AllanKelly
www.allankelly.net

Software Development is Upside Down

Allan Kelly
@allankellynet
allan@allankelly.net
http://www.allankelly.net

Craft Conf, Budapest, May 2018

Allan Kelly

Bringing technology & business together

Inspiring Agile Teams

- Writing
- Training
- Advising
- Troubleshooting

Business Patterns for
Allan Kelly
Clearly defines the route to business strategy and operations
Includes over 38 strategy patterns
Explains how to implement
Changing Software Develo
LEARNING TO
Allan
Allan Kelly
A Little Book about Requirements and User Stories
Continuous Digital
An agile alternative to projects for digital business
Software development for the digital business
Team based development without projects
Guided by value
Allan Kelly
Allan Kelly
Xanpan
Team centric Agile software development
Combining Kanban and XP - inspiration for creating your own hybrid

Mental models

Maybe…

... we need to...

... rethink

Organization &
Management models?

Mental models

5

Diseconomies of Scale

Software development…

- Does NOT have economies of Scale
- Development has DISECONOMIES of scale

Milk is cheapest in BIG cartons

Software is cheapest in lots of small cartons

And small cartons of software reduce risk

Project A: Risk = 30% Value at risk = £1m
Therefore risk weighted value = £300,000

Consi r a large project

Prj B: Risk = 15%
Value @ risk = £½m
Therefore … = £75,000

Prj C: Risk = 15%
Value @risk = £½m
Therefore … = £75,000

E: Risk = 6%
@risk = £20
Therefore = £

F: Risk = 6%
@risk = £200
Therefore = £1

G: Risk = 6
@risk = £2
Therefore =

H: Risk = 6%
@risk = £20
Therefore = £

I: Risk = 6%
@risk = £200k
Therefore = £12k

Software development…

- Does NOT have economies of Scale
- Development has DISECONOMIES of scale

Therefore

- Stop thinking BIG
- Start thinking SMALL

Optimize for lots of Small

- Small batch size (limited amount of work)
- Small code bases
- Small releases
- Small tests
- Small teams
- Small funding
  - Allocate £$€ in small batches

Higher quality is faster

Quality… makes all things possible

"**Quality has much in common with sex.**

- Everyone is for it. (Under certain conditions of, course)
- Everyone feels they understand it. (Even though they wouldn't want to explain it)
- Everyone thinks execution is only a matter of following natural inclinations. (After all, we do get along somehow)

And, of course, **most people feel that all problems in these areas are caused by other people.**"

