

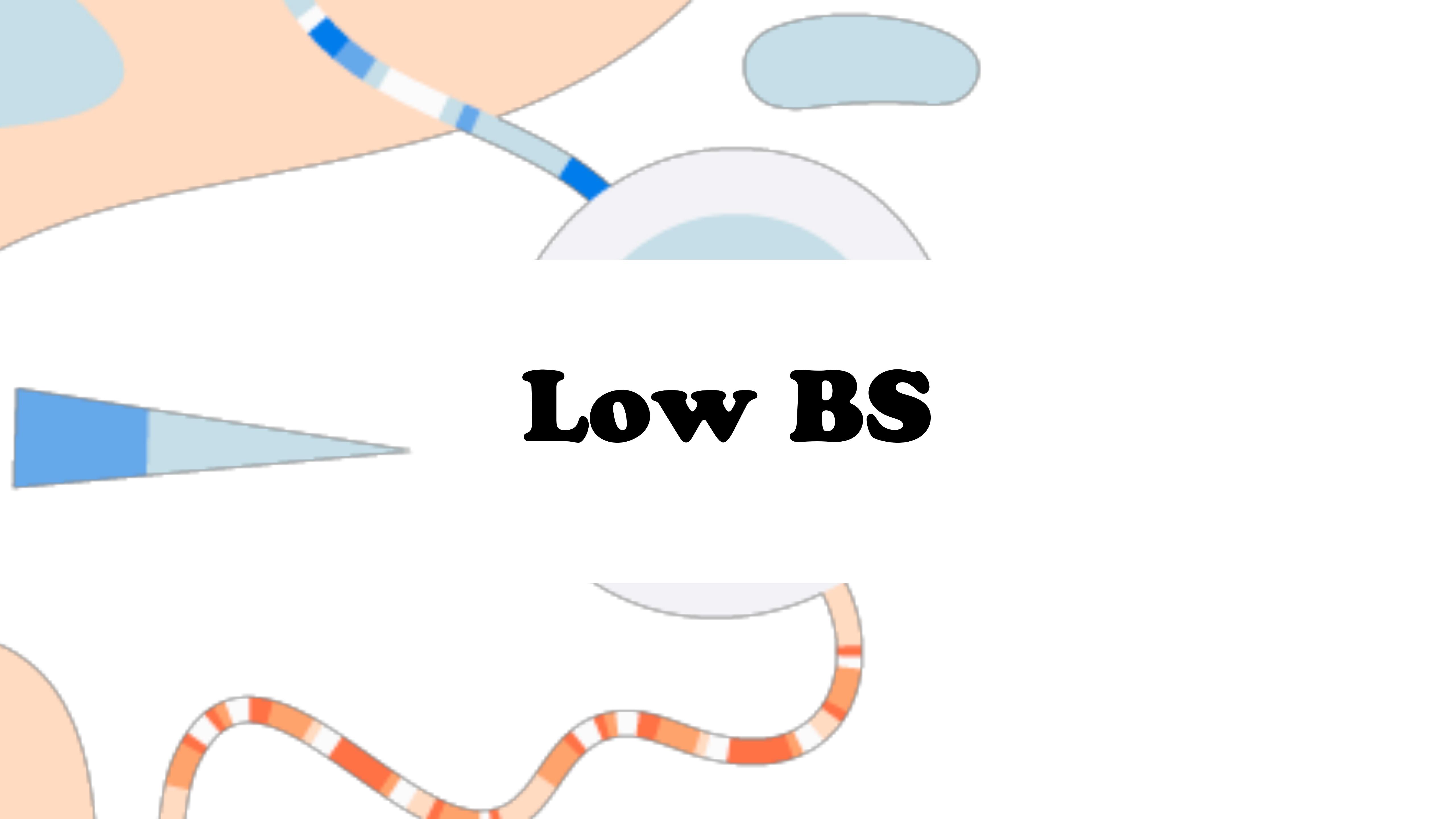
DOMAIN  
DRIVEN  
DESIGN  
EUROPE

# Domain Modelling towards First Principles

@cyriux  
Cyrille Martraire

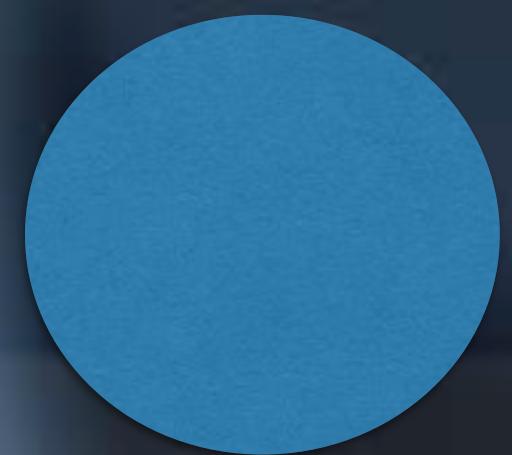
arolla

# **DDD Europe**



**Low BS**

# DIGITAL Transformation



**DDD practitioners: we  
need you for Digital  
Transformations!**

# Passionate developer Deliberate Designer

PARIS

Since 1999

@cyriux

Cyrille Martraire

# Paris Software Crafters

## Community since 2011



<http://www.meetup.com/paris-software-craftsmanship/>



arolla.fr @arollafr

# TDD BDD DDD

[arolla.fr](http://arolla.fr) @arollafr

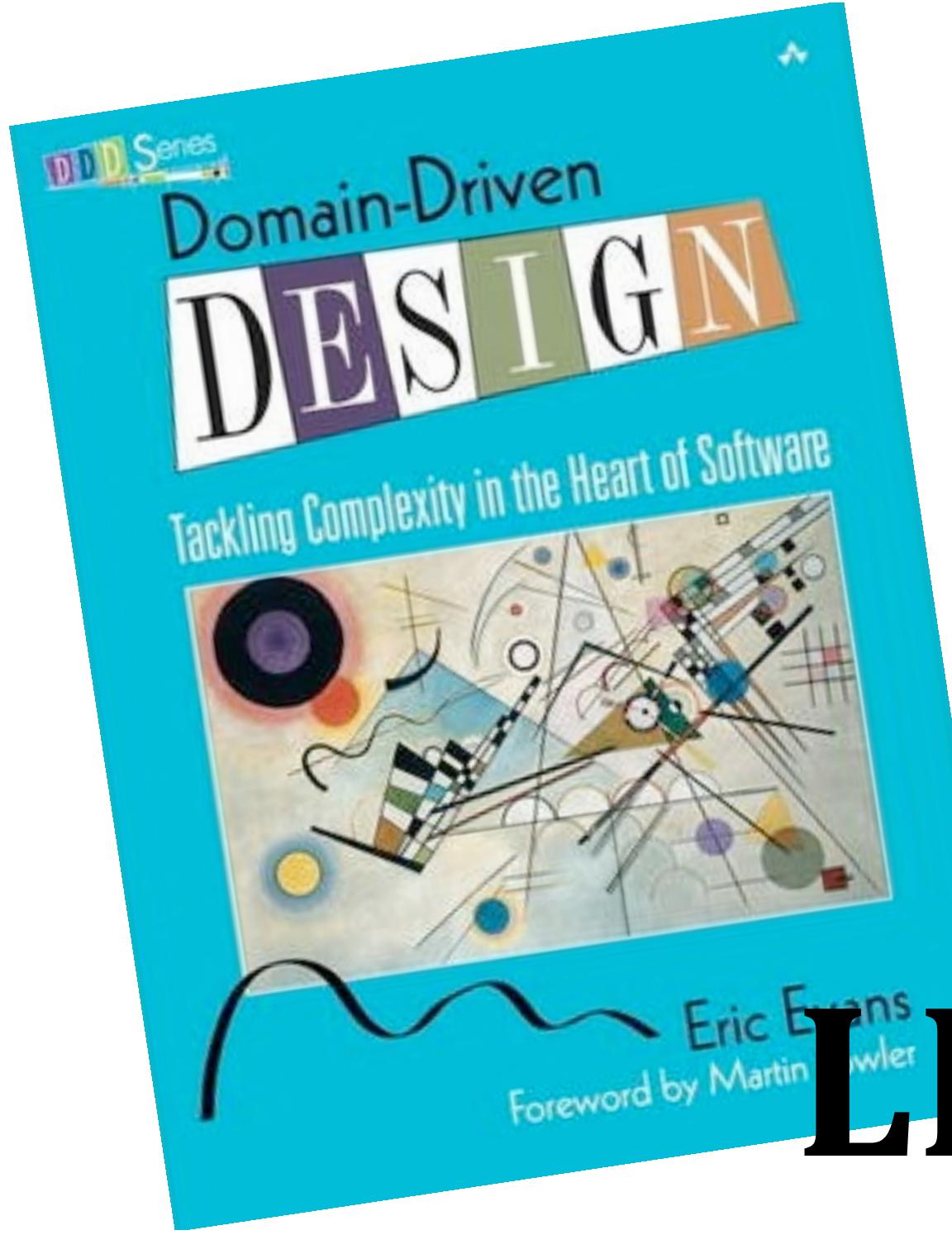
**“The DDD guy from  
Paris”**



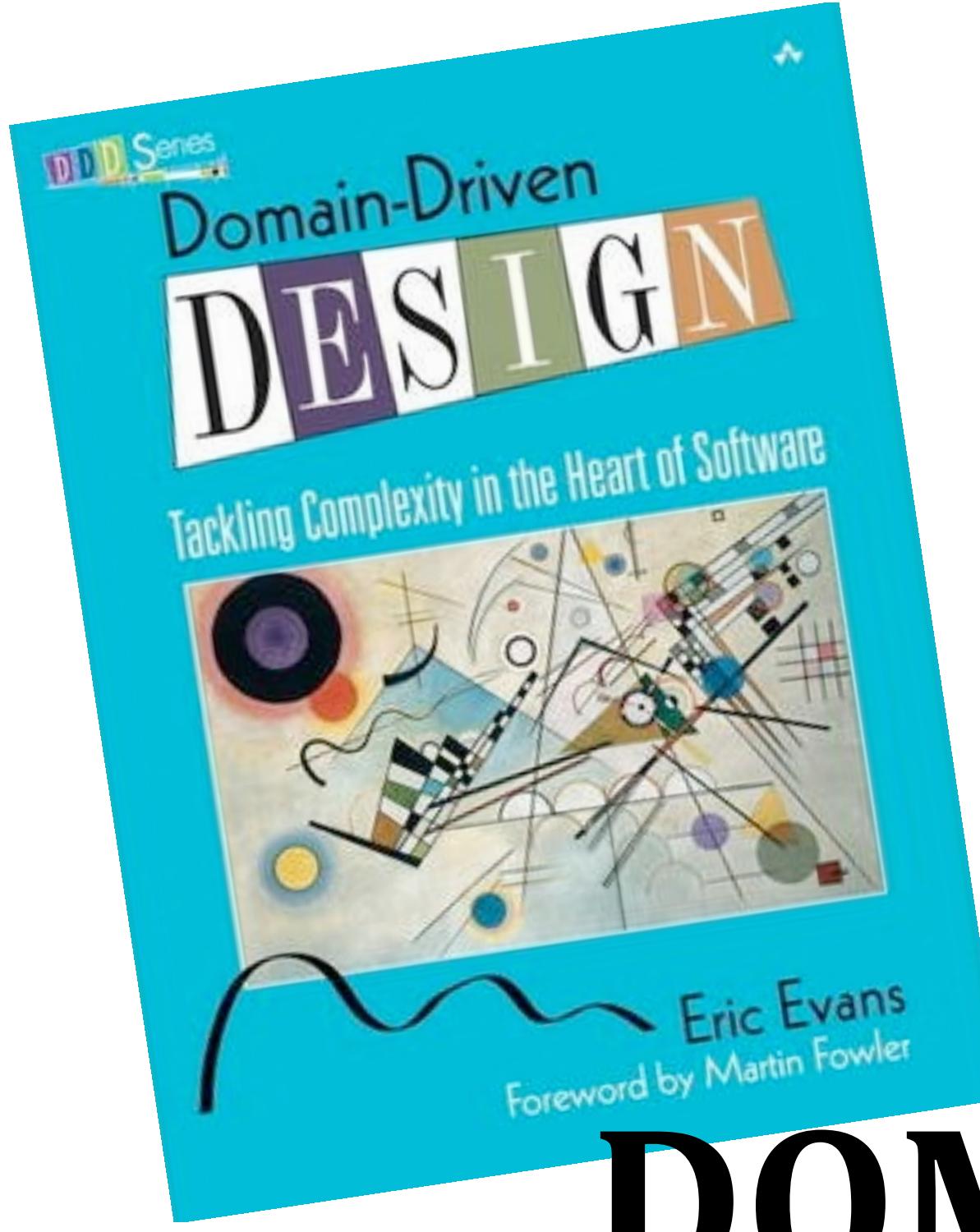
# MY KEY MESSAGE

# **DOMAIN-DRIVEN**

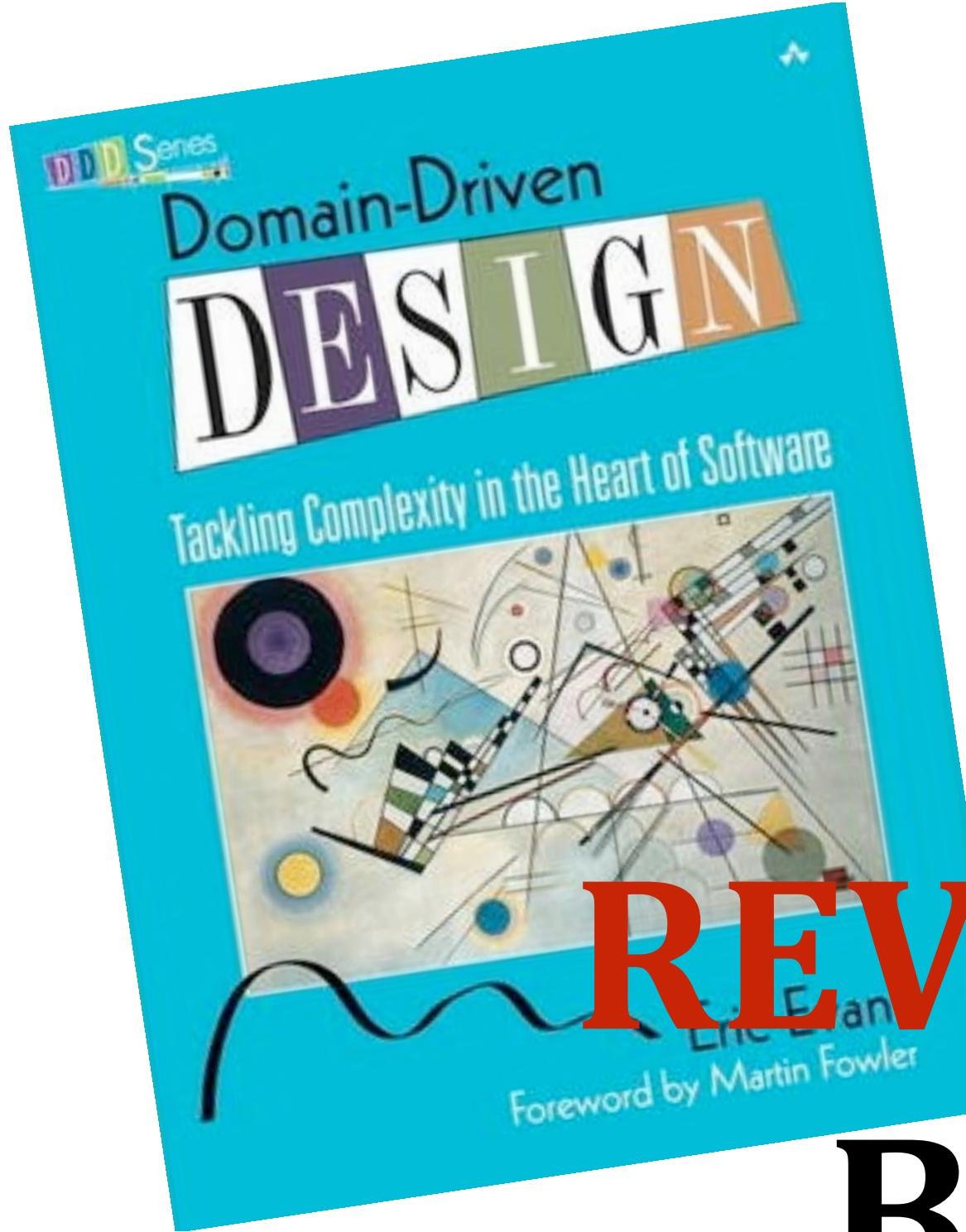
DOMAIN-DRIVEN  
too gentle



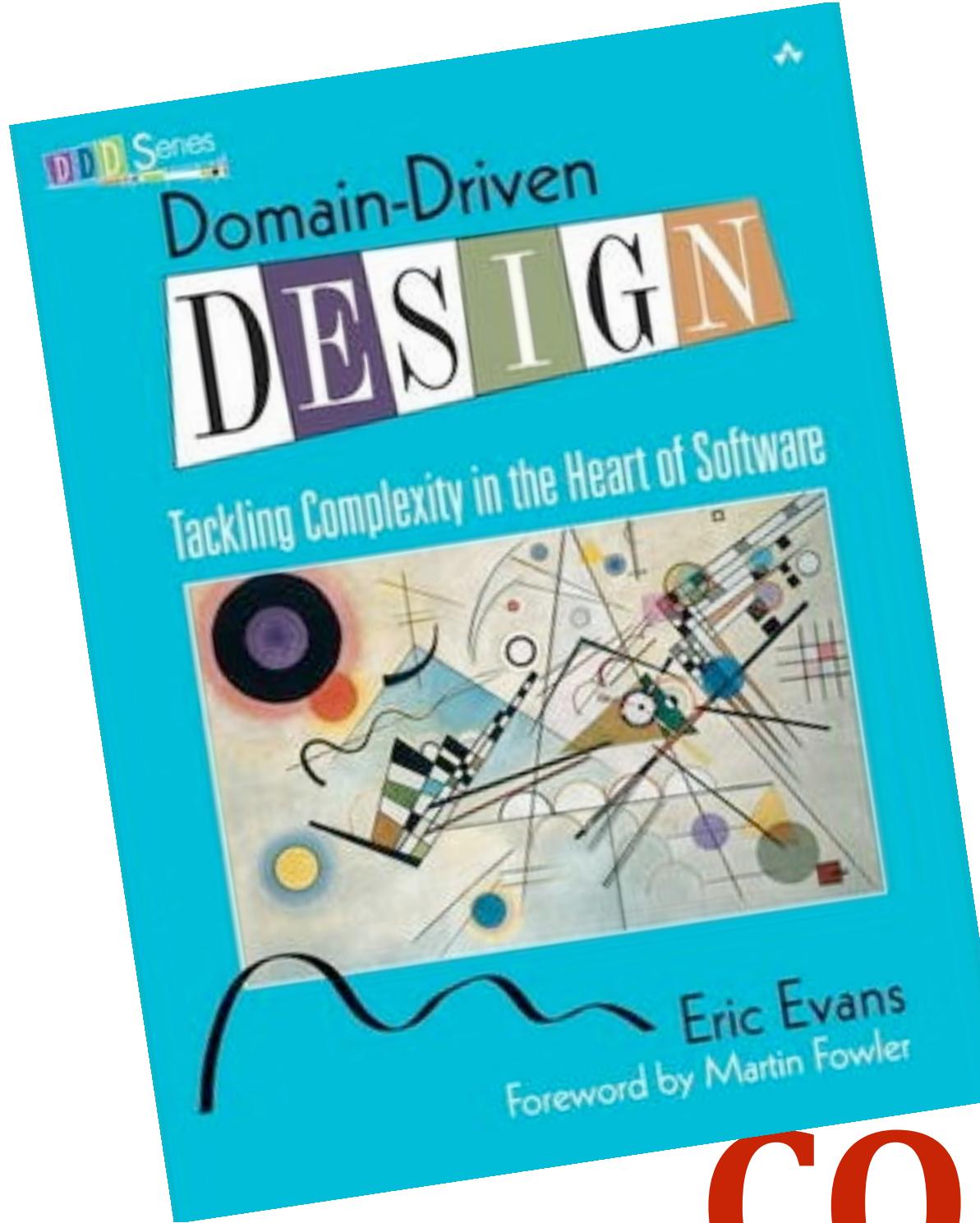
# LEARN THE DOMAIN



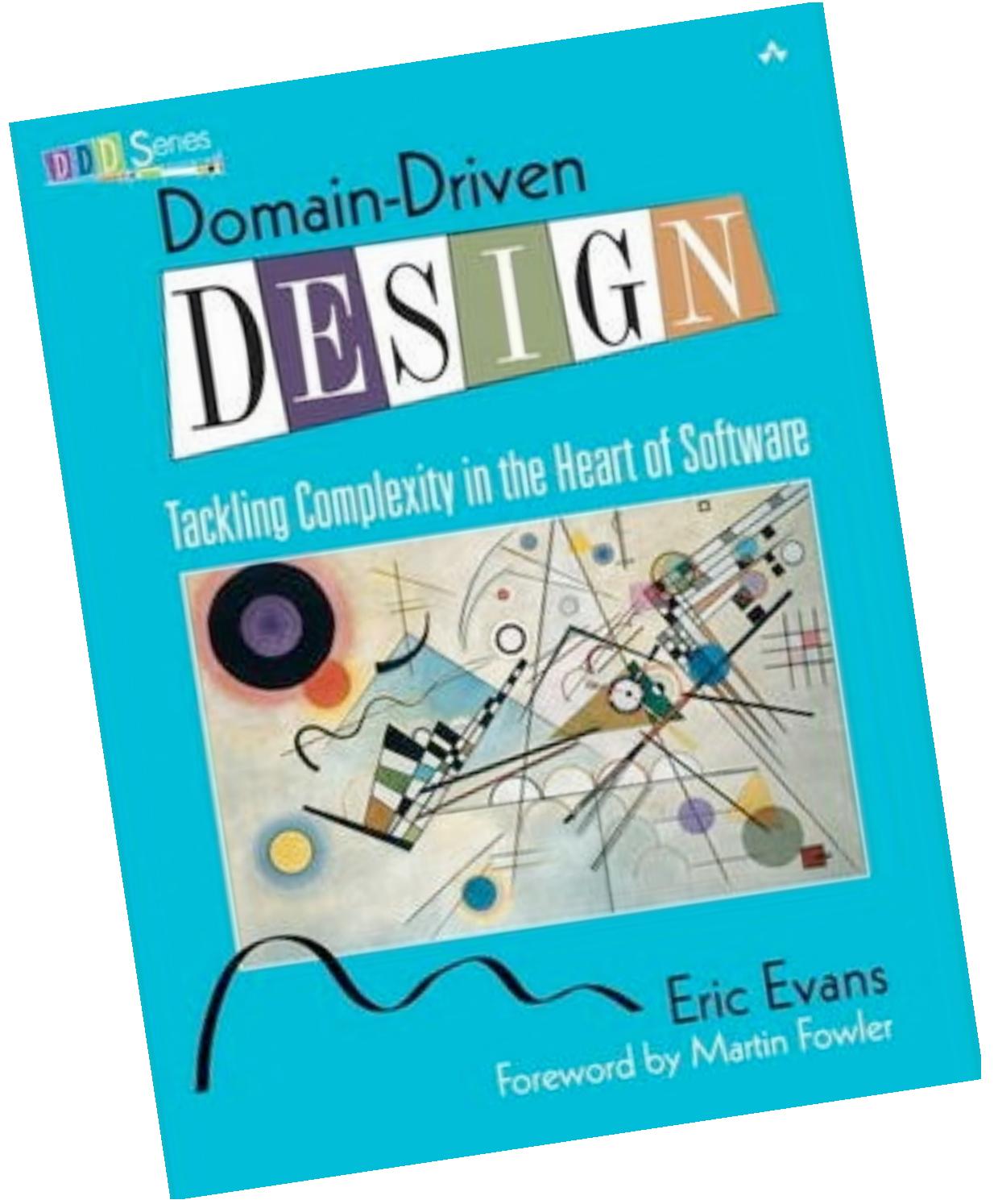
**USE YOUR  
DOMAIN-DRIVEN SKILLS**



# **REVERSE-ENGINEER THE BUSINESS DOMAINS**

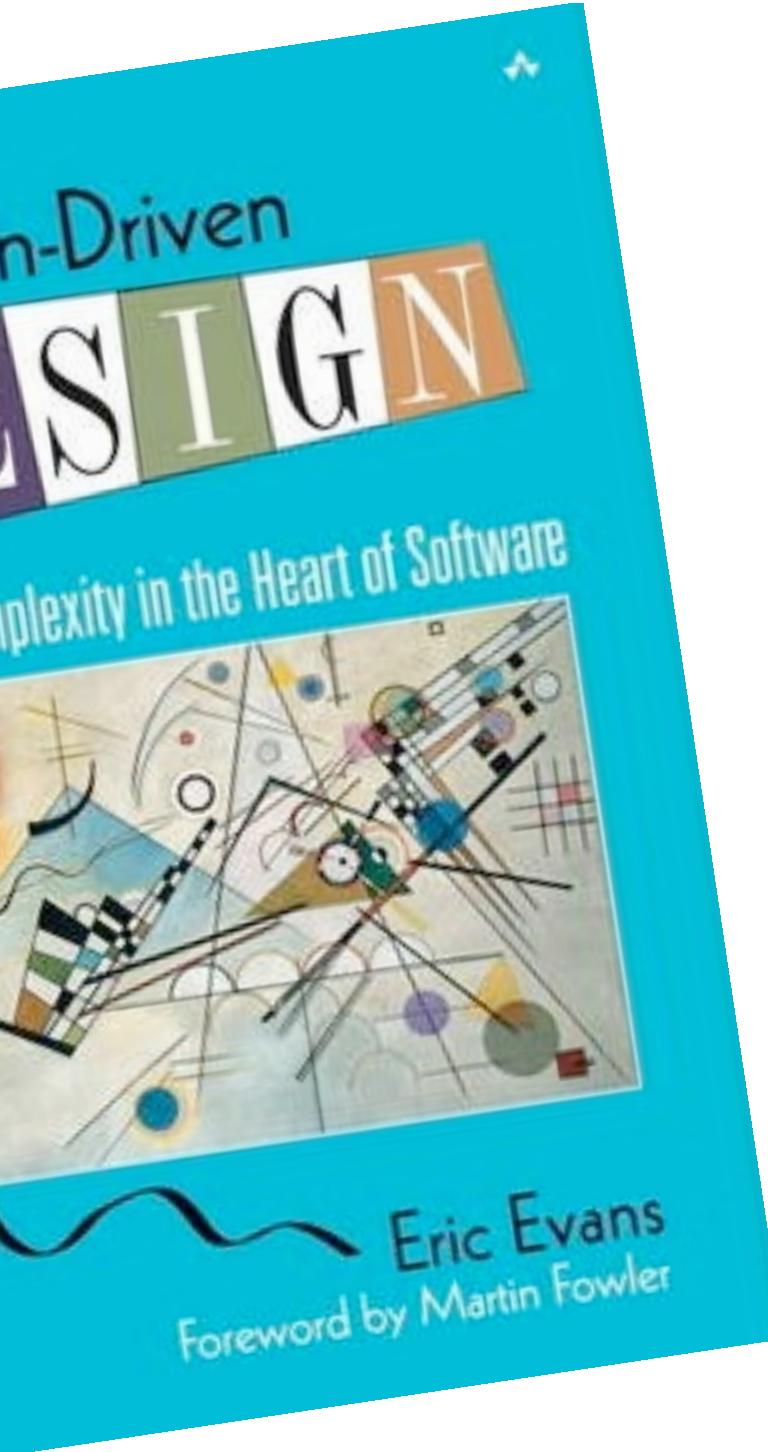


# INTO USEFUL CONCEPTUAL MODELS



ALL THIS

too gentle



# CONCEPTUALISE A DOMAIN MODEL AS A THEORY

**SPOT THE  
FIRST PRINCIPLES  
BEHIND THE THEORY**

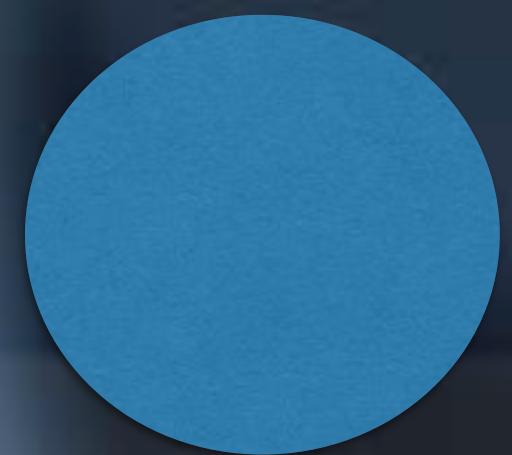
**CHALLENGE THE  
FIRST PRINCIPLES**

**TO SUGGEST CHANGES TO  
THE BUSINESS PROCESSES**

**TO TURN THEM TO  
DIGITAL-NATIVE BUSINESSES**

**TO EMBRACE**

# DIGITAL Transformation



# DIGITAL TRANSFORMATION

TOLD YOU SO!

why this talk?

**SERIOUSLY**

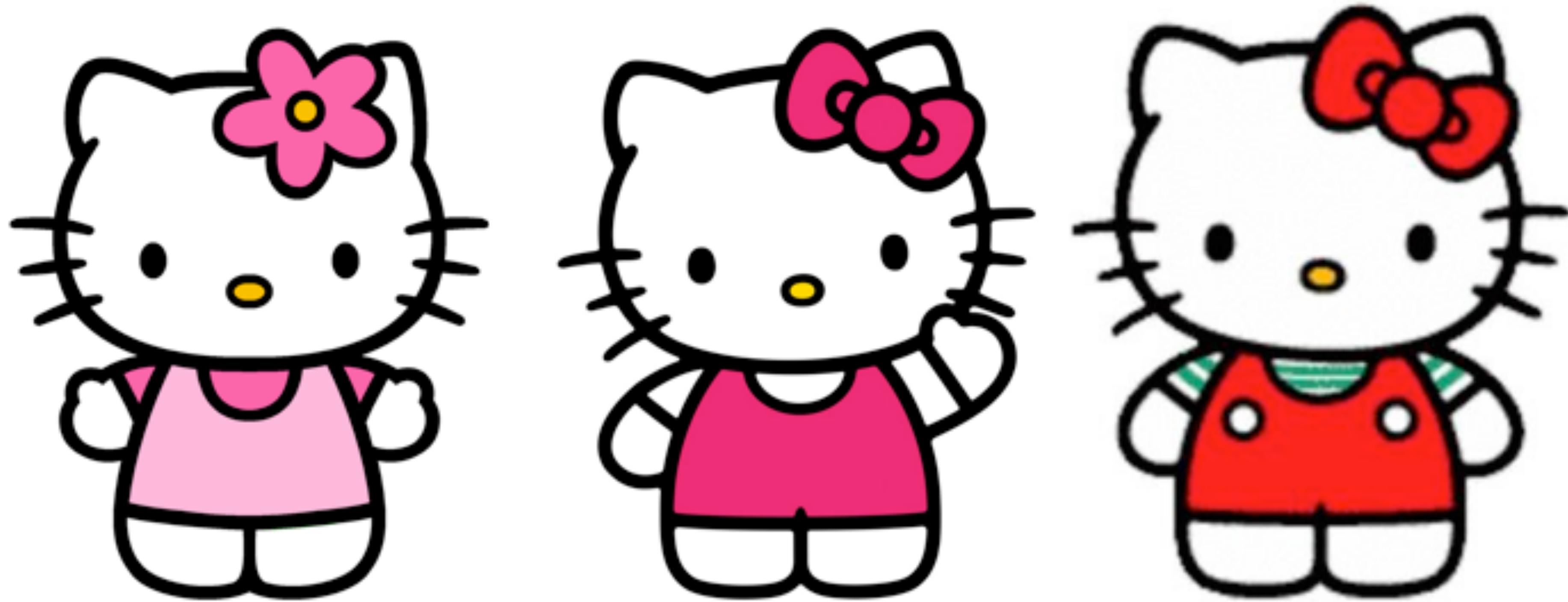


**I'M FED UP**

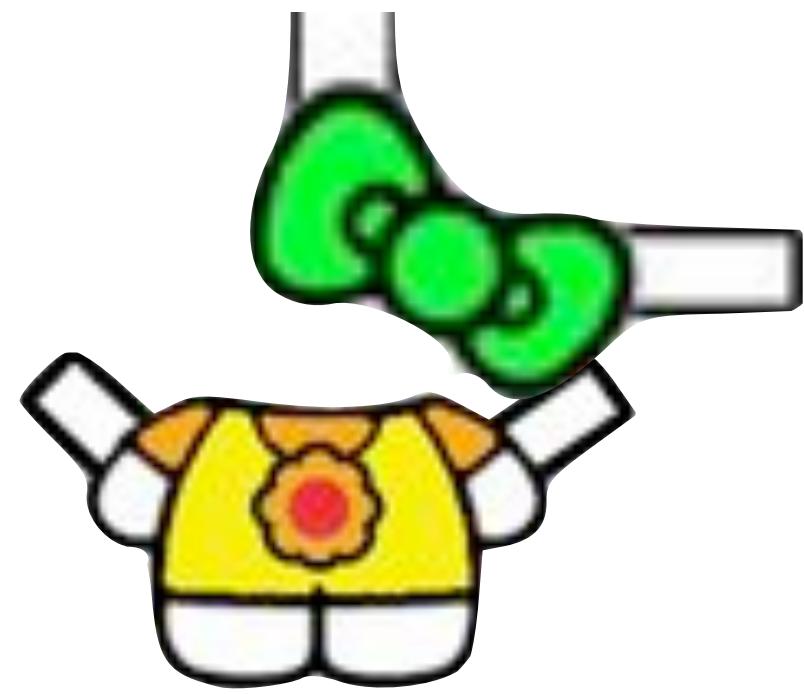
*Memes Happen*

**We are given work**

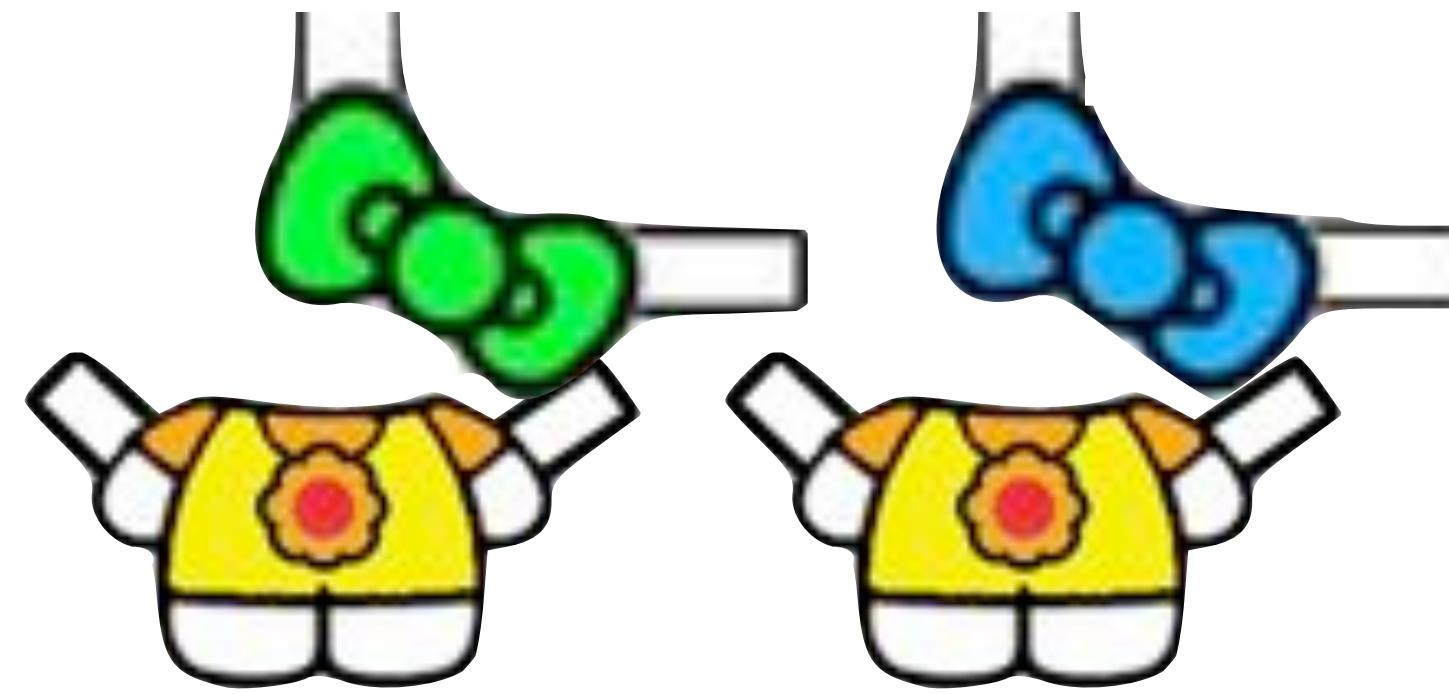
# Case by Case



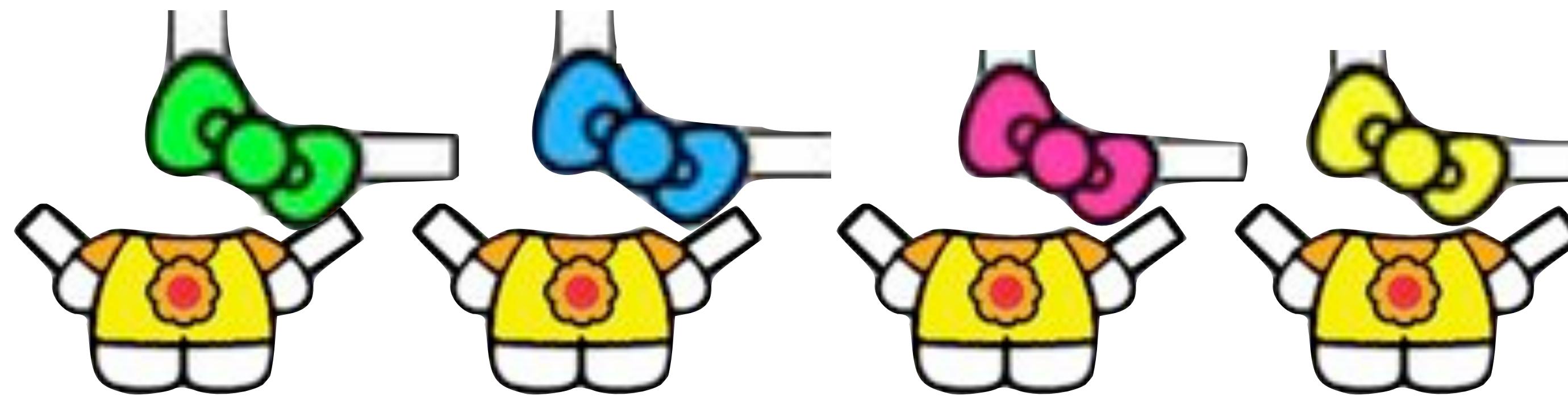
# One User Story



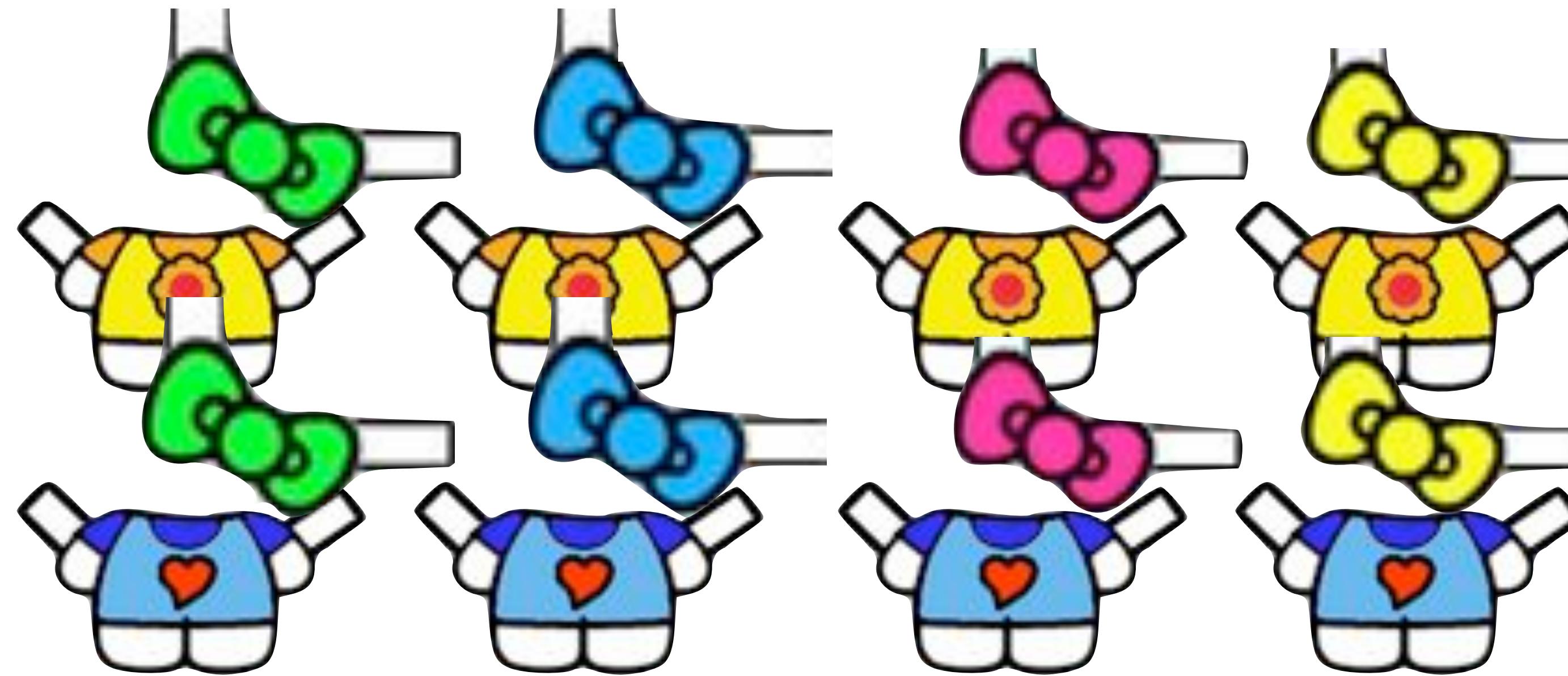
# Another User Story



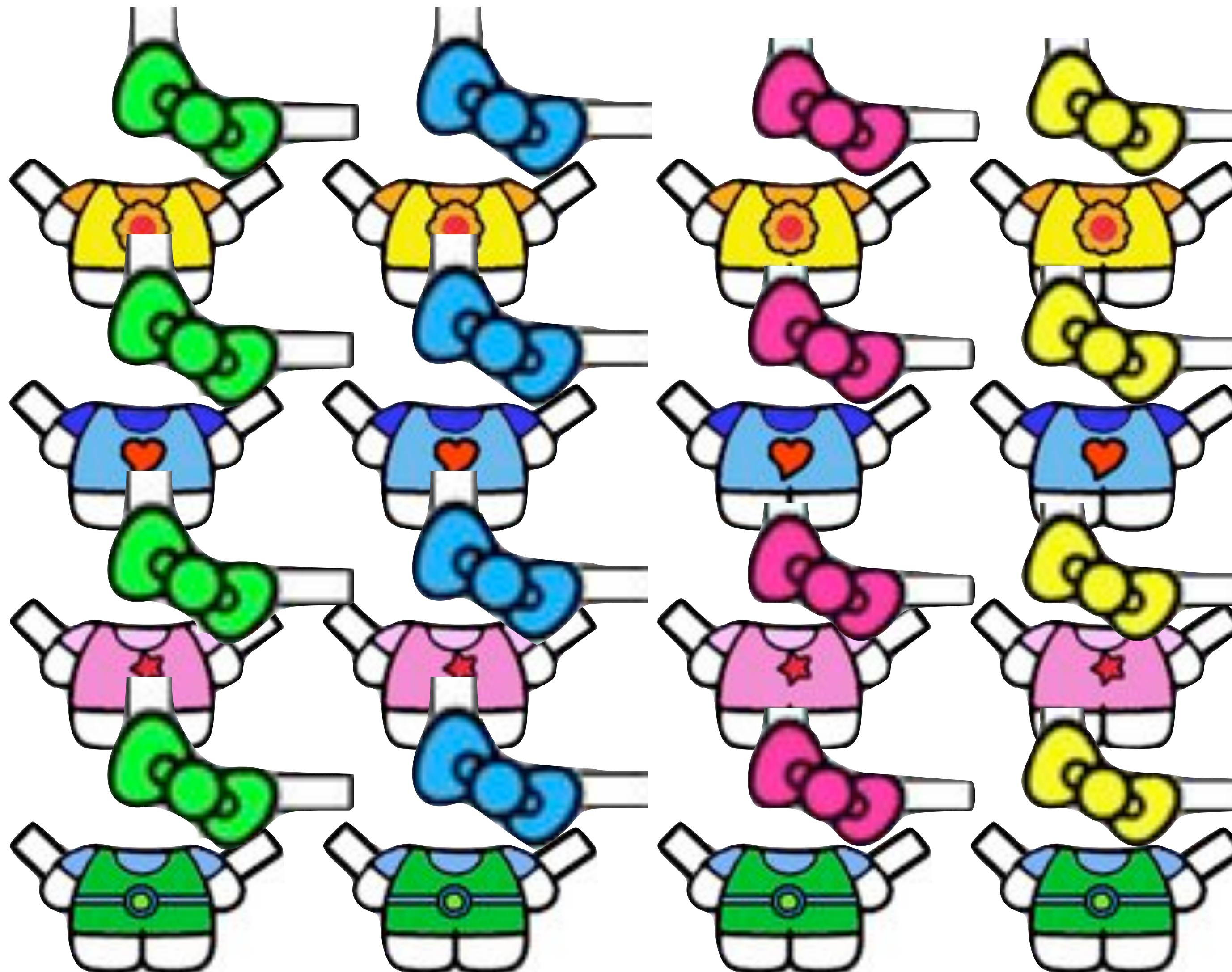
# Some Other User Stories



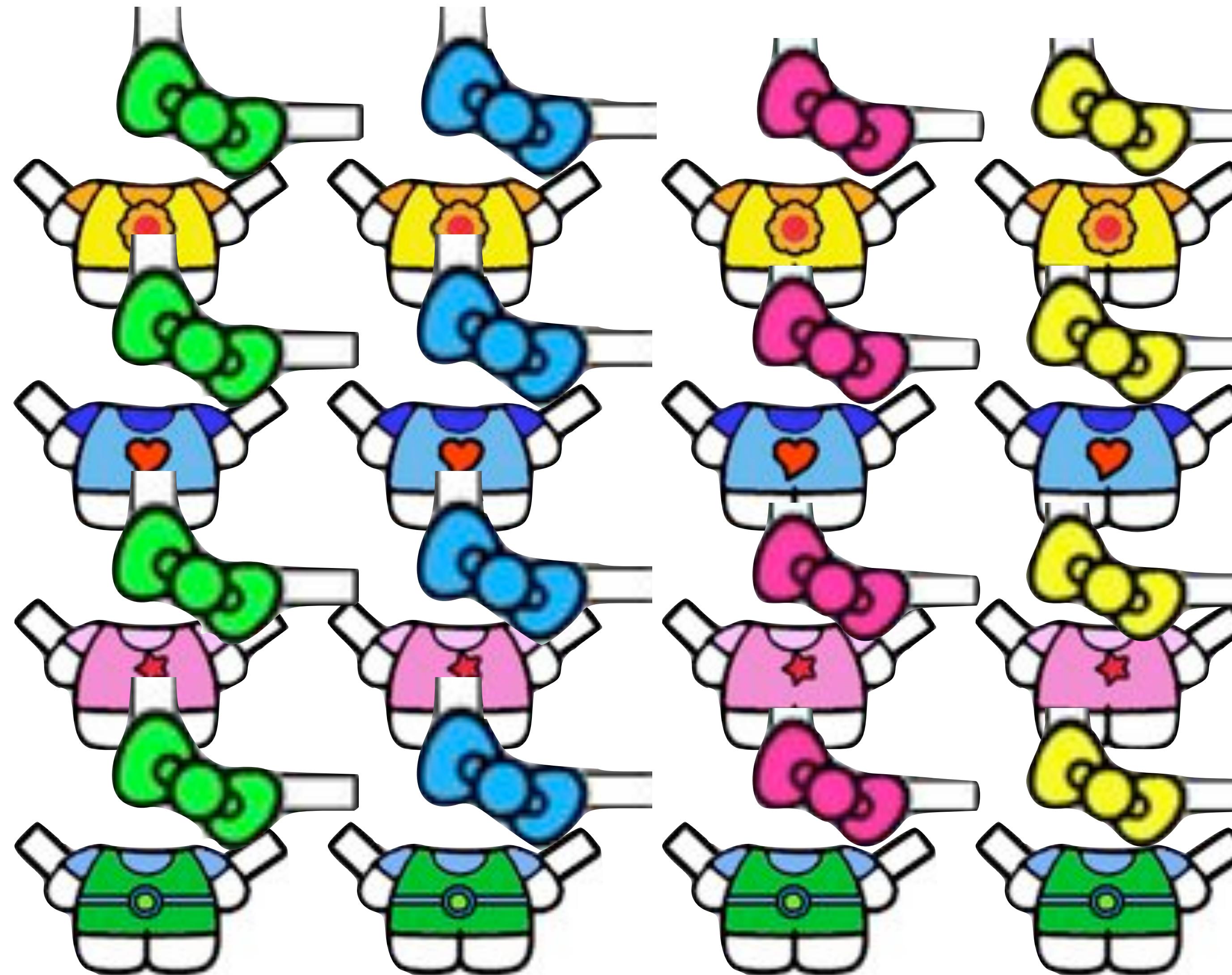
# Even More User Stories



# A LOT of User Stories...

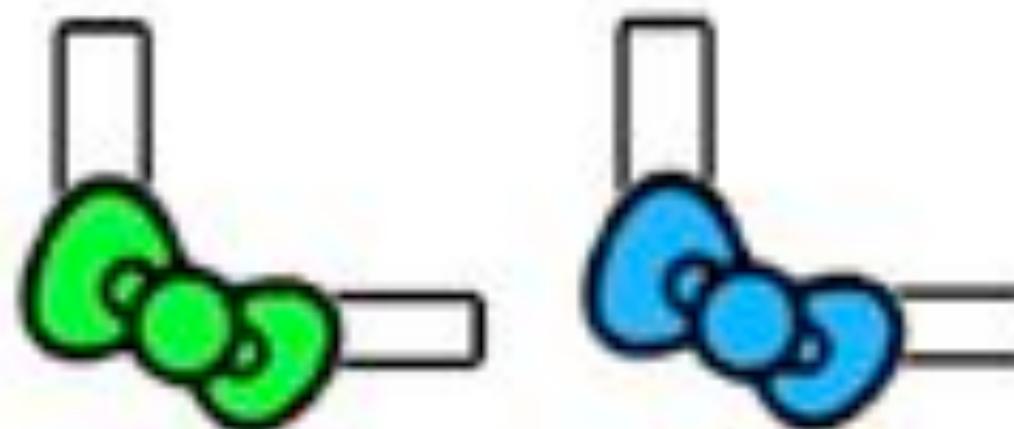
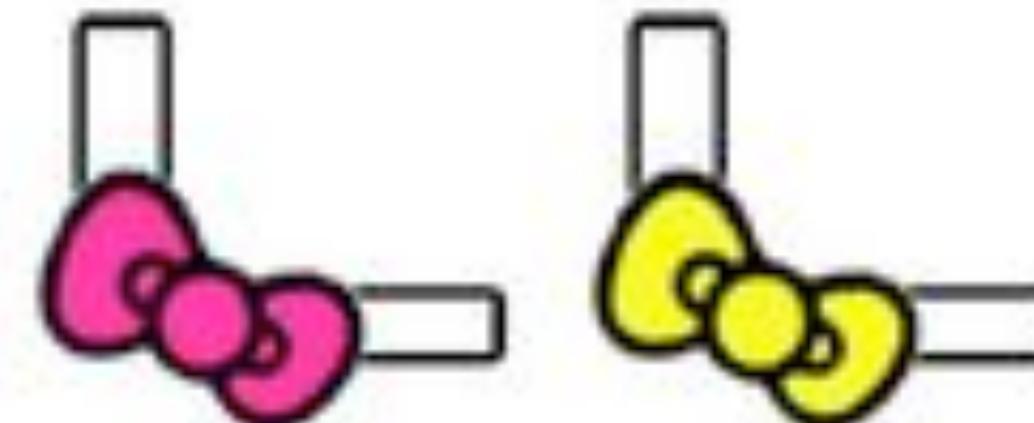


