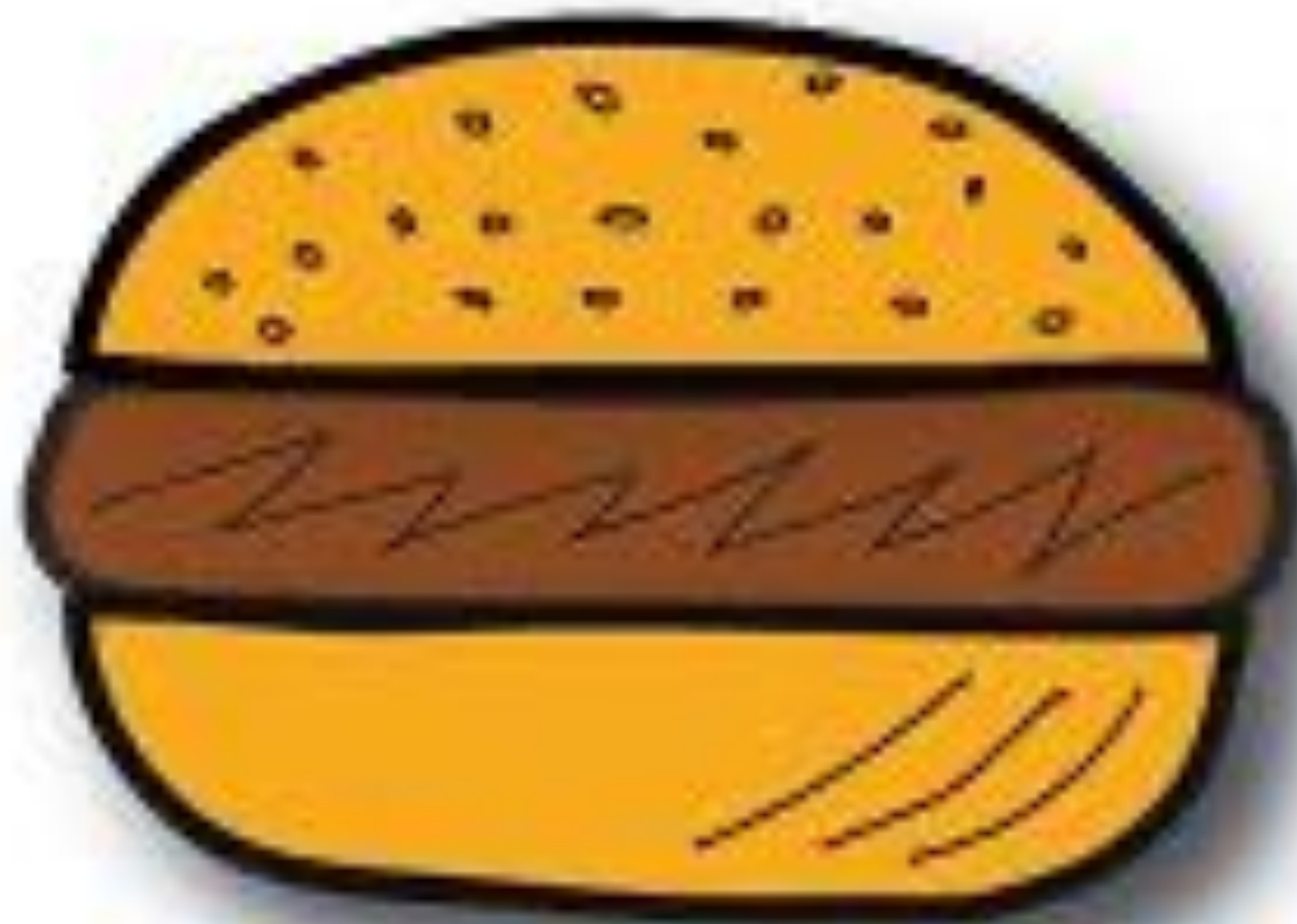
# HOW TO MAKE A GOOD CODE REVIEW



TO FIND SOMETHING POSITIVE

TO FIND SOMETHING POSITIVE

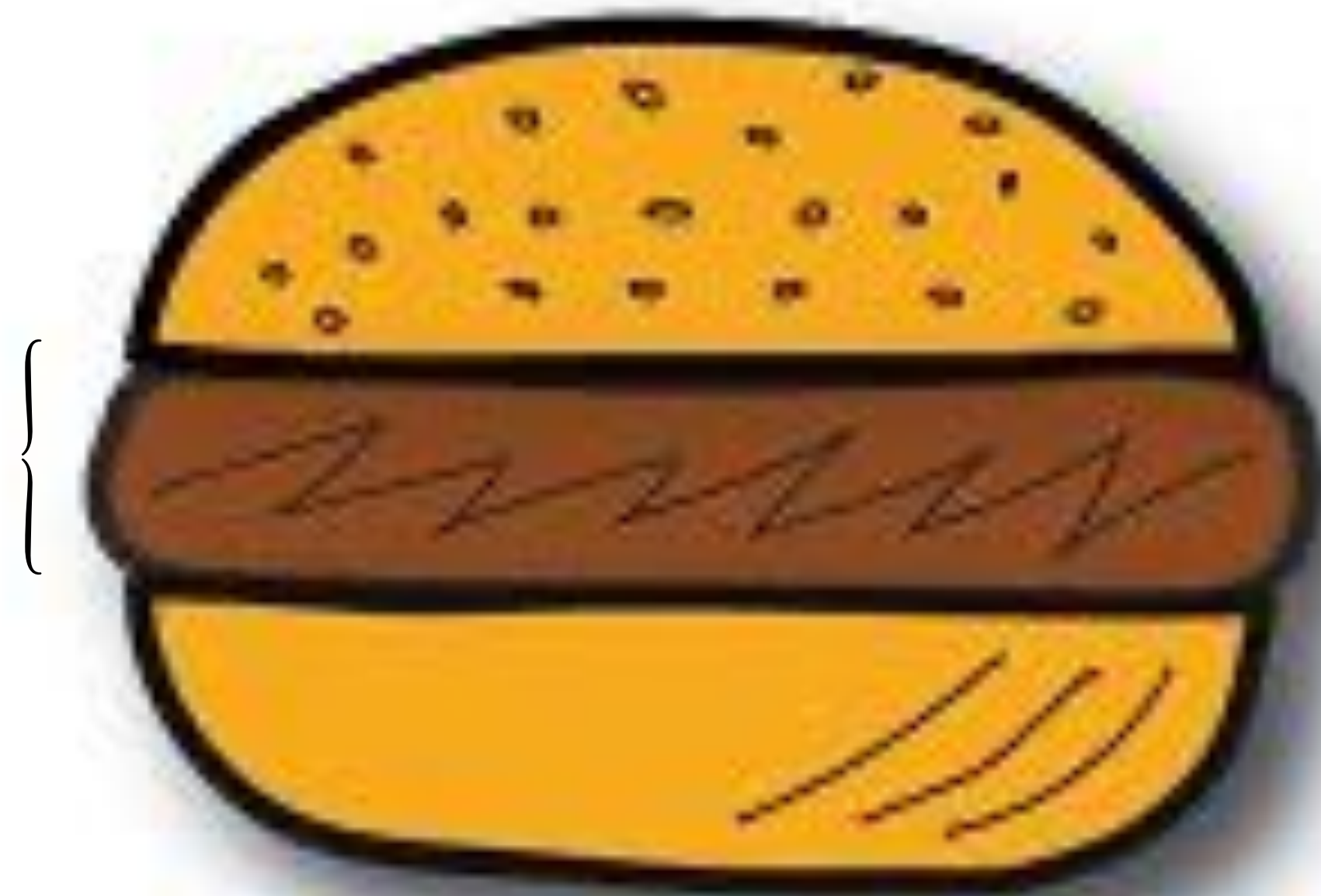
# The Feedback Sandwich



**positive**  
**negative**  
**positive**