Philip Crosby

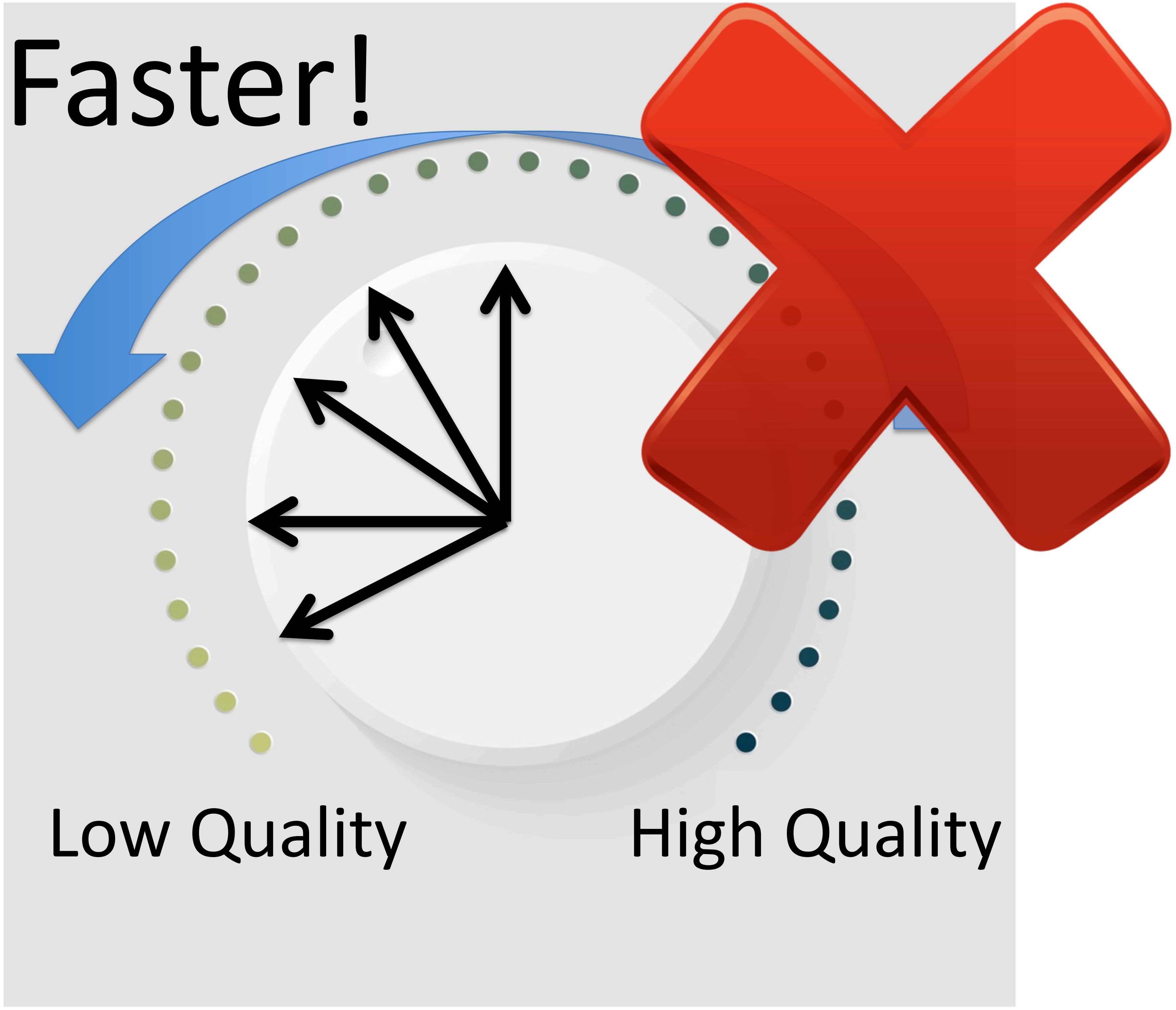
```
public class RecentlyUsedList {
  private List<string> list;
  public RecentlyUsedList() {
    list = new List<string>();
  }
  public string this[int index] {
    get {
    int position = 0;
    foreach (string value in list) {
      if (position == index)
        return value;
      ++position; }
    throw new ArgOutOfRngExcpt();
  }
```

```
public int Count {
      get {
        int size = list.Count;
        return size; } }

public void Add(string newItem) {
      if (list.Contains(newItem)) {
      int position =
              list.IndexOf(newItem);
      string existingItem =
list[position];
      list.RemoveAt(position);
      list.Insert(0, existingItem);
      } else {
        list.Insert(0, newItem); }
    }
} }
```

```
public class RecentlyUsedList {
  private List<string> list = new List<string>();
  public void Add(string newItem) {
    list.Remove(newItem);
    list.Add(newItem); }
  public int Count {
    get {
      return list.Count; }
  }
  public string this[int index] {
    get {
      return list[Count - index - 1]; }
} }
```

Code & refactoring from Kevlin Henney – www.curbralan.com


Faster!
Low Quality
High Quality

Faster!
Low Quality
High Quality

Defects are not free. Somebody makes them, and gets paid for making them

John Cage

How do you improve quality?