# ... all Special Cases of a General Case!



$$= \begin{matrix} & \text{Character 1} & \text{Character 2} \\ \text{Character 1} & & \\ & \text{Character 2} & \end{matrix} \times \begin{matrix} & \text{Character 3} & \text{Character 4} \\ \text{Character 3} & & \\ & \text{Character 4} & \end{matrix}$$

# What you have in mind



Ribbon

X



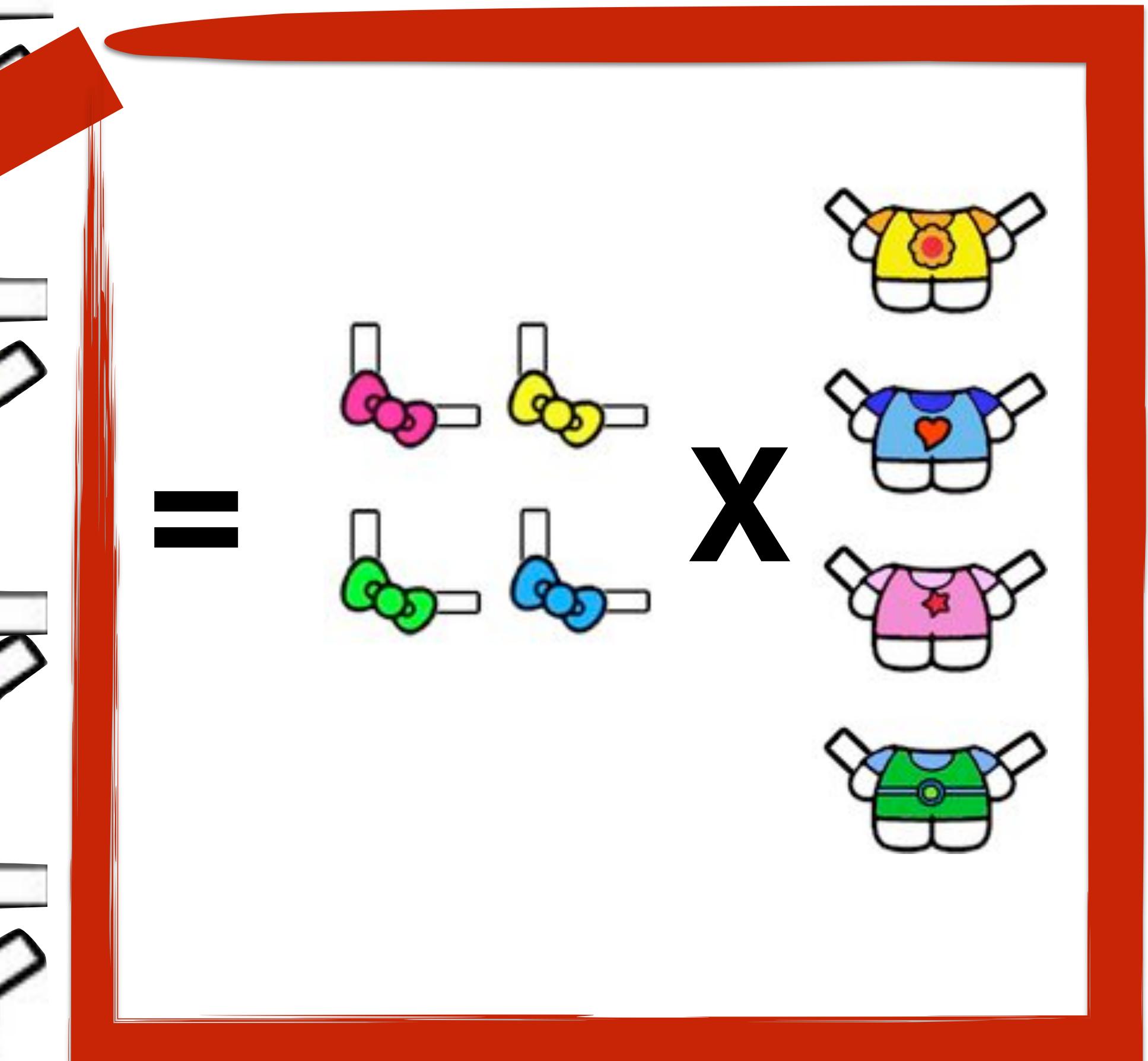
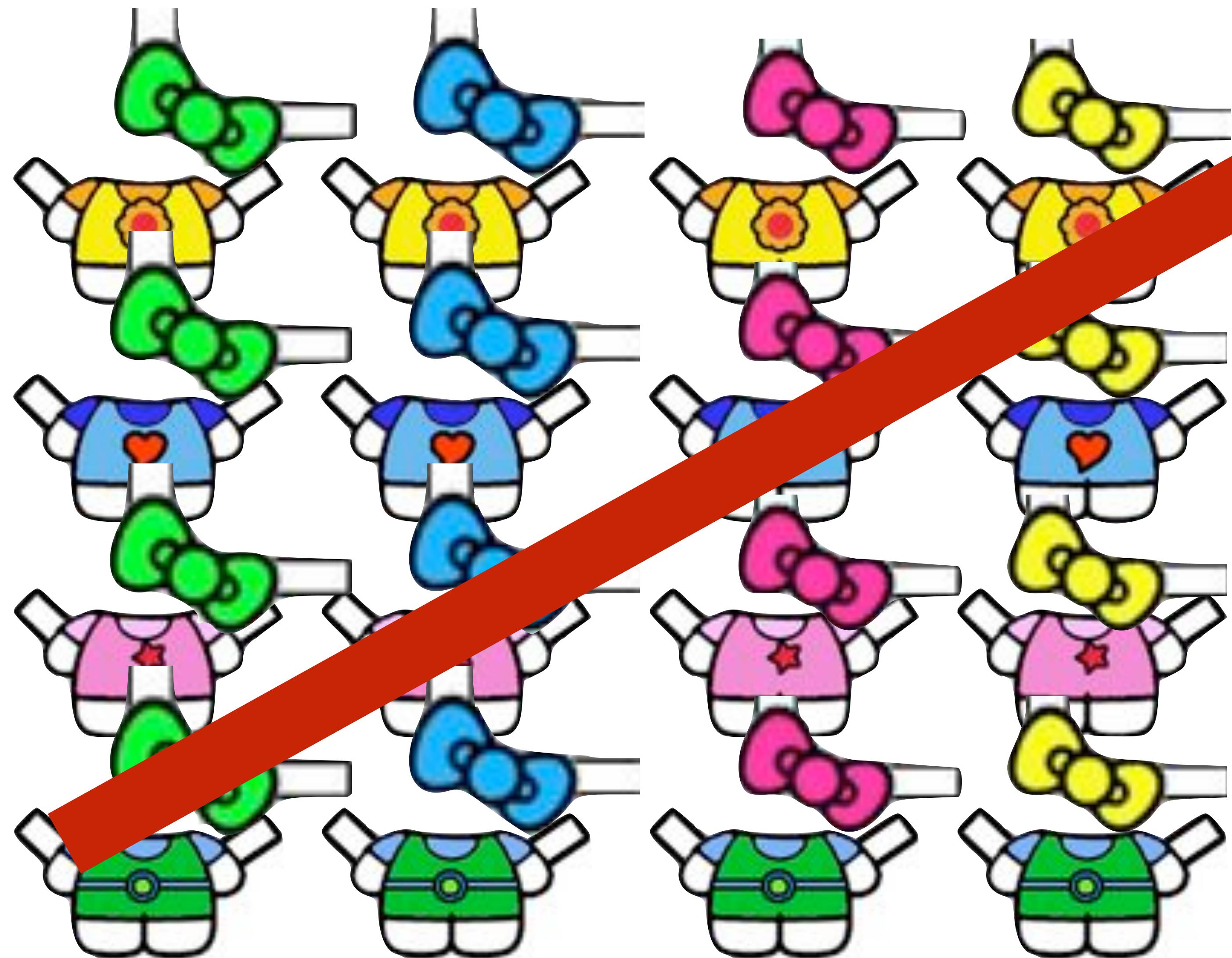
Costume

**Anti-pattern**

# **HUMAN COMPILER**

**Please describe the problem, not  
its multiple consequence**

# Please give me the general case!



**This is a job for machines,  
not for Developers**

**WHY?**

# SOLUTION ENVY

Everyone wants to be the problem  
'solver', even when it's not their skills

# LACK OF TRUST

"They're not able to understand"

# BUSINESS RULES FOR **DUMMIES**

**IDIOT-PROOF SPECS,  
PERFECT FOR DEVELOPERS**

*A Reference for the Rest of Us!*™



# **INTRODUCING THE WATERLINE**

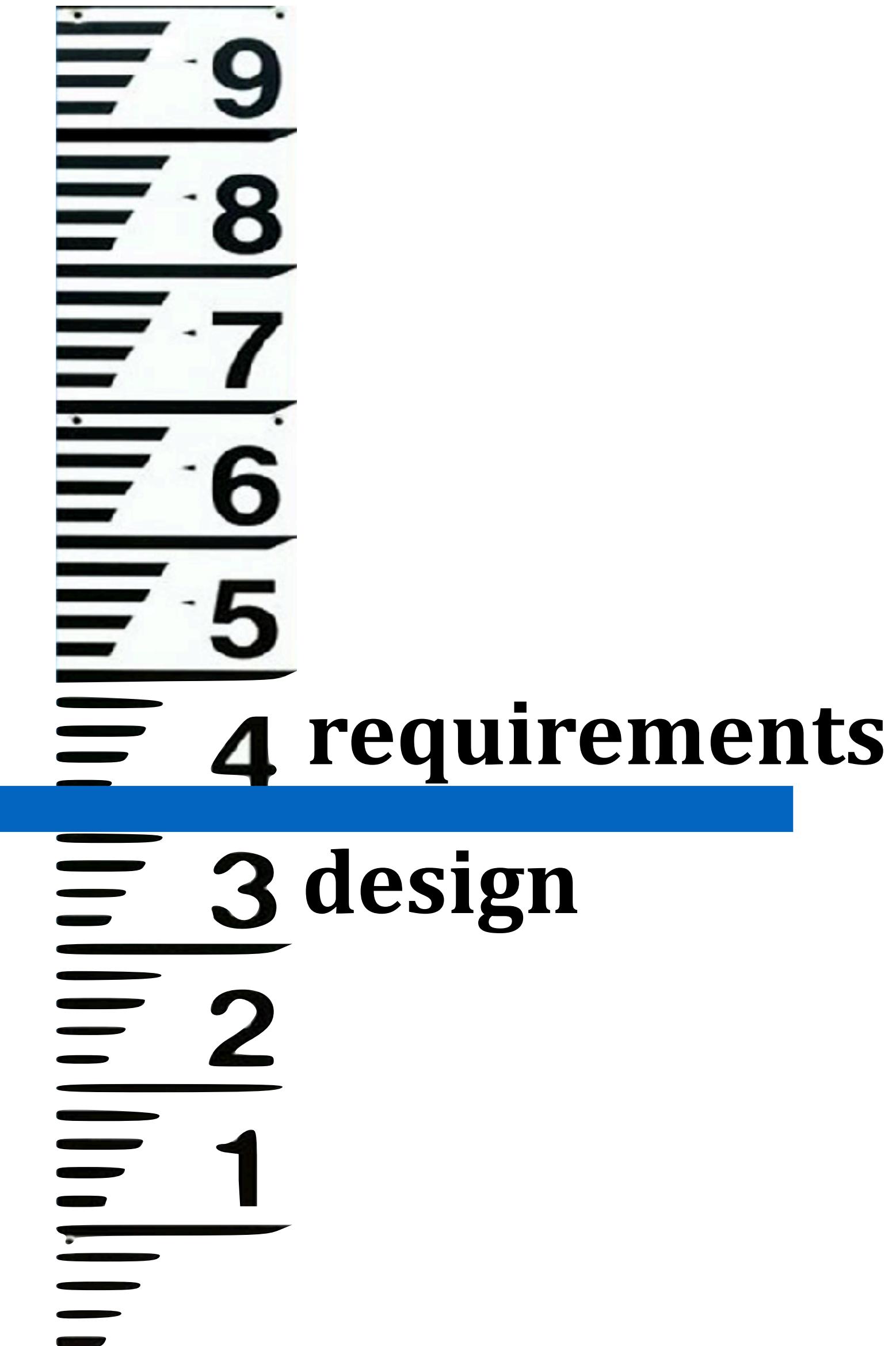
# Design vs Requirement

”

*If it's your decision to make,  
then it's 'design'; if it's not,  
then it is a requirement*

**— Alistair Cockburn**

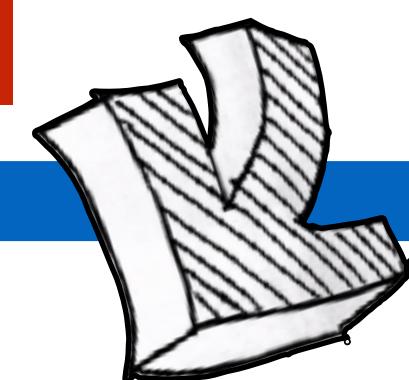
# The design waterline



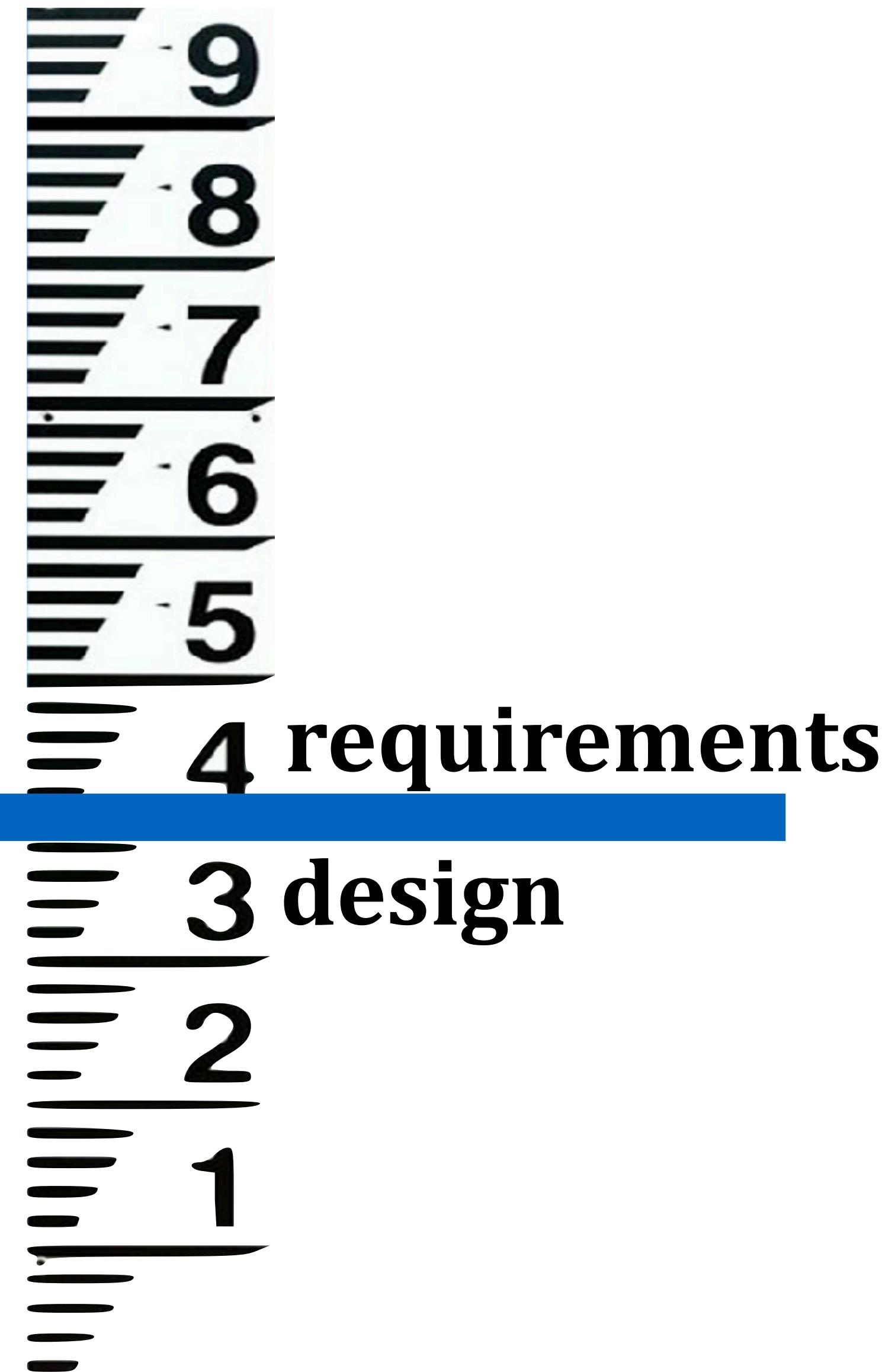
# What happens too often

HOW BUSINESS  
PEOPLE SPECIFY

LABORIOUS  
CODE



current  
waterline



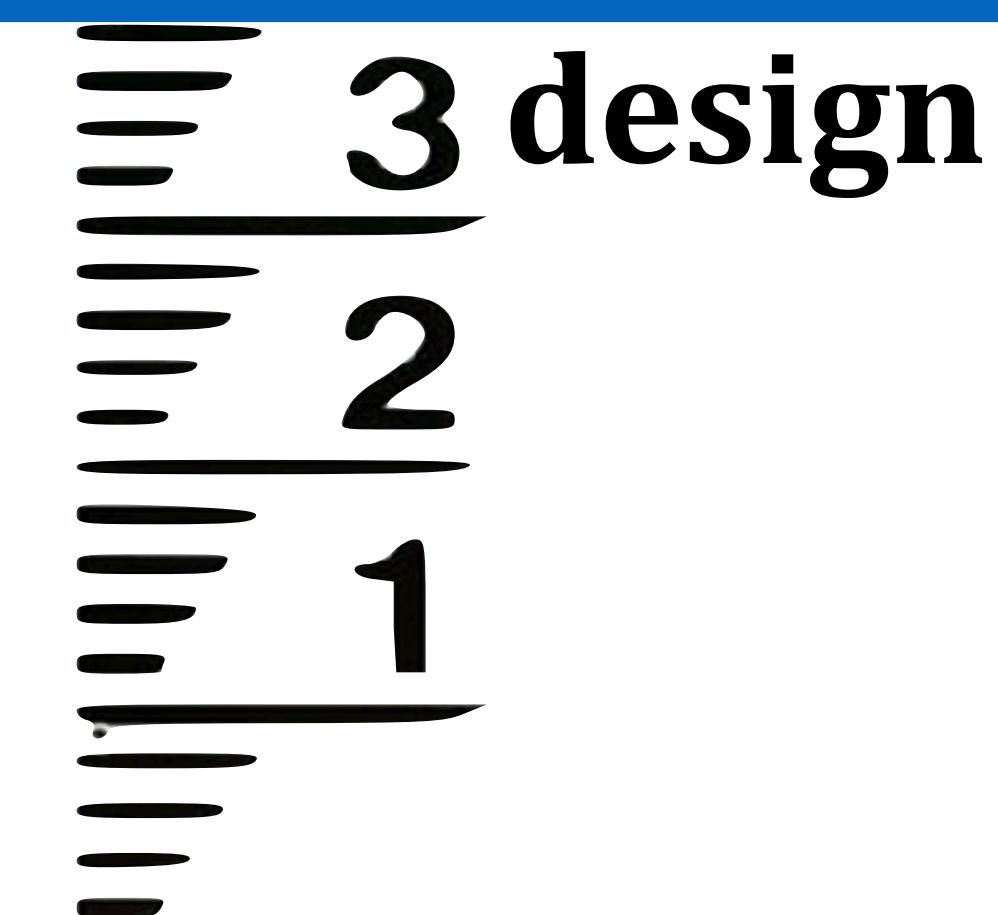
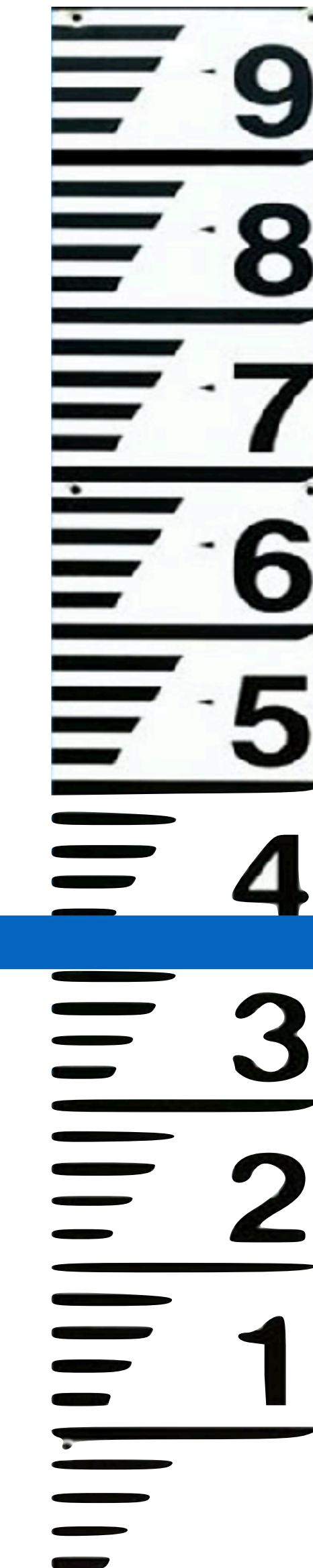
# Accidental Complexity (Biz people)

HOW BUSINESS  
PEOPLE THINK

HOW BUSINESS  
PEOPLE SPECIFY



current  
waterline



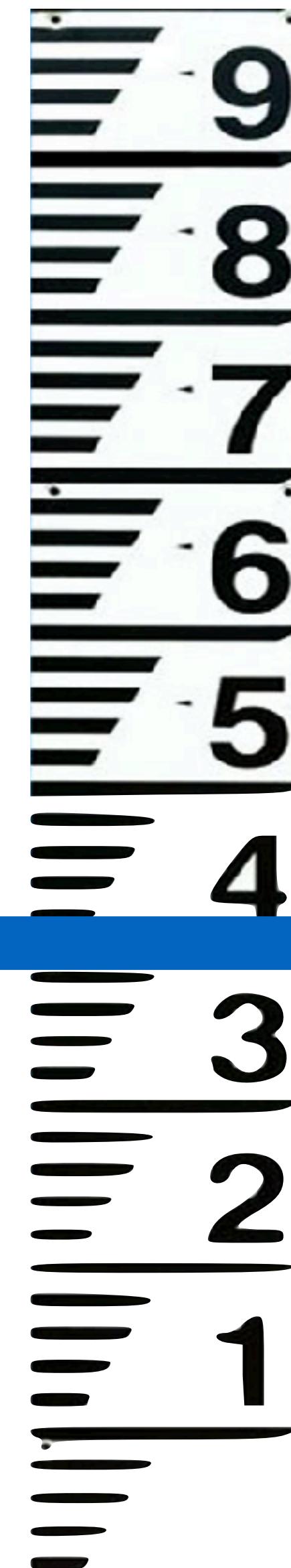
# Accidental Complexity (Biz people)

ONE GENERAL CASE

MANY CONSEQUENCES



current  
waterline

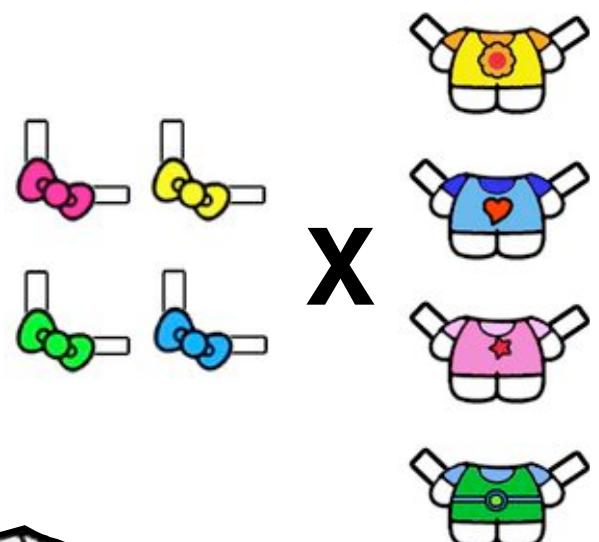


4 requirements

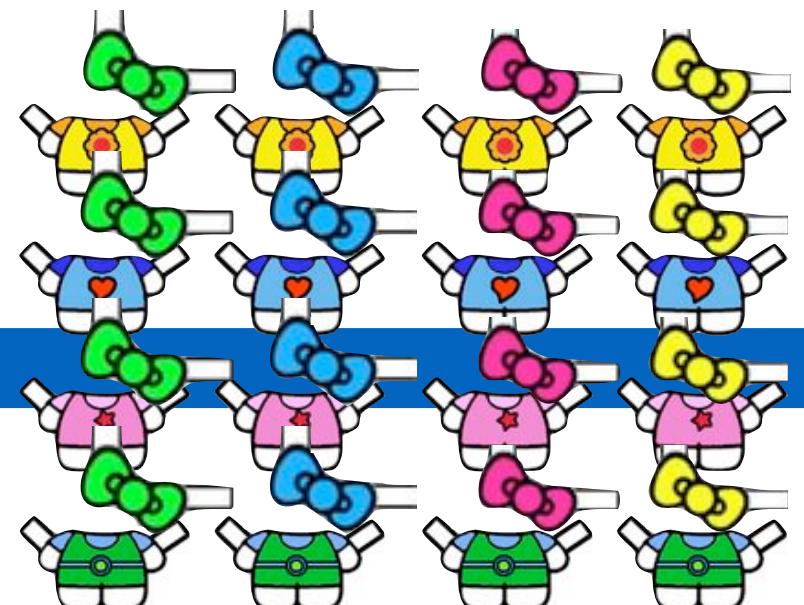
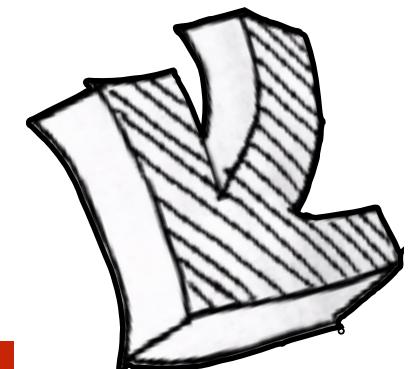
3 design

# Accidental Complexity (Biz people)

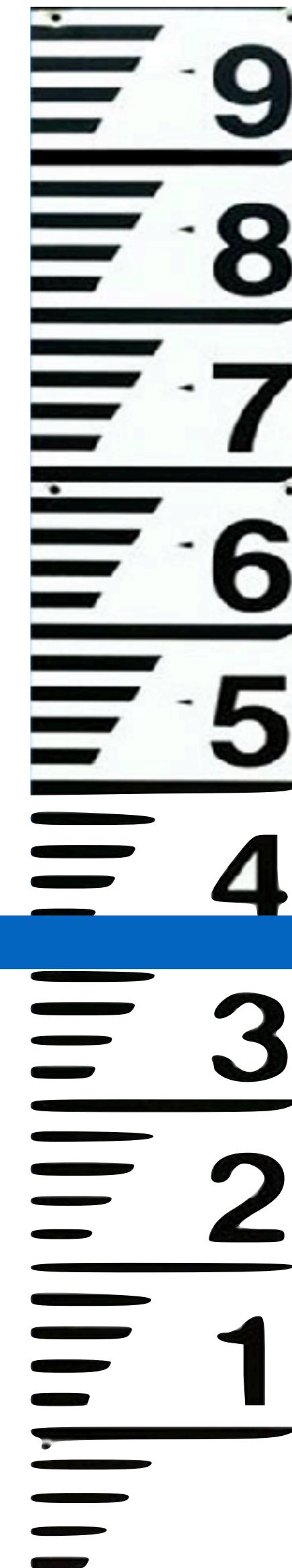
ONE GENERAL CASE



MANY CONSEQUENCES



current waterline

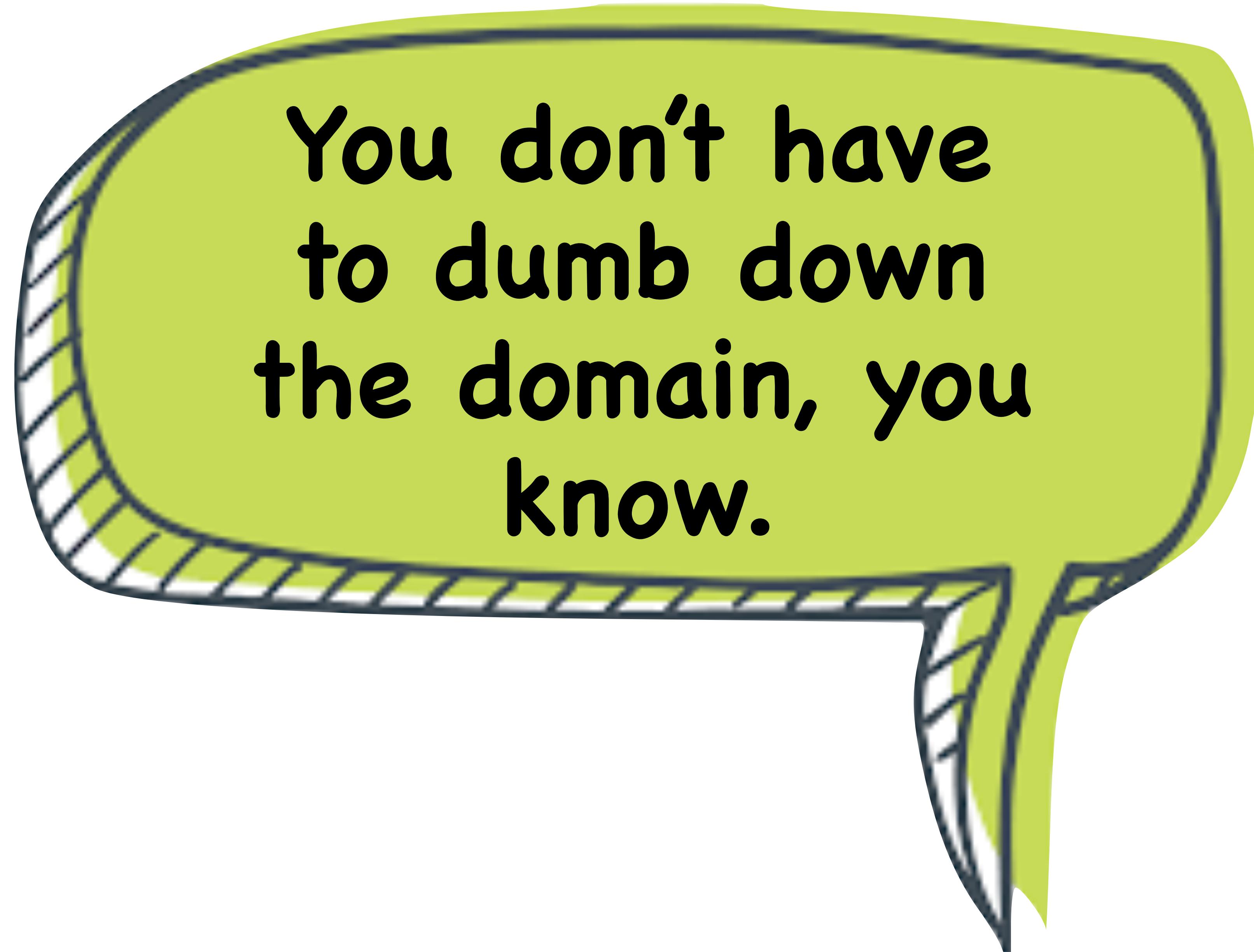


4 requirements

3 design

**RAISE THE  
WATERLINE**

**RAISE THE  
WATERLINE  
(mutually)**



You don't have  
to dumb down  
the domain, you  
know.

# Mutual Higher-Level Communication

HOW BUSINESS  
PEOPLE SPECIFY

SMARTER  
CODE

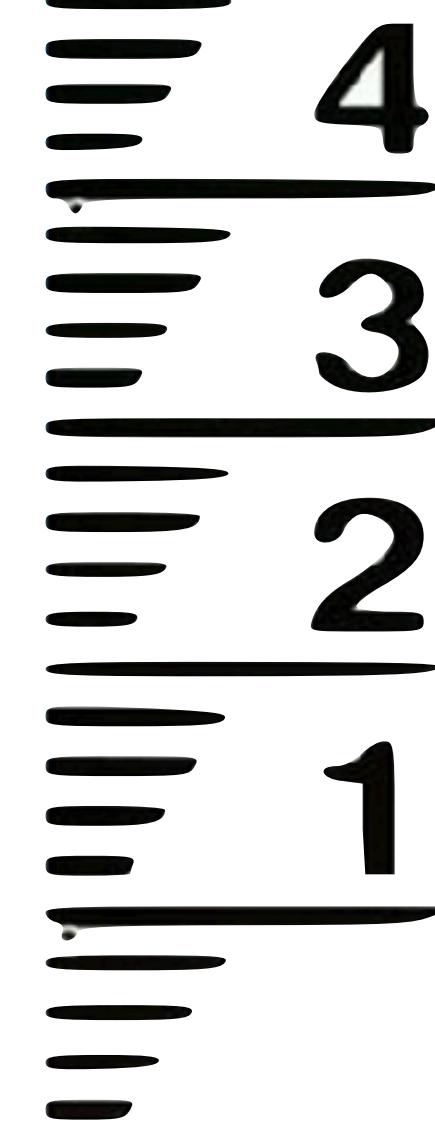
new  
waterline



requirements



design



**More generally...**

# **SOLUTION STATEMENT**

The expressed need is frequently a solution dressed up as a problem

**Challenging  
the expressed "needs"**

**How do I do that?**

**POWERFUL  
QUESTIONS**



# Facilitation Questions Cards

WHEN WOULD  
THIS PROPERTY  
**NOT BE  
TRUE**

SOUNDS LIKE A  
**SOLUTION**  
DRESSED UP AS A  
**PROBLEM**

@arollafr

arolla

arolla.fr

@arollafr

arolla

arolla.fr

→ in your goodies bags

**SERIOUSLY**



**I'M FED UP**

*Memes Happen*

**Sometime we developers  
become human  
compilers too...**

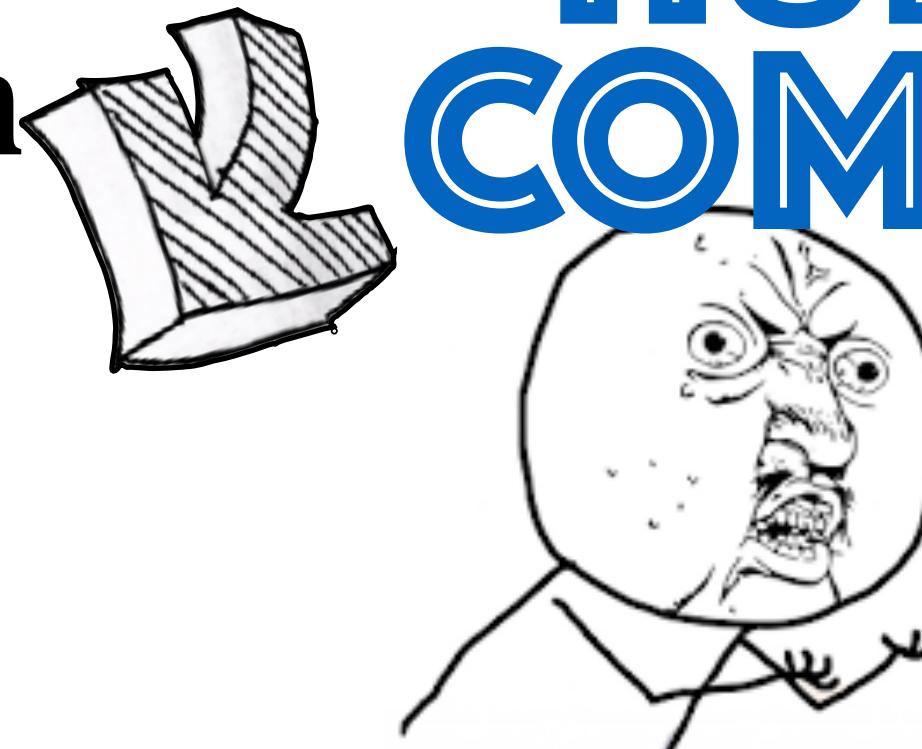
# BTW: Accidental Complexity (devs)

HOW BUSINESS  
PEOPLE SPECIFY

deductive  
implementation

LABORIOUS  
CODE

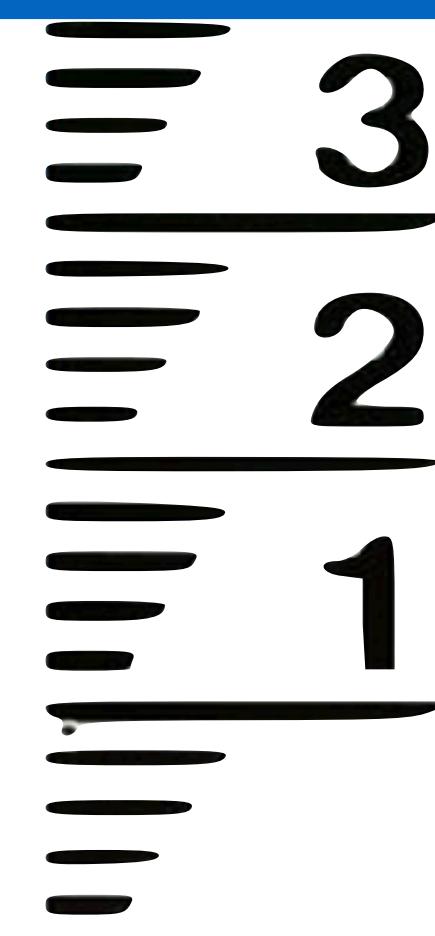
HUMAN  
COMPILER



waterline



4 requirements



3 design

```
If Val(Left(sFormNumber, 2)) <> 16 And Val(Left(sFormNumber, 2)) <> 99 Then  
    '*****Start*****  
    If Val(Left(sFormNumber, 2)) = 10 Or Val(Left(sFormNumber, 2)) = 11 Or _  
        Val(Left(sFormNumber, 2)) = 12 Or Val(Left(sFormNumber, 2)) = 13 Or Val(Left(sFormNumber, 2)) = 17 Or Val(Left(sFormNumber, 2)) = 23 Or Val(Left(sFormNumber, 2)) = 43 Or Val(Left(sFormNumber, 2)) = 46 Or Val(Left(sFormNumber, 2)) = 71 Or Val(Left(sFormNumber, 2)) = 72 Or Val(Left(sFormNumber, 2)) = 75 Or Val(Left(sFormNumber, 2)) = 76 Or Val(Left(sFormNumber, 2)) = 79 Or Val(Left(sFormNumber, 2)) = 83 Or Val(Left(sFormNumber, 2)) = 96 Or Val(Left(sFormNumber, 2)) = 97 Or Val(Left(sFormNumber, 2)) = 186 Or Val(Left(sFormNumber, 2)) = 187 Or Val(Left(sFormNumber, 2)) = 188 Or Val(Left(sFormNumber, 2)) = 191 Or Val(Left(sFormNumber, 2)) = 194 Or Val(Left(sFormNumber, 2)) = 195 Or Val(Left(sFormNumber, 2)) = 198 Or Val(Left(sFormNumber, 2)) = 199 Or Val(Left(sFormNumber, 2)) = 280 Or Val(Left(sFormNumber, 2)) = 290 Or Val(Left(sFormNumber, 2)) = 490 Or Val(Left(sFormNumber, 2)) = 491 Or Val(Left(sFormNumber, 2)) = 505 Or Val(Left(sFormNumber, 2)) = 810 Or Val(Left(sFormNumber, 2)) = 813 Or Val(Left(sFormNumber, 2)) = 814 Or Val(Left(sFormNumber, 2)) = 951 Then
```

What's wrong  
with that?

What's wrong  
with that?

*Too much  
code!*

What's wrong  
with that?

*Too much  
code!*

*Can't code each  
possible case!*

Not maintainable!

What's wrong  
with that?

*Too much  
code!*

Can't code each  
possible case!

Not maintainable!

Why  
w  
ing

Too mu  
code!

Can't code each  
possible case!