# The Feedback Sandwich

Meaningful Feedback



**positive**

**negative**

**positive**



# The Feedback Sandwich

Waste



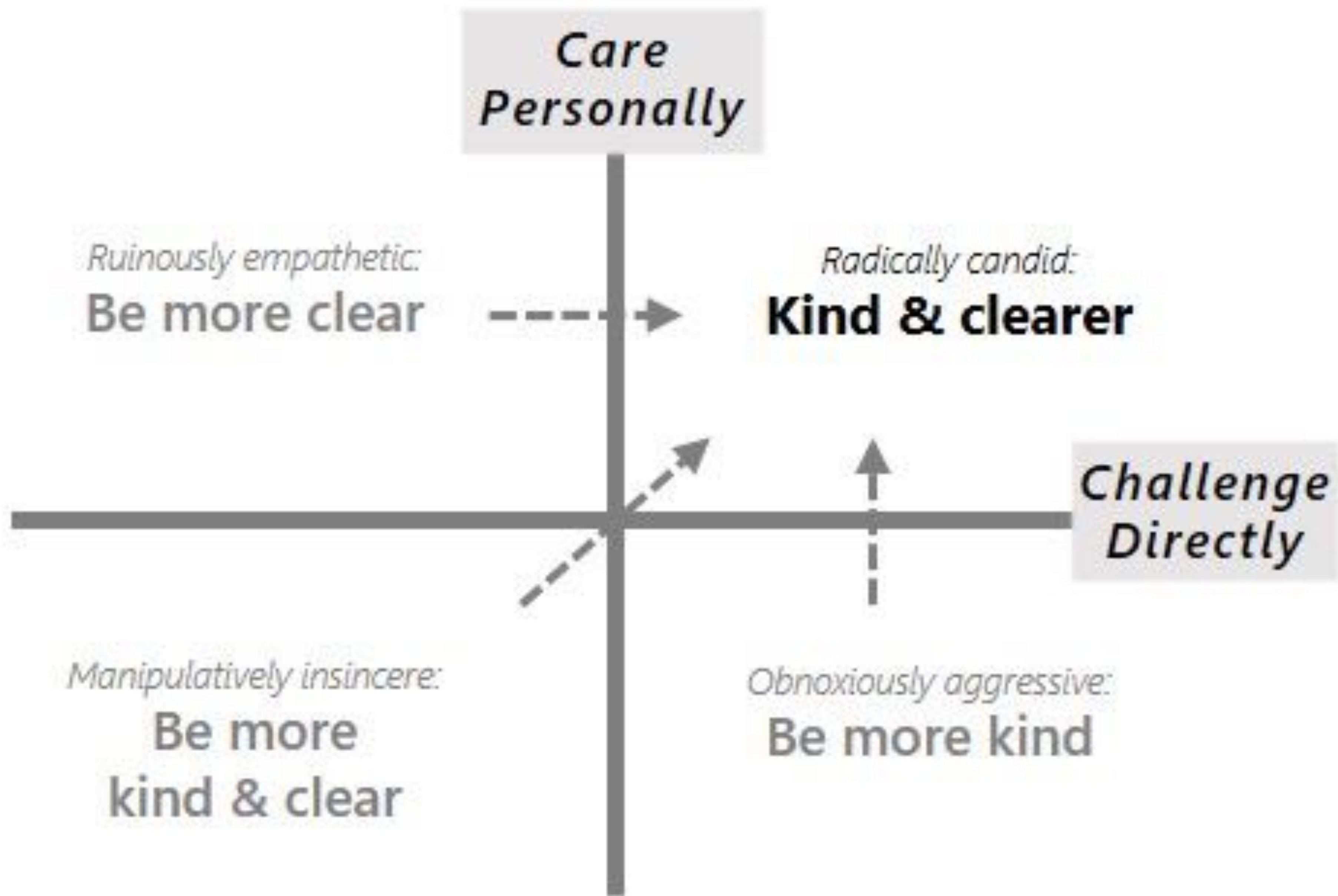
positive

negative

positive

Meaningful Feedback

- An order of magnitude more code than you've ever written
- Complexity scales super-linearly
- Review time already limited



# How the Course Will Be Run in F'20



<https://www.ccs.neu.edu/home/matthias/4500-f20/index.html>

go to my CCIS homepage (google), follow the link