T D D

A T D D

B D D

Quickest way to **learn** is to **do**

Planning is learning

Planning is valuable

But…

IBM 360

luseum
Media

46 years …

1970 OS/360 model 195

- 10,000 KIPS (10 MIPS)
- 4096kb (4Mb)
- COBOL on OS/360
- IMS database
- Monthly rental $250,000

  (Approx. $1.25m in 2016 prices)

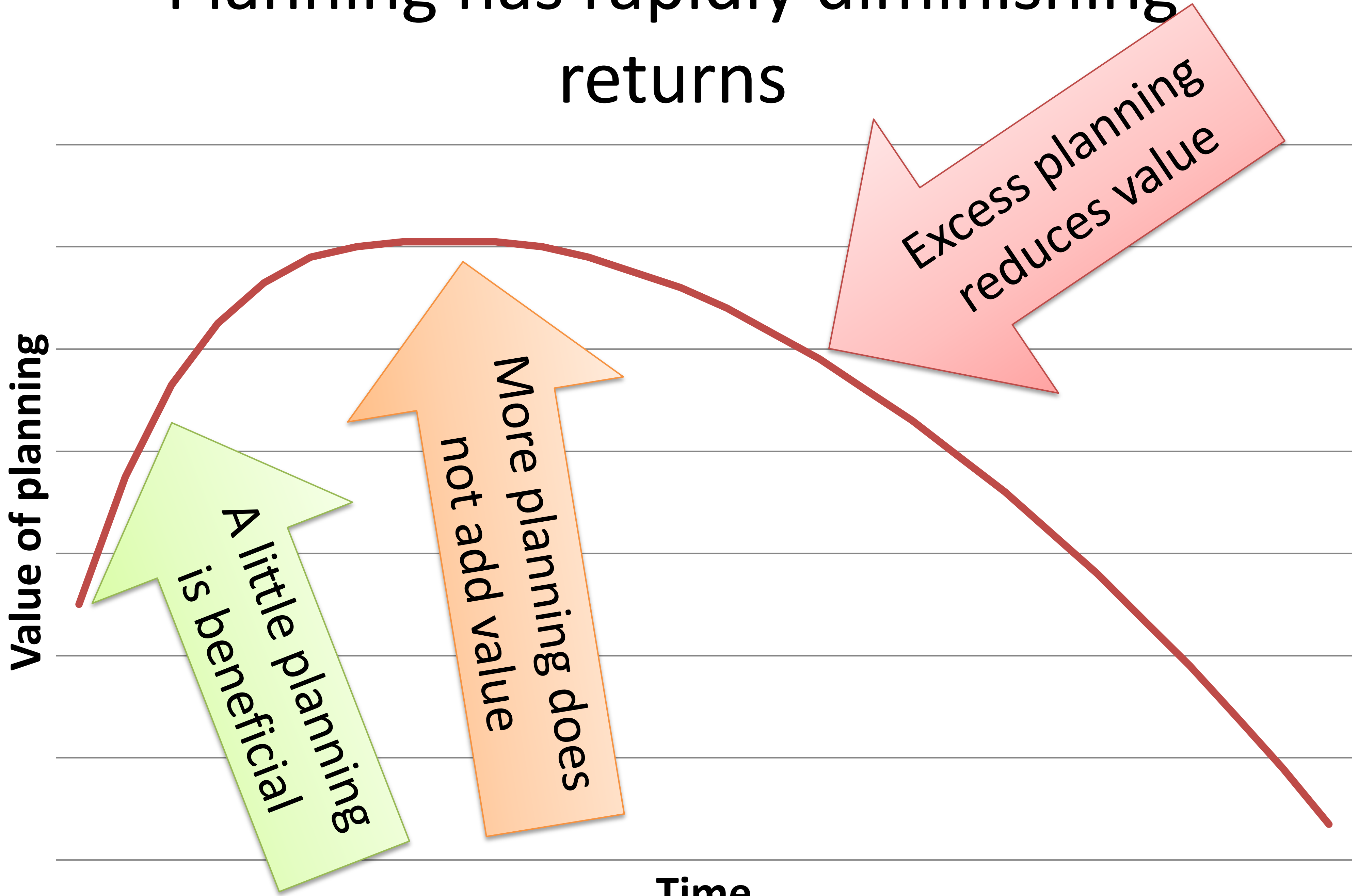
2016 Raspberry Pi 2

- 4,744 MIPS
- 1 Gb
- Linux
- Python, Scala, Ruby, …
- SQL, NoSQL
- Buy $35

CPU cycles €€€ ->
Planning is cheap

CPU cycles €€€ ->
Planning is expensive

Planning has rapidly diminishing returns



Planning is learning

Planning is valuable

But…

Planning is expensive