**You know the antidote**

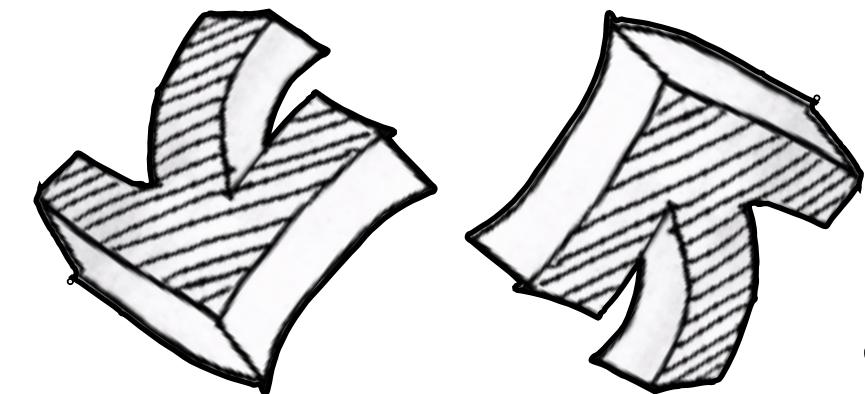
# **Refactoring skills**

# **Test-Driven Development skills**

# Concrete-Scenario-Driven (TDD, BDD)

HOW BUSINESS  
PEOPLE SPECIFY

derive  
concrete  
scenarios  
(deduction)



TEST CASES

refactor to  
general code  
(induction)

waterline

requirements

design



TRIANGULATION

Our **super power** is to be  
good at exploiting  
**regularities** through  
refactoring

Which means we can **help**  
business people at that

**SERIOUSLY**



**I'M FED UP**

*Memes Happen*

**Domain Experts**  
**not really experts**

**RAISE THE  
WATERLINE  
(yourselves)**

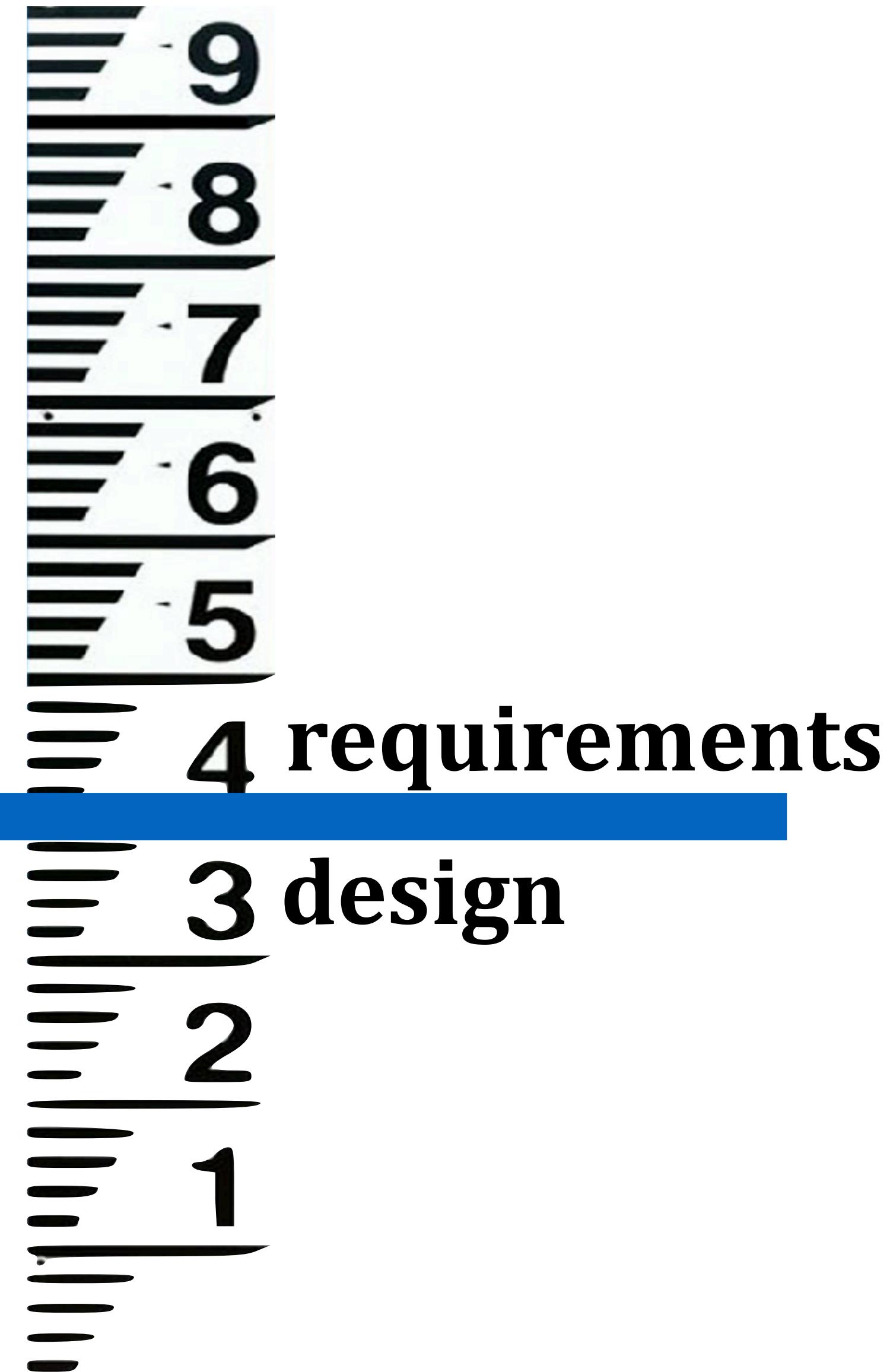
# Current situation

HOW BUSINESS  
PEOPLE SPECIFY

LABORIOUS  
CODE



current  
waterline

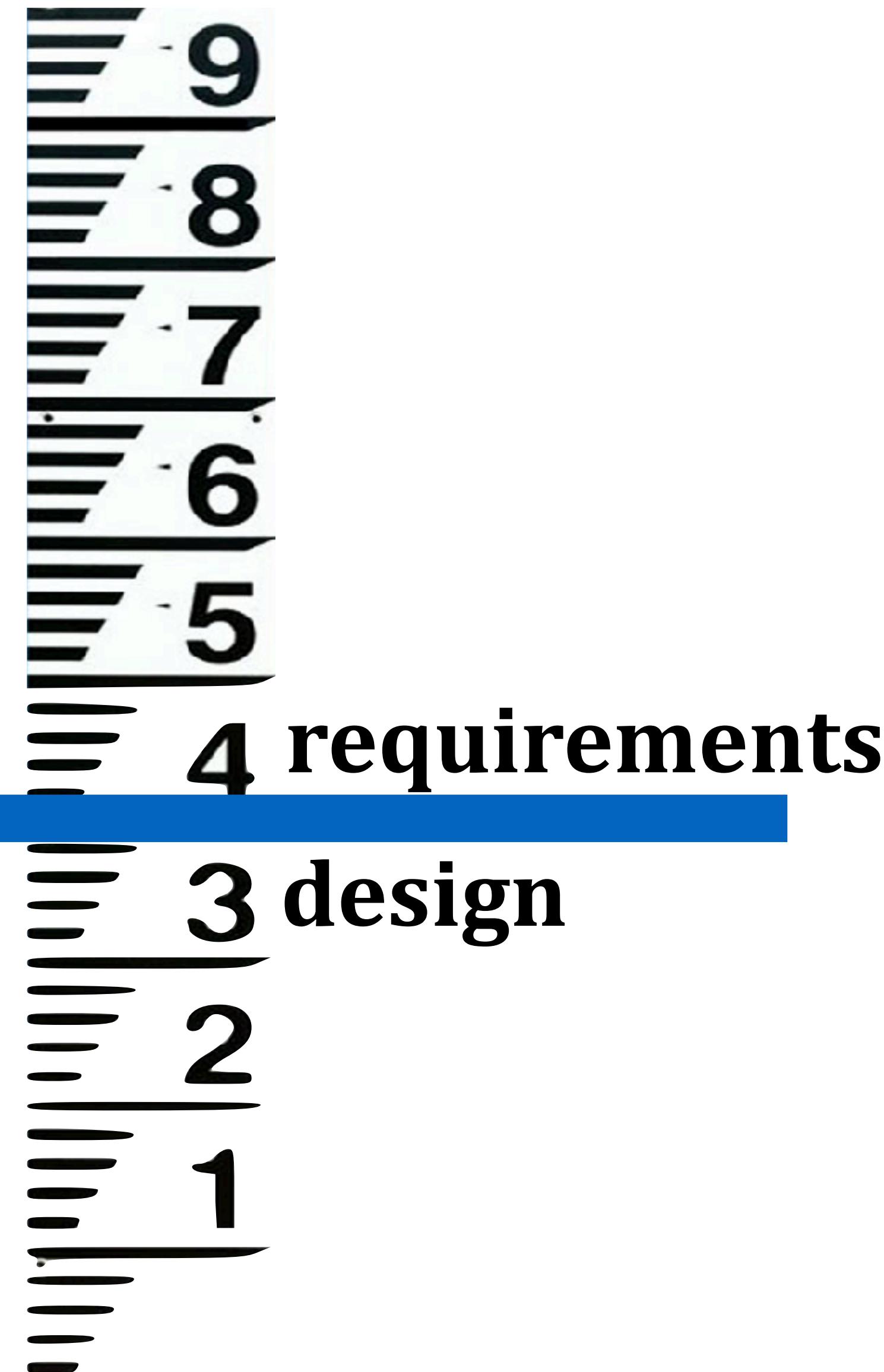


# Reverse-Engineering Investigation

(Progressing towards Higher Principles)

THE WAY BUSINESS  
REALLY WORKS  
(TACIT KNOWLEDGE)

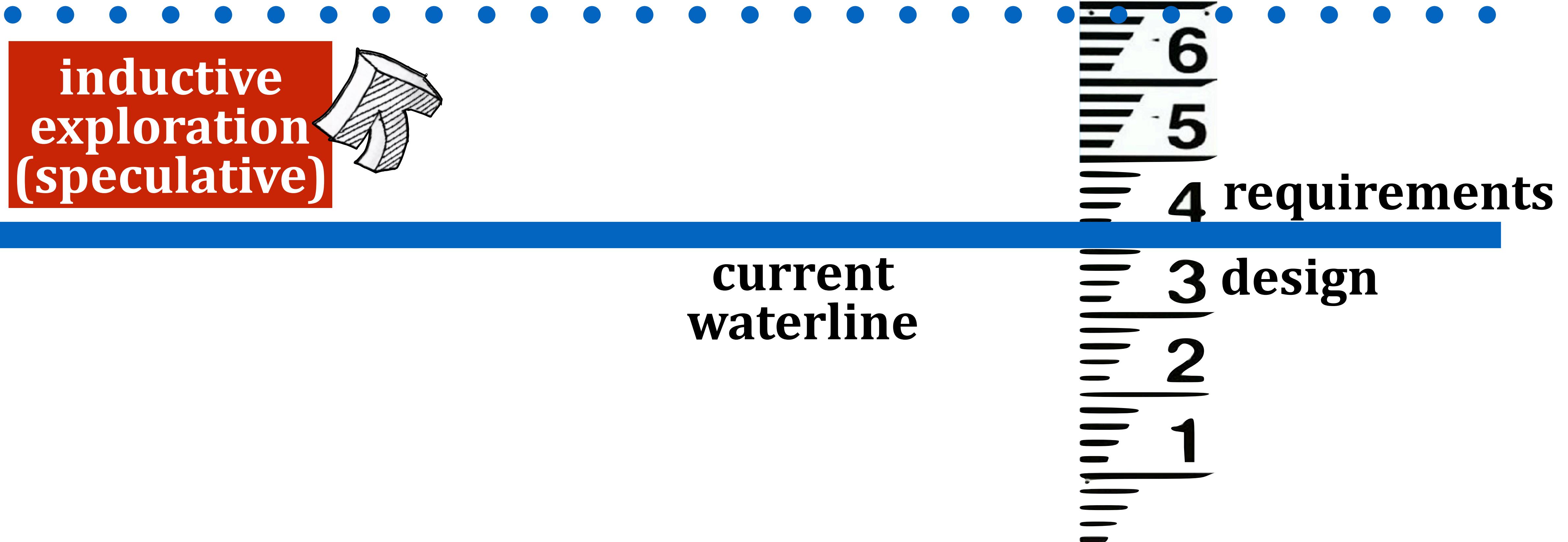
current  
waterline



# Reverse-Engineering Investigation

(Progressing towards Higher Principles)

THE WAY BUSINESS  
PEOPLE THINK  
(MAY BE TACIT)



# Reverse-Engineering Investigation

(Progressing towards Higher Principles)

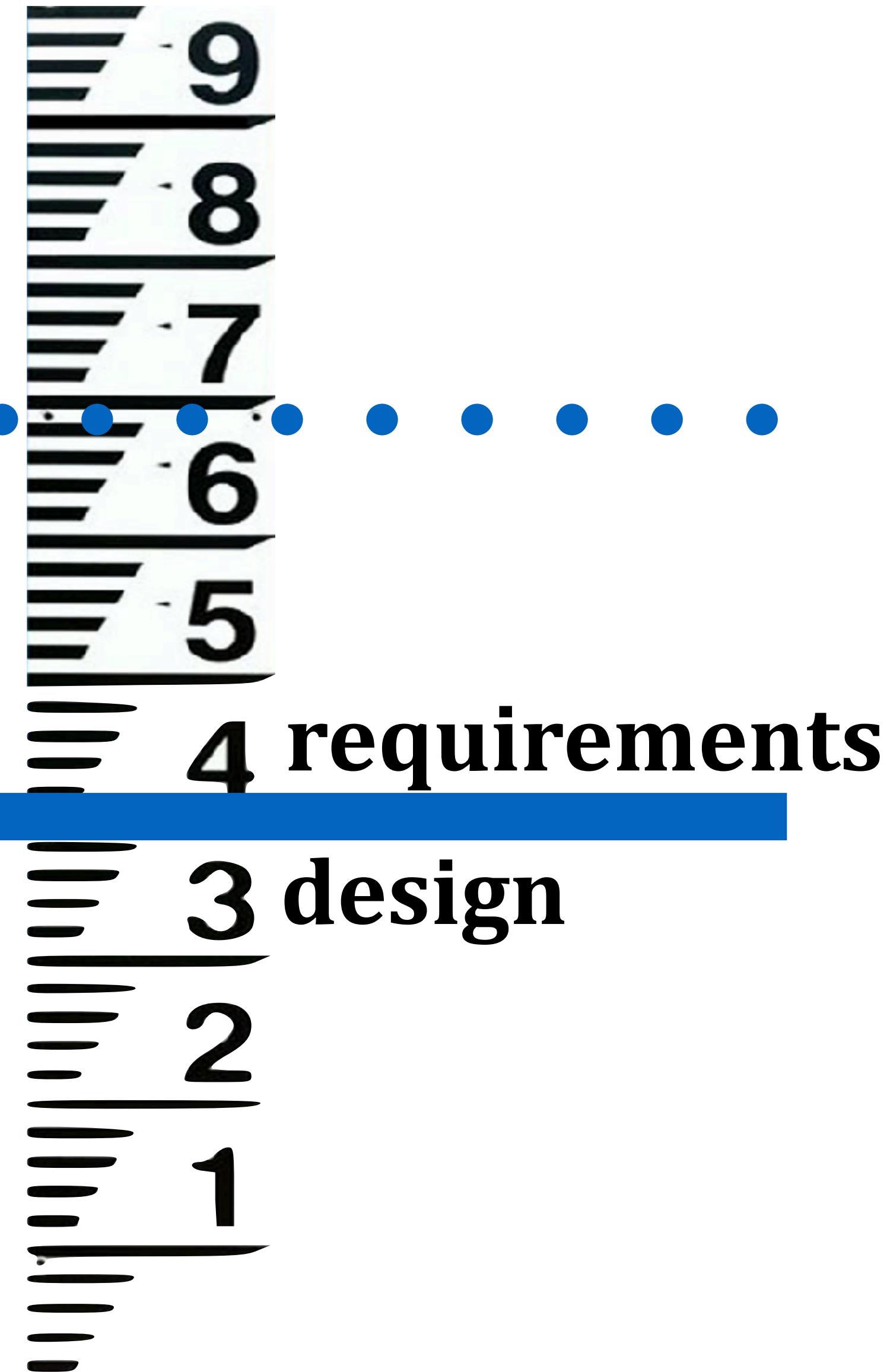
THE WAY BUSINESS  
PEOPLE THINK  
(MAY BE TACIT)

inductive  
exploration  
(speculative)



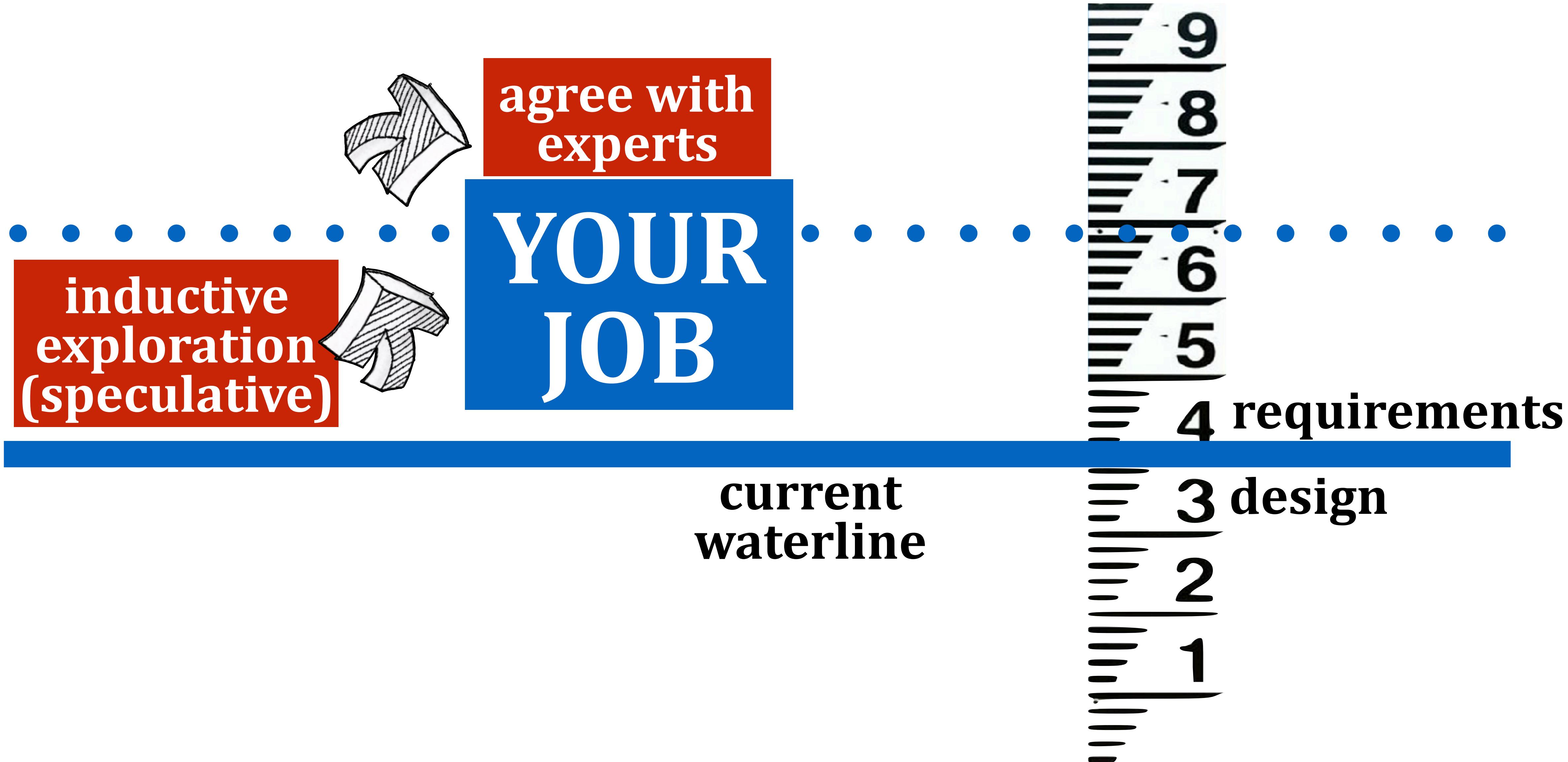
YOUR  
JOB

current  
waterline



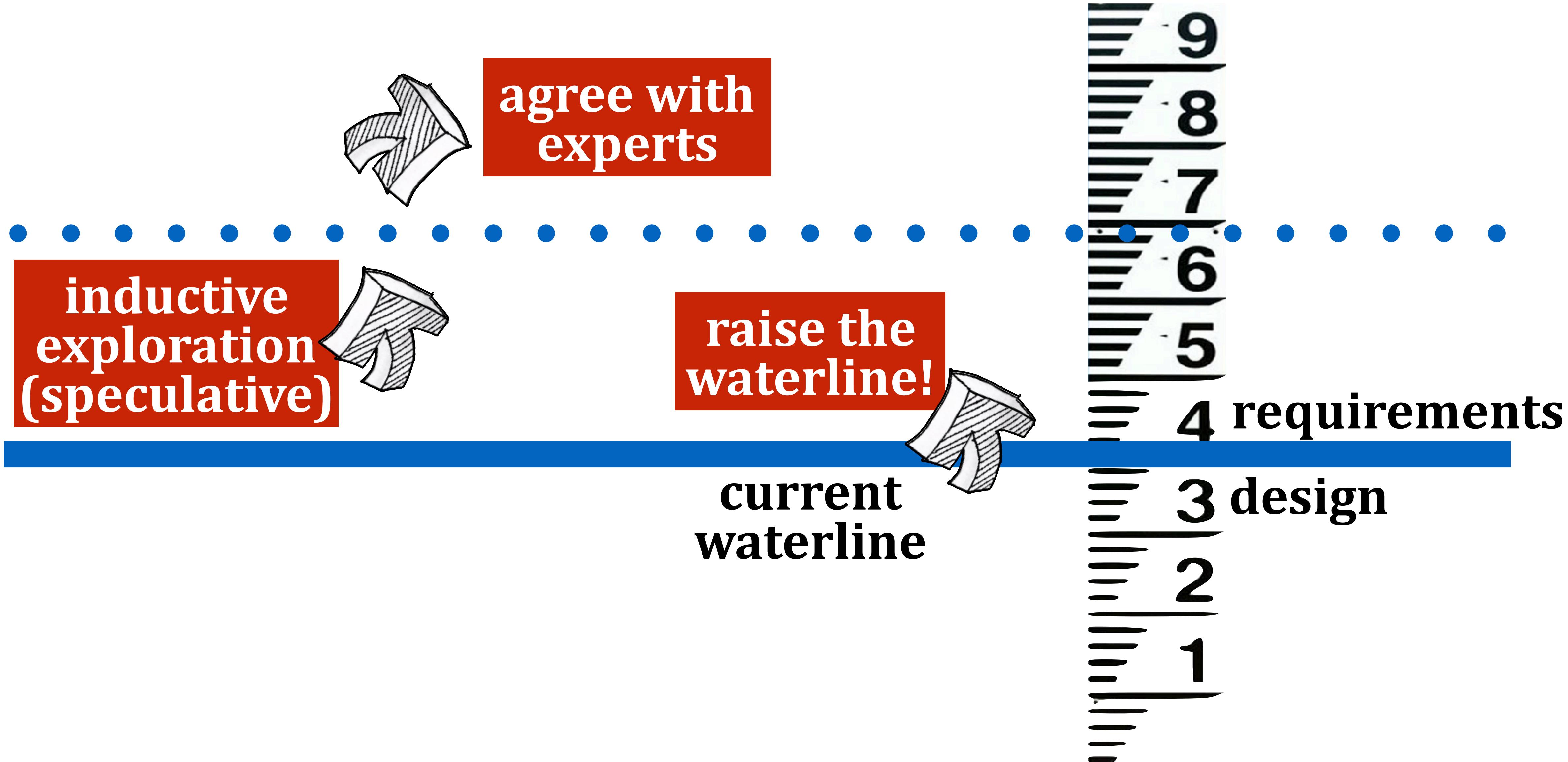
# Reverse-Engineering Investigation

(Progressing towards Higher Principles)



# Reverse-Engineering Investigation

(Progressing towards Higher Principles)



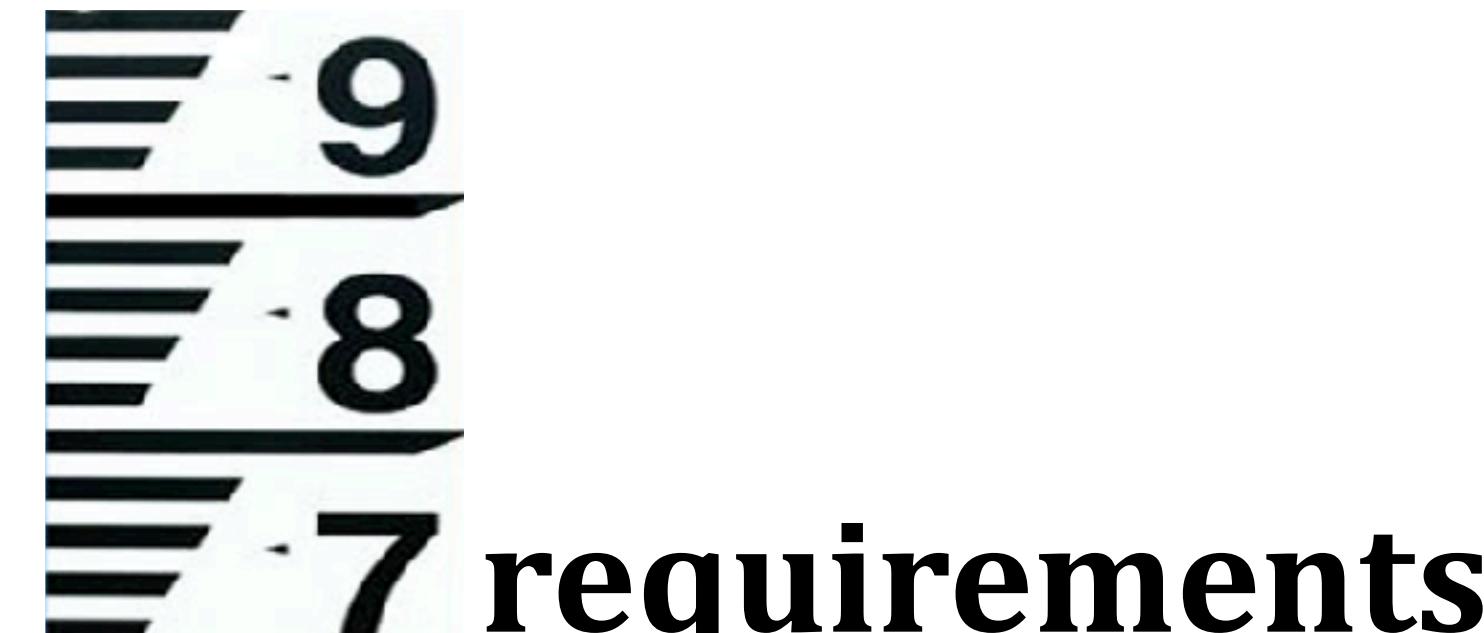
# Reverse-Engineering Investigation

(Progressing towards Higher Principles)

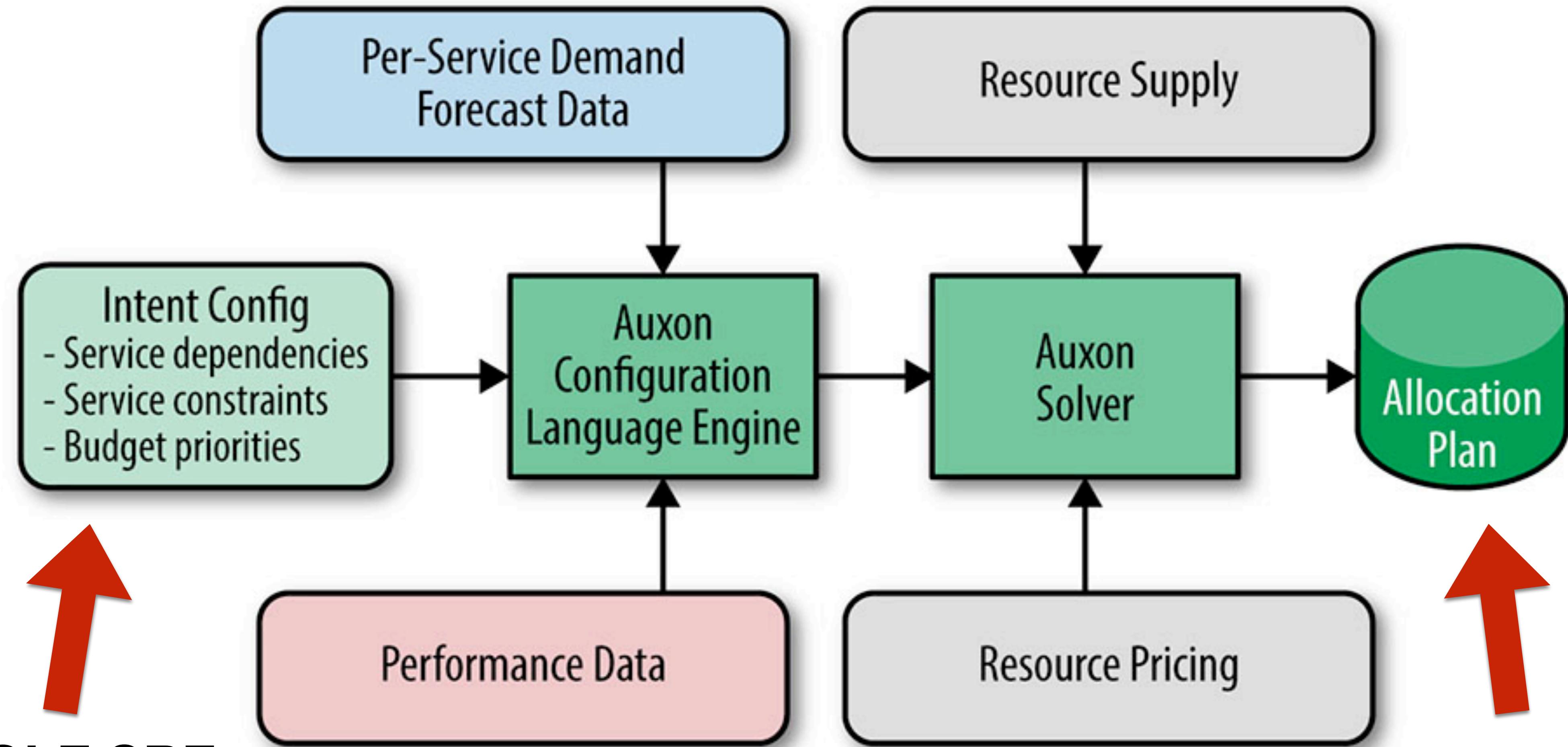


MORE POTENTIAL FOR  
SMARTER DESIGN

new  
waterline



**Smarter domain description,  
Smarter implementation**

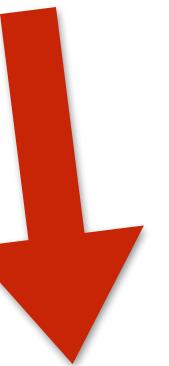


**GOOGLE SRE  
CONFIGURES  
HERE**

**MOST TEAMS  
CONFIGURE  
THIS MANUALLY**

# GOOGLE Auxon - DC resources allocations

Moving from concrete resource demands to motivating reasons in order to arrive at the true capacity intent often requires several layers of abstraction.

1. *I want 50 cores in clusters X, Y, and Z for service Foo.* **WHY?** 
2. *I want a 50-core footprint in any 3 clusters in geographic region YYY for service Foo.* **WHY?** 
3. *I want to meet service Foo's demand in each geographic region, and have N + 2 redundancy.* **WHY?** 
4. *I want to run service Foo at 5 nines (99.999%) of reliability A*

In Google's experience, we tend to achieve the best wins as they cross to step 3. Particularly sophisticated services may aim for step 4.

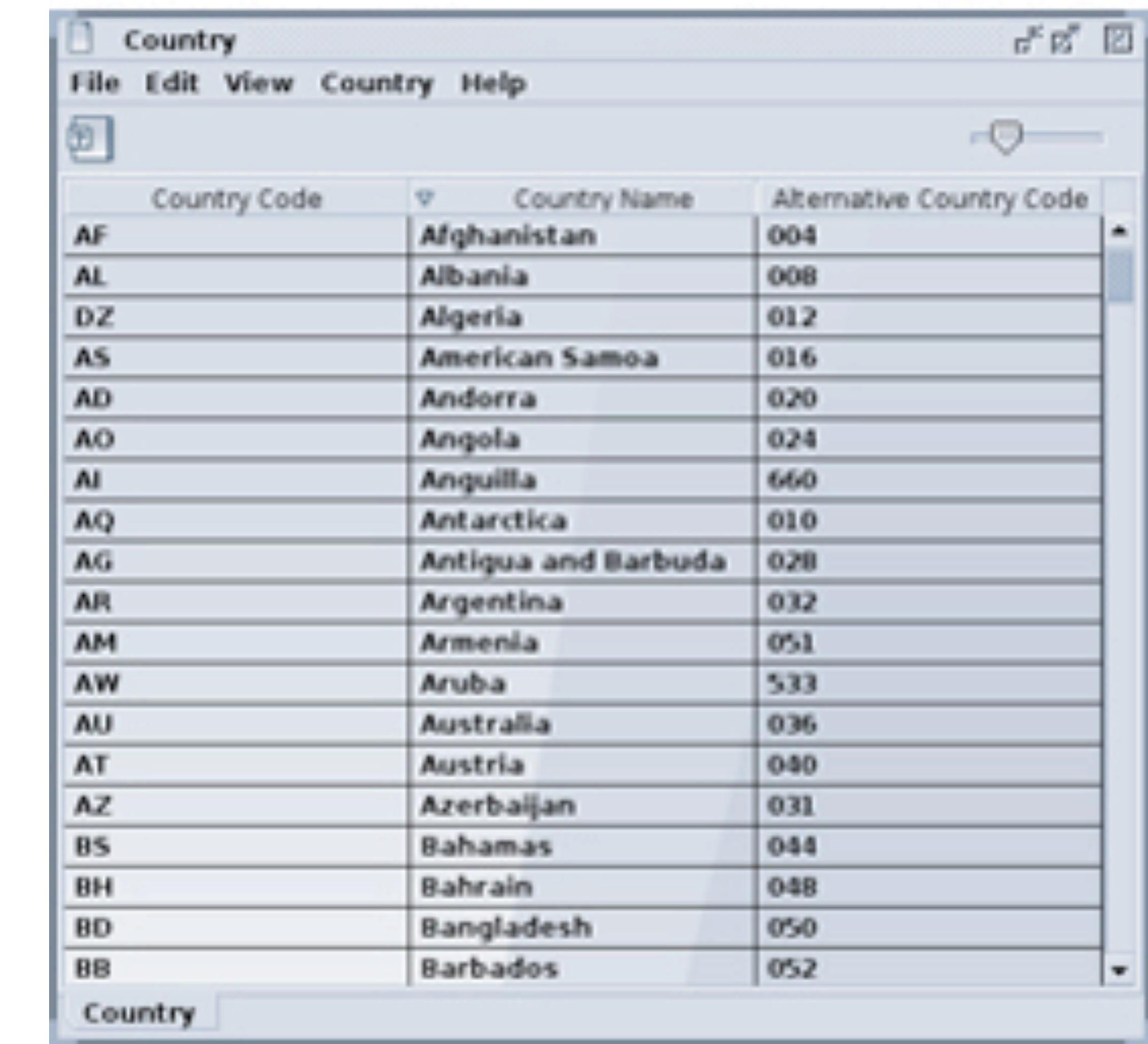
Will hardly happen unless you  
drive that

**Developers are  
the professionals  
of induction**

**Example please!**

# Configuration Lists

Teams members  
Workers assignments  
Categories  
Bank holidays  
Currencies  
Industry Classifications  
Countries, Regions  
Include / Exclude lists  
Preferences  
Access rights



The screenshot shows a software application window titled "Country". The menu bar includes File, Edit, View, Country, and Help. A toolbar with icons for search, refresh, and other functions is visible. The main area is a grid table with three columns: "Country Code", "Country Name", and "Alternative Country Code". The table contains 24 rows of data, each representing a country with its ISO code, name, and alternative code. The data is as follows:

Country Code	Country Name	Alternative Country Code
AF	Afghanistan	004
AL	Albania	008
DZ	Algeria	012
AS	American Samoa	016
AD	Andorra	020
AO	Angola	024
AI	Anquilla	660
AQ	Antarctica	010
AG	Antigua and Barbuda	028
AR	Argentina	032
AM	Armenia	051
AW	Aruba	533
AU	Australia	036
AT	Austria	040
AZ	Azerbaijan	031
BS	Bahamas	044
BH	Bahrain	048
BD	Bangladesh	050
BB	Barbados	052

# Lists in admin website

Reference Datasets    [Ontologies](#)    [Tasks](#)    [Browse](#)    [Search](#)    [Administration](#)

## Reference Datasets that you manage



[Country Codes](#)  
Created by [Administrator](#) on Nov 19, 2014 2:32:13 PM. Last changed by [Administrator](#) on Nov 19, 2014 2:35:42 PM. 1 working copies, 7 members.  
Main entity (class): [Country Code](#)  
ISO 3166 Country Codes, including both 2-alpha and 3-alpha codes.



[Currency Codes](#)  
Created by [Administrator](#) on Nov 19, 2014 2:42:49 PM. Last changed by [Administrator](#) on Nov 19, 2014 2:43:59 PM. 0 working copies, 7 members.  
Main entity (class): [Currency Code](#)  
ISO 4217 Currency Codes downloaded directly from ISO website.

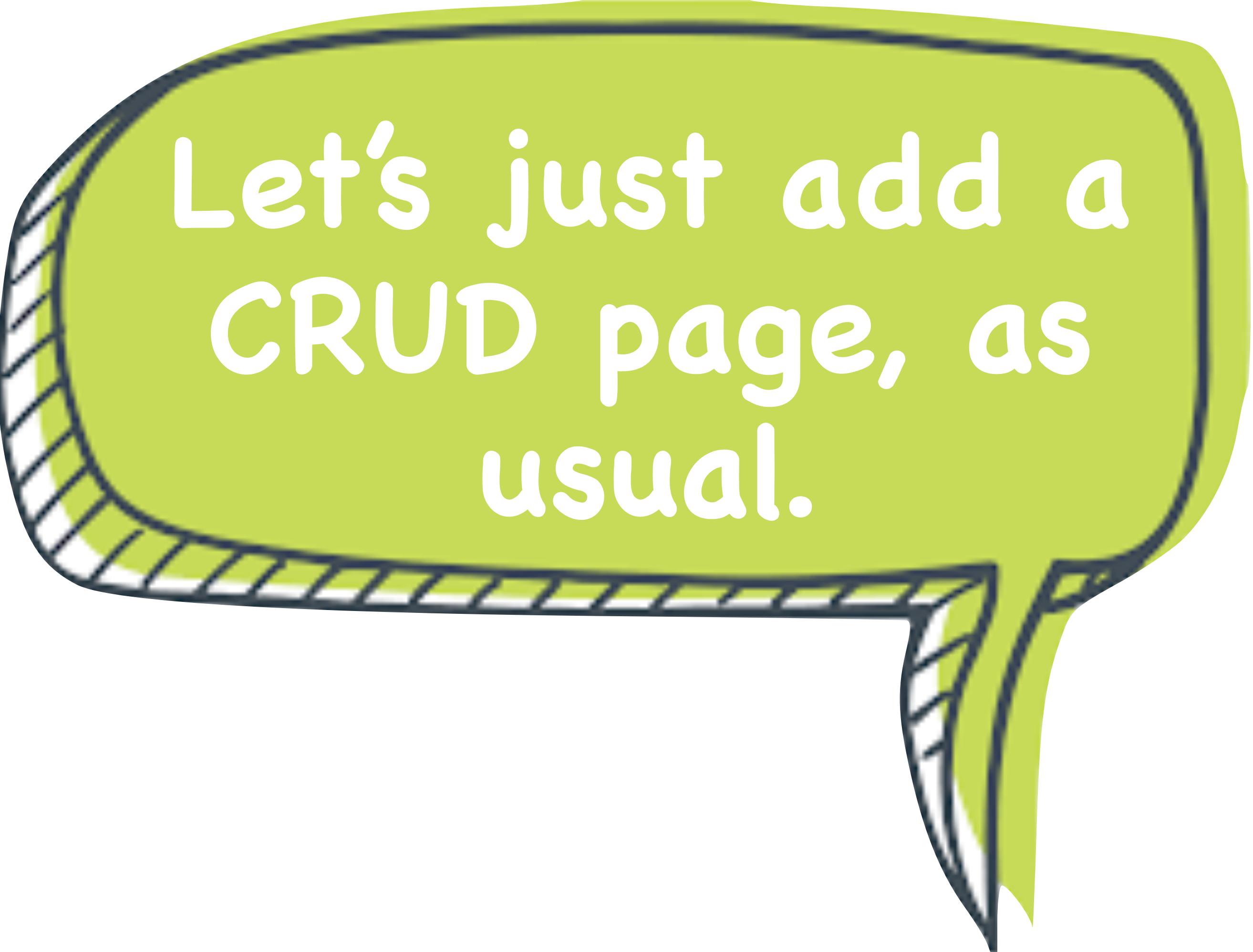


[Market Identifier Codes](#)  
Created by [Administrator](#) on Nov 19, 2014 2:37:21 PM. Last changed by [Administrator](#) on Nov 19, 2014 2:41:04 PM. 0 working copies, 7 members.  
Main entity (class): [Market Identifier Code](#)  
ISO 10383 Market Identifier codes downloaded directly from ISO website.

[Create New Reference Dataset](#)

# Somebody has to manage all these big lists...





Let's just add a  
CRUD page, as  
usual.

**RAISE THE  
WATERLINE**



WHY this list?



WHERE does it  
come from?

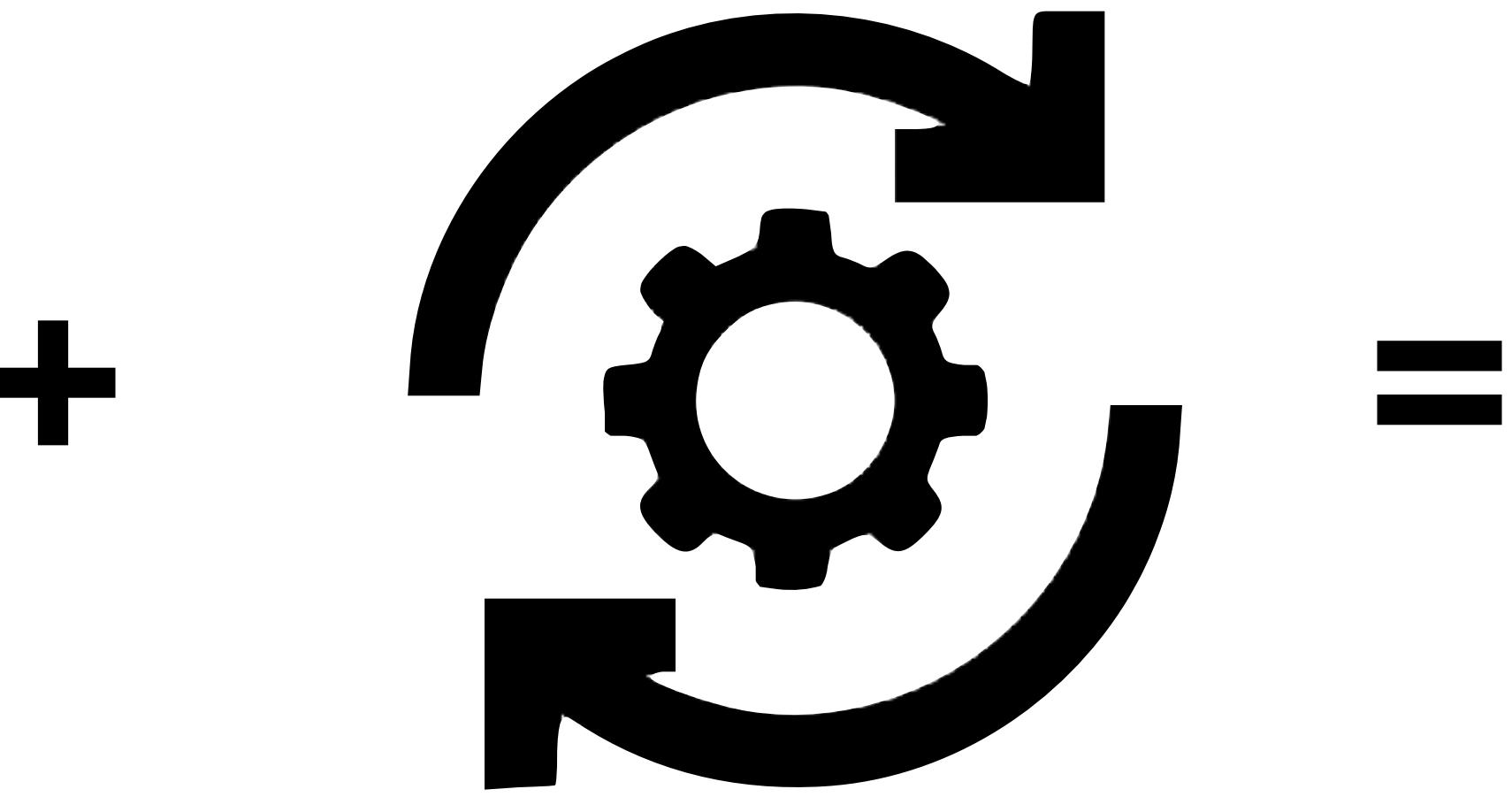


From other  
lists, post-  
processed!

Web feeds  
Spreadsheets  
Data already in DB

- merged
- filtered

Category	Recommended	Packed	Returned
Blankets or sheets	8		
Clothing, laundry, clothing, laundry & accessories	8		
Books	2		
Long pencils (pencils, pencil cases, colour pencils)	2		
Planners (journals and calendar)	2		
Hairs or leathers	16		
Motor vehicles	16		
Furniture	—		
Hairs or running shoes/leathers			
Blankets			
Mat for car protection			
Woven shoes			
Non-pen (pencil, books)			
<b>Electronics</b>			
Face masks			
Hand masks			
Health items			
Electro-therapeutic products			
Other health products			
Items more than 100 m higher			
Items & equipment			
Leather, vinyls & documents			
Plastic bags for other supplies			
Writing supplies			
Automotive engine products			
<b>Building</b>			
Sleeping bag			
Tool boxes for single cut saw			
Pillows and pillows cases			
House blocks for cold nights			
<b>Equipment</b>			
Country's tea			
Watch/gift and extra batteries			
Water bottle			
Self-adhesive stamped envelopes and writing paper			
Small tools, needles, small papers, pencils			
<b>Optional equipment &amp; clothing</b>			
Pair of training boots			
Big pocket or backpack			
Non-jackets or jackets for drivers			
Wind jacket for ice day			
Small hat			
Work, 1 piece, continuous case			
Canvas, blue, bathmats			
Blankets			
Small bags/pouches			
Extra sheets, plastic sheets & zipcases for insulation			



raw lists

processing

THE *list.*

MY list

**Imagine you work as a dev for an  
insurance company...**

# **Insurance Broker Portfolio Configuration**

"Each Insurance Broker has a list of contracts they manage: their "portfolio" of contracts."

They configure it manually on a web page.



# Insurance Broker Portfolio Configuration



"Each Insurance Broker has a list of contracts they manage. This is done via a web page."

**Where does this list come from?**

They configure it manually on a web page.



WHERE does it  
come from?

## Turns out

- They did searches during their setup
  - Search all contracts by corporate, for the few corporates they manage
- Their portfolio is every contracts with these corporates.
- Except the VIP contracts, managed by the head of sales.



From other  
lists, post-  
processed!

```
SELECT * FROM contracts  
WHERE contract.corporate IN my-accounts  
AND !contract.isVIP
```



Nothing fancy here.

```
SELECT * FROM contracts  
WHERE contract.corporate IN my-accounts  
AND !contract.isVIP
```



**SMALL list**

A red arrow points upwards from the word "list" towards the word "accounts" in the SQL query.

And now for a new pedantic word...