Planning has rapidly diminishing returns

If you want to finish sooner

Then

Start building sooner

Do it right,

<u>**then**</u>

Do the right thing

Yesterday

1) Do the right thing
2) Do it right

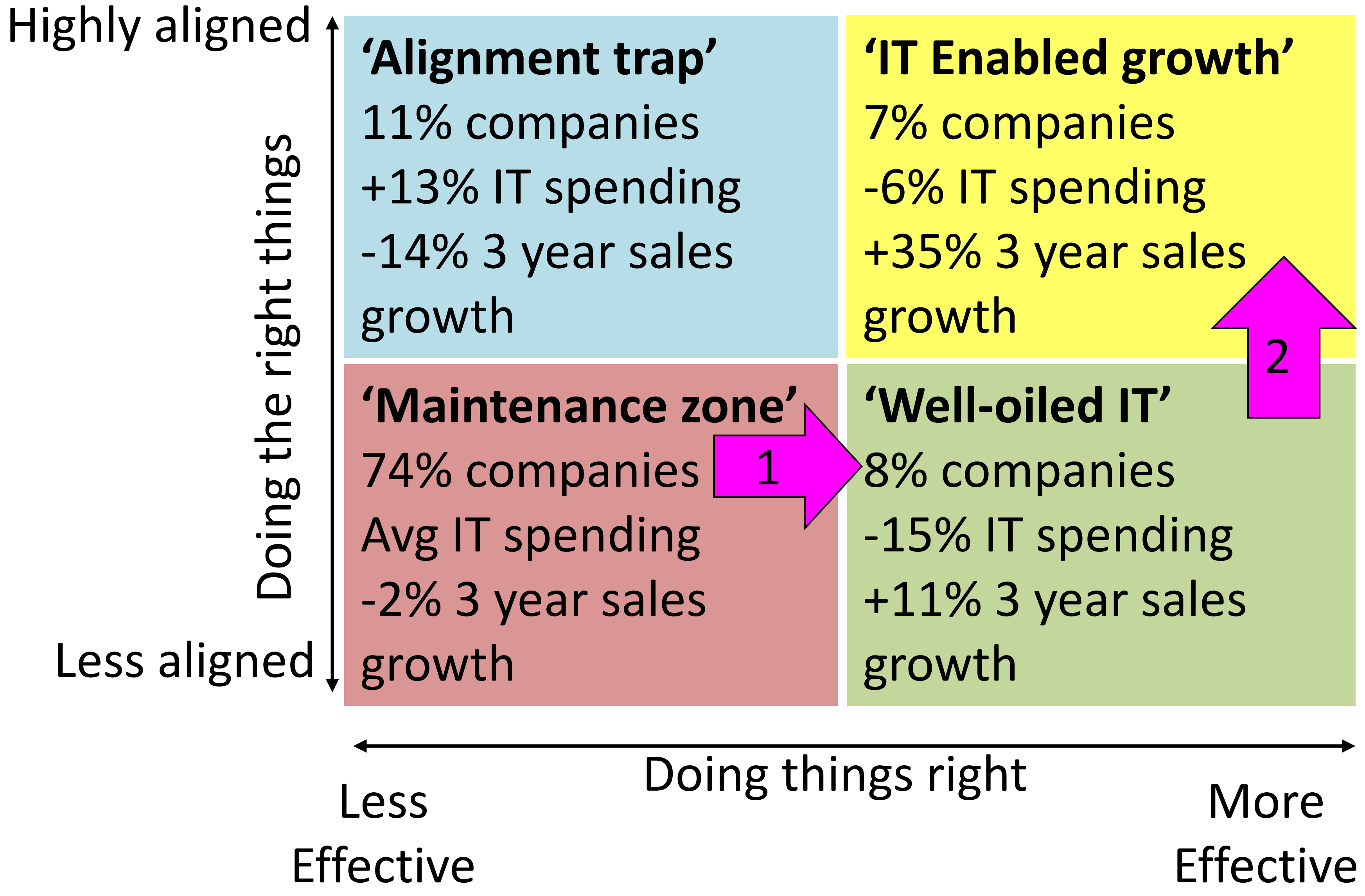
Decide what the right thing is

Build it the right way

“Users do not know what they want until they see working software”

Humphrey's Law
Watts Humphrey

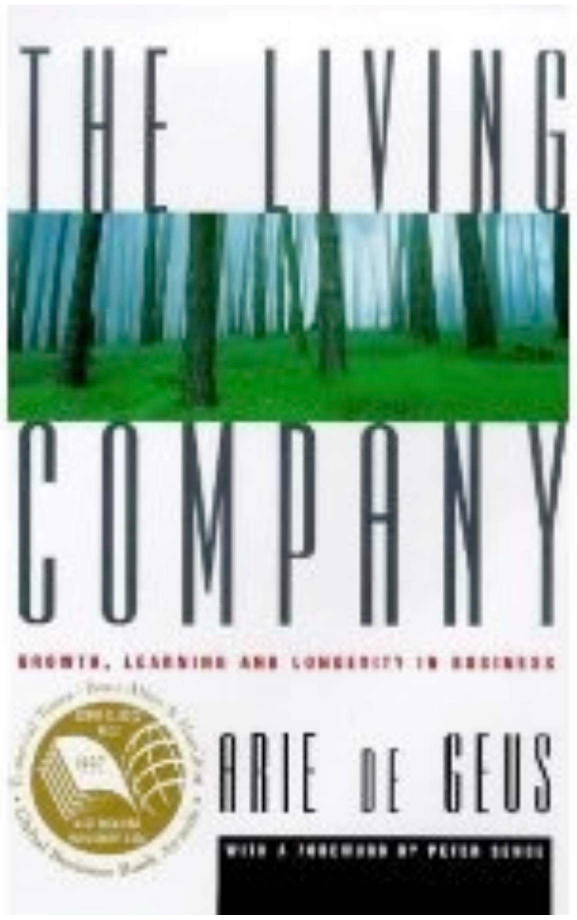
The Alignment Trap



	Less Effective	More Effective
Highly aligned	**‘Alignment trap’** 11% companies +13% IT spending -14% 3 year sales growth	**‘IT Enabled growth’** 7% companies -6% IT spending +35% 3 year sales growth
Less aligned	**‘Maintenance zone’** 74% companies Avg IT spending -2% 3 year sales growth	**‘Well-oiled IT’** 8% companies -15% IT spending +11% 3 year sales growth

Source: Shpilberg, Berez, Puryear, Shah: MIT Sloan Review, Fall 2007

He who learns fastest wins


THE LIVING COMPANY
GROWTH, LEARNING AND LONGEVITY IN BUSINESS
ARIE DE GEUS
WITH A FOREWORD BY PETER SENGE

"We understand that the only competitive advantage the company of the future will have is its **managers' ability to learn faster than their competitors**."

Arie de Geus, *The Living Company* 1988

Learn by doing – iterate!

Today

1) **Do the right thing**

   Build a machine which can iterate

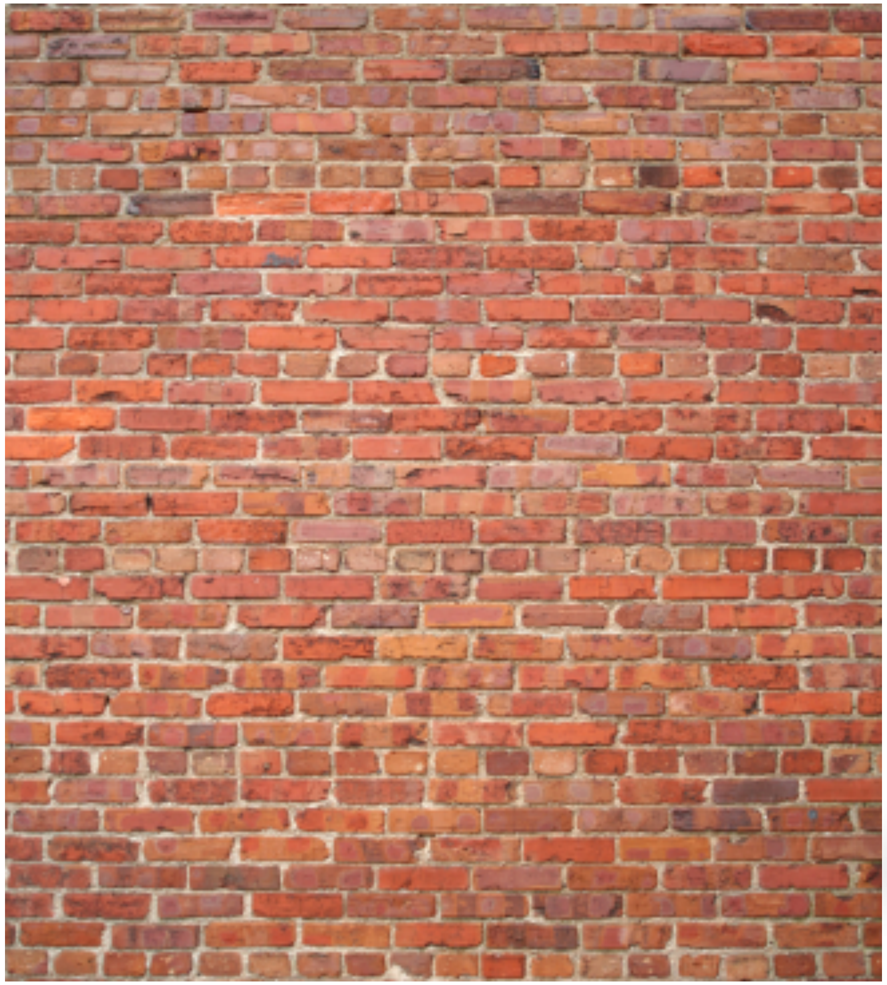
   A learning machine

2) **Do it right**

   Use the machine to iterate your way to the right thing