# Portfolio defined as...

Category	Requirement	Planned	Picked	Returned
Shirts	Short sleeve shirts	As needed		
	Long sleeve shirts including button & sweatshirts	As needed		
Shorts	Shorts	As needed		
	Long pants (drama, school pants, cotton pants)	As needed		
Pajamas	Sleepwear and cool weather	As needed		
Hats or Beanie	Hats or Beanie	As needed		
Motherwear	Motherwear	As needed		
Underwear	Underwear	As needed		
Blankets	Blankets	As needed		
Face of running shoes/athletic	Face of running shoes/athletic	As needed		
Health supplies	Health supplies	As needed		
Not for educational	Not for educational	As needed		
Skate shoes	Skate shoes	As needed		
Non-peer (adult, travel)	Non-peer (adult, travel)	As needed		
Furniture	Furniture	As needed		
Textiles	Textiles	As needed		
Hand towels	Hand towels	As needed		
Beach towels	Beach towels	As needed		
Laundry detergent/softener	Laundry detergent/softener	As needed		
Blanket/sleeping bags	Blanket/sleeping bags	As needed		
Bounce balls (B7V # 10 or higher)	Bounce balls (B7V # 10 or higher)	As needed		
Swim items	Swim items	As needed		
Books, documents, documents	Books, documents, documents	As needed		
Plastic, plastic for classroom supplies	Plastic, plastic for classroom supplies	As needed		
Writing supplies	Writing supplies	As needed		
Resinous paper products	Resinous paper products	As needed		
Bedding	Bedding	As needed		
Sleeping bag	Sleeping bag	As needed		
Bed sheets for single and twin	Bed sheets for single and twin	As needed		
Pillows and pillow cases	Pillows and pillow cases	2 pillow cases		
Blanket/towel for cold nights	Blanket/towel for cold nights	As needed		
Equipment	Equipment	As needed		
Clothing Bag	Clothing Bag	As needed		
Flashlight and extra batteries	Flashlight and extra batteries	As needed		
Water bottle	Water bottle	As needed		
Self-adhesive stamp envelopes and writing paper	Self-adhesive stamp envelopes and writing paper	At least 4		
Class time activities books, small games, pencils	Class time activities books, small games, pencils	As needed		
Optional equipment & clothing	Optional equipment & clothing	As needed		
Pair of riding boots	Pair of riding boots	As needed		
Ring packed or kept off	Ring packed or kept off	As needed		
New clothes or costumes for dances	New clothes or costumes for dances	As needed		
Velvet shorts for ice skates	Velvet shorts for ice skates	Optional		
Small fan	Small fan	As needed		
Skirt, 3 pieces, costume, coat	Skirt, 3 pieces, costume, coat	As needed		
Camera, film, batteries	Camera, film, batteries	As needed		
Jump ropes	Jump ropes	As needed		
Small backpacks	Small backpacks	As needed		
Extra sheets, plates, sheets & cutlery for both women	Extra sheets, plates, sheets & cutlery for both women	As needed		

Category	Requirement	Value
390	$10^{1173}$	Nonagintatrecentillion
391	$10^{1176}$	Unnonagintatrecentillion
392	$10^{1179}$	Duononagintatrecentillion
393	$10^{1182}$	Trenonagintatrecentillion
394	$10^{1185}$	Quattuoragintatrecentillion
395	$10^{1188}$	Quinnonagintatrecentillion
396	$10^{1191}$	Senonagintatrecentillion
397	$10^{1194}$	Septenonagintatrecentillion
398	$10^{1197}$	Octononagintatrecentillion
399	$10^{1200}$	Novenonagintatrecentillion
400	$10^{1203}$	Quadringtonillion
401	$10^{1206}$	Unquadringtonillion
402	$10^{1209}$	Duoquadringtonillion
403	$10^{1212}$	Tresquadringtonillion
404	$10^{1215}$	Quattuorquadringtonillion
405	$10^{1218}$	Quinquadringtonillion
406	$10^{1221}$	Sesquadringtonillion
407	$10^{1224}$	Septenquadringtonillion
408	$10^{1227}$	Octoquadringtonillion
409	$10^{1230}$	Novenquadringtonillion
410	$10^{1233}$	Deciquadringtonillion
411	$10^{1236}$	Undeciquadringtonillion
412	$10^{1239}$	Duodeciquadringtonillion
413	$10^{1242}$	Tredeciquadringtonillion
414	$10^{1245}$	Quattuordeciquadringtonillion
415	$10^{1248}$	Quindeciquadringtonillion
416	$10^{1251}$	Sedeciquadringtonillion

raw lists



= THE *list*.

MY list

EXTENSIONAL

An intensional  
definition ... by  
specifying necessary  
and sufficient  
conditions

**"bachelor"**

=

**"unmarried man"**

# PREDICATE

evaluate(something): boolean

An extensional  
definition ... by  
**listing everything**

**"bachelor"**

=

**listing of all the  
unmarried men in  
the world**

**"bachelor"**

=

**listing a  
unmarrie  
the v**

**Ooops!!!!**



# Enumerable

for each: element

**SERIOUSLY**



**I'M FED UP**

*Memes Happen*

**EXTENSIONAL  
OBSESSION**



**Nat Pryce**  
@natpryce

Following



When programmers don't understand a problem, their reaction is to add end-user configuration.

**Giovanni Asproni** @gasproni

Driverless cars could let you choose who survives in a crash [newscientist.com/article/215033...](http://newscientist.com/article/215033...)

10:38 PM - 16 Oct 2017

**126** Retweets **182** Likes



**HEURISTICS:  
CONSIDER THE  
INTENSIONAL  
ALTERNATIVE**

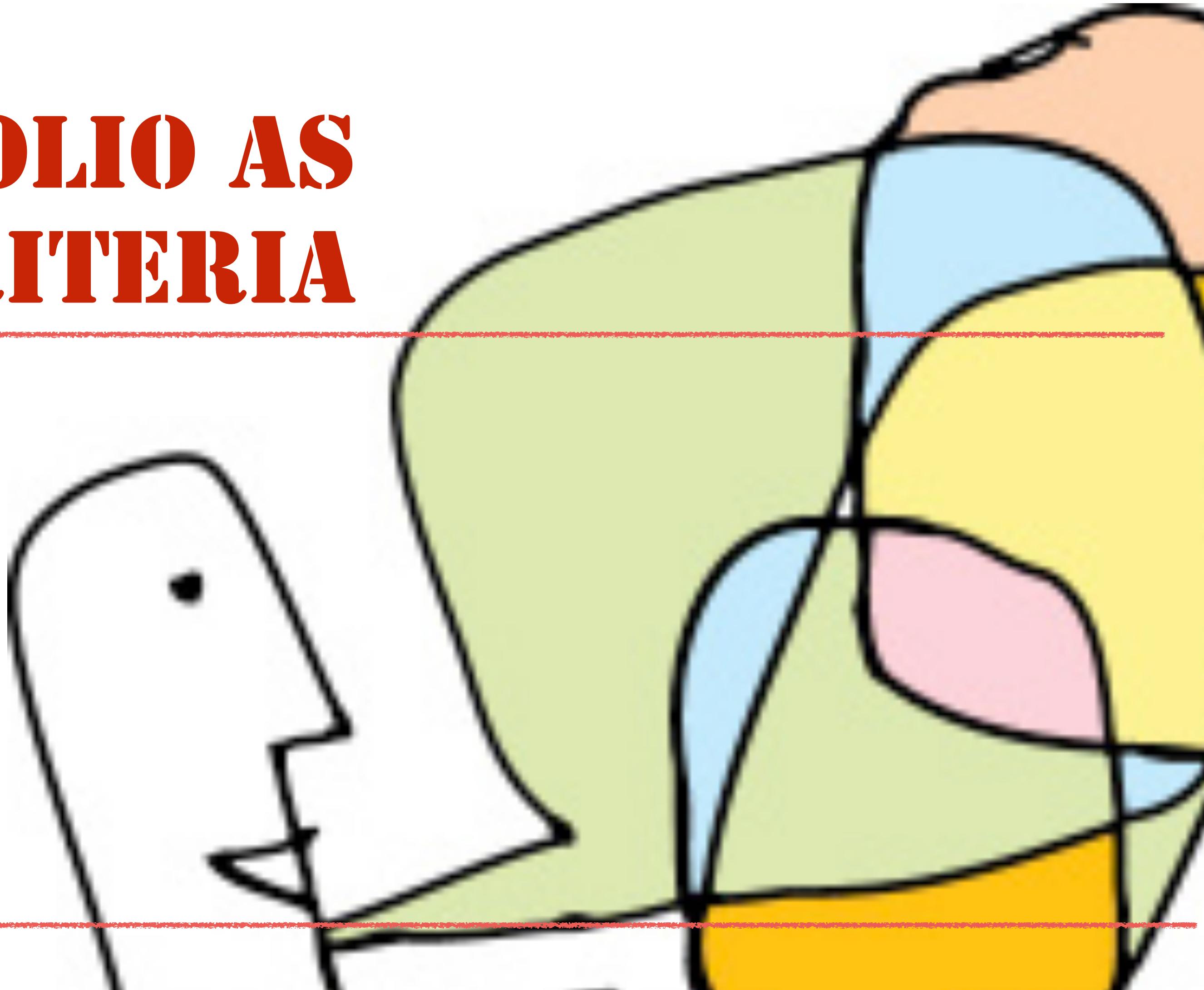
# HIGHER LEVEL of LANGUAGE

**PORTFOLIO AS  
CRITERIA**

---

**PORTFOLIO AS  
LIST OF CONTRACTS**

---



# PORTFOLIO AS LIST OF CONTRACTS

	Recommended	Packed	Returned
<b>Clothing</b>			
Short sleeve shirts	3		
Long sleeve shirts including t-shirts & sweatshirts	4		
T-shirts	2		
Long pants (jeans, cargo pants, cotton pants)	3		
Harem pants (jean and cool varieties)	1		
Pants or shorts	1.5		
Underwear	1.5		
Socks	1		
Pairs of running shoes/sneakers	2 or 3		
Beach towels	1		
Mat for sun protection	1		
Water shoes	1		
Rain gear (jacket, boots)	1		
<b>Toiletries</b>			
Face wash	1		
Hand towels	2		
Beach towels	1		
Deodorant/odor control products	1		
Shampoo/conditioner	1		
Bath, shampoo & shower gel	1		
Plastic bottle for shower supplies	1		
Shaving supplies	As needed		
Antiseptic/Insect repellent	As needed		
<b>Bedding</b>			
Sleeping bag	1		
Bed sheets for single and twin	2		
Pillow and pillow cases	2 (allow 3 total)		
Blanket/towel for cold nights	1		
<b>Equipment</b>			
Travel bag	1		
Flashlight and extra batteries	1		
Water bottle	1		
Golf addressed stamped envelopes and writing paper	at least 14		
Used note books/books, small papers, pencils	1		
<b>Optional Equipment &amp; Clothing</b>			
Pair ofinking boots	1		
Bug pocket or bug net	1		
Non-latex or condoms for clients			
White shirt for an size	optional		
Small fan			
Short, 1 piece swimwear (size)			
Chinless, thin, buttless			
Swimsuit			
Small backpack			
Travel mirror, plastic sheets & impaxas for last minute			

# INTENSIONAL

```
SELECT * FROM contracts  
WHERE contract.corporate IN my-accounts  
AND !contract.isVIP
```

# EXTENSIONAL



# PORTFOLIO AS CRITERIA

Category	Implemented	Picked	Returned
Short duration contracts	8		
Long duration contracts including industrial & construction	8		
Total	16		

managed  
list of corporates

+

VIP tag on some contracts

**problem solved :)**

**SERIOUSLY**



**I'M FED UP**

*Memes Happen*

This modelling  
trick is nice...

But what  
about all the  
other business  
rules?

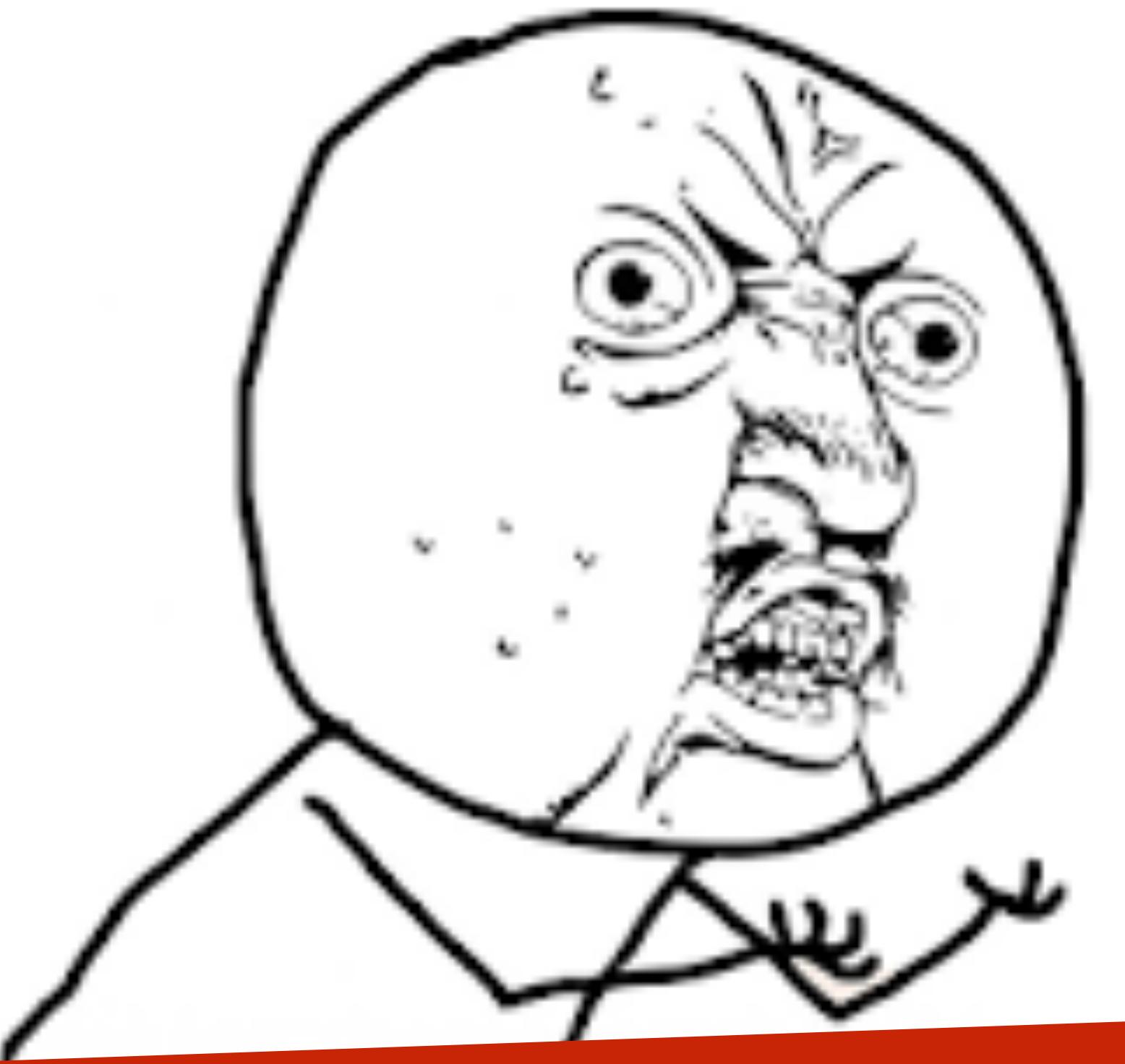
But what  
about ALL THE  
other business  
rules?

But what  
about all the  
other business  
rules?

**ARBITRARY  
RULES  
OBSESSION**

**ARBITRARY  
RULES  
OBSESSION**

**Business**  
= **CRUD**  
+ **rules**



**ARBITRARY  
RULES  
OBSESSION**

**Business**  
=  
**CRUD**  
+  
**rules**

Bunch of  
data fields

Piles of IF  
statements

Domain model as-a

# **BIG BAG OF BUSINESS RULES**

You cannot get  
smarter domain models if  
you believe the domain is  
mostly arbitrary!

Do you believe the  
world is just noise,  
or full of hidden  
regularities?

Code can be small  
YET still supports a  
LOT of behaviours

**Domain Modelling is  
a quest to eradicate  
arbitrary business  
rules!**

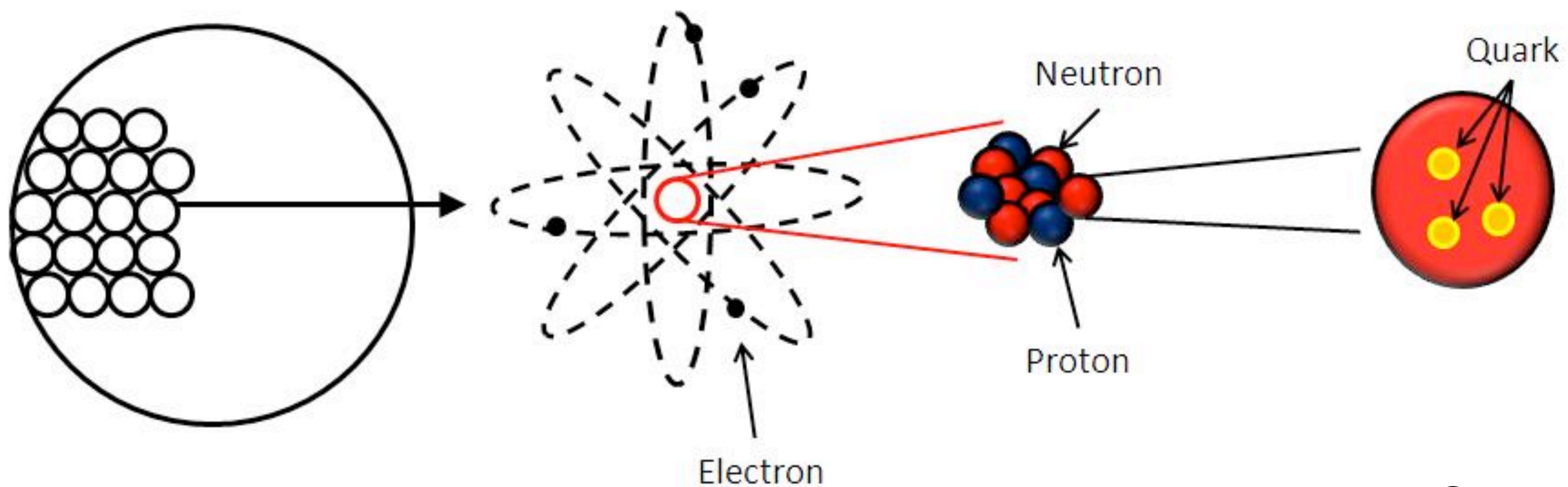
# **Programming as Theory-Building**

**— Peter Naur (1985)**

# DOMAIN MODEL as-a THEORY

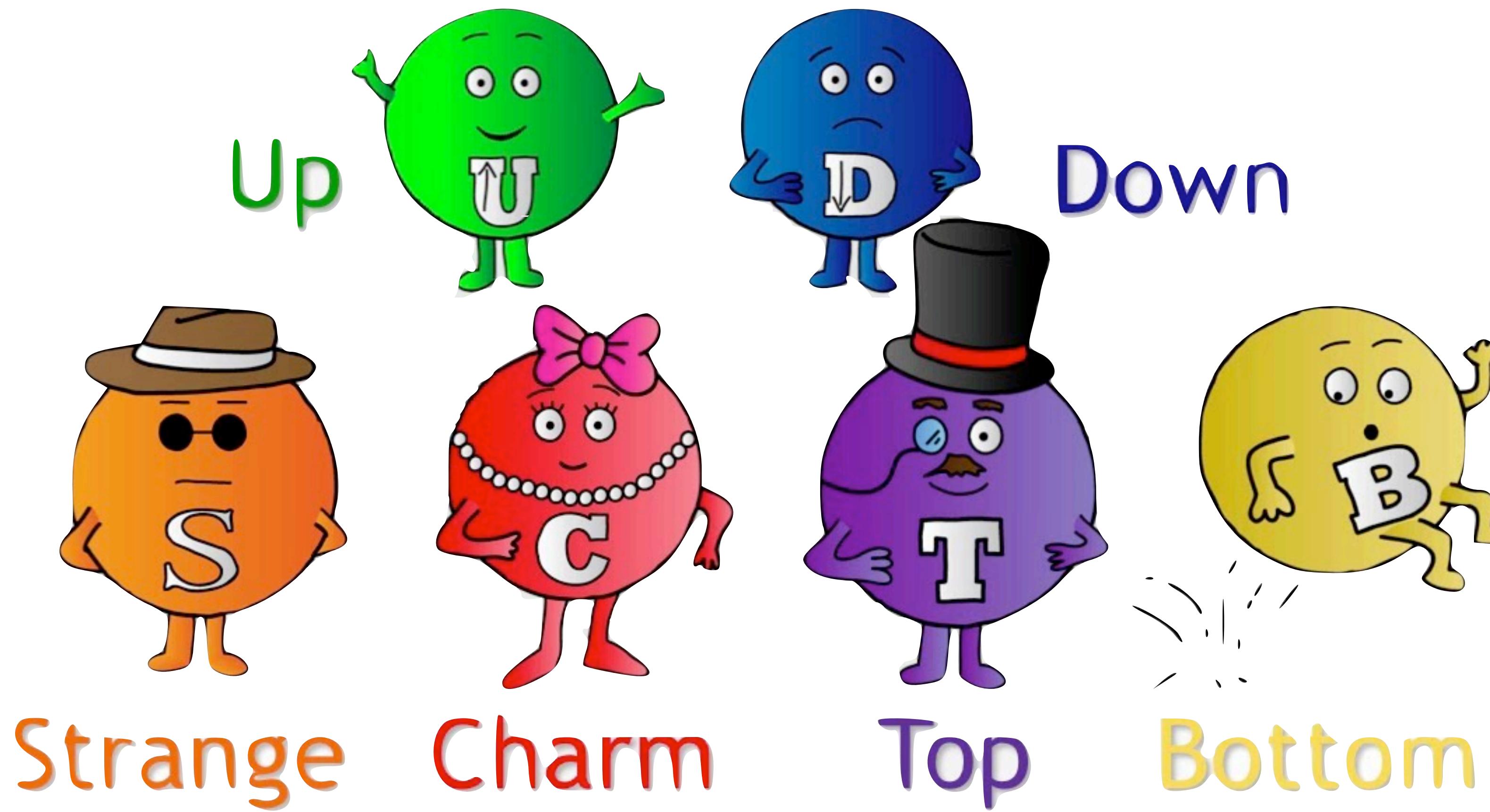
**To understand  
To explain  
To predict**

# A widely accepted theory

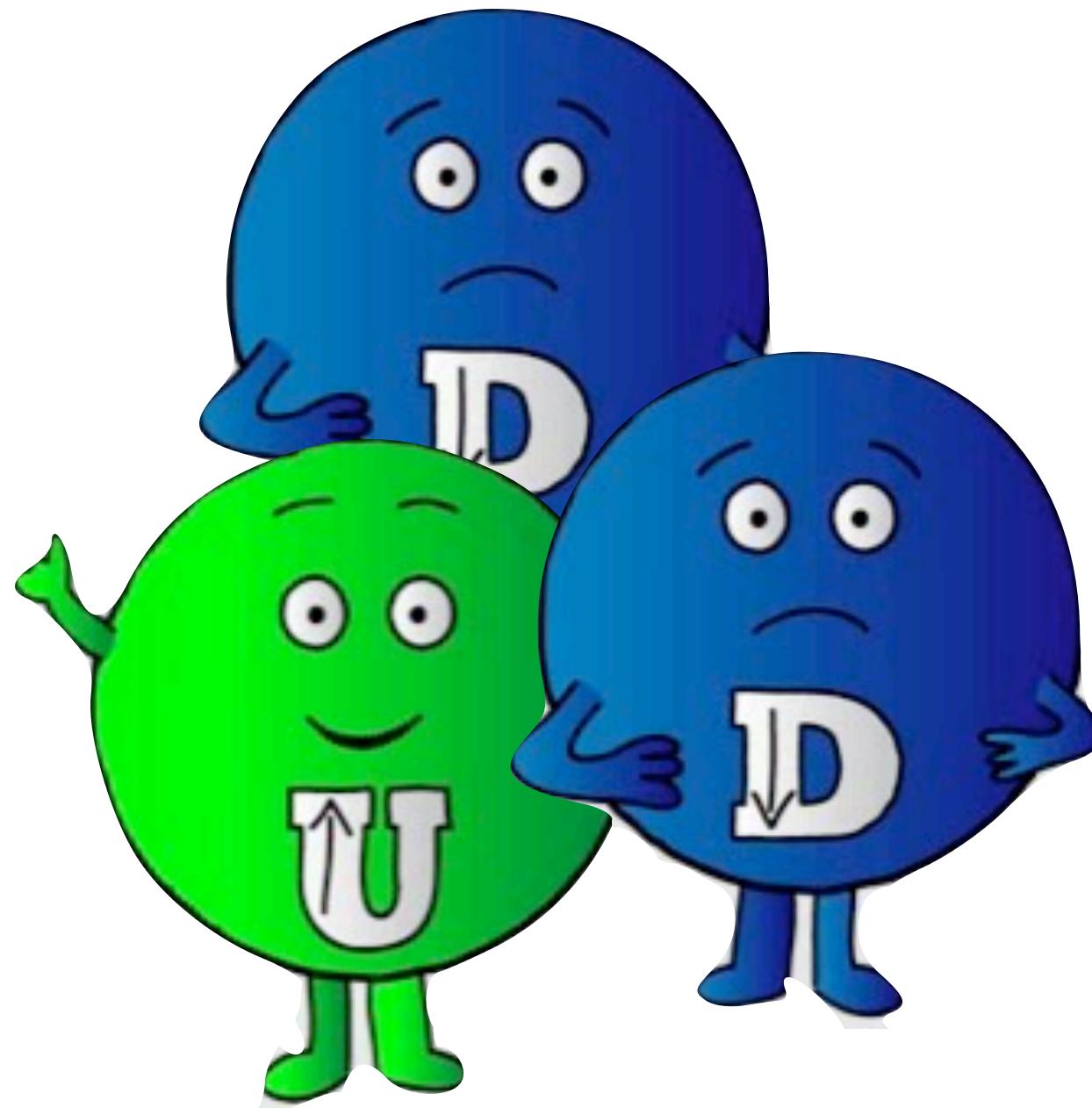


Murray Gell-Mann  
Nobel 1969

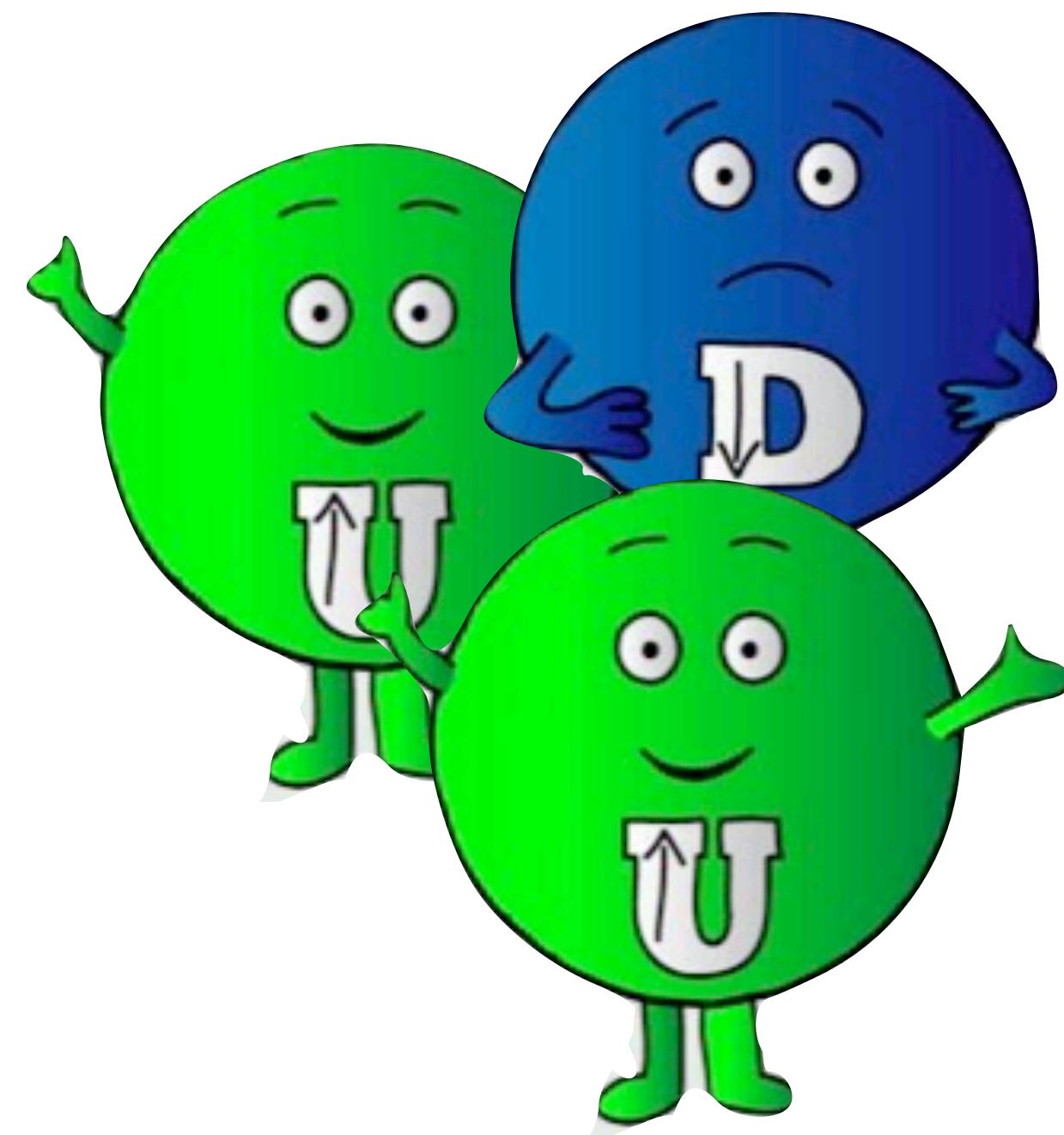
# A Finite Set of Quarks



# Quarks Compose into bigger particles

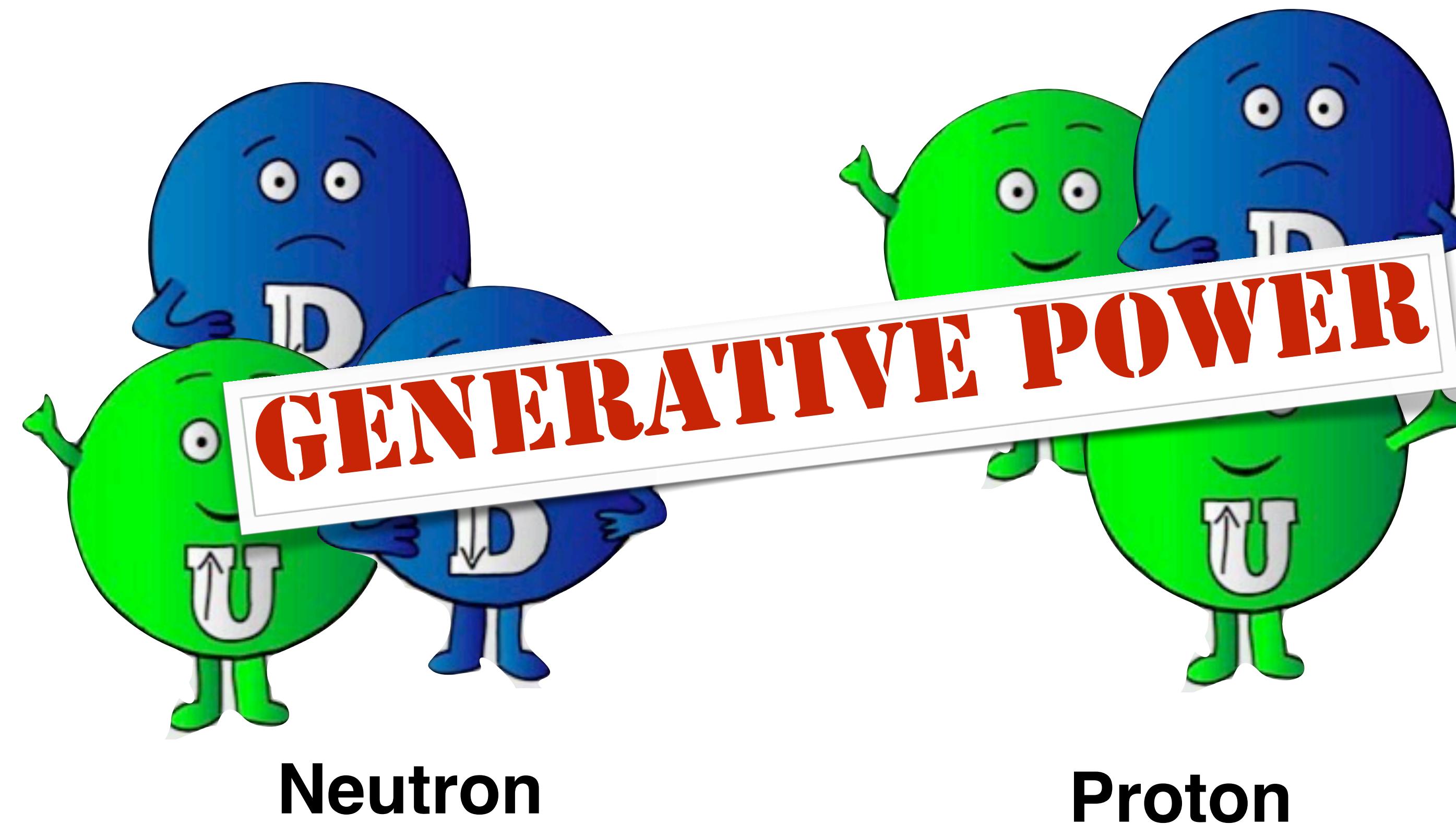


Neutron



Proton

# Quarks Predict Unknown particles



Quark Composition  
of Several Hadrons

Particle	Quark Composition
<hr/>	
Mesons	
$\pi^+$	$\bar{d}u$
$\pi^-$	$u\bar{d}$
$K^+$	$\bar{s}u$
$K^-$	$u\bar{s}$
$K^0$	$\bar{s}\bar{d}$
<hr/>	
Baryons	
p	uud
n	udd
$\Lambda^0$	uds
$\Sigma^+$	uus
$\Sigma^0$	uds
$\Sigma^-$	dds
$\Xi^0$	uss
$\Xi^-$	dss
$\Omega^-$	sss

and  
many others...

# **Why it matters?**

a Theory is  
**CONCISE**

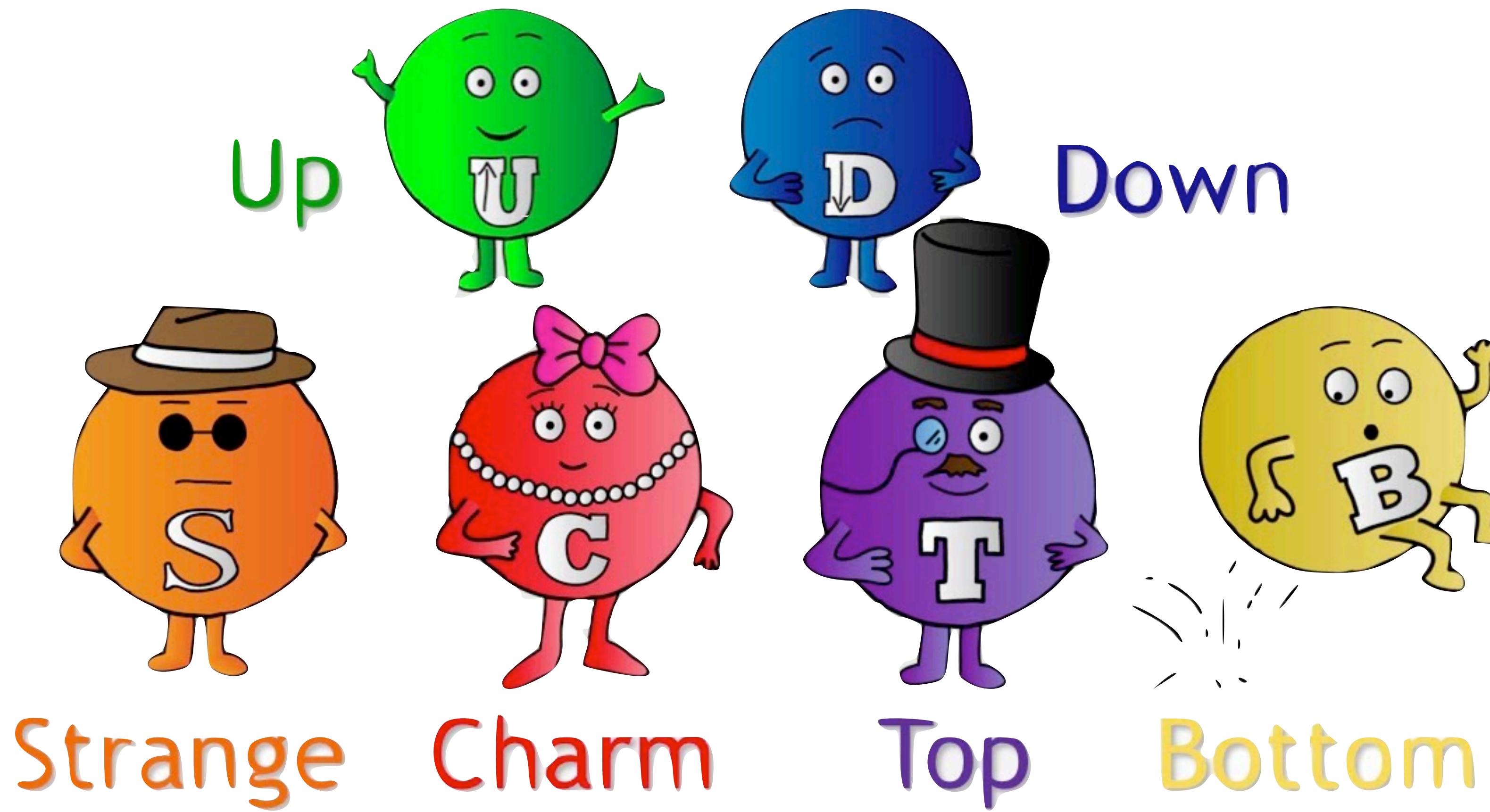
but their consequences are  
**EXTENSIVE**

a Theory is  
**CONCISE**

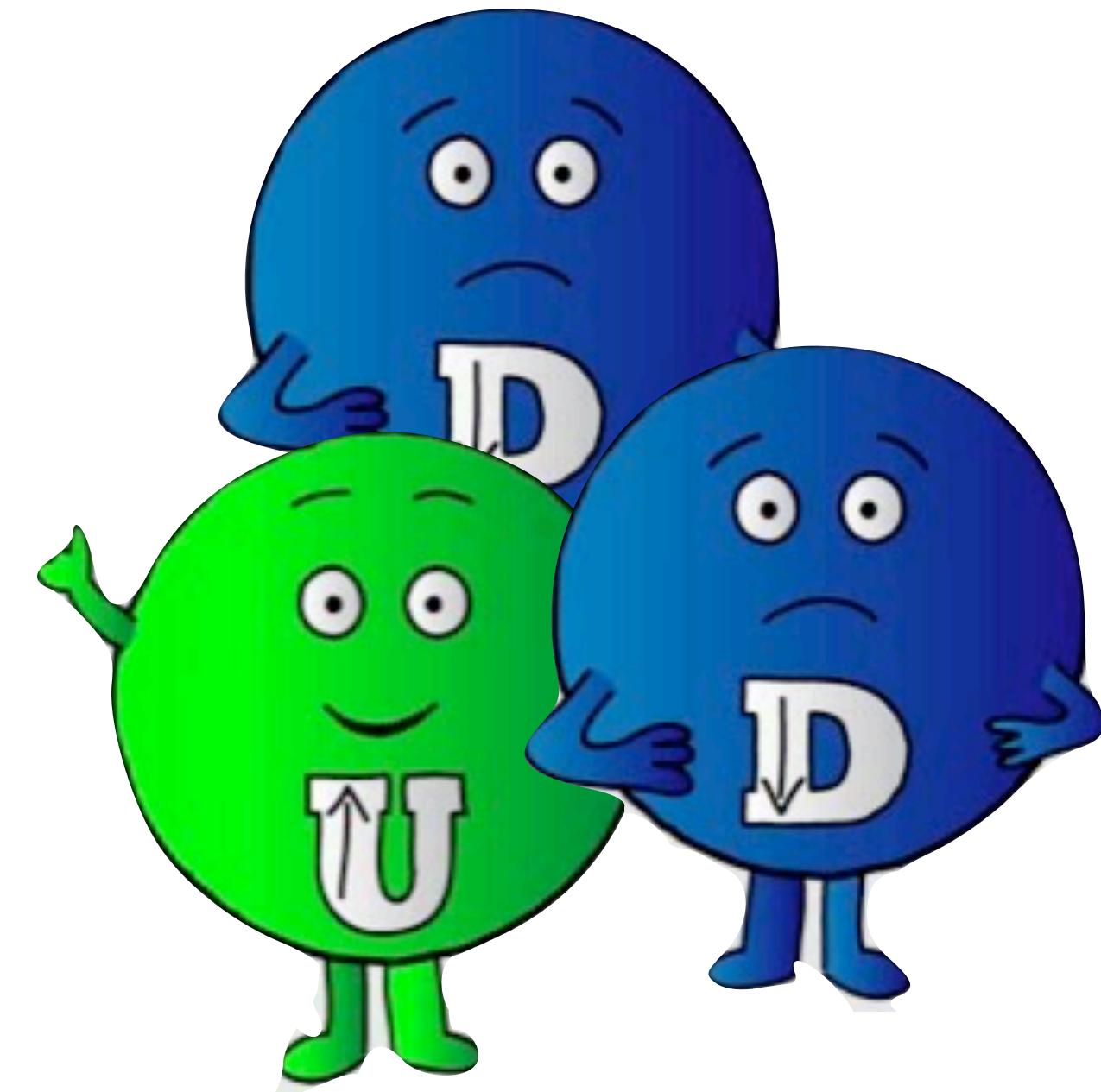
but its consequences are

**EXTENSIVE**

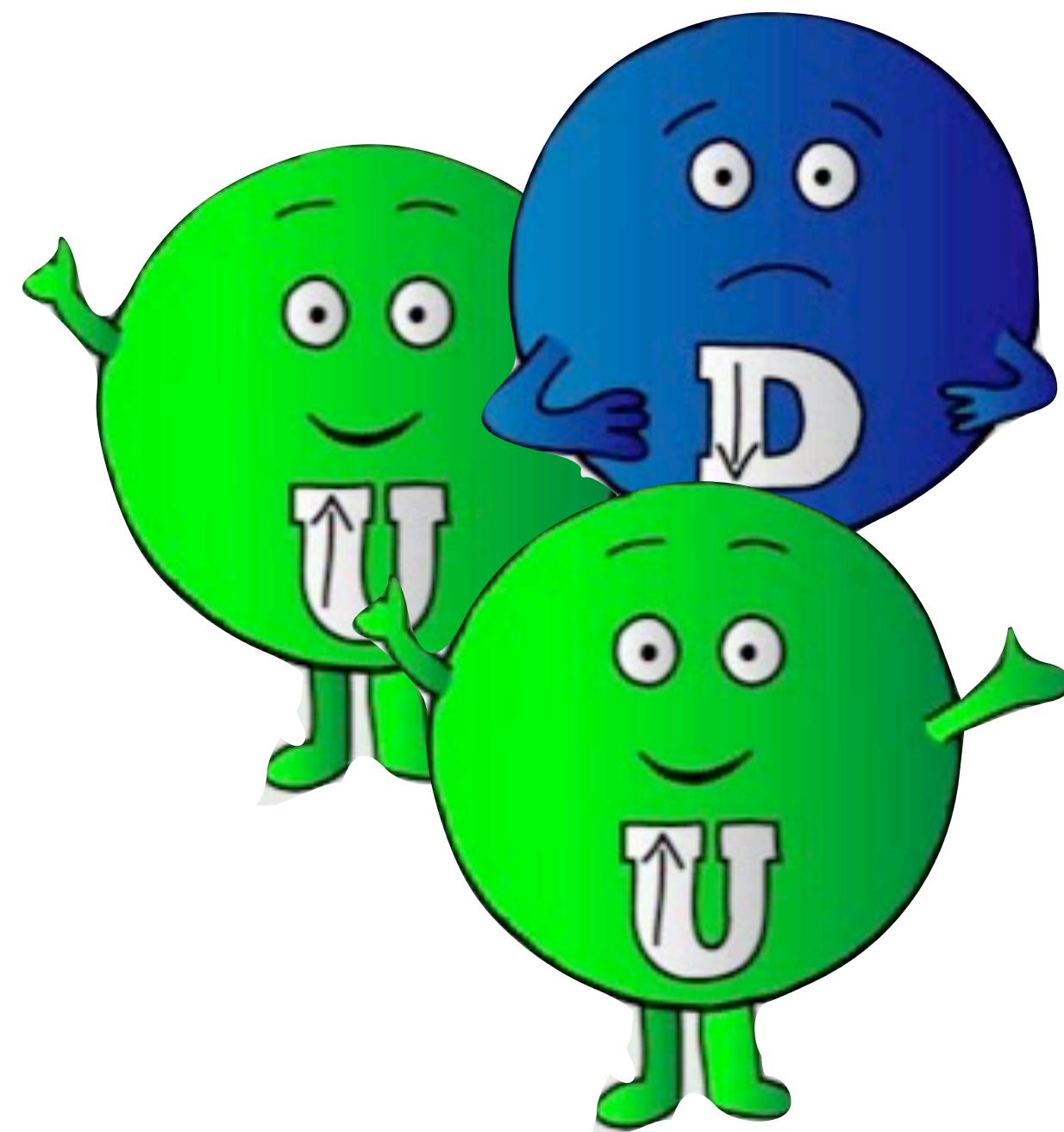
# A Finite Set of Quarks



# Many particles



Neutron



Proton

## Quark Composition of Several Hadrons

Particle	Quark Composition
<b>Mesons</b>	
$\pi^+$	$\bar{d}u$
$\pi^-$	$ud$
$K^+$	$\bar{s}u$
$K^-$	$\bar{u}s$
$K^0$	$\bar{s}d$
<b>Baryons</b>	
p	uud
n	udd
$\Lambda^0$	uds
$\Sigma^+$	uus
$\Sigma^0$	uds
$\Sigma^-$	dds
$\Xi^0$	uss
$\Xi^-$	dss
$\Omega^-$	sss

and  
many others...

a Theory is  
*'CONCISE'*

but its consequences are

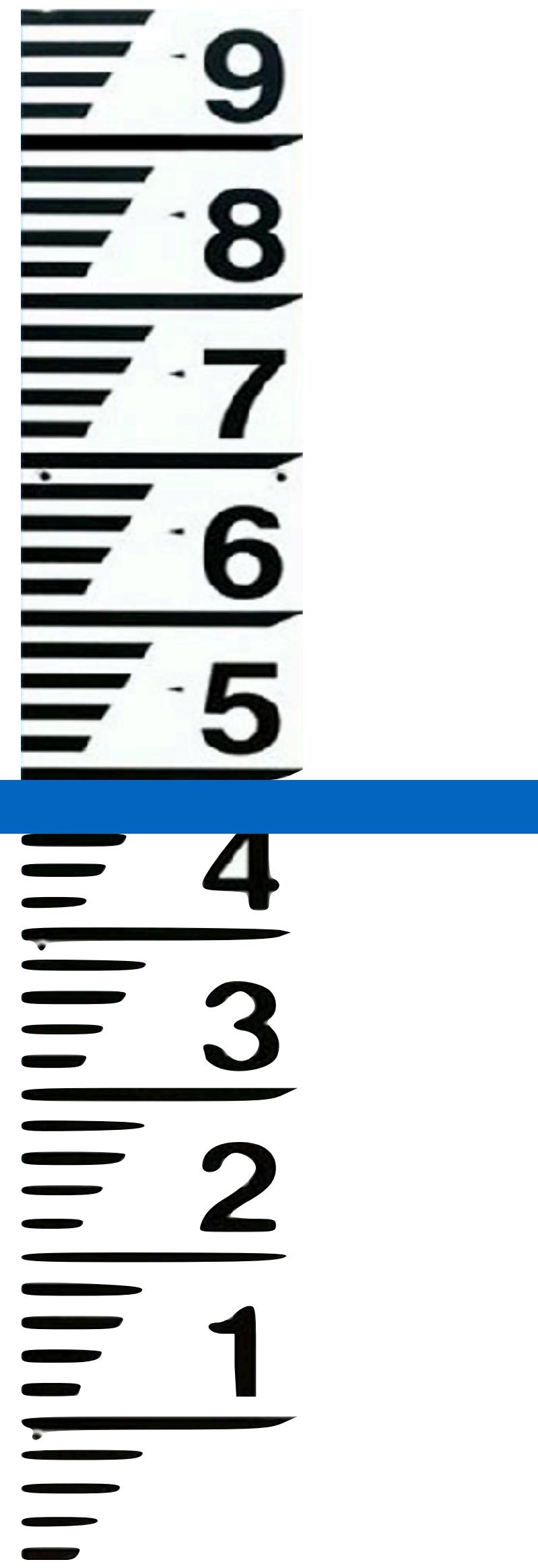
**EXTENSIVE**

# Surface of the waterline

requirements

design

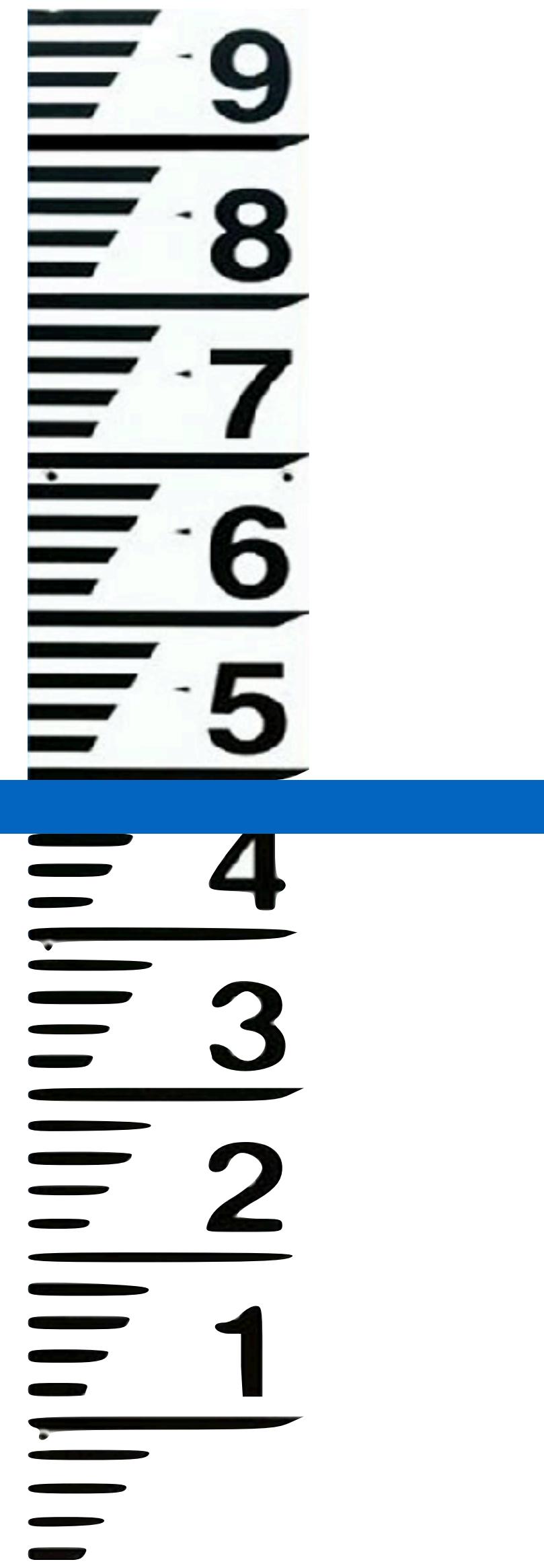
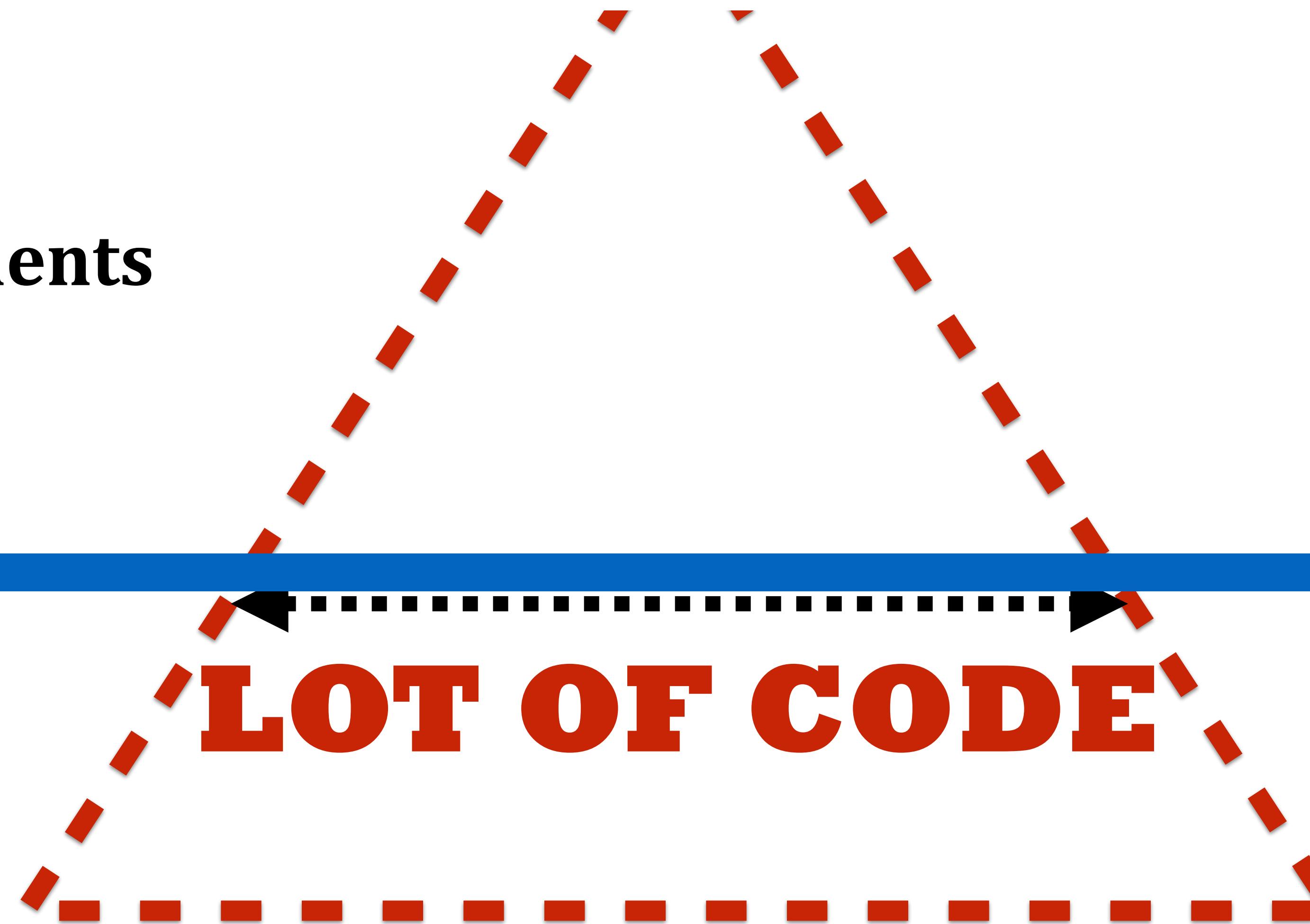
**EXTENSIVE**



# Surface of the waterline

requirements

design

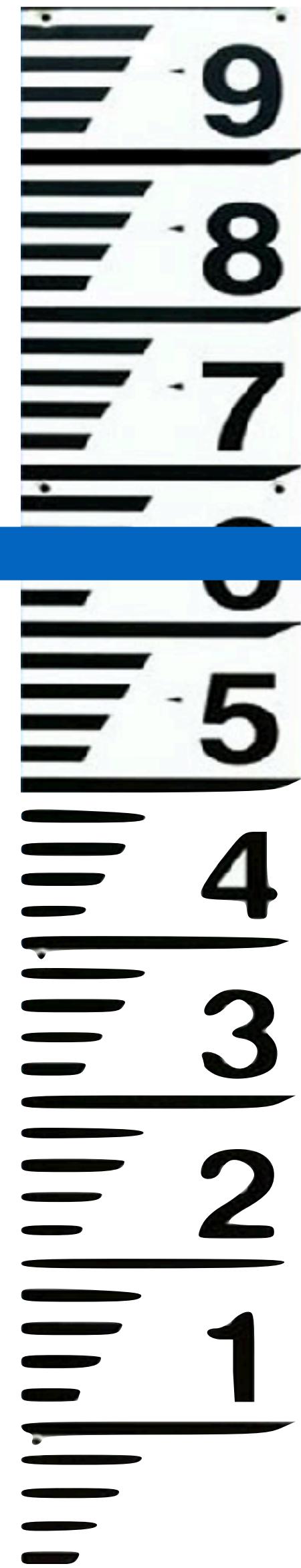
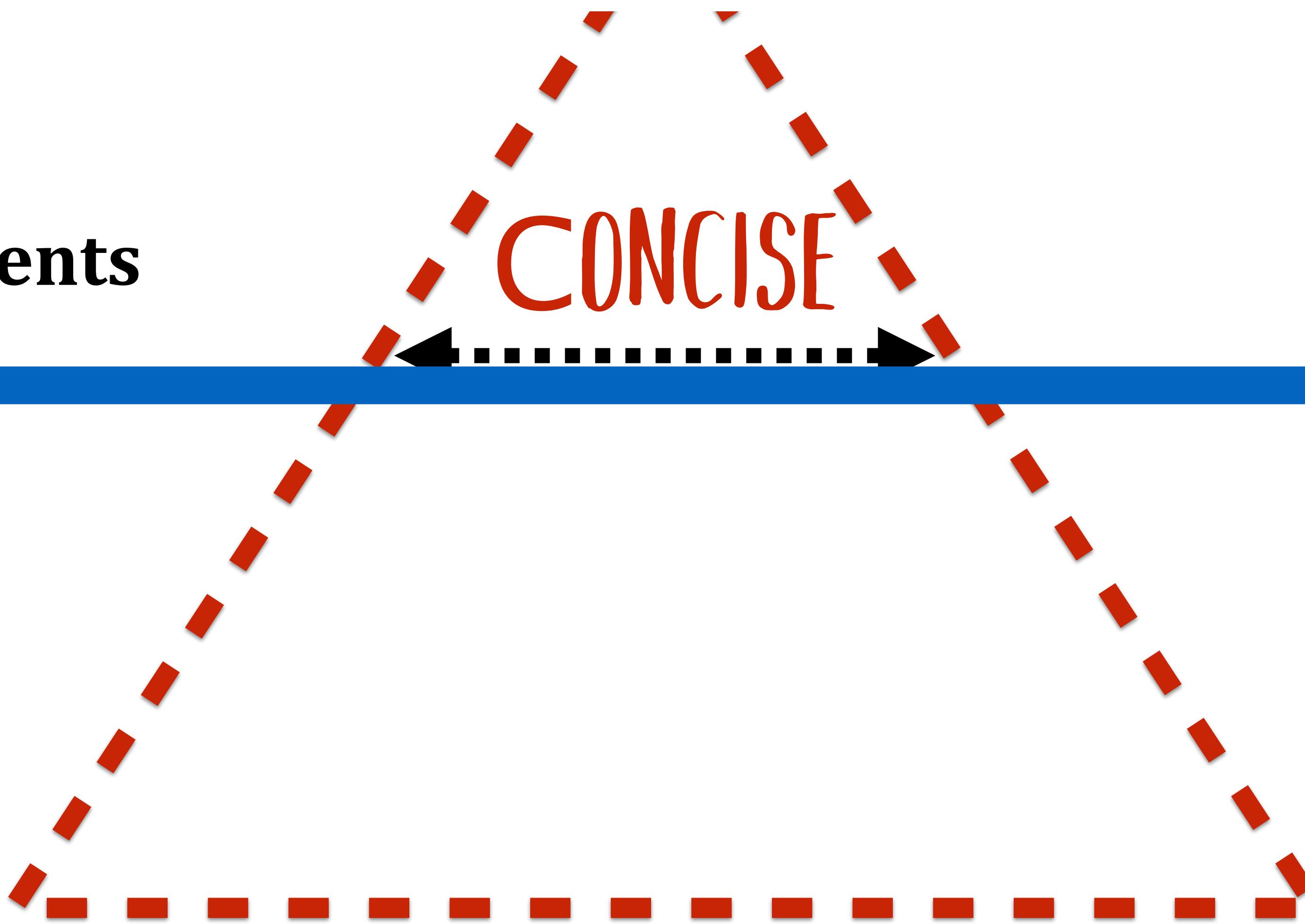


# Surface of the waterline

requirements

design

CONCISE



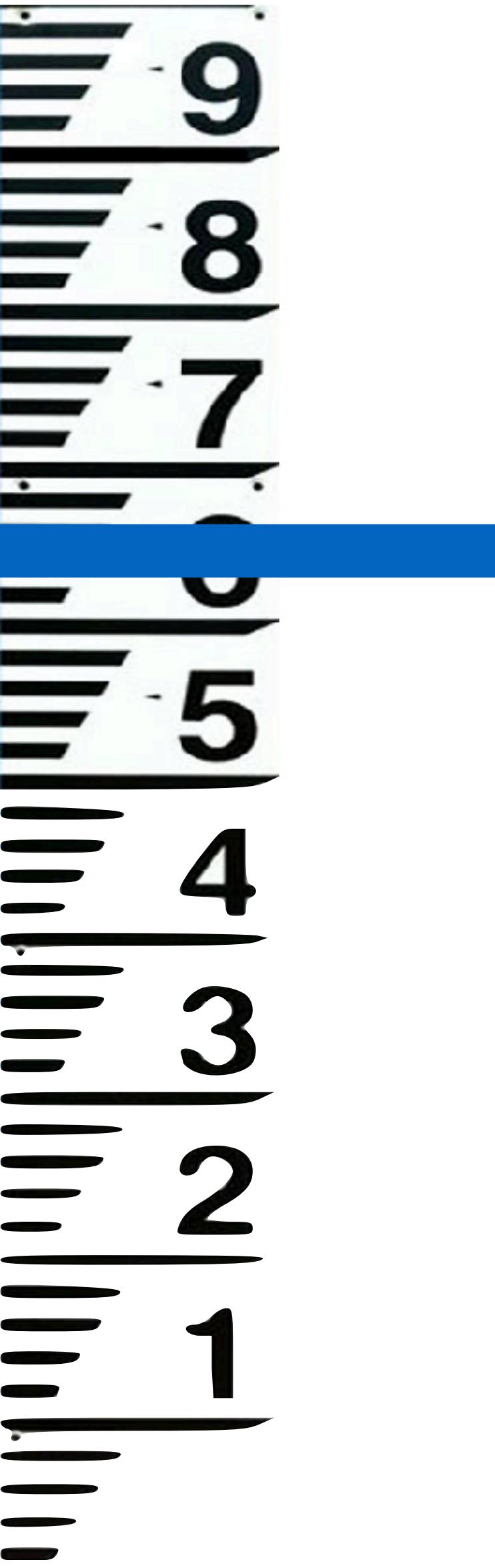
# Surface of the waterline

requirements

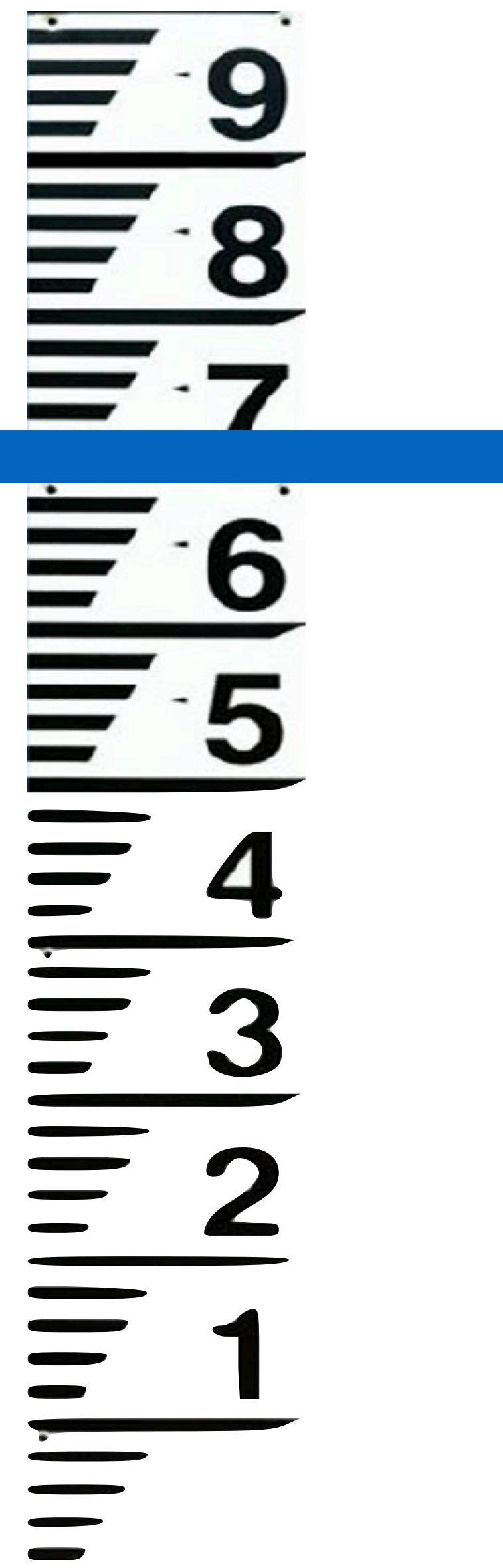
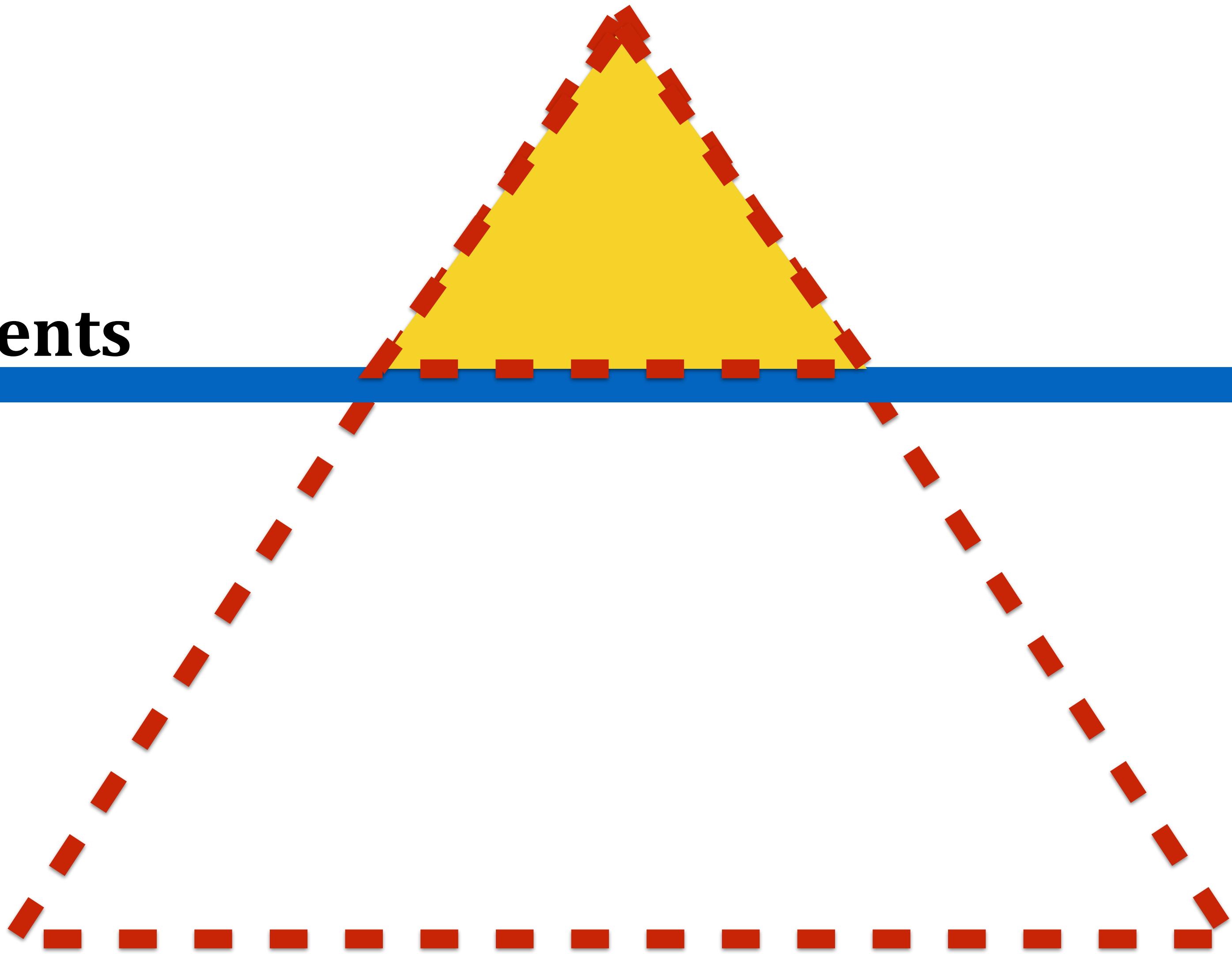
design

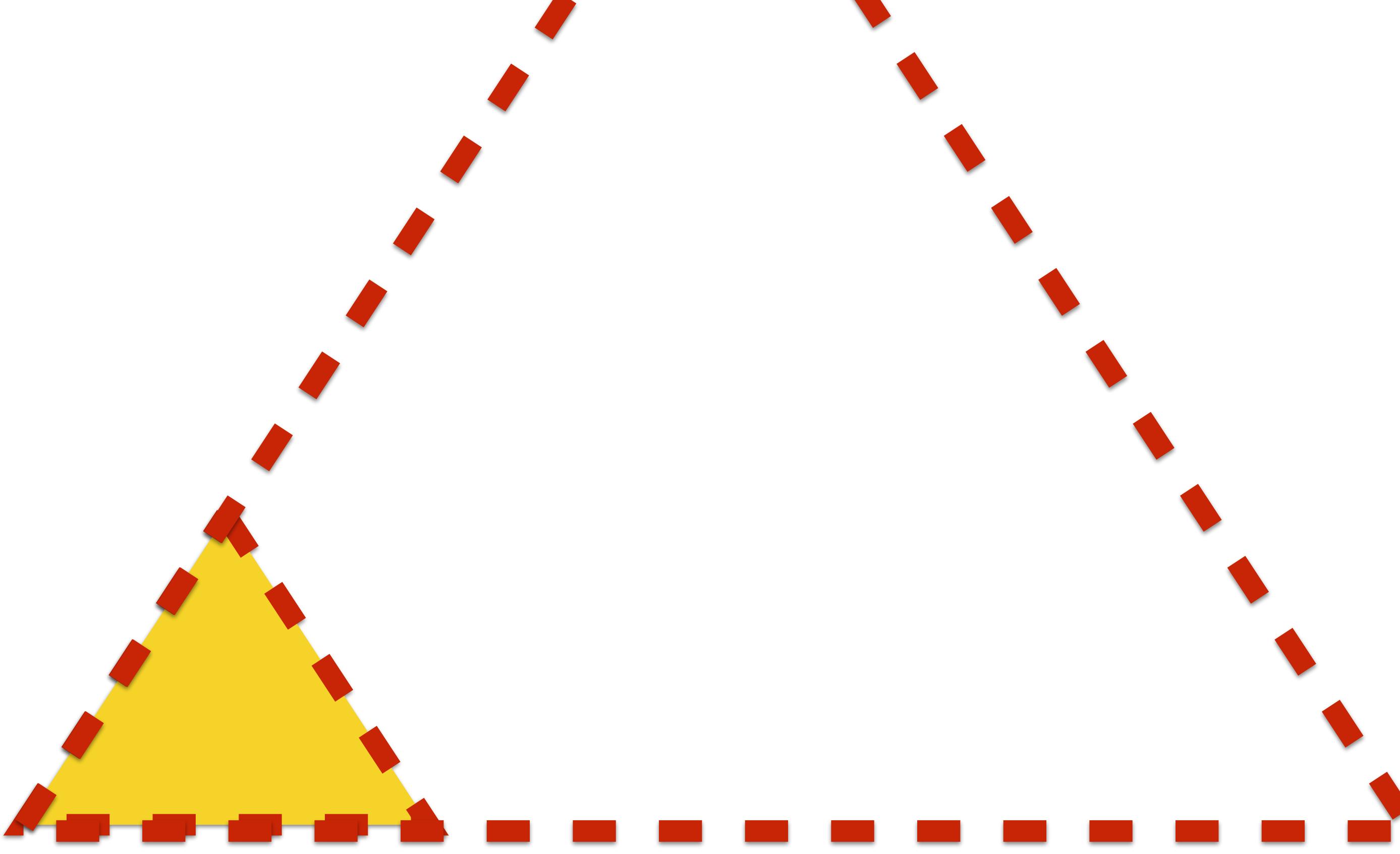
CONCISE

FEWER CODE



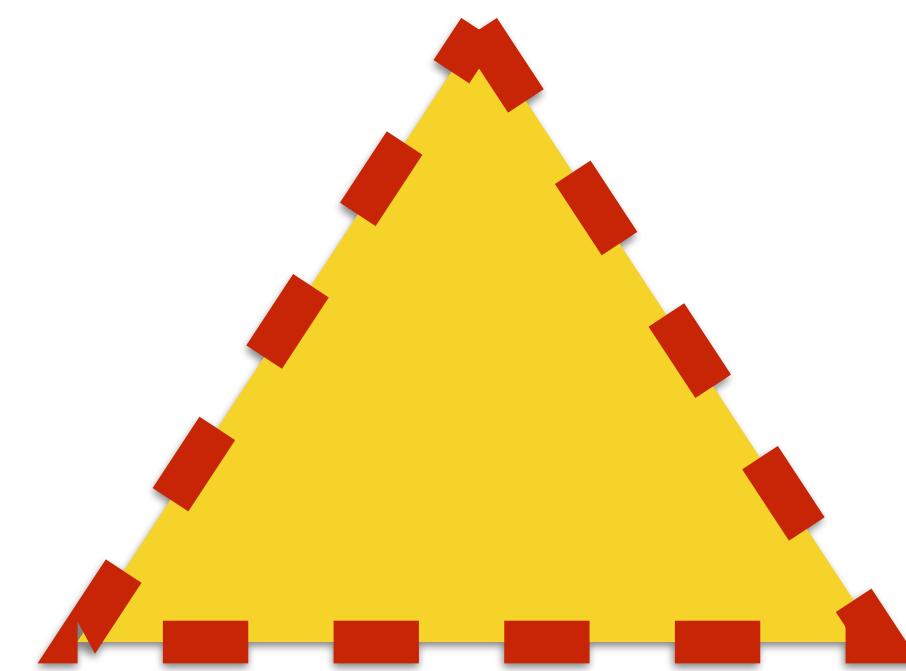
**requirements**  
**design**



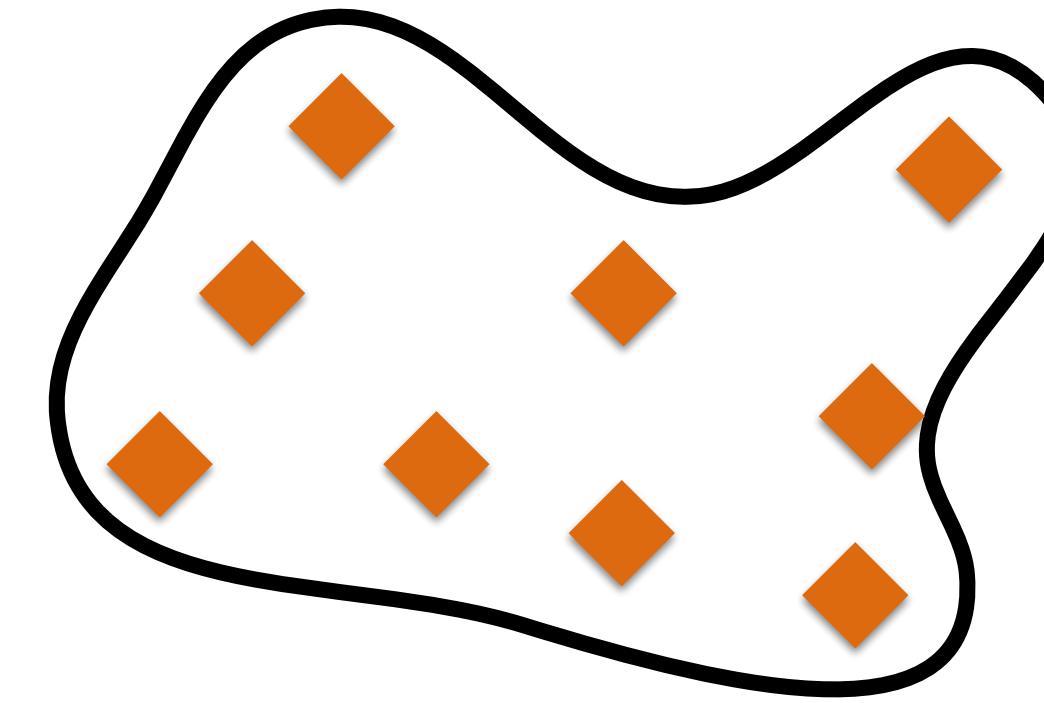


# Theory

# But many business domains are messy



+



Regularities  
Found

Residual  
Mess

# Arbitrary = Messy Domains

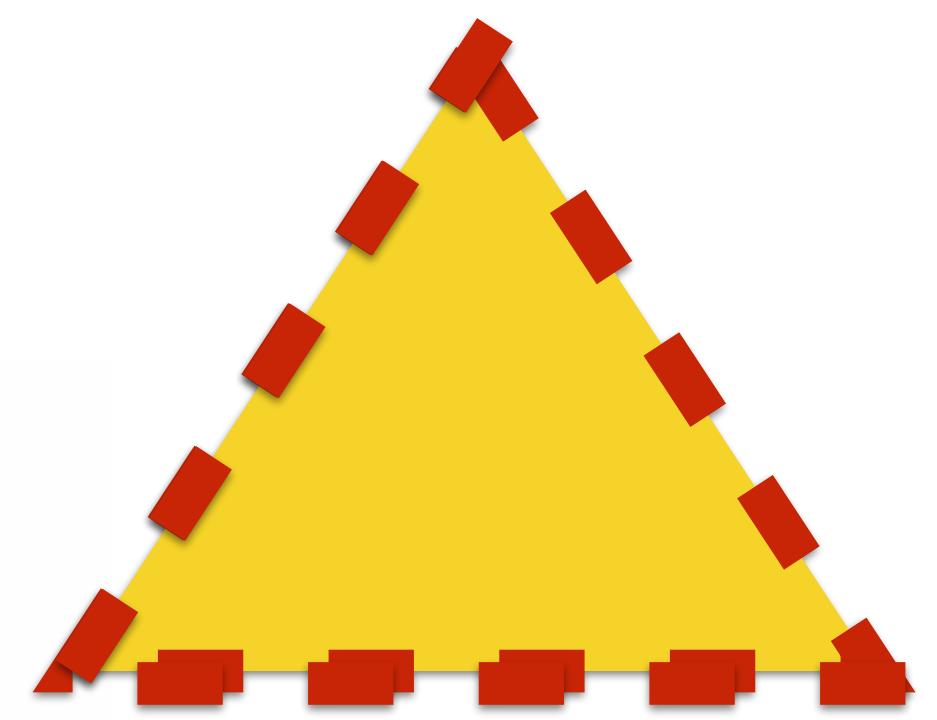
*The Spill Zone pattern: to cover the remaining messy cases, e.g. a function `casesWeHaventFiguredOut()` with only IF-THEN inside – Eric Evans at DDDEU 2017*

(like Circuit Breaker,  
but for design)



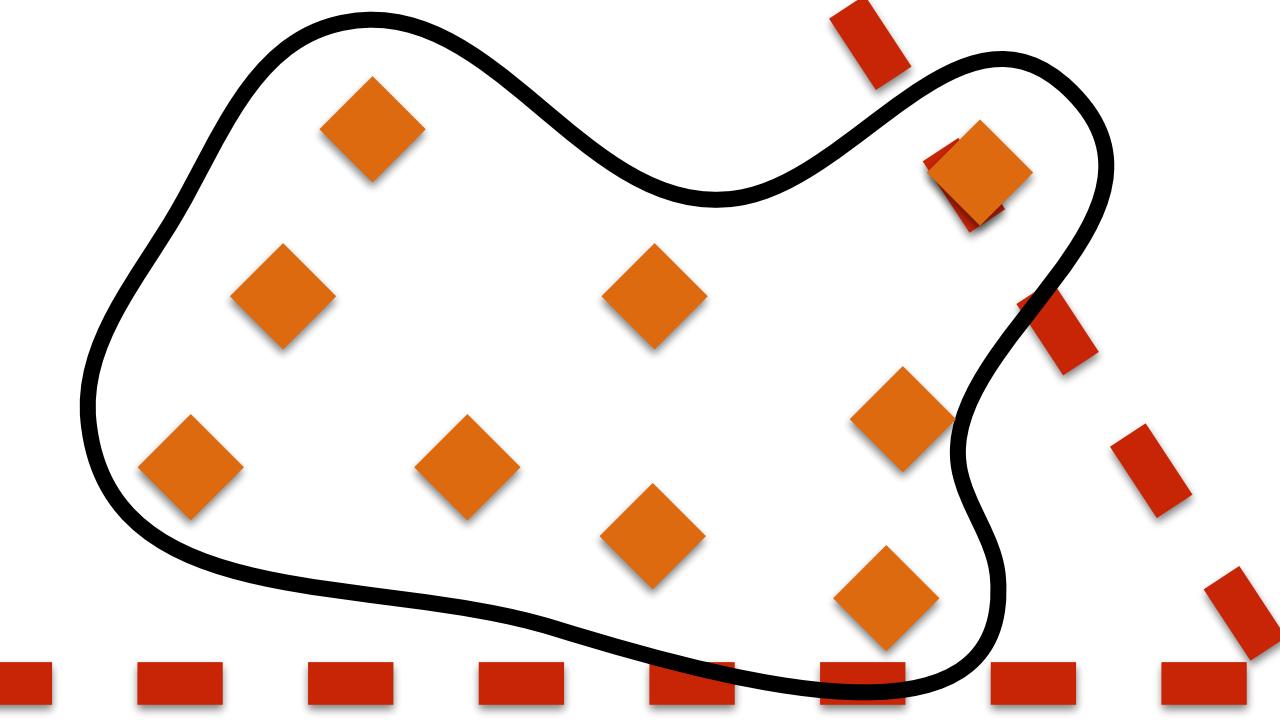
**WIN!**

Theory

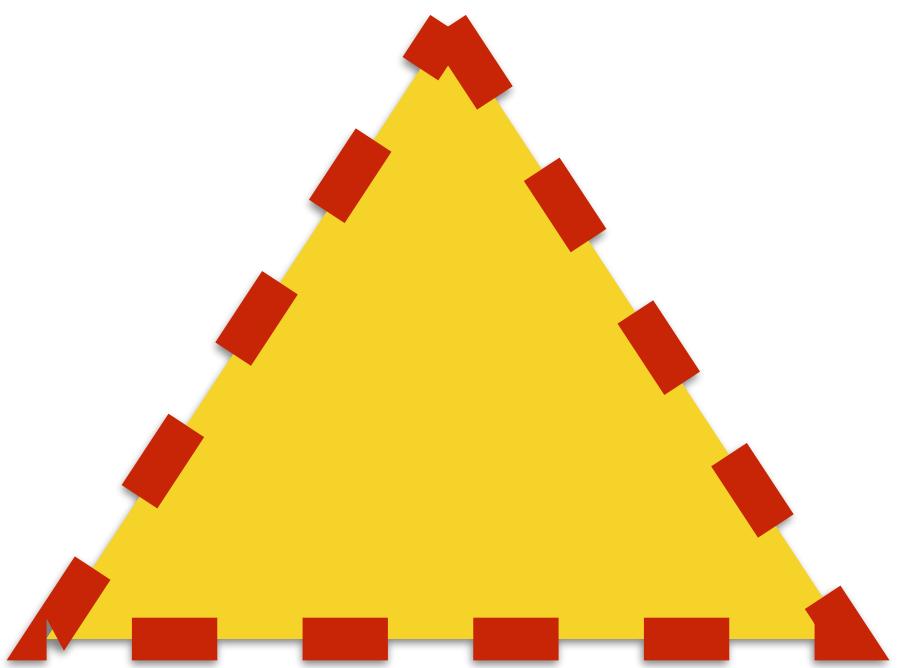


+

Residual  
Mess

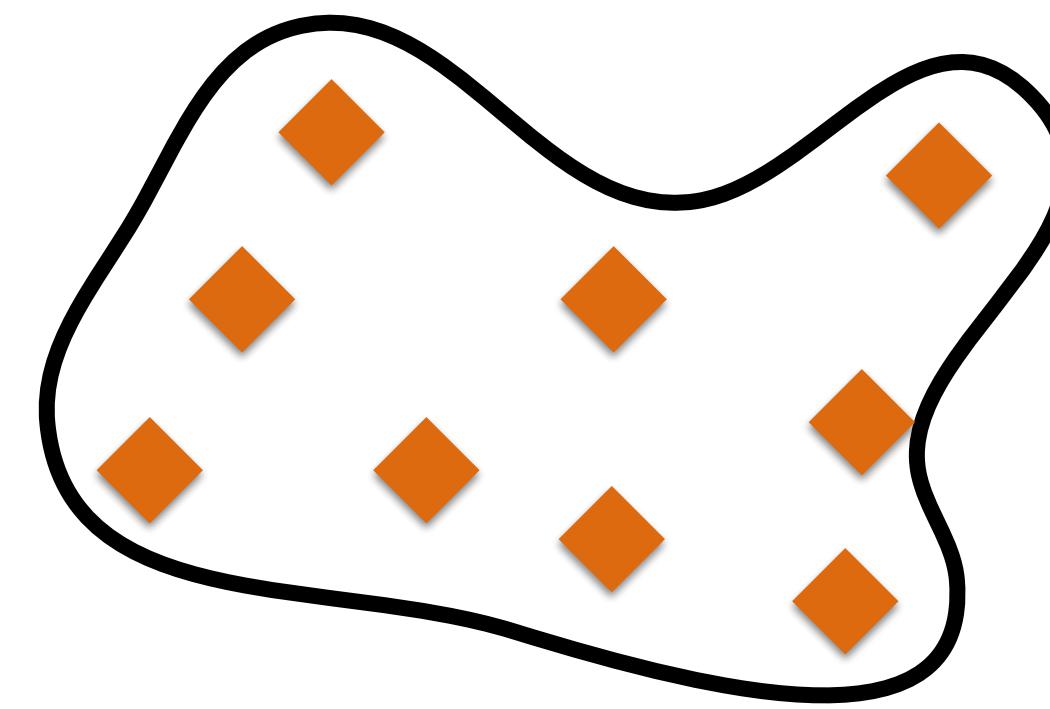


**SMALL**



+

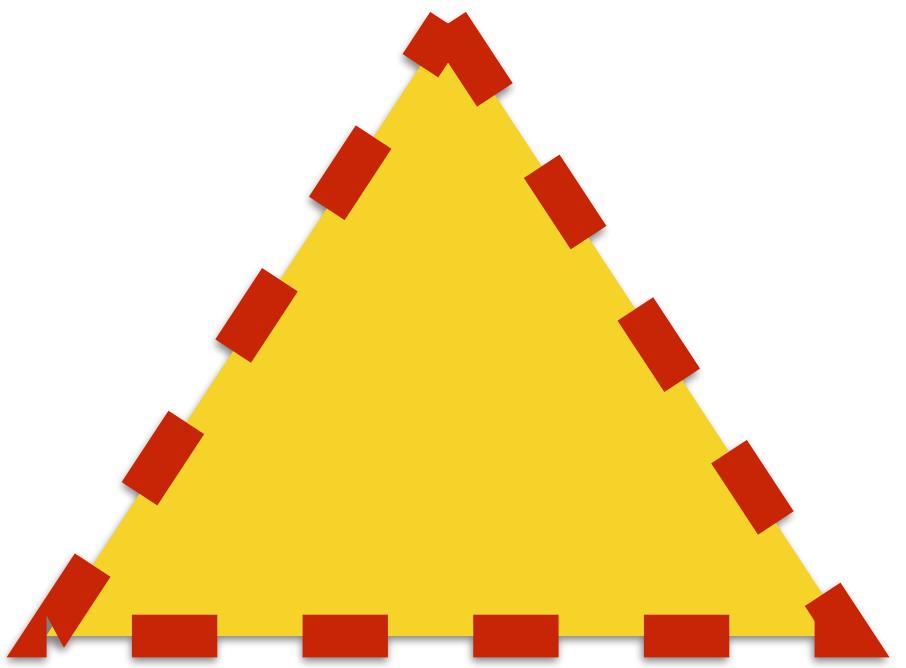
**SMALL**



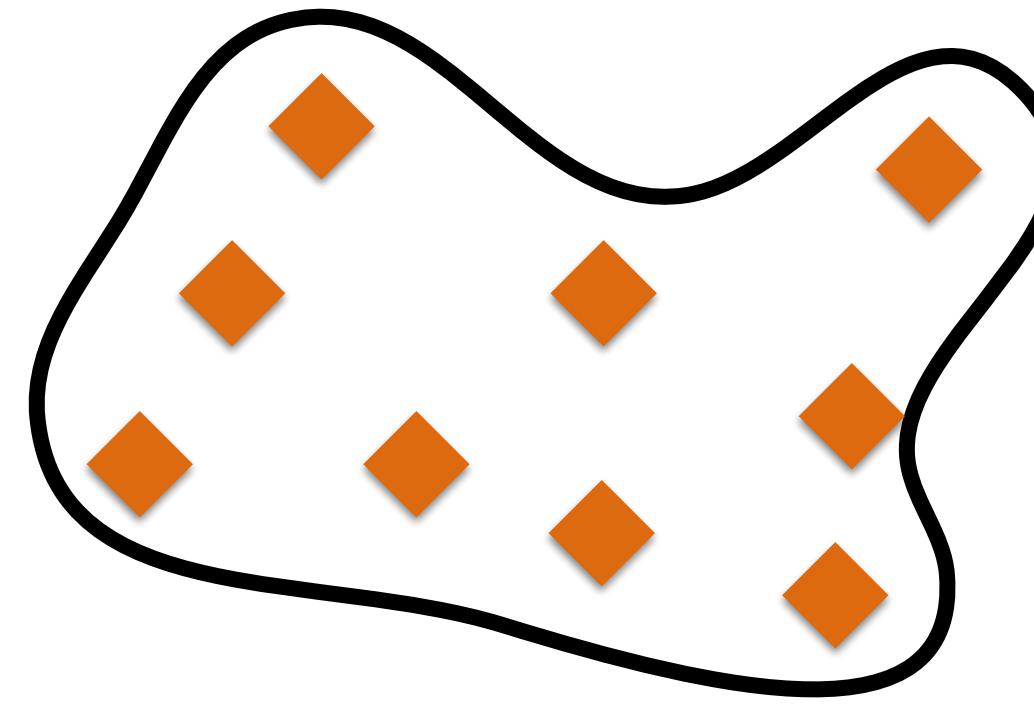
**Theory**

**Residual  
Mess**

**TESTABLE**



**TESTABLE**

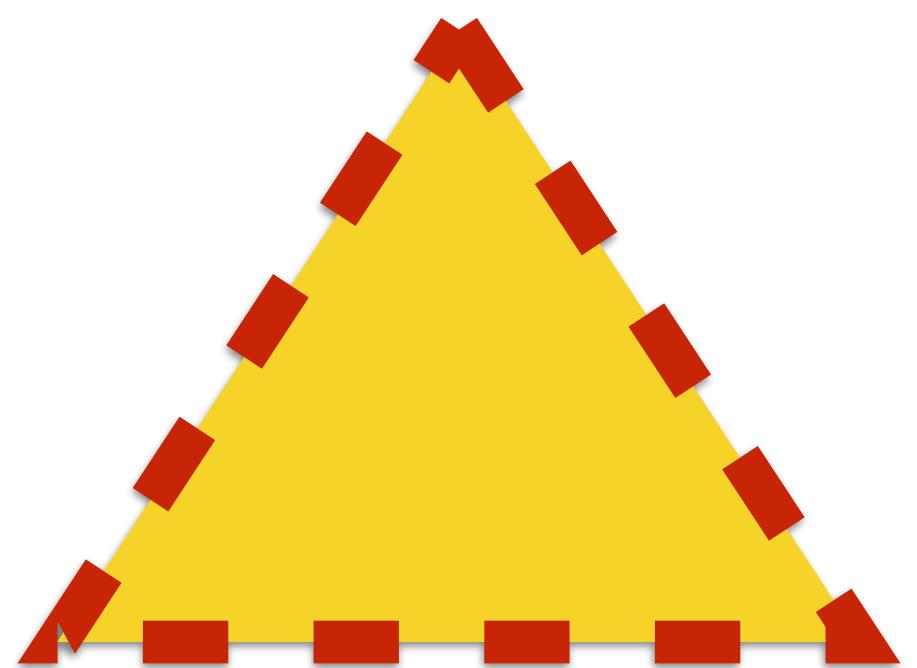


+

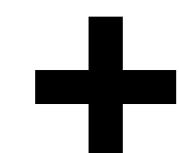
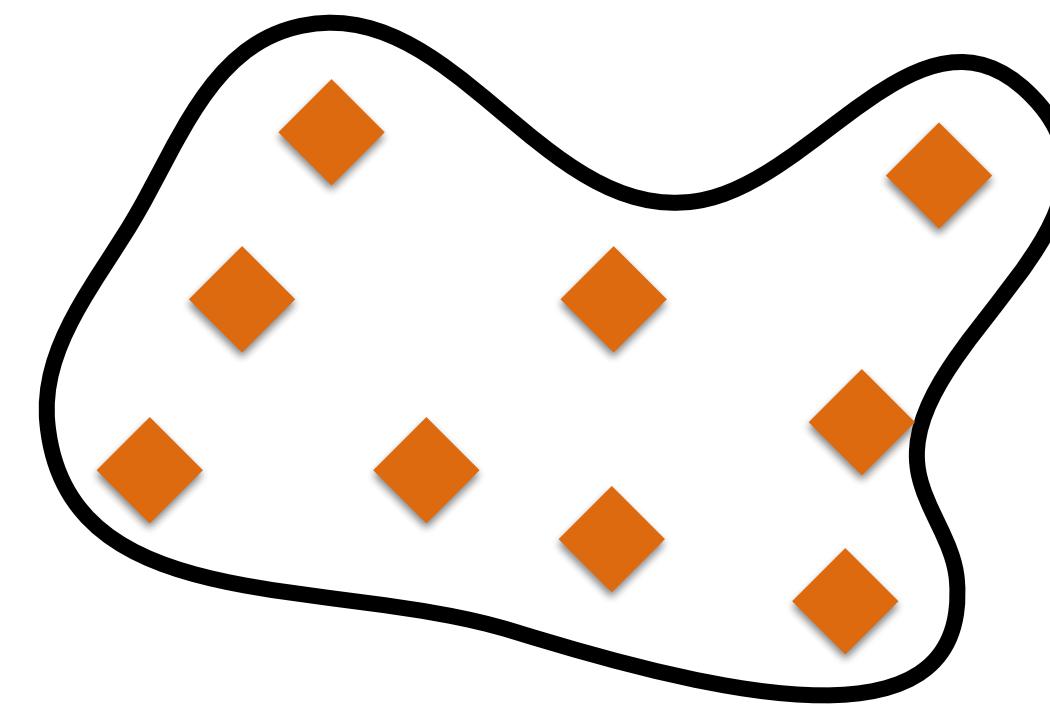
**Theory**

**Residual  
Mess**

**MAINTENABLE**



**MAINTENABLE**

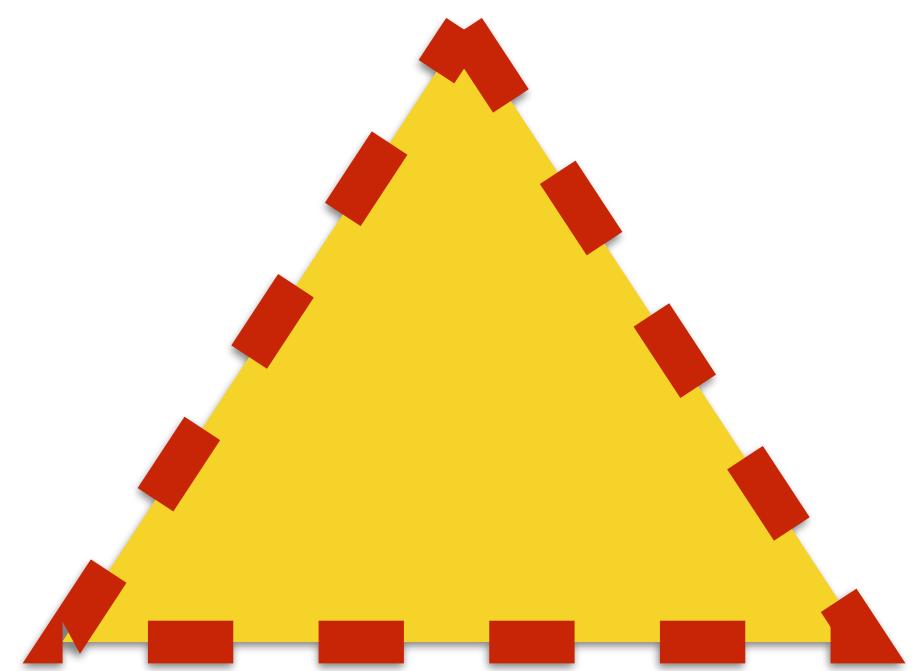


**Theory**

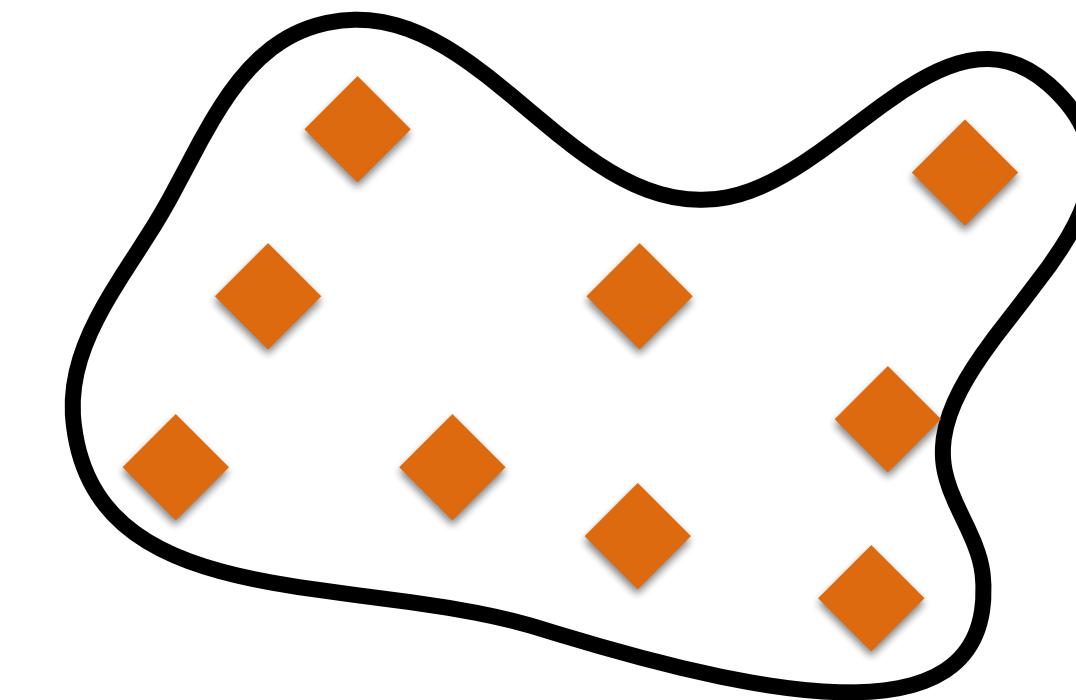
**Residual  
Mess**

**CHALLENGEABLE!**

**CHALLENGEABLE!**



+



**Theory**

**Residual  
Mess**

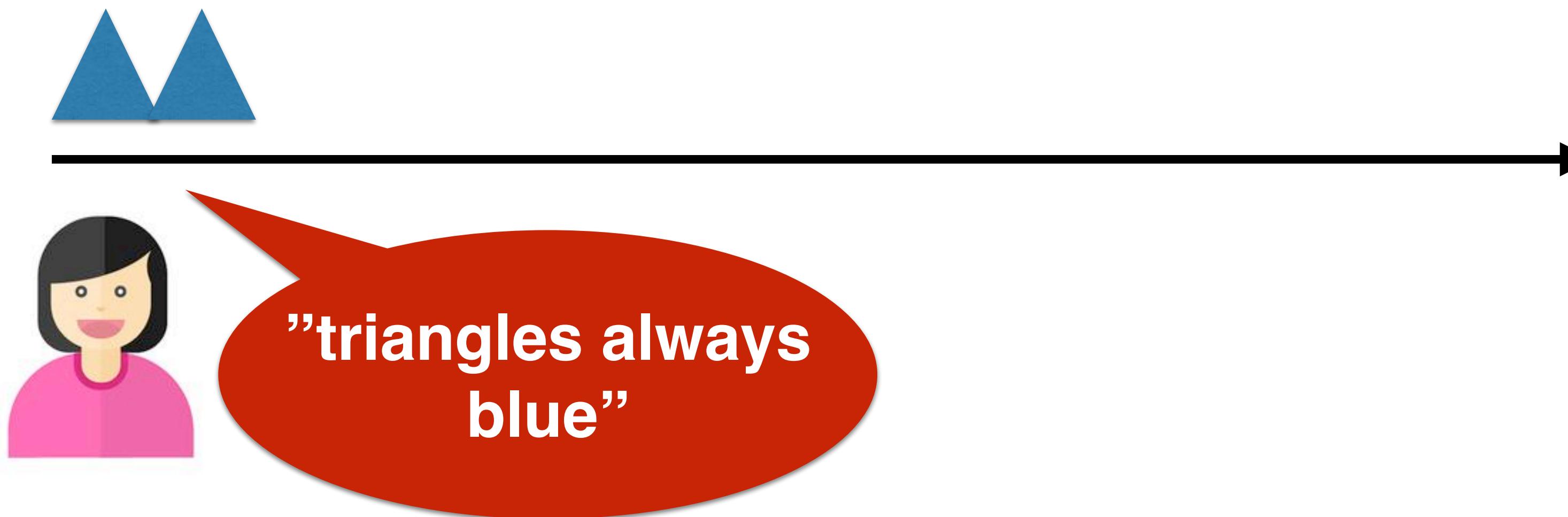


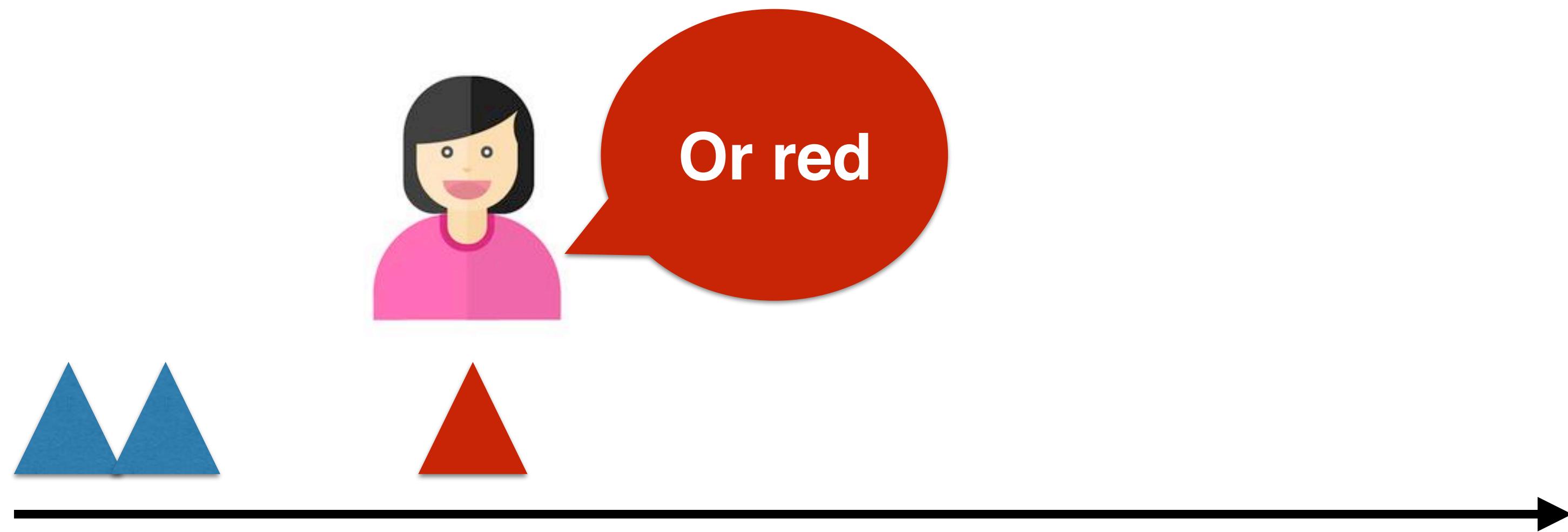
**But to reverse-engineer it all...**



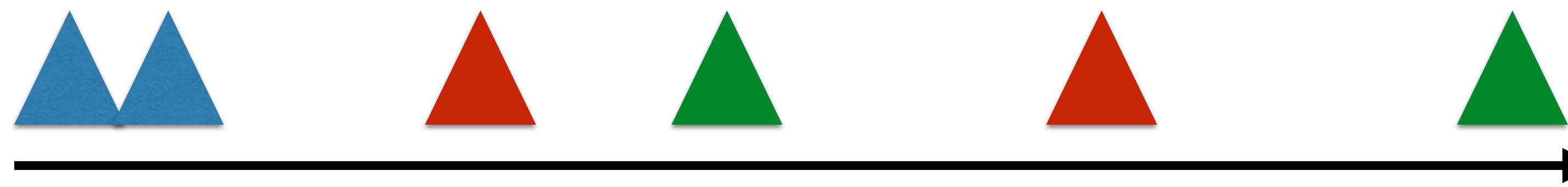
...you must have the whole  
picture in mind.

**MIND THE  
DISCOVERY PATH**





# Discovery with the full memory

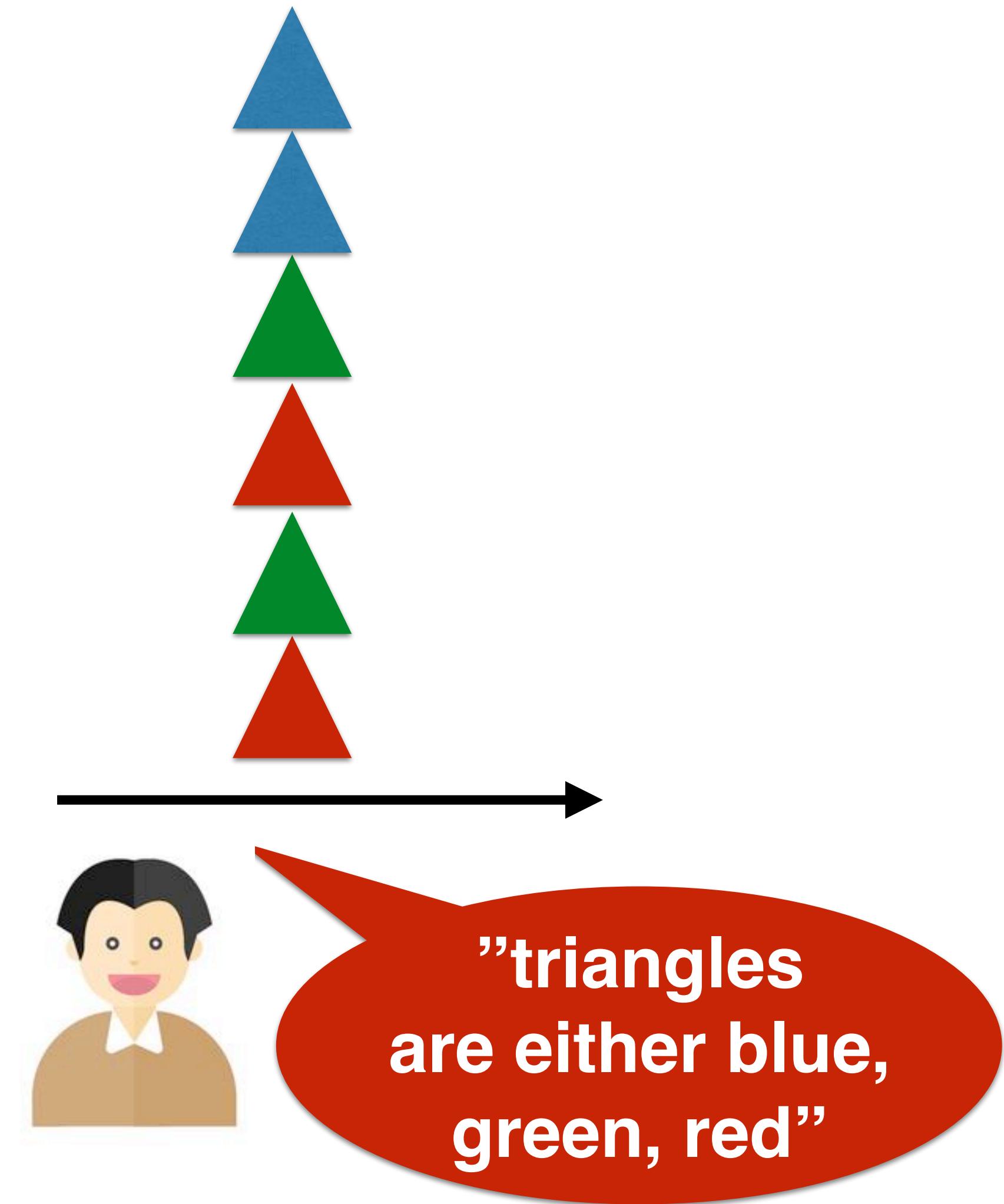


“triangles  
are either blue,  
green, red”

# One case at a time: no insight

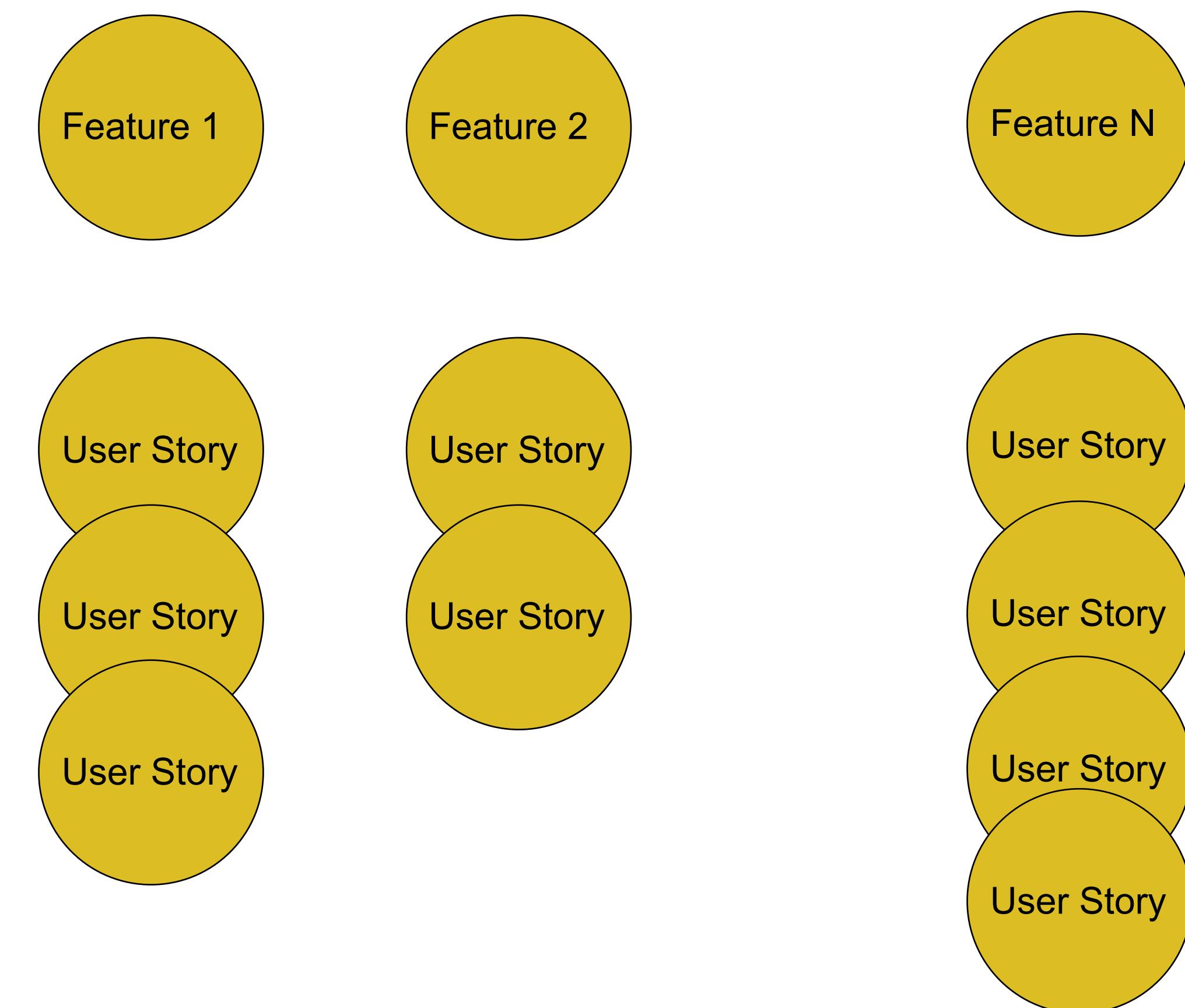


# Observing all diversity at once makes insights obvious

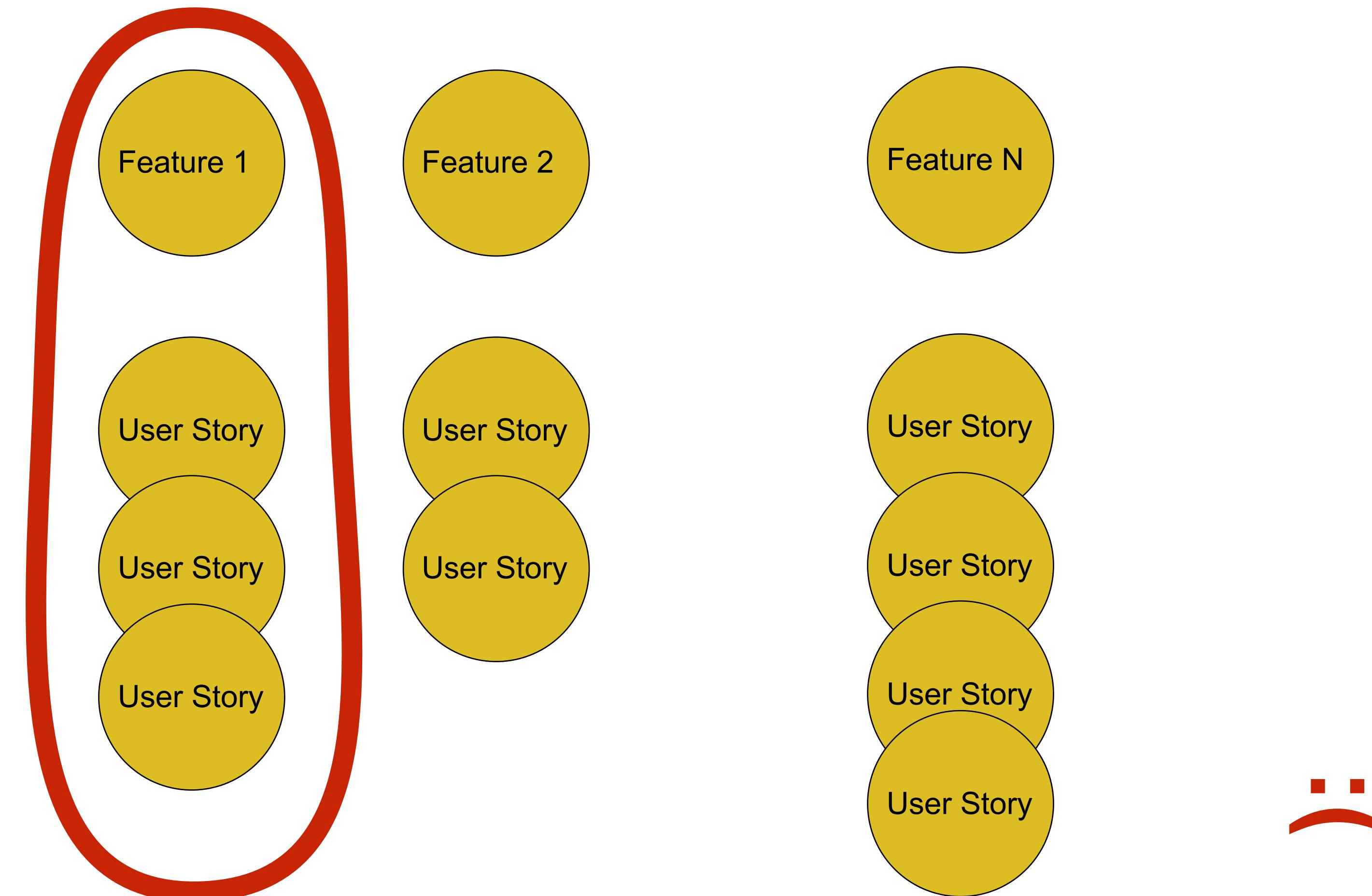


Observing the  
**DIVERSITY** at once  
makes insights  
obvious

# Story map

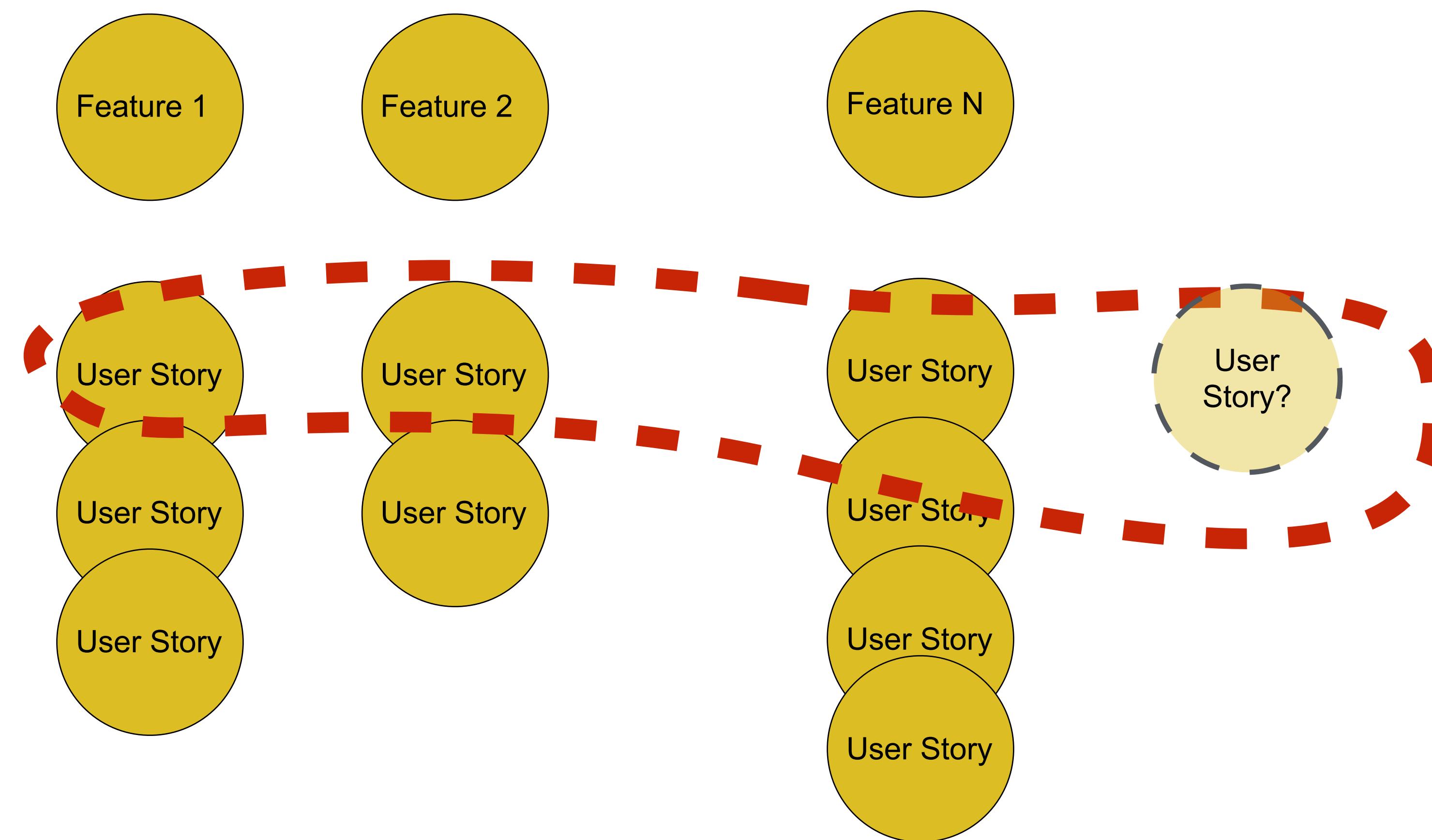


# Story map



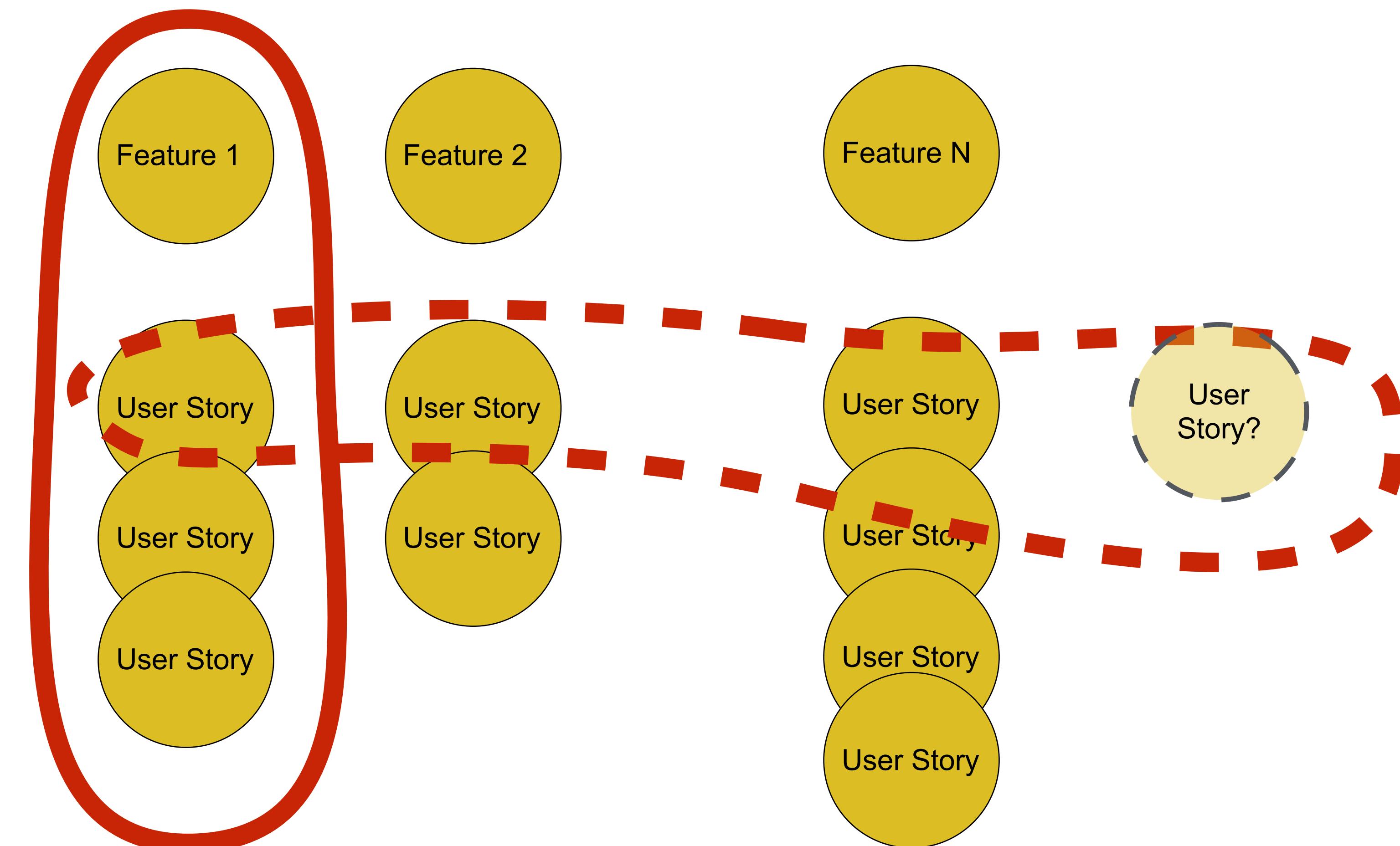
**INITIAL DESIGN ON 1 SINGLE CASE**

# Story map



**EXPLORING QUICKLY WHAT'S NEXT**

# Story map



A BIT OF BOTH



# T-SHAPED EXPLORATION

# LIKE IN TDD!

*Driving the  
implementation and  
its design by ordering  
the tests strategically*

# BUT CYRILLE...

*We're agile, we should  
be able to change  
anything anytime!*



**First layer of bricks matters more...**



**This should not be true but it is,  
for many bad reasons.**

**SERIOUSLY**



**I'M FED UP**

*Memes Happen*

**The worst domain expert is the one whose expertise was built from the intricacies of the existing systems**



**LEGACY  
CORRUPTION**



“We've always  
done like that!”

# Legacy Corruption

People distorted their business concepts to cope with the constraints of their legacy software

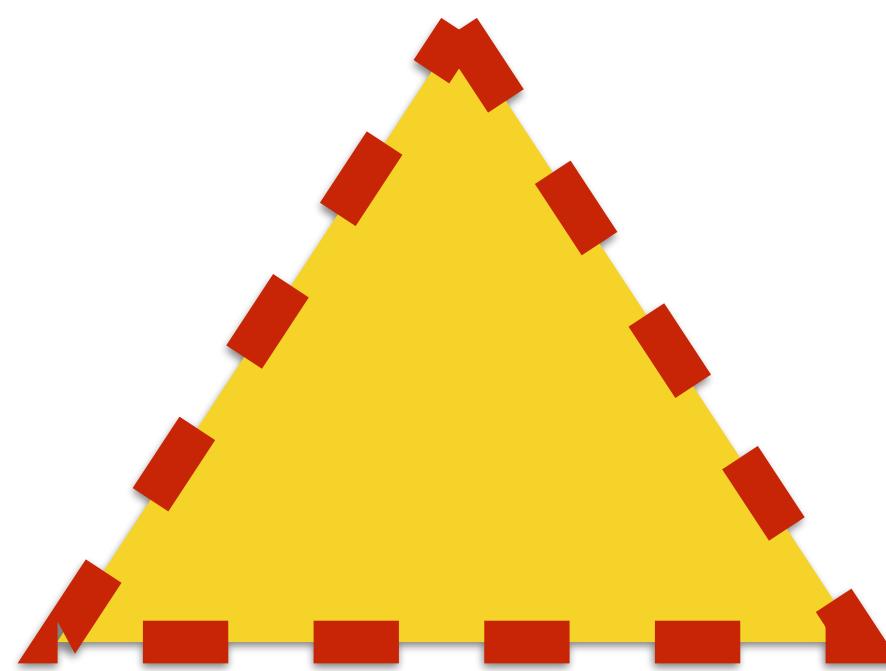
*"A Tricycle is a kind of Truck because only the type Truck has the field to specify the 'number of wheels'"*

# Legacy Assumptions

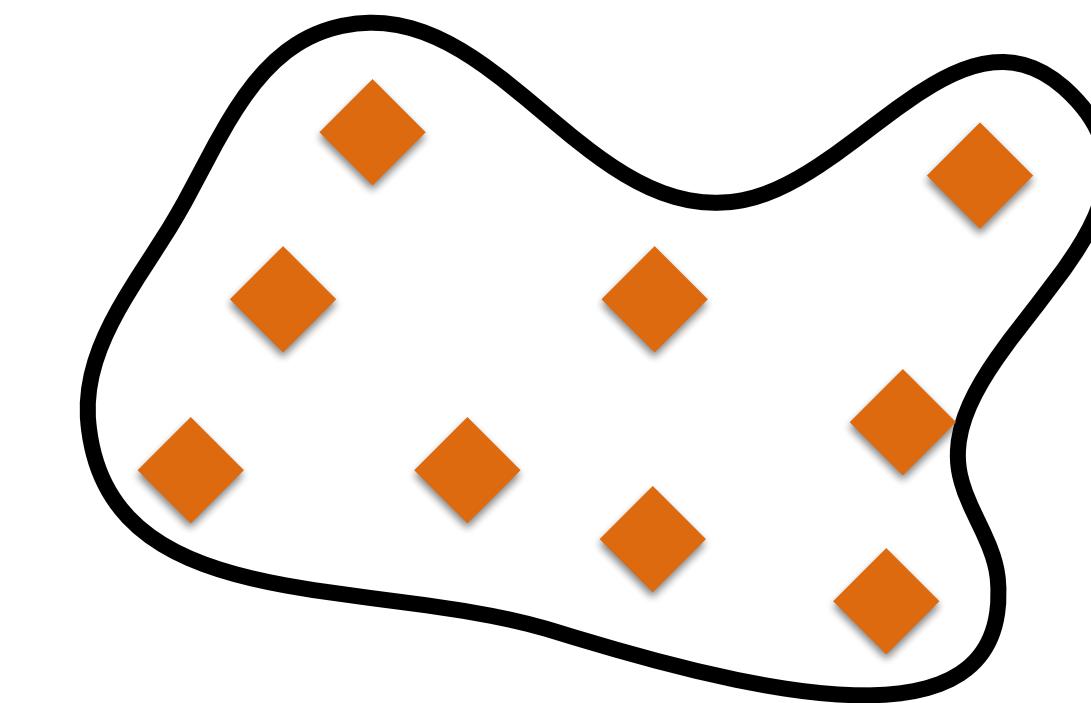
**BATCHES** take a lot of time and fail often, so we need to store all data at each step for human inspection and recovery (*Data Obsession*)

**ANTIDOTE**

**DETOX**



+



Theory

Residual  
Mess



+ LEGACY  
CORRUPTION



**SERIOUSLY**



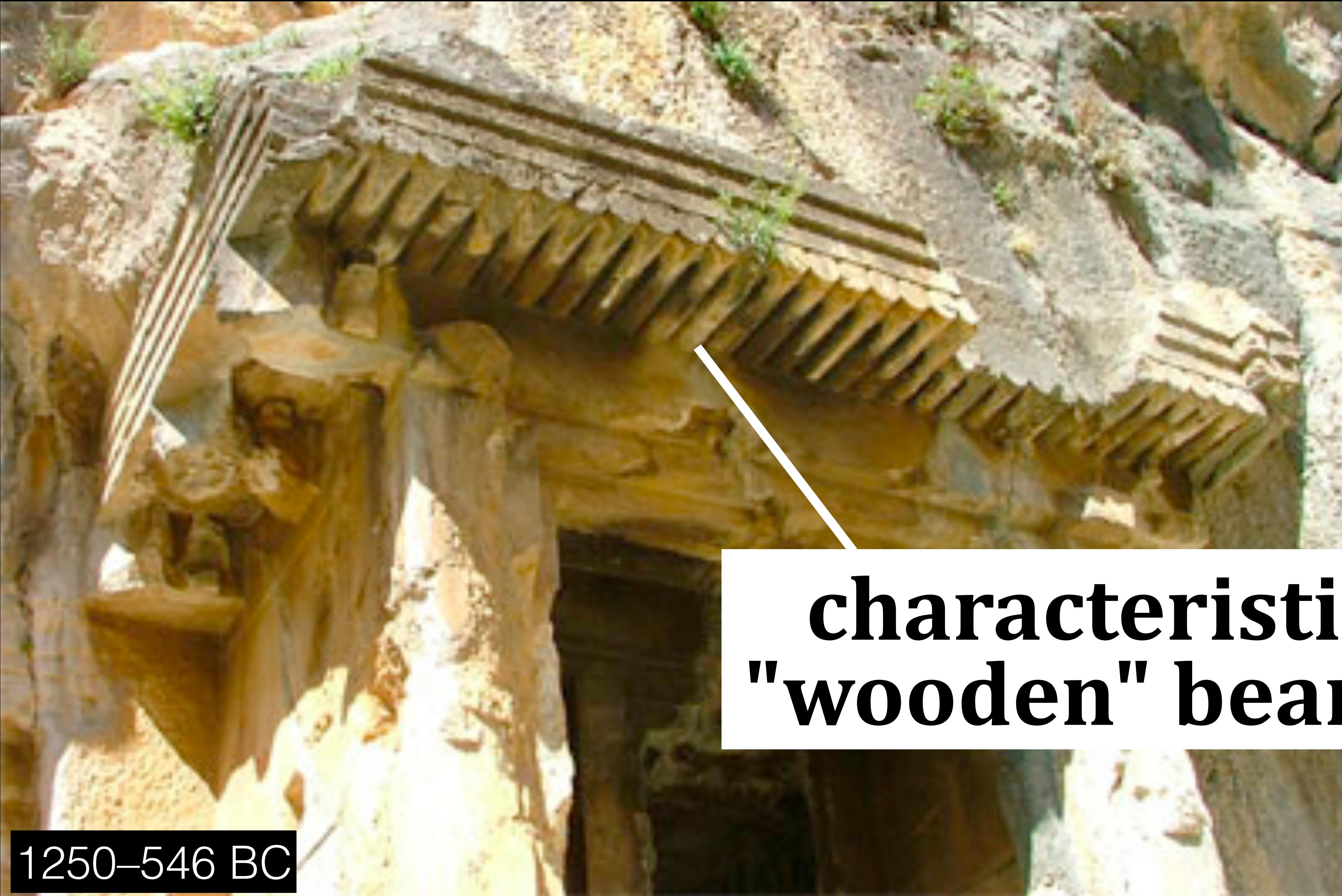
**I'M FED UP**

*Memes Happen*

**SKEUOMORPHIC  
MODEL**



**”Preservation of the familiar  
through technological change”**



1250–546 BC

characteristic  
"wooden" beams

**At some point we have to learn to  
rid of skeuomorphism**

# PATH-DEPENDENT EVOLUTION

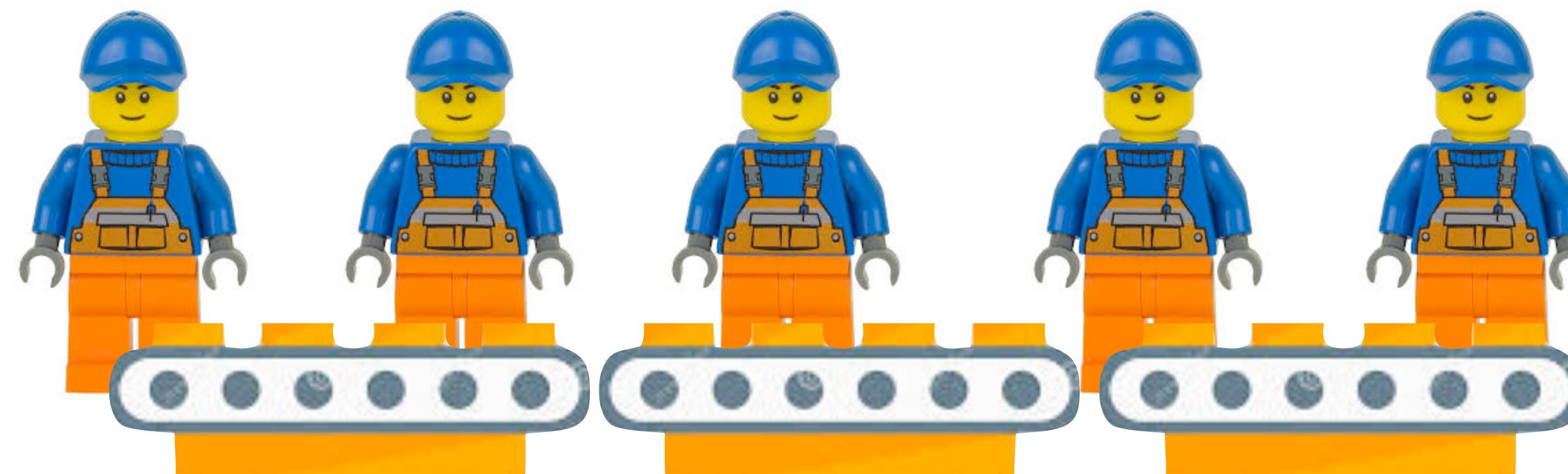
# Long ago



# Mass production



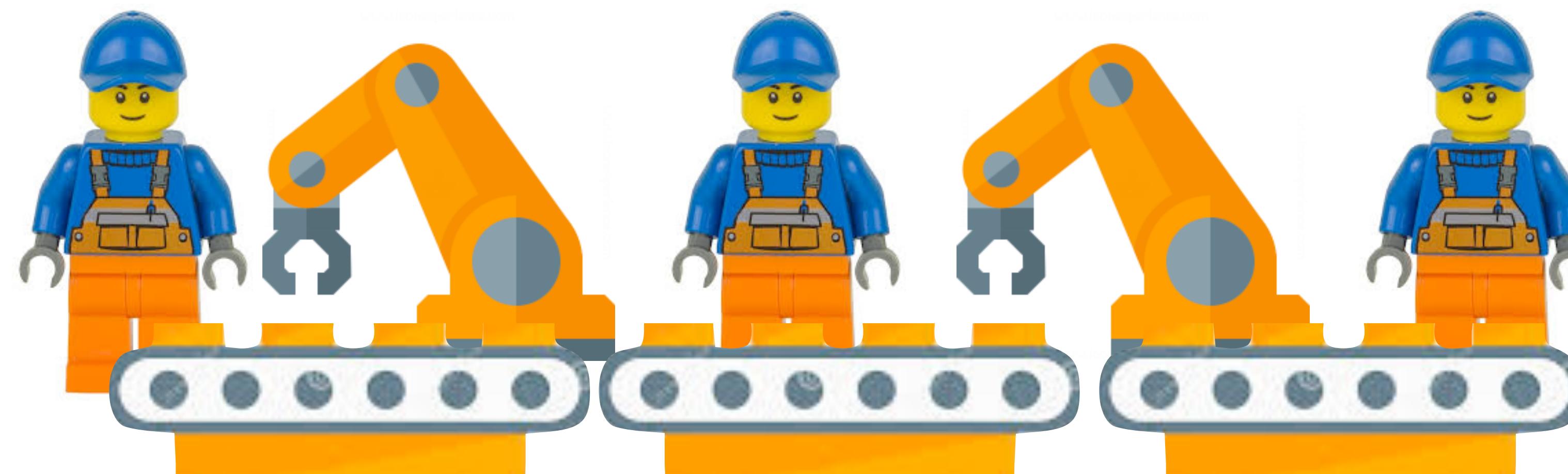
# The factory line



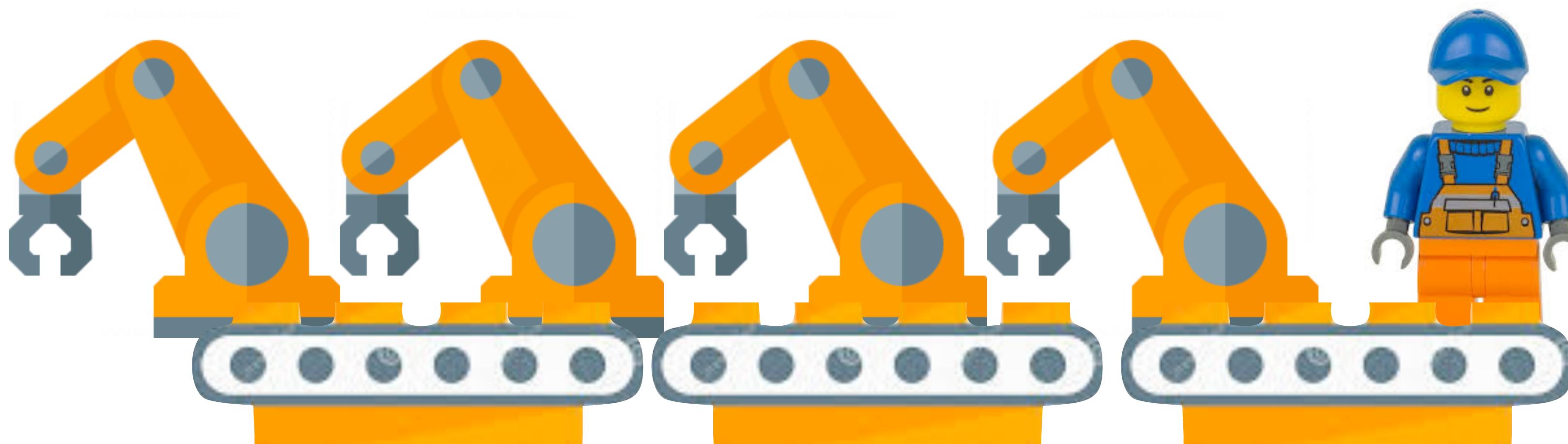
# NEW: robot!



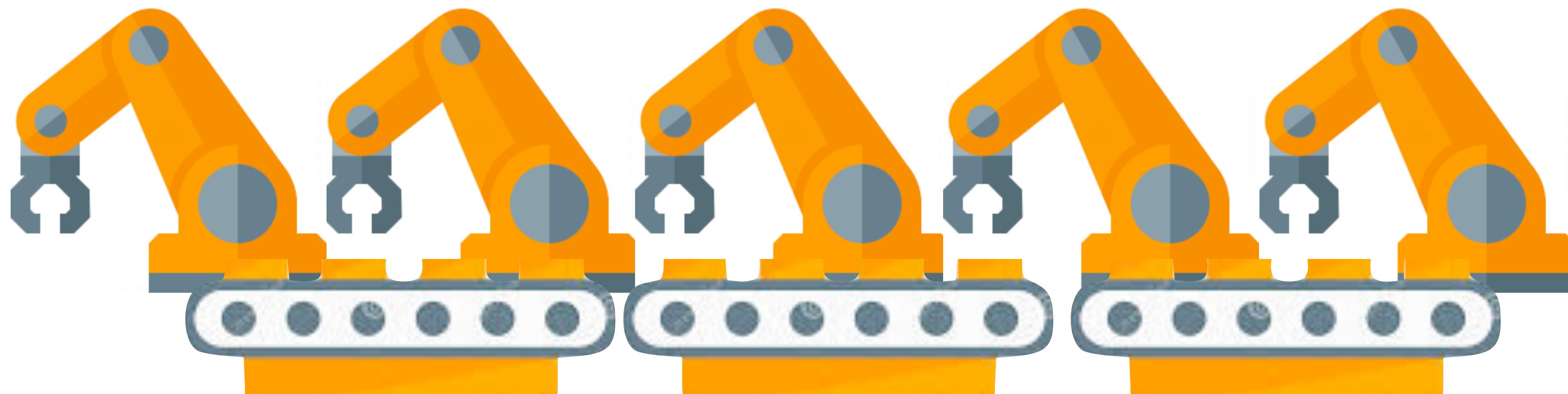
# More robots!!



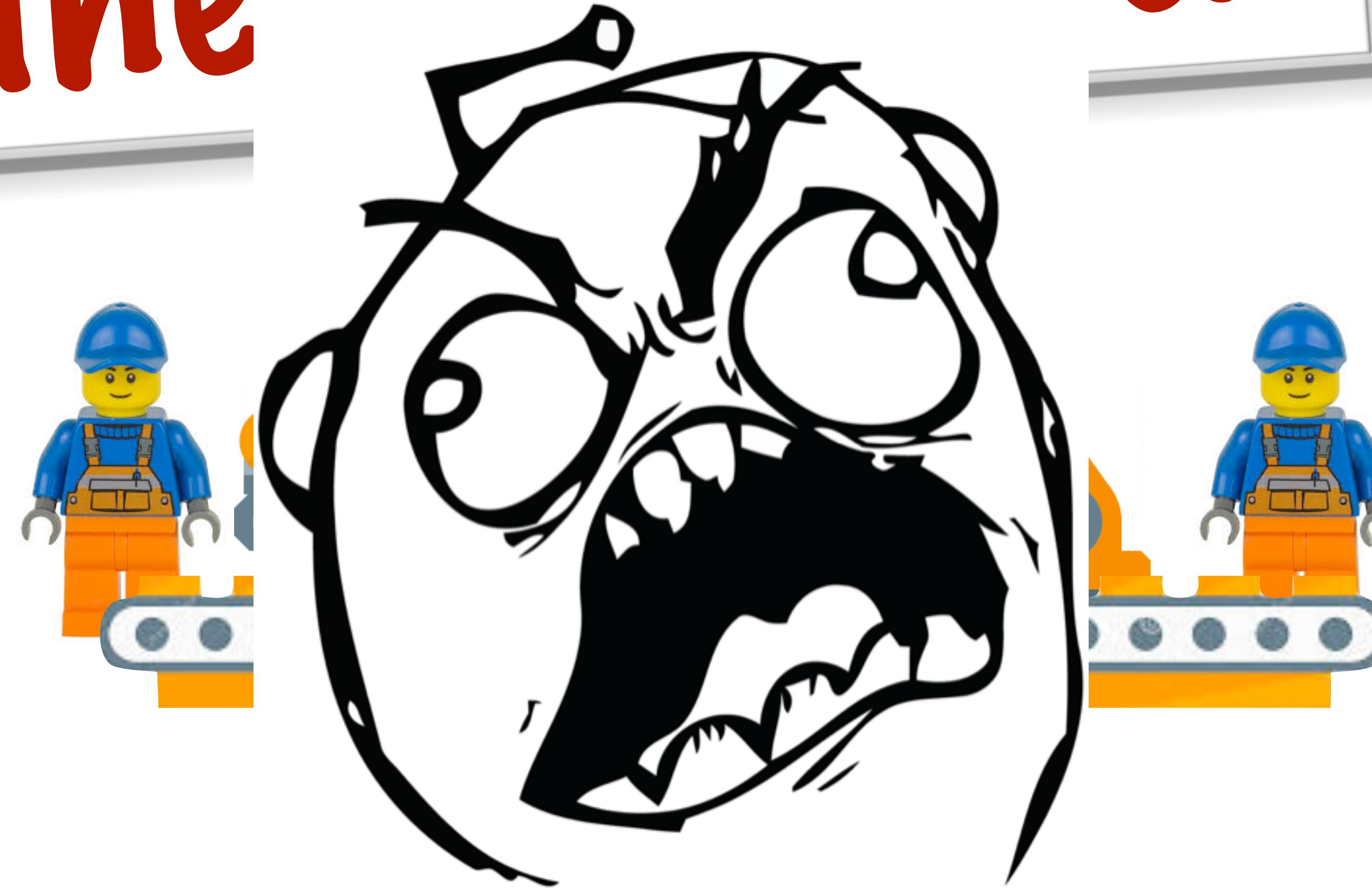
# More robots!!!



# A line of robots...



A line of rhots...



**Sum of LOCAL  
optimizations**

**≠**

**GLOBAL  
Optimization**

A line of robots...

COULD DO  
BETTER  
THAN THIS



**THINK  
DIGITAL-  
NATIVE  
FAB**

# Factory reboot



laser cutter



Versatile  
Robot

# Digital Fab-native



laser cutter



Versatile  
Robot

**BOOXI**  
T  
BOOBUGGY.COM

**SCORPIK**



# Fewer Machines



laser cutter



Versatile  
Robot

# Mass Customization



laser cutter



Versatile  
Robot

**Opportunity to review /  
change the business  
processes**

Theory build on top of  
principles

**FIRST-  
PRINCIPLES**

# first prin·ci·ples

**noun**

noun: **first principle**

the fundamental concepts or assumptions on which a theory, system, or method is based.

"I think we have to start again and go right back to first principles"



Translations, word origin, and more definitions

**Aristotle:** a *first principle* is the "first basis from which a thing is known"

# First Principles Thinking

*Deconstruct then reconstruct.*

*...one of the most effective strategies you can employ for **breaking down complicated problems and generating original solutions.***

# Elon Musk on First Principles



“

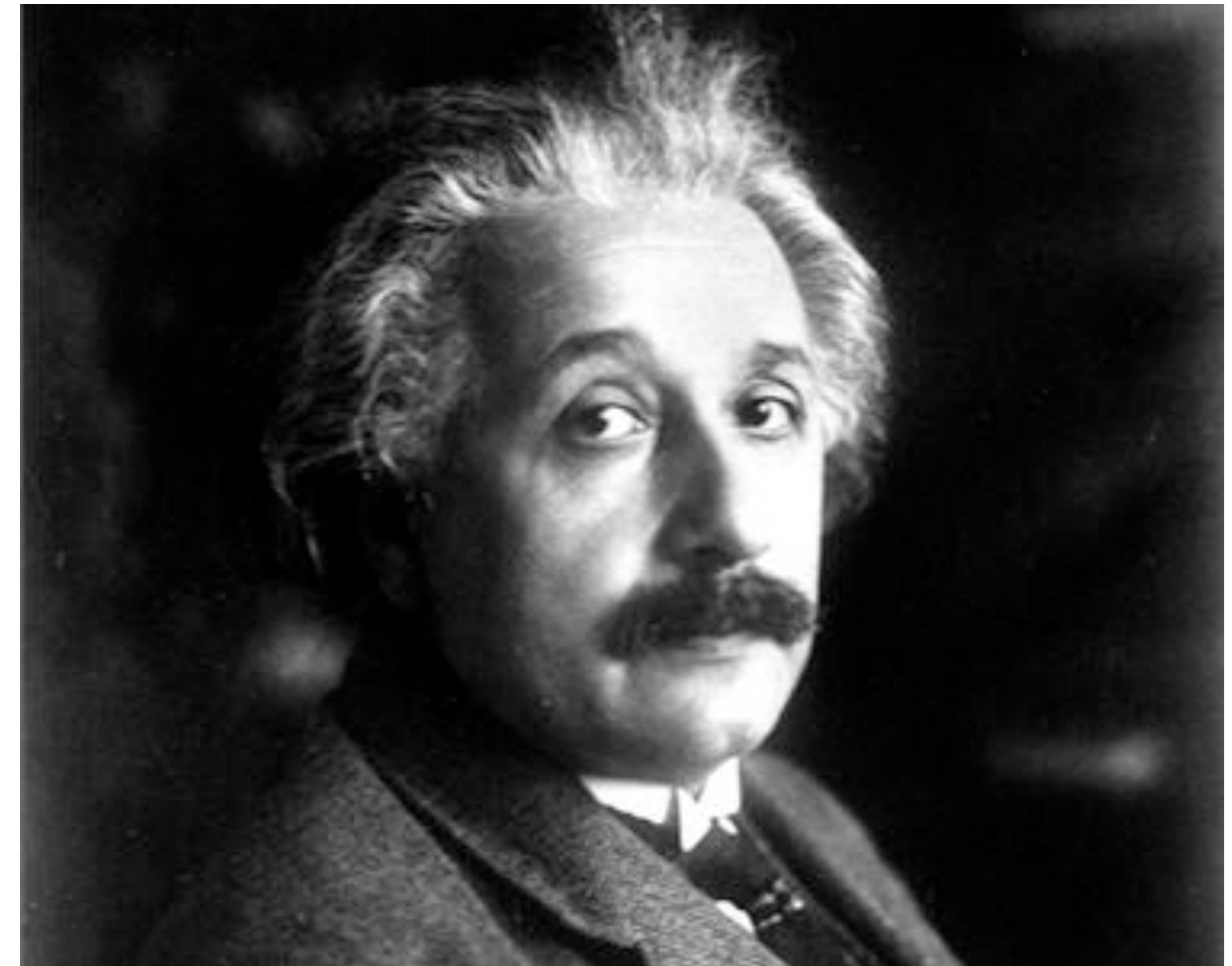
*[With first principles] you boil things down to the most fundamental truths ... and then reason up from there.”*

**Principles that we can  
challenge...**

# First Principles Thinking / Einstein

*Einstein questioned the assumption that time and space were absolute.*

1905



# Not continuous improvements

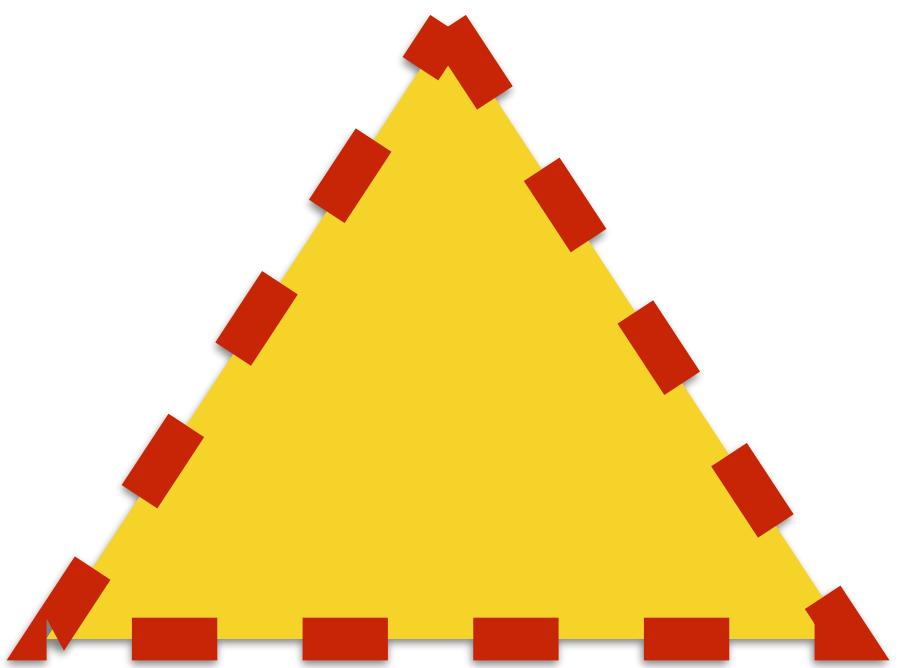
*Continuous improvement tends to occur **within the boundary set by the original vision.***



when designing software

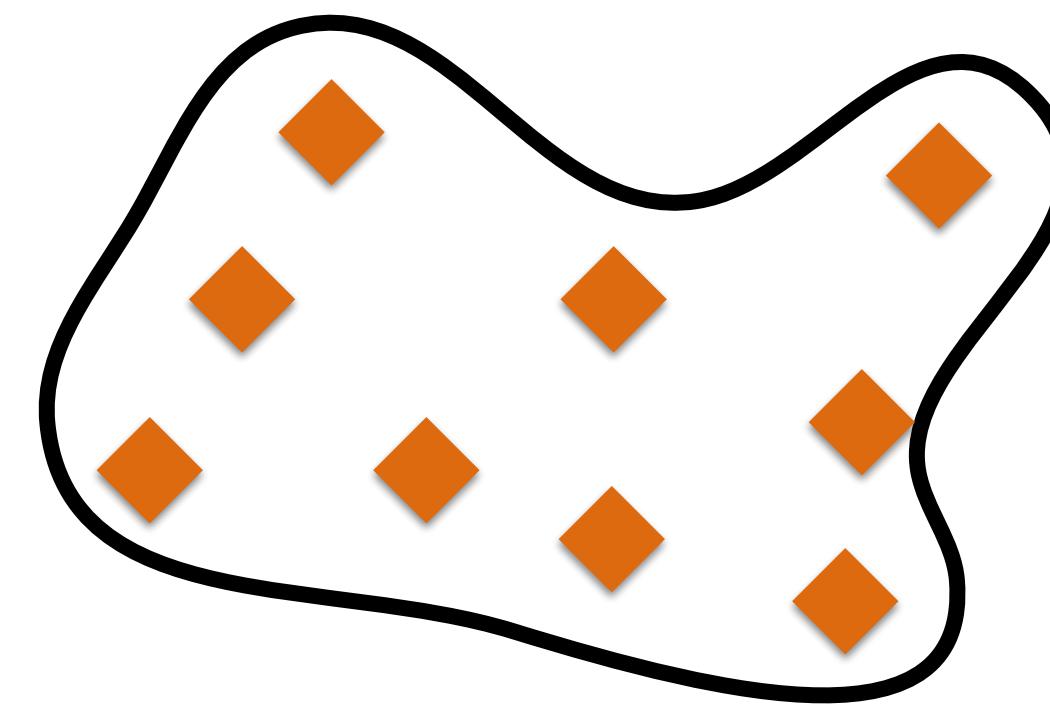
Deconstruct **past** models  
to reconstruct  
new **innovative** models

**SMALL**



+

**SMALL**

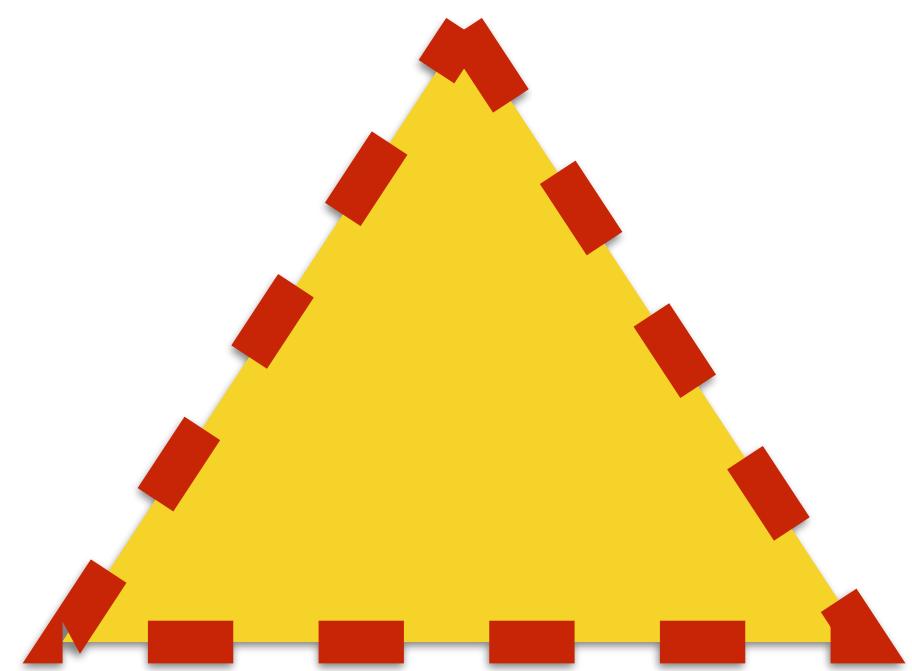


**Theory**

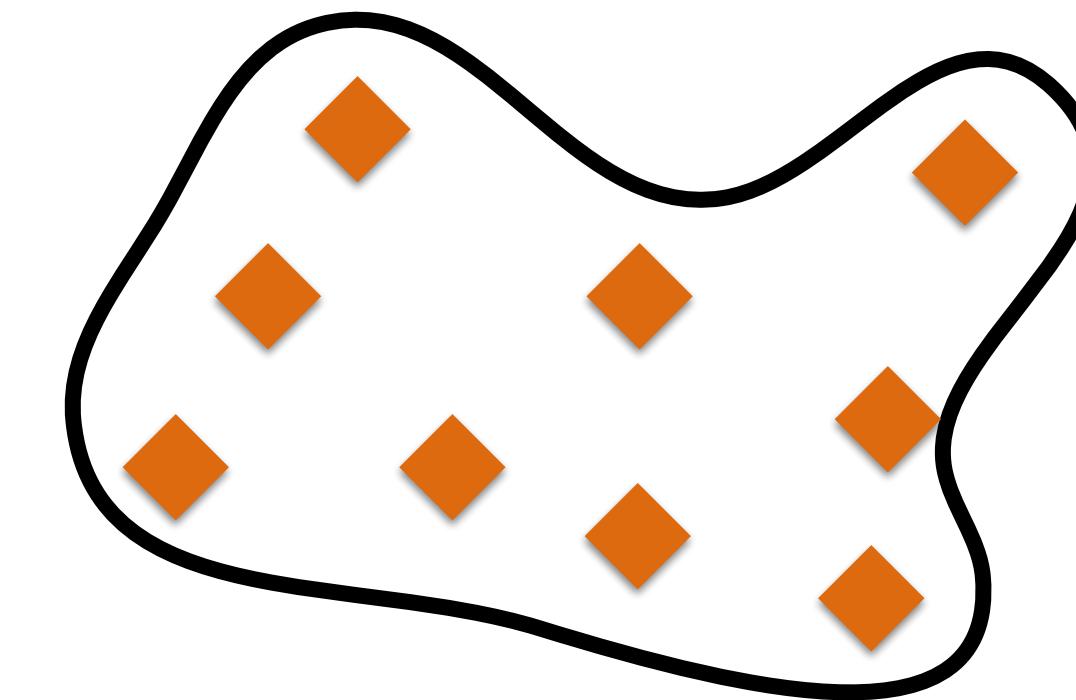
**Residual  
Mess**

**CHALLENGEABLE!**

**CHALLENGEABLE!**



+



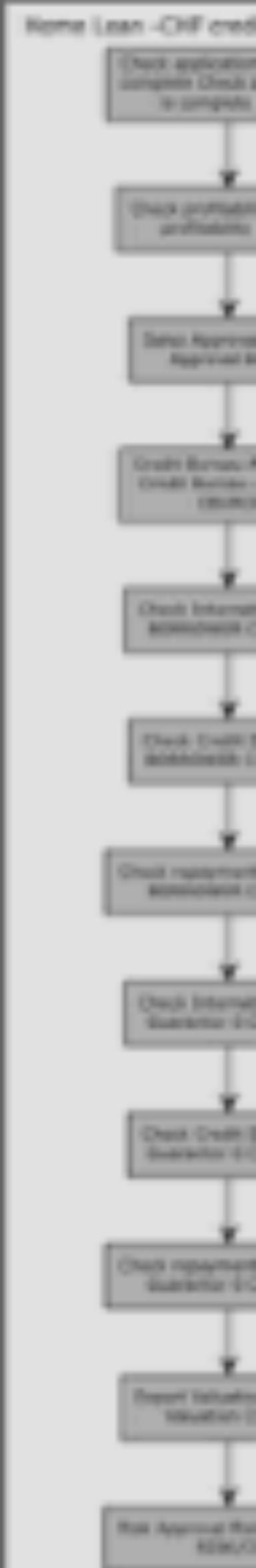
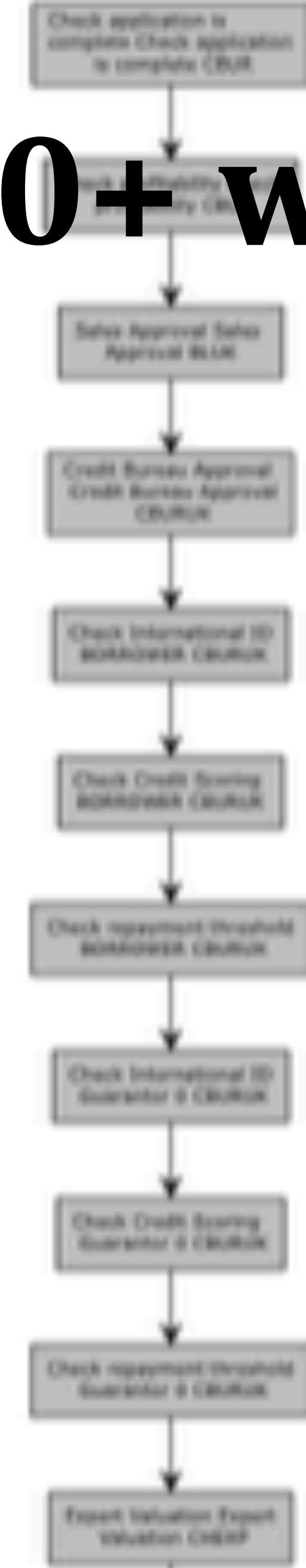
**Theory**

**Residual  
Mess**

# DIGITAL-NATIVE BUSINESSES

# **LOAN APPROVAL WORKFLOWS**

# We have 80+ workflows



Foreign\_House\_Loan\_-UK\_credit\_(Swiss\_House)\_with\_guarantor.png

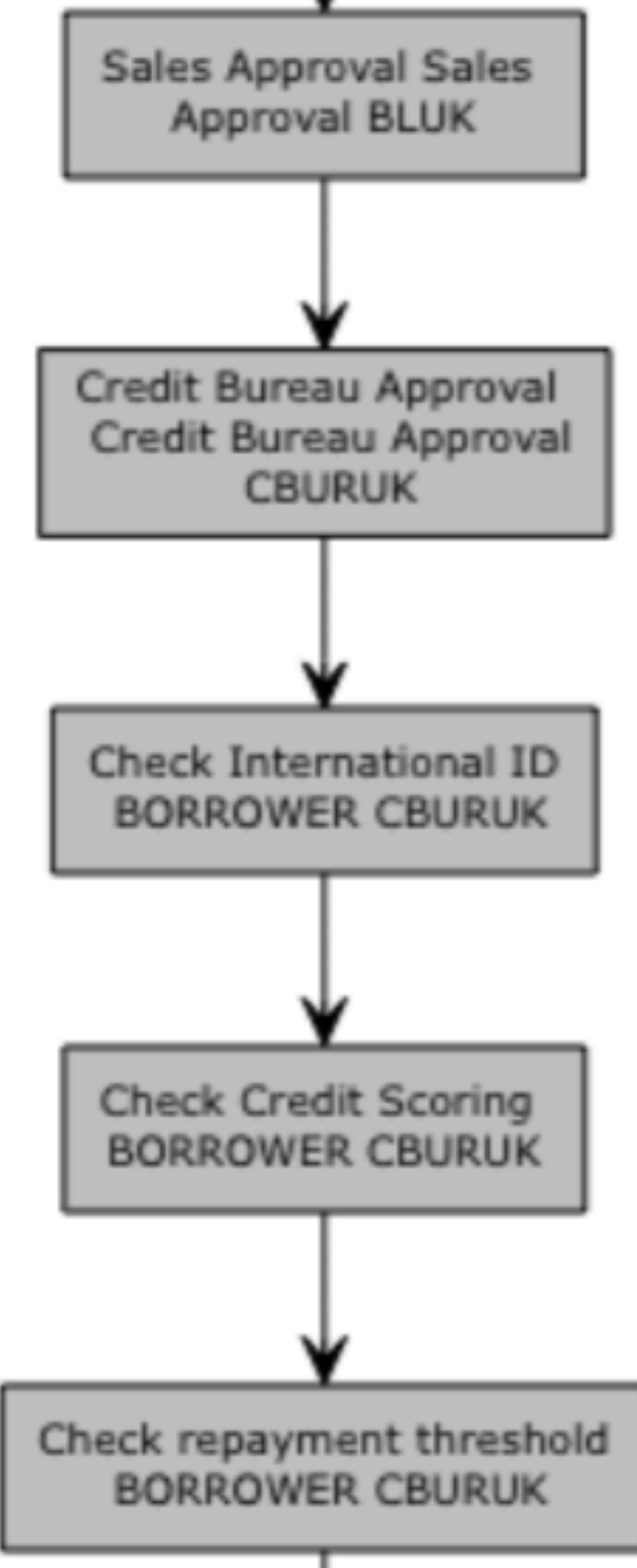
MANY WORKFLOWS  
UNCONSISTENCIES

MAINTENANCE

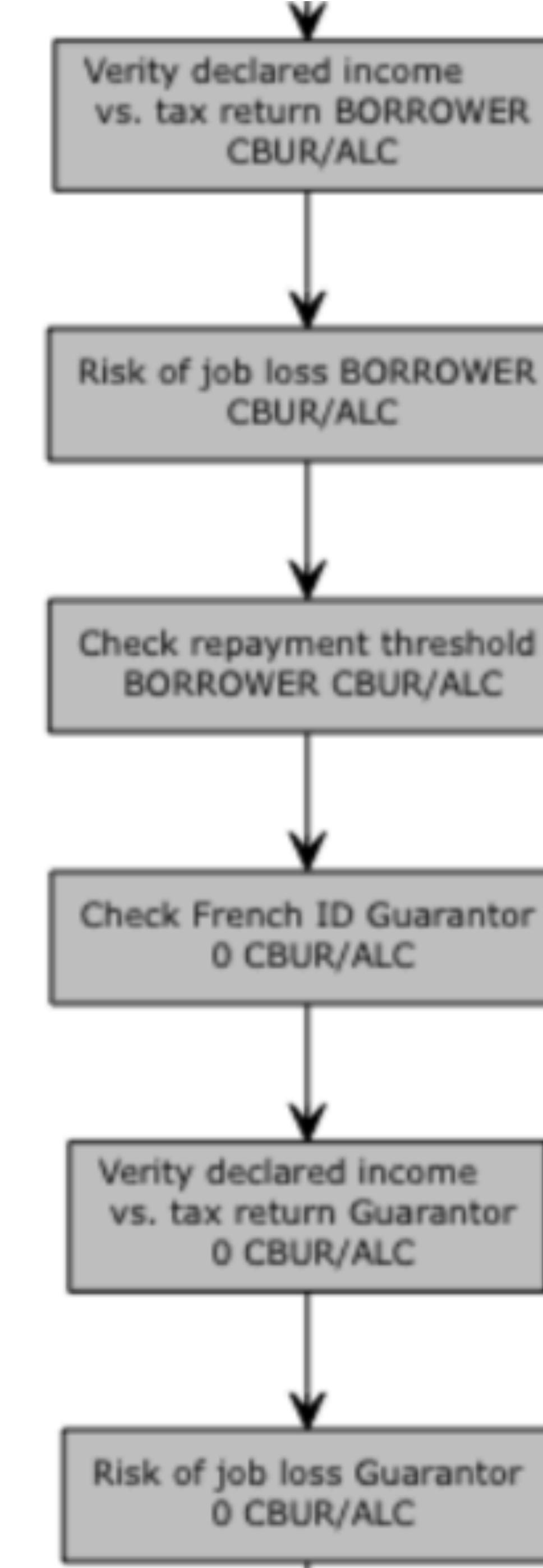
TOO SEQUENTIAL

ANYONE LATE DELAYS  
EVERYONE LATER

**RAISE THE  
WATERLINE**



**SIMILAR STEPS OVERALL  
BUT NOT ALWAYS THERE  
BY DIFFERENT TEAMS**



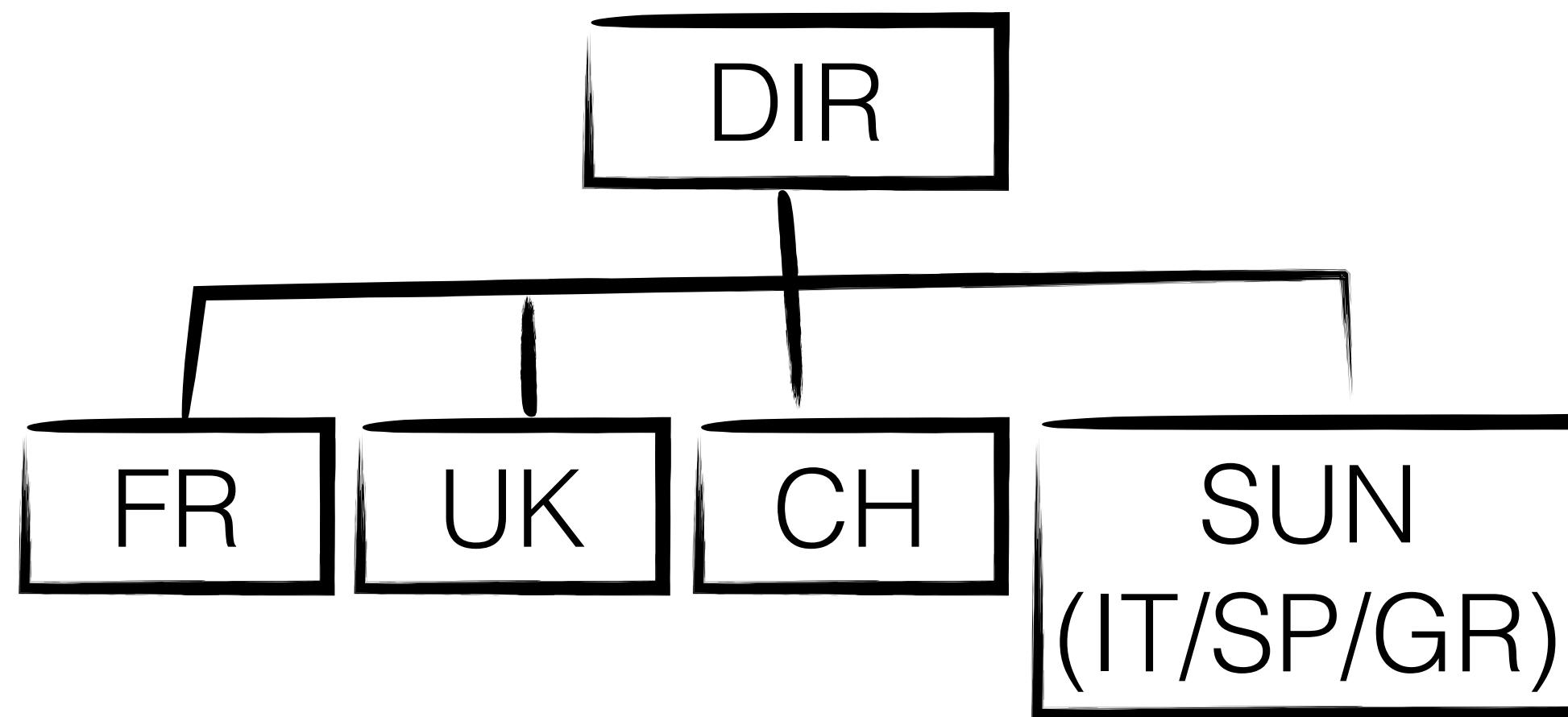
**MANY WORKFLOWS.  
FEW RATIONALES.**

**MANY WORKFLOWS.  
FEW RATIONALES.**

**TURNS OUT ALL  
WORKFLOW ARE PRETTY  
MUCH THE SAME**

# TEAM SEGMENTATION

## PARAMETERIZED WORKFLOW



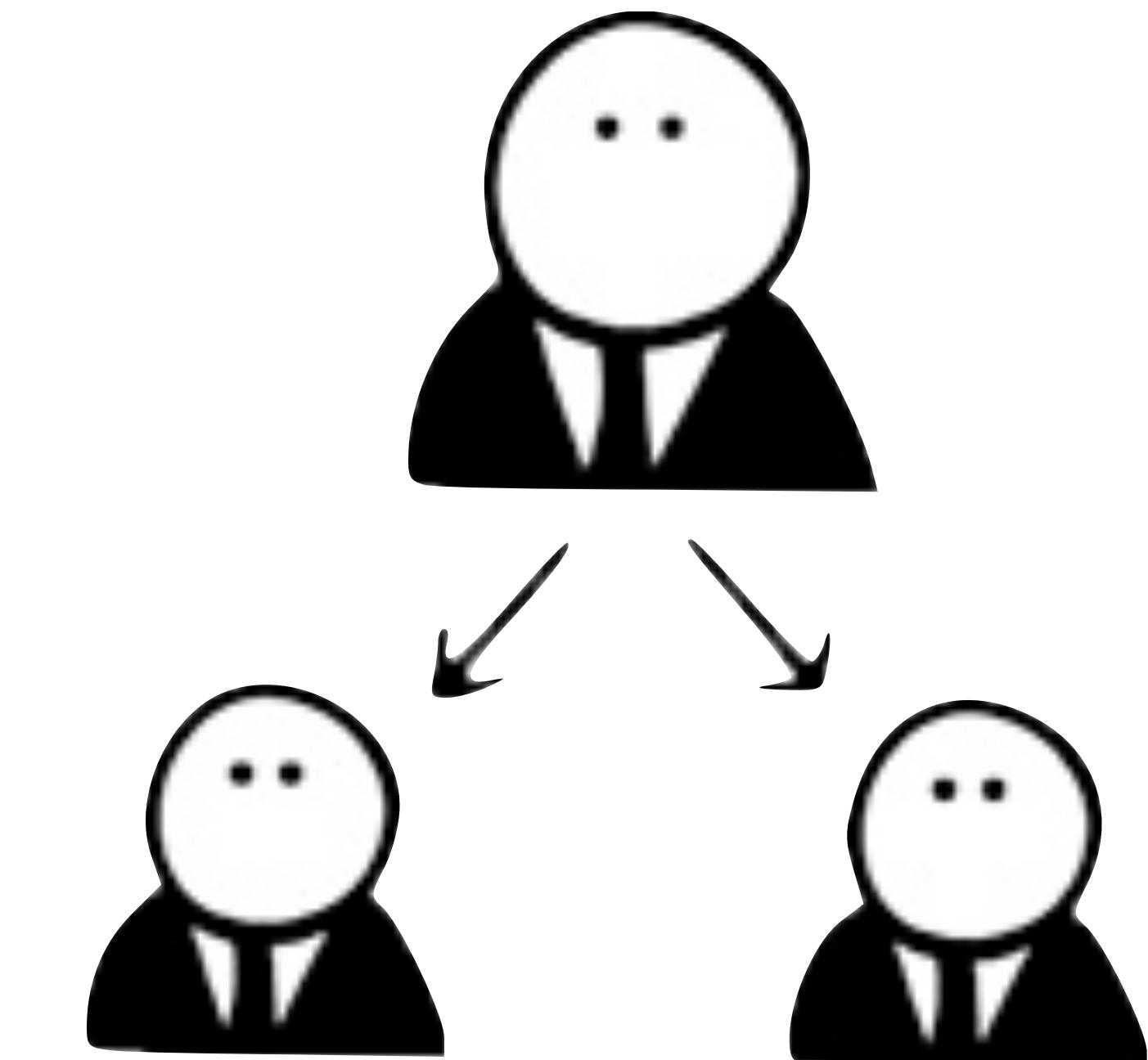
→ TEAM  
chosen  
by location

## CLIENT COVERAGE

# DELEGATION MECHANISM

→ TEAM  
replaced  
by delegation

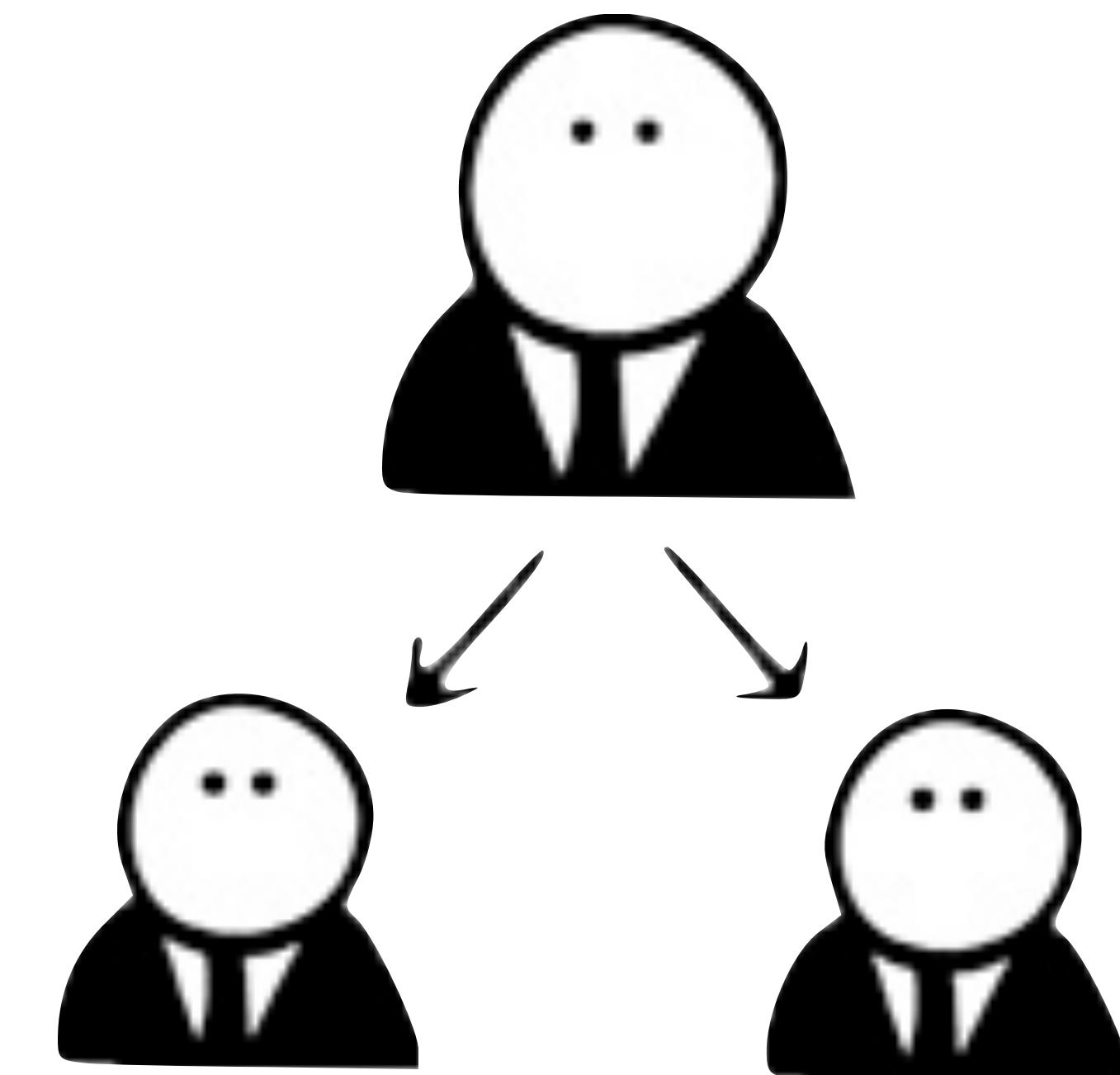
→ Steps by  
same team  
merged



DELEGATION

# DELEGATION MECHANISM

Delegation by amount  
→ optional extra steps for big amounts



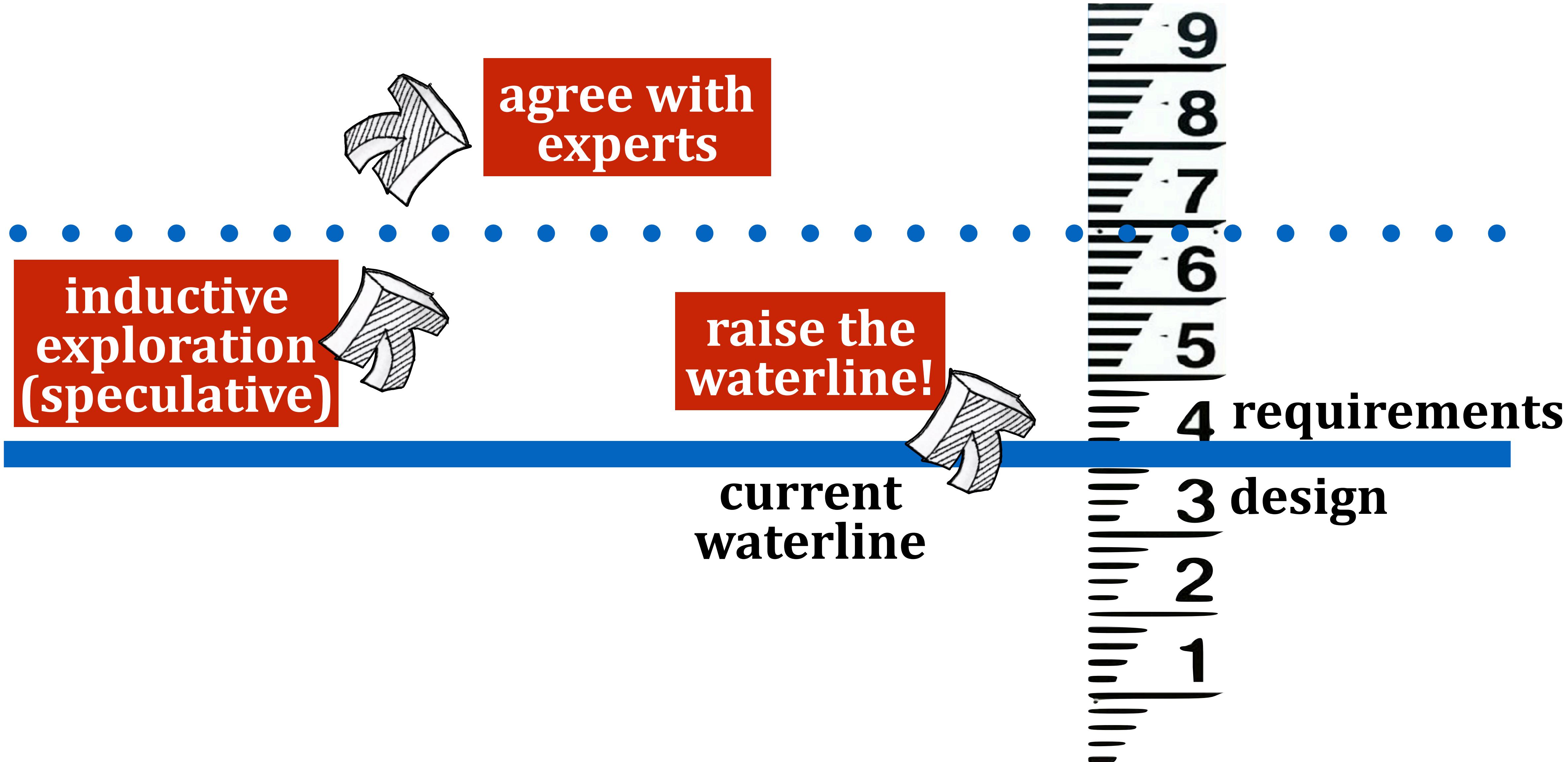
DELEGATION

# Builder Pattern as a Virtual B.A.

**Reasoning**  
To build  
on-the-fly  
**concrete workflow**

# Reverse-Engineering Investigation

(Progressing towards Higher Principles)



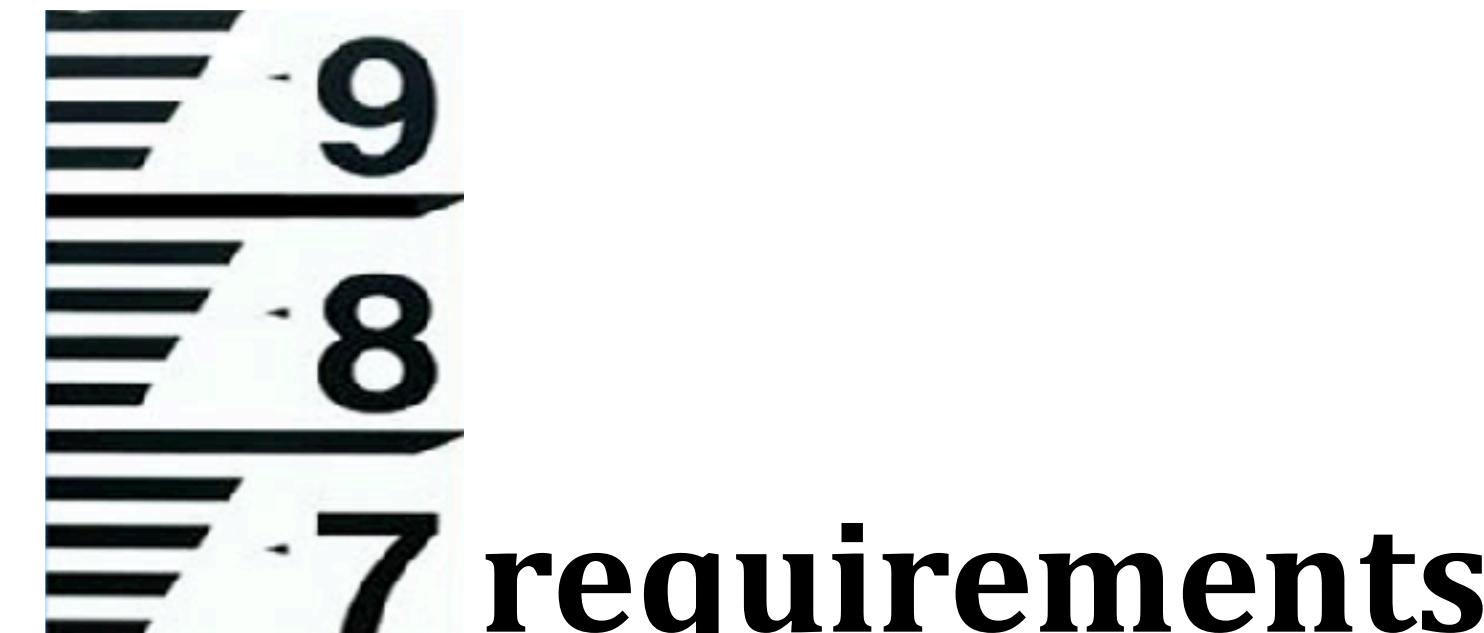
# Reverse-Engineering Investigation

(Progressing towards Higher Principles)



MORE POTENTIAL FOR  
SMARTER DESIGN

new  
waterline



**Going further...**

**What does the  
workflow tries to  
optimise for?**

**MOST  
LIKELY  
TO FAIL  
ASAP**

**DISTURB  
SENIOR  
ROLES  
LAST**

**Declare the preferences.**  
**Let the tool find out the**  
**workflow.**

**MOST  
LIKELY  
TO FAIL  
ASAP**

**DISTURB  
SENIOR  
PEOPLE  
LAST**

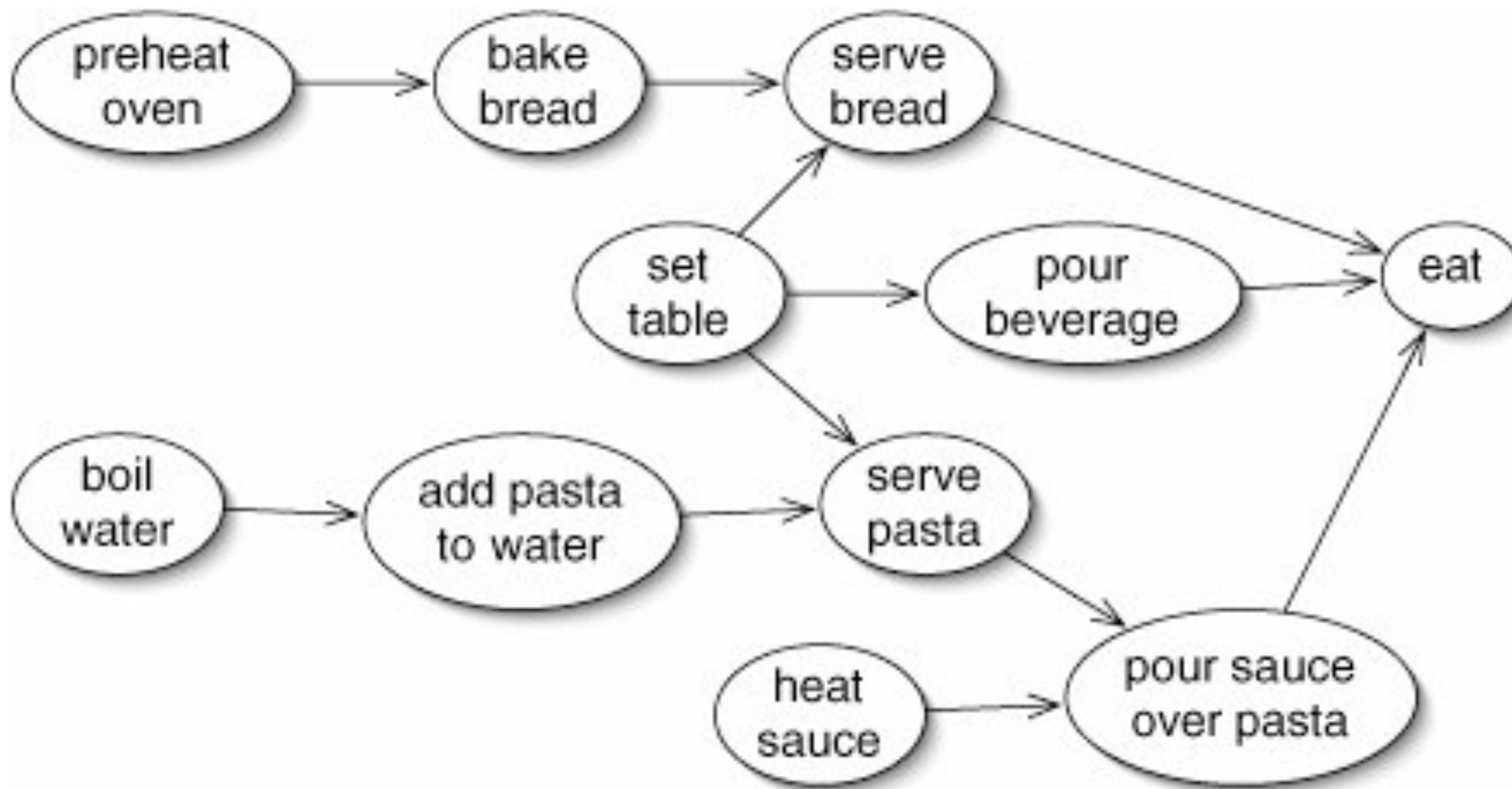
Declare the preferences.  
Let the tool find out the

**WE CALL THAT  
"SORTING"**

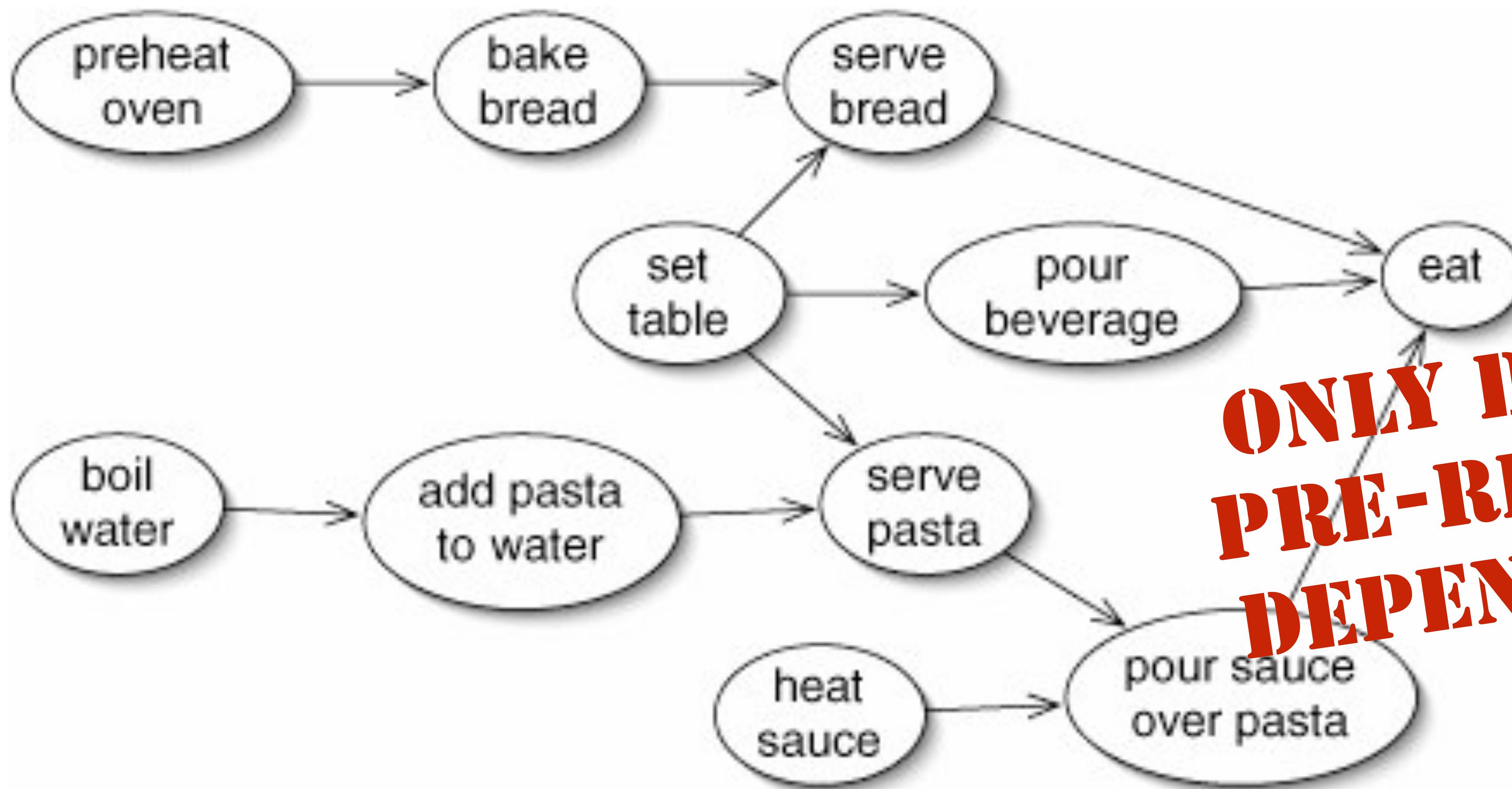
**SENIOR  
PEOPLE  
LAST**

**FAIL  
ASAP**

# Topological Sorting

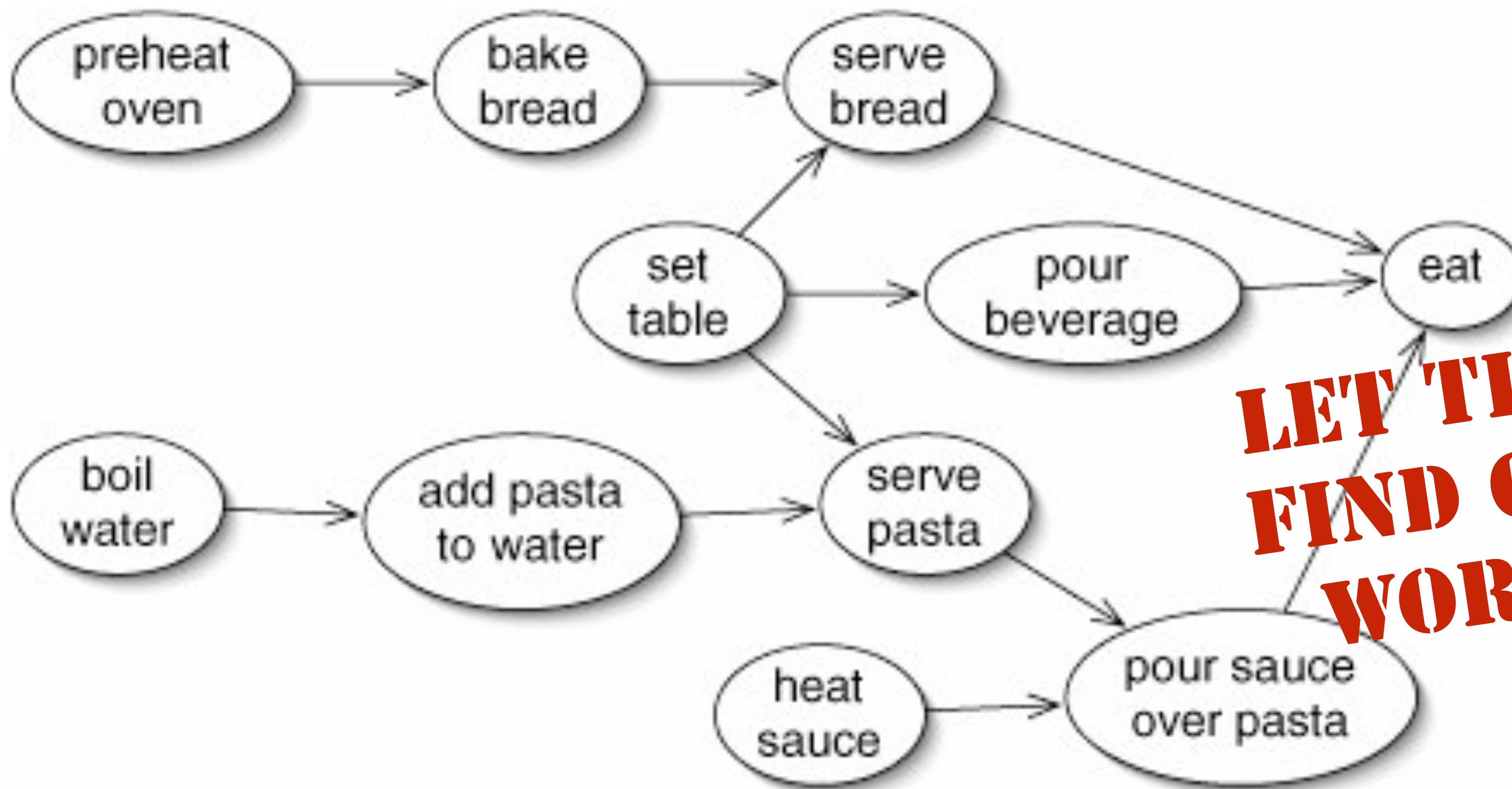


# Topological Sorting

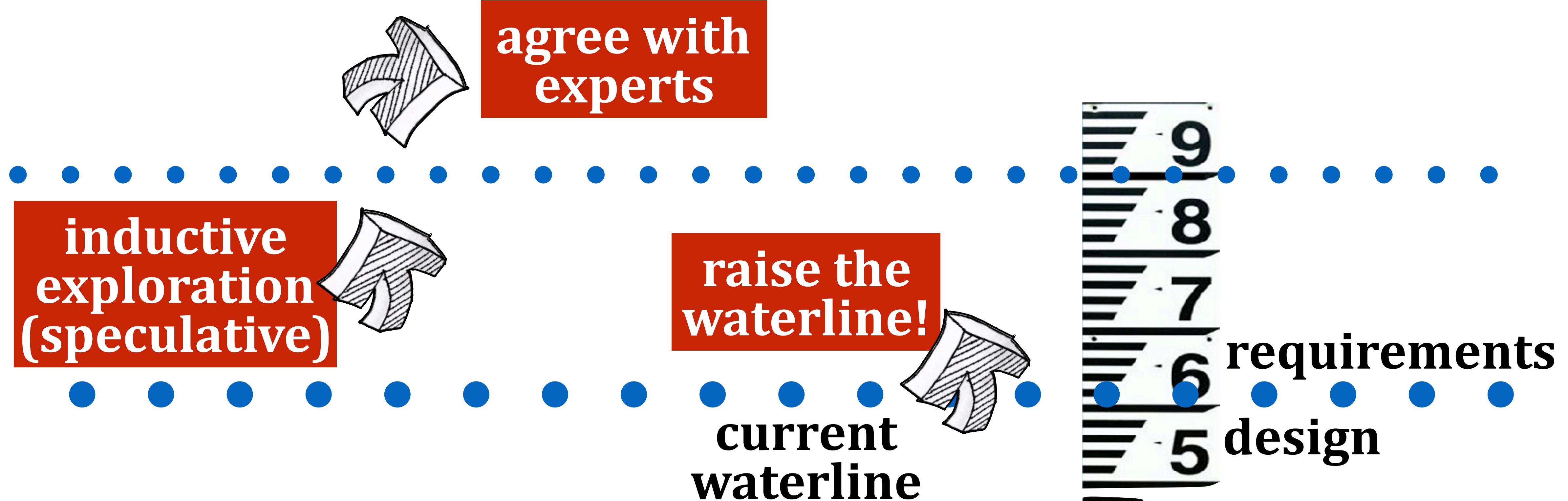


**ONLY DECLARE  
PRE-REQUISITE  
DEPENDENCIES**

# Topological Sorting



**Domain modelling  
can literally express  
human psychology.**



# Reverse-Engineering Again

(Progressing towards Higher Principles)

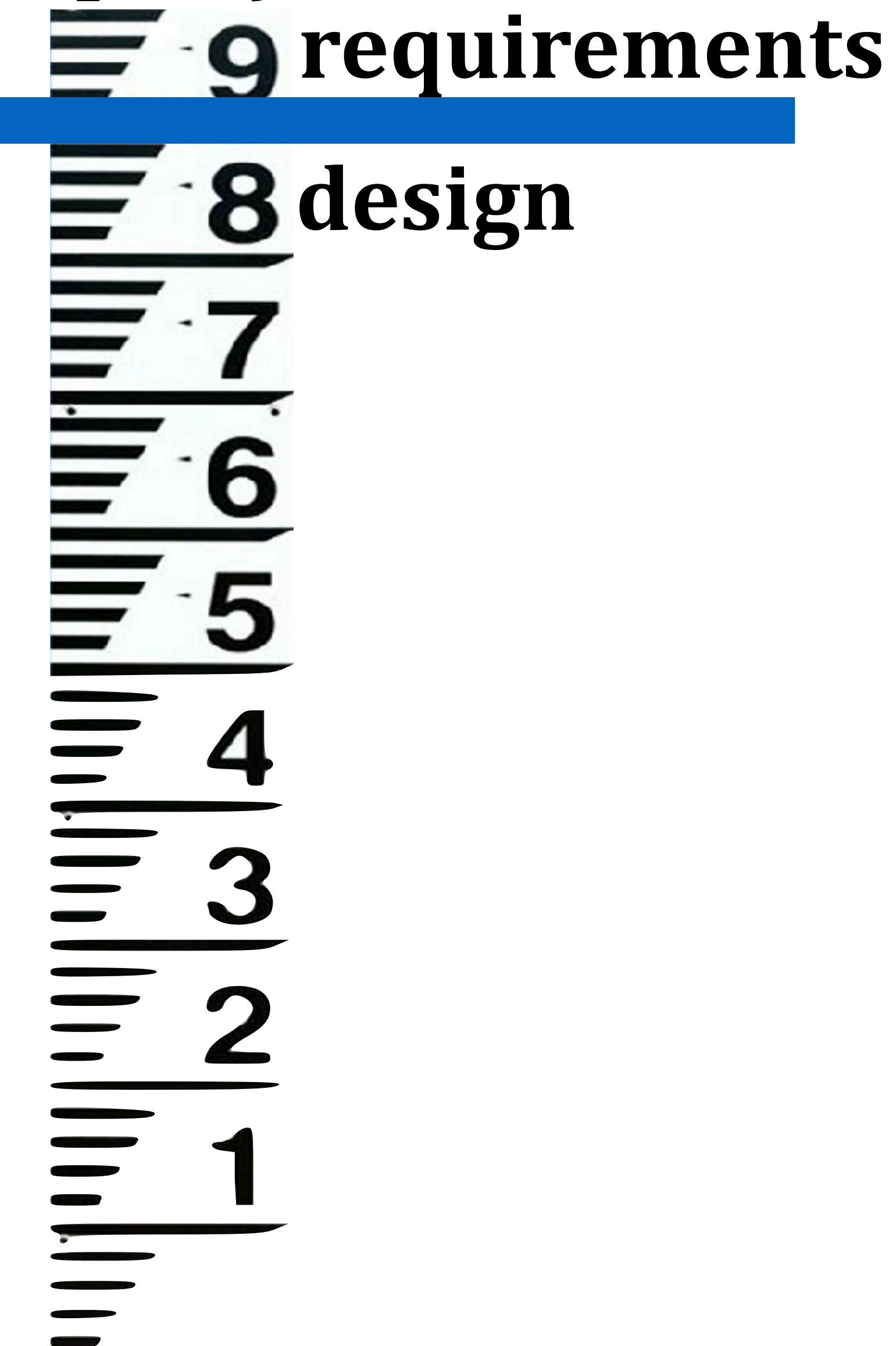
# Reverse-Engineering Investigation

(Progressing towards Higher Principles)



MORE POTENTIAL FOR  
SMARTER DESIGN

new  
waterline



**Going further...**

**When 80% of requests are  
fully automated...**

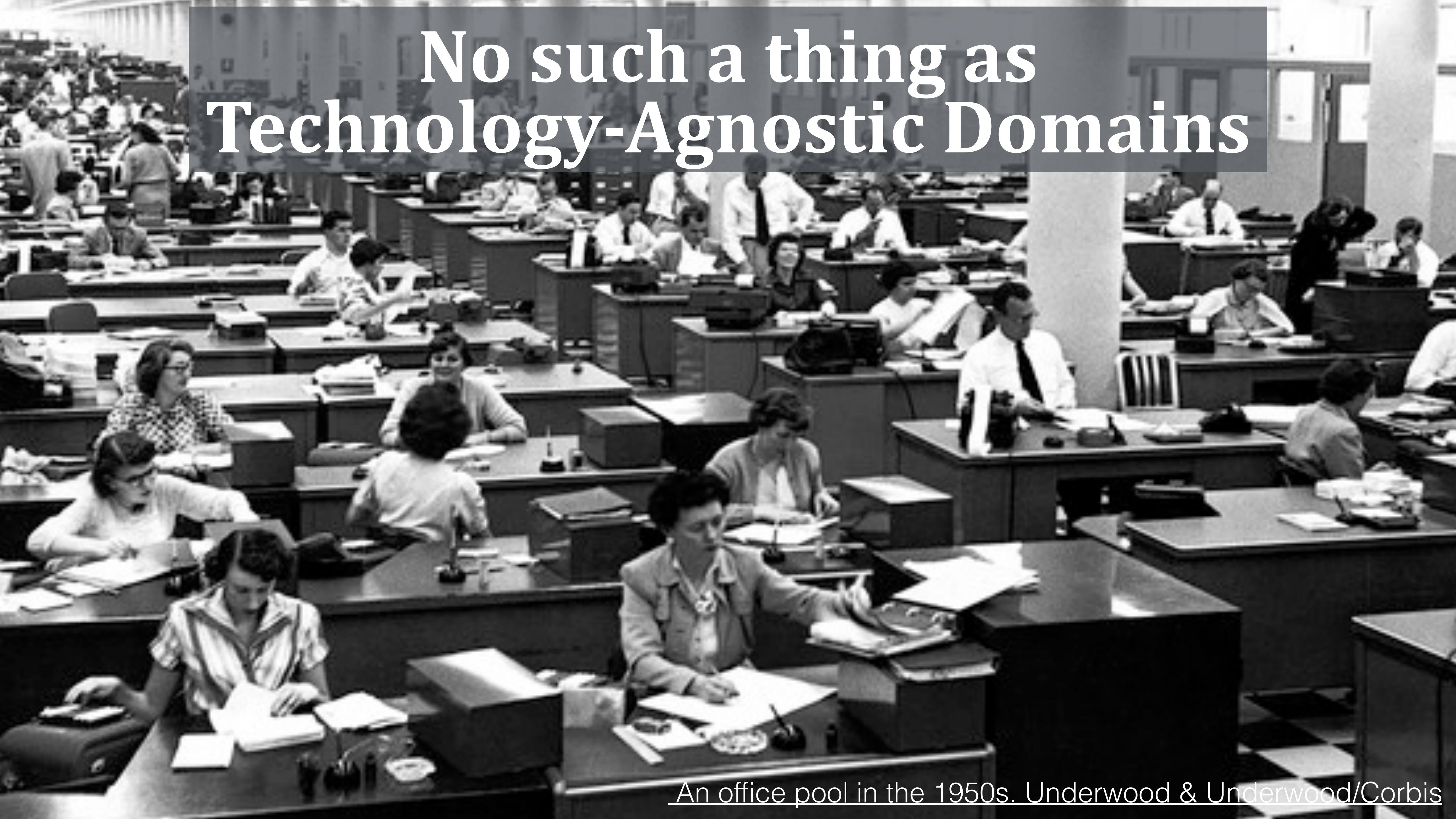
**...does it still makes sense  
to have a workflow  
between (mostly) function  
calls?**

**#NoWorkflow**

# UNDERLYING TECHNOLOGY

as-an-influence

# No such a thing as Technology-Agnostic Domains



An office pool in the 1950s. Underwood & Underwood/Corbis



Domains are still shaped by  
the previous technology:  
bureaucracy

An office pool in the 1950s. Underwood & Underwood/Corbis

# RADICAL RE-INVENTION

from first principles

**What does the  
workflow tries to  
optimise for?**

# CREDIT RISK FROM FIRST PRINCIPLES

## IS IT BALANCED?



MONEY LENT  
OVER TIME

RISK BUREAU

GUARANTEES  
VALUE OVER TIME

# PAYMENT FROM FIRST PRINCIPLES

**"NON REPUDIATION"**



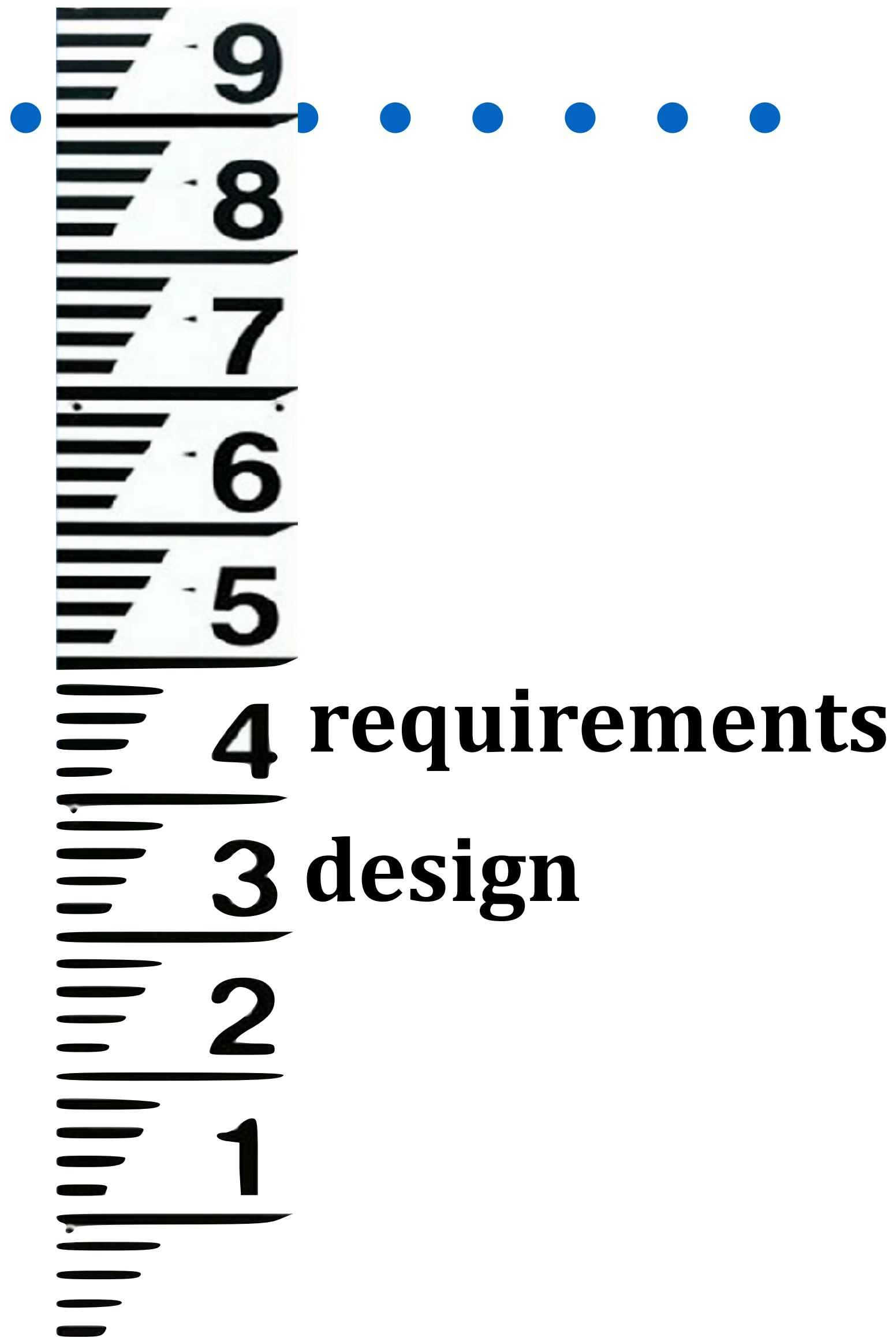
# Radically From First Principles

(Ignoring the traditions)

FIRST  
PRINCIPLES OF  
THE BUSINESS



CO-DESIGNED  
SOFTWARE



# The Case for Learned Index Structures

Tim Kraska\*

MIT

Cambridge, MA

[kraska@mit.edu](mailto:kraska@mit.edu)

Alex Beutel

Google, Inc.

Mountain View, CA

[alexbeutel@google.com](mailto:alexbeutel@google.com)

Ed H. Chi

Google, Inc.

Mountain View, CA

[edchi@google.com](mailto:edchi@google.com)

Jeffrey Dean

Google, Inc.

Mountain View, CA

[jeff@google.com](mailto:jeff@google.com)

Neoklis Polyzotis

Google, Inc.

Mountain View, CA

[npolyzotis@google.com](mailto:npolyzotis@google.com)

## Abstract

Indexes are models: a B-Tree-Index can be seen as a model to map a key to the position of a record within a sorted array, a Hash-Index as a model to map a key to a position of a record within an unsorted array, and a BitMap-Index as a model to indicate if a data record exists or not. In this exploratory research paper, we start from this premise and posit that all existing index structures can be replaced with other types of models, including deep-learning models, which we term *learned indexes*. The key idea is that a model can learn the sort order or structure of lookup keys and use this signal to effectively predict the position or existence of records. We theoretically analyze under which conditions learned indexes

# The Case for Learned Index Structures

Tim Kraska\*

MIT

Cambridge, MA

kraska@mit.edu

Alex Beutel

Google, Inc.

Ed H. Chi

CA.com

#BEFORE:  
ENGINEERING TIME IS PRECIOUS,  
THEREFORE GENERIC INDEXES  
#AFTER:  
DEEP-LEARNING CAN REPLACE  
HUMAN ENGINEERING TIME

... can be seen as a model to map a key to the position of a record ... may, a Hash-Index as a model to map a key to a position of a record within an unsorted array, and a BitMap-Index as a model to indicate if a data record exists or not. In this exploratory research paper, we start from this premise and posit that all existing index structures can be replaced with other types of models, including deep-learning models, which we term *learned indexes*. The key idea is that a model can learn the sort order or structure of lookup keys and use this signal to effectively predict the position or existence of records. We theoretically analyze under which conditions learned indexes

# **!Political Risk!**

**Who's gonna sign off the decision to make radical changes to business processes?**

*Discovering a modelling breakthrough is only half of the total effort; the other half is communicating and convincing other people (business sponsors, managers...) to accept the associated risk of novelty.*

in closing

**RAISE THE  
WATERLINE**

**EXPECT  
UNTOLD  
REGULARITIES**

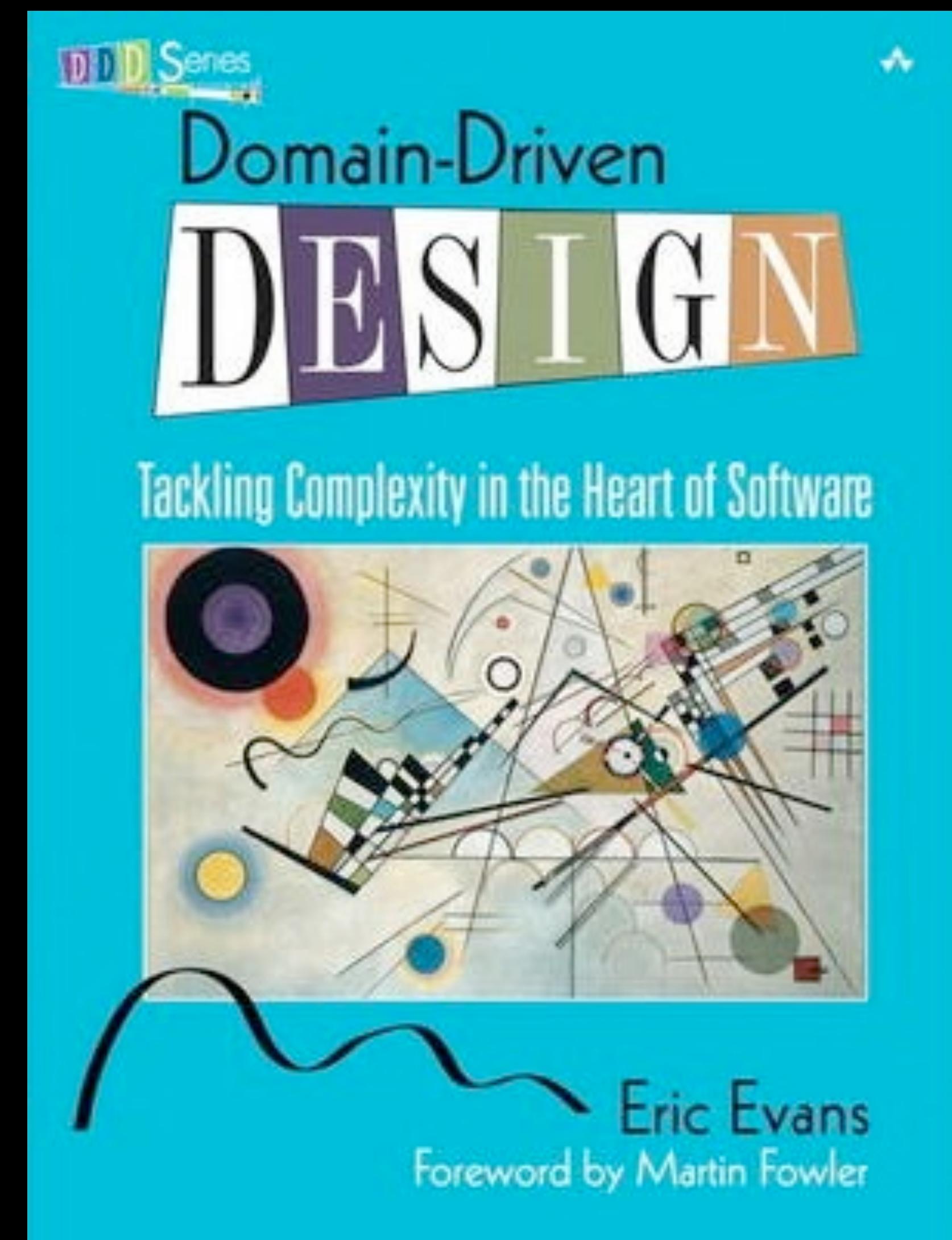
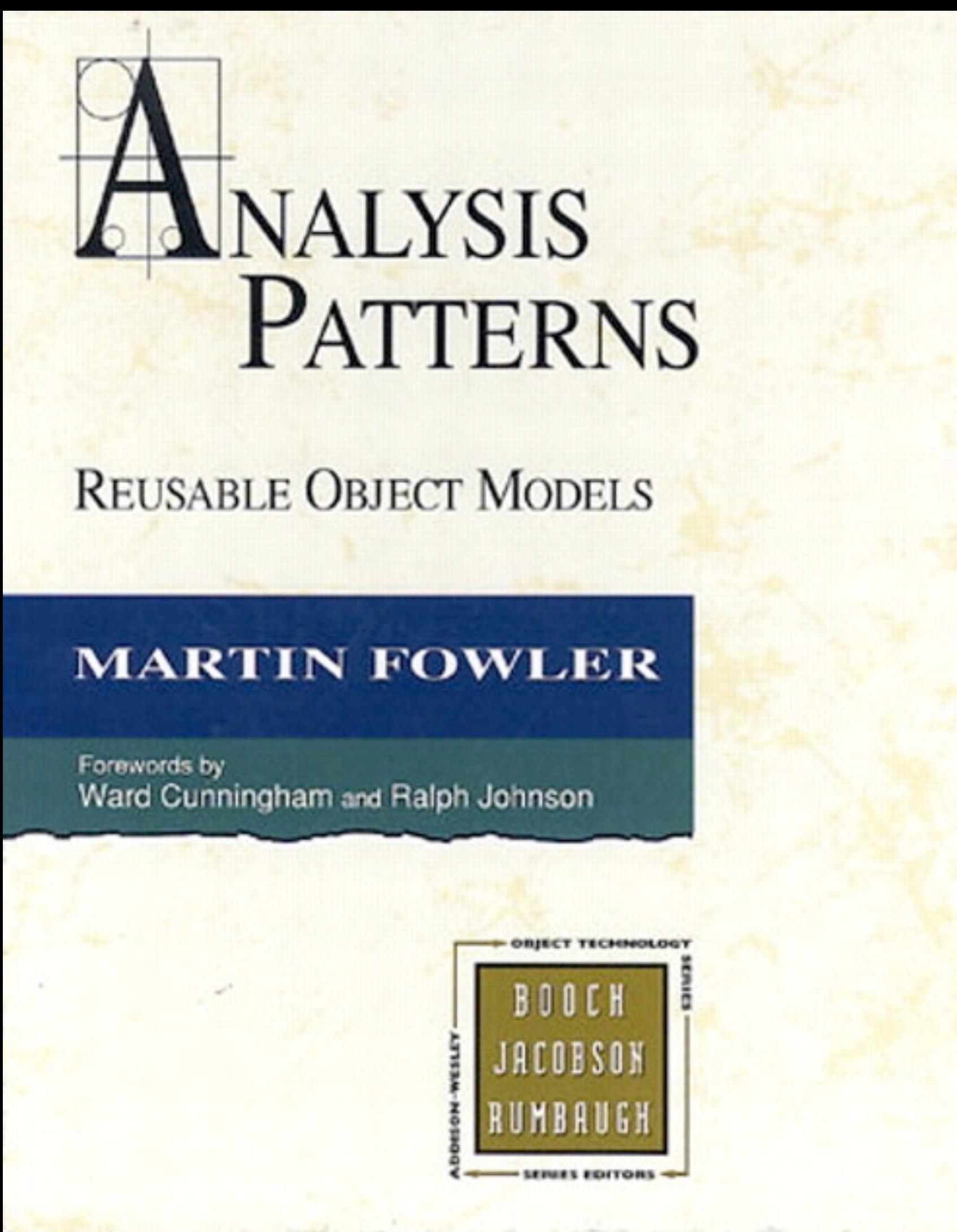
# **PRACTICE TDD**

# **PRACTICE**

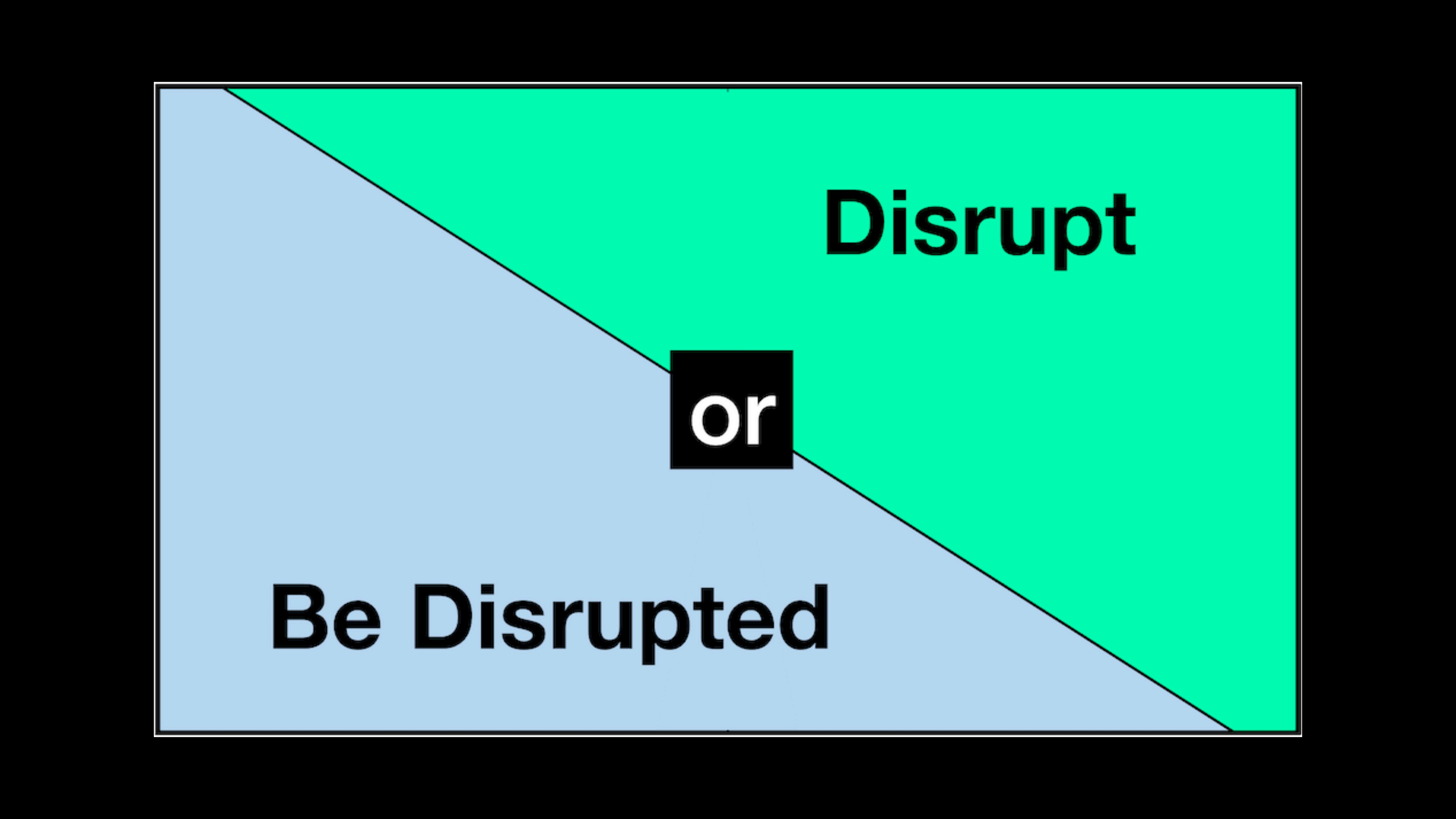
# **Domain-Driven**

# **Design**

**Build theories,  
not just lists of  
business rules**



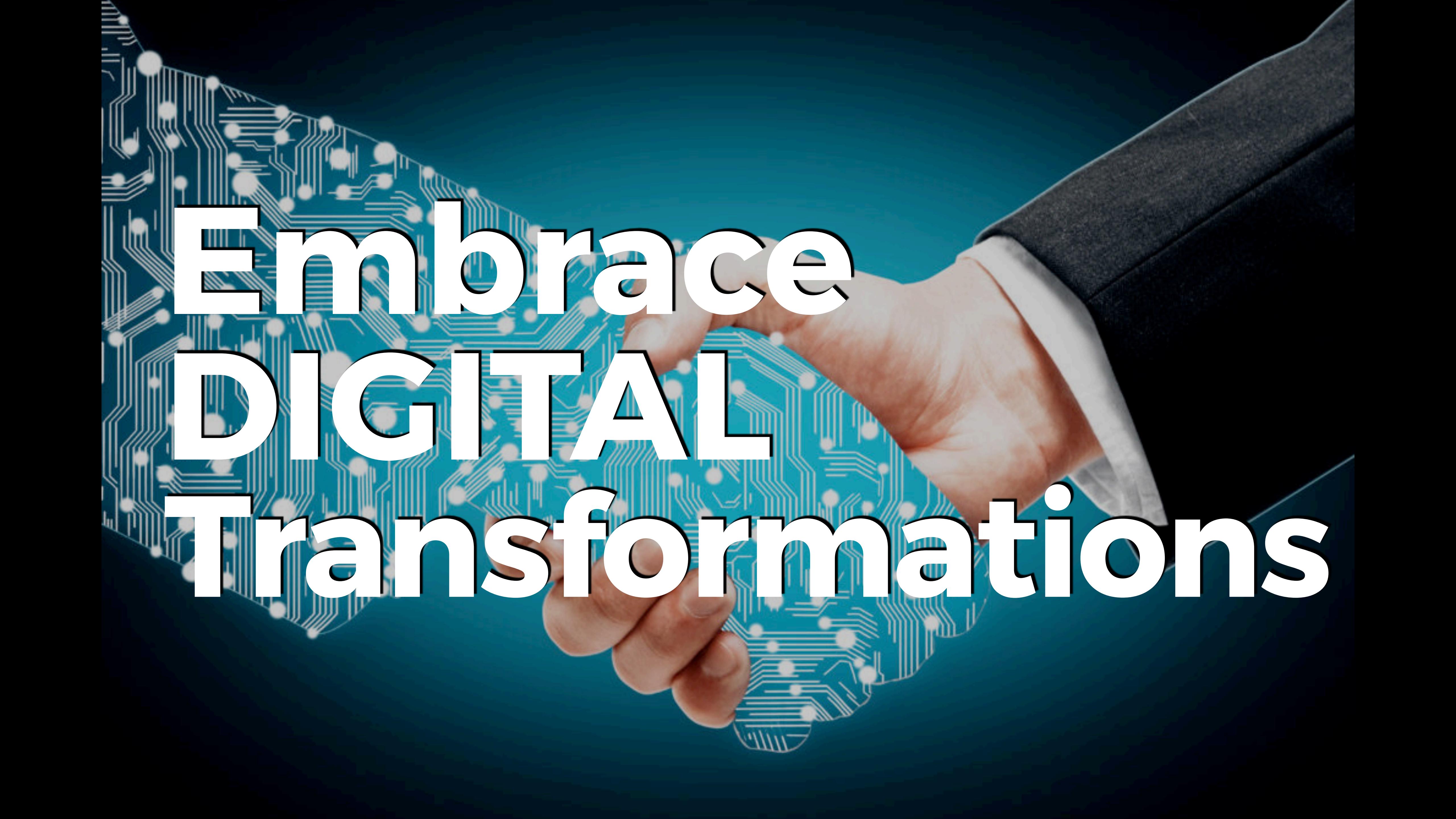
**LEARN TO  
THINK FROM  
FIRST  
PRINCIPLES**



**Disrupt**

**or**

**Be Disrupted**

A close-up photograph of a person's hand holding a dark-colored smartphone. The background features a vibrant blue and green abstract pattern that resembles a circuit board or a digital signal. Overlaid on this image is the text "Embrace DIGITAL Transformations" in a large, bold, white sans-serif font.

**Embrace  
DIGITAL  
Transformations**

# **to SOFTWARE- AUGMENTED BUSINESSES**





DOMAIN  
DRIVEN  
DESIGN  
EUROPE

JAN 31<sup>ST</sup> - FEB 3<sup>RD</sup> 2017

AMSTERDAM



Thanks!

@cyriux

Cyrille Martraire



arolla

**Follow me @cyriux**

**Slides: [slideshare.net/cyriux](https://www.slideshare.net/cyriux)**

**Blog: [cyrille.martraire.com](http://cyrille.martraire.com)**

**We offer training, coaching & consulting on DDD, BDD, TDD**