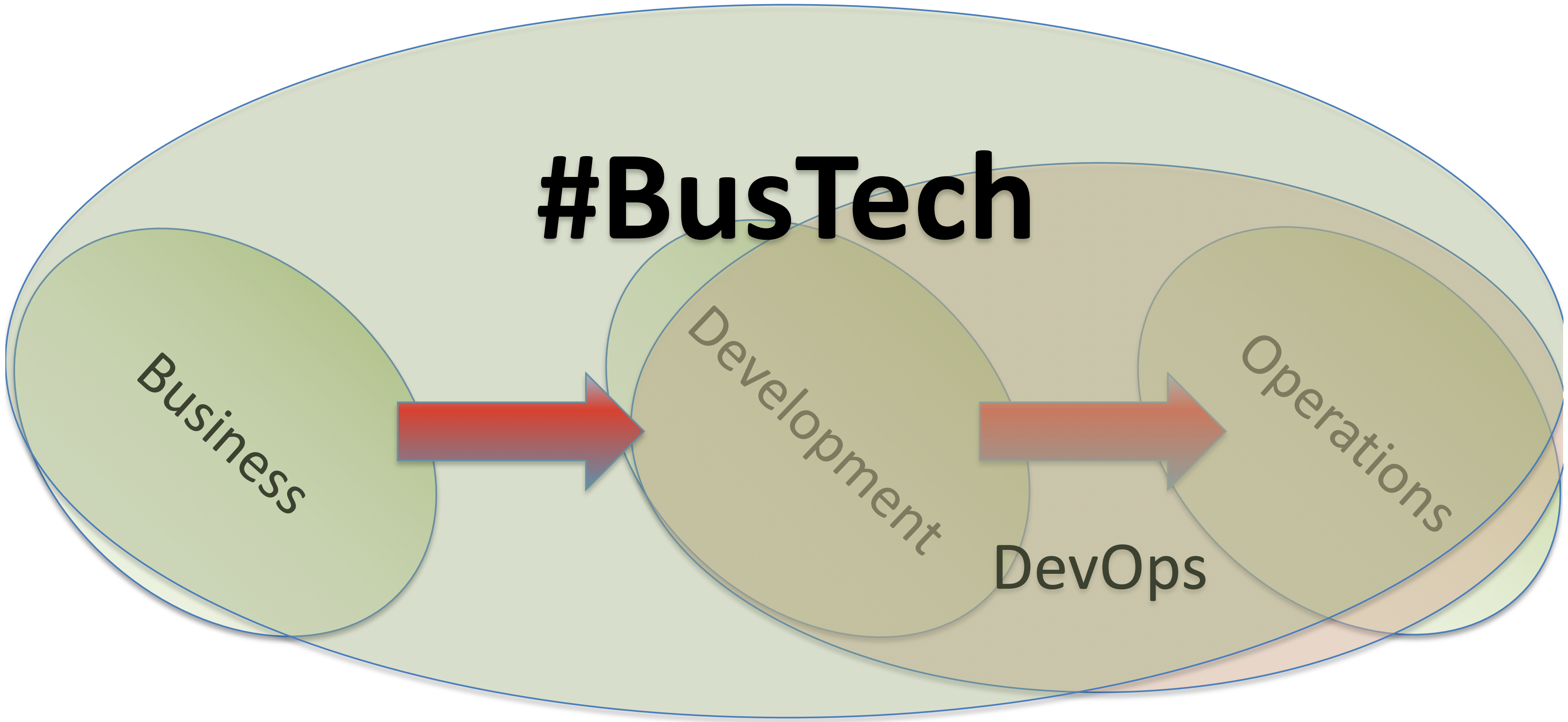
Bring everyone together

Business



Technology

#BusTech



1 Team – No divide

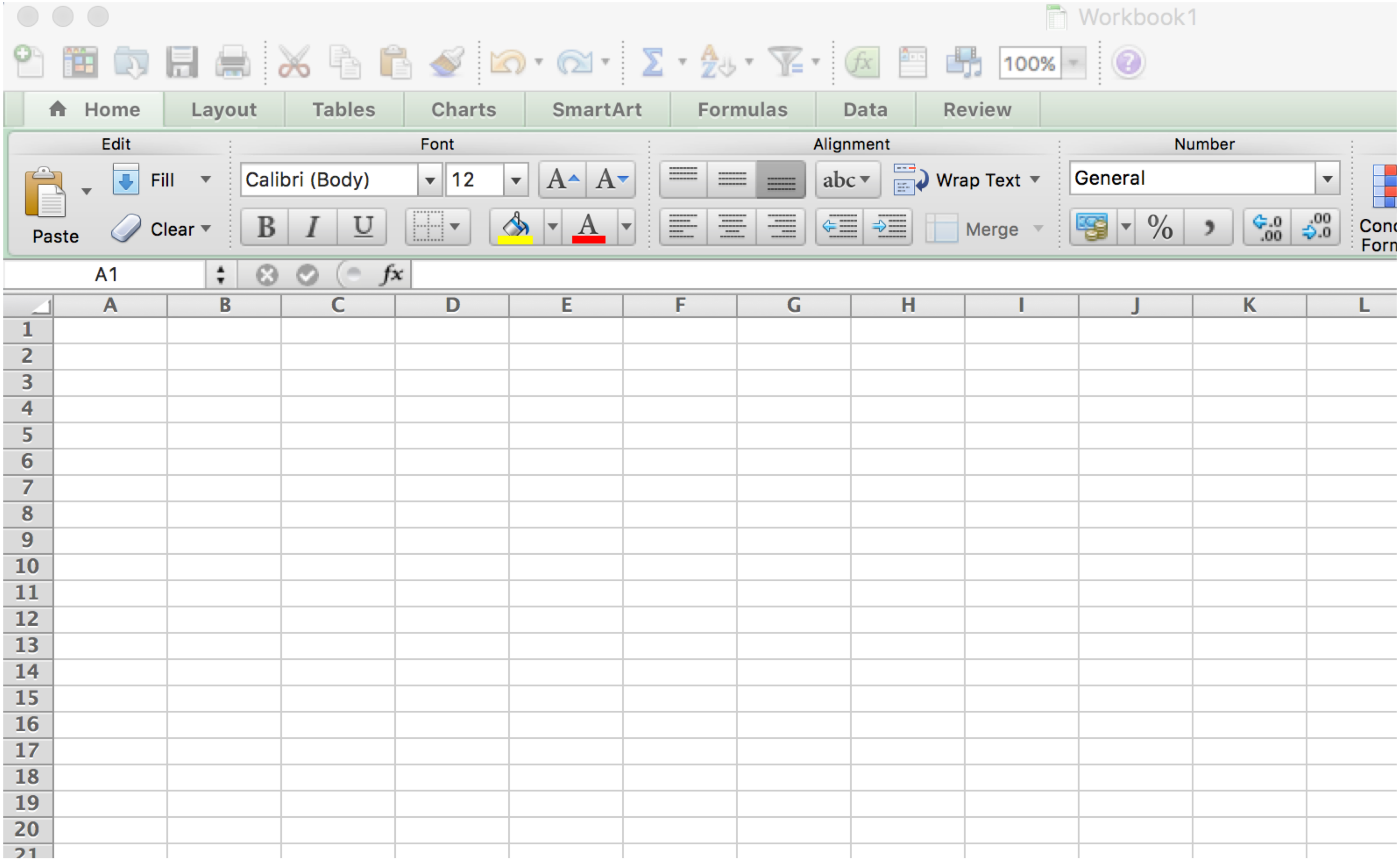
#BusTech

"It is time to open up the development process to include

business people as first class citizens."

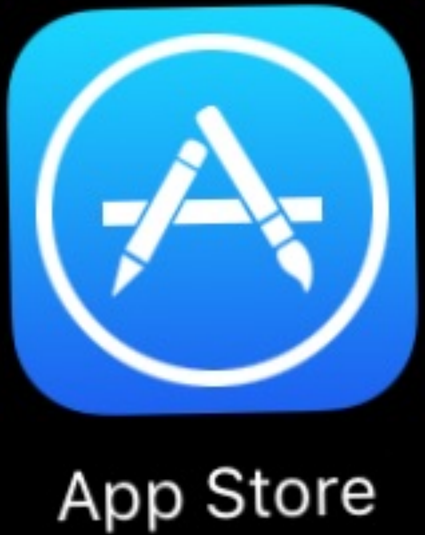
Mel Conway, CraftConf, May 2018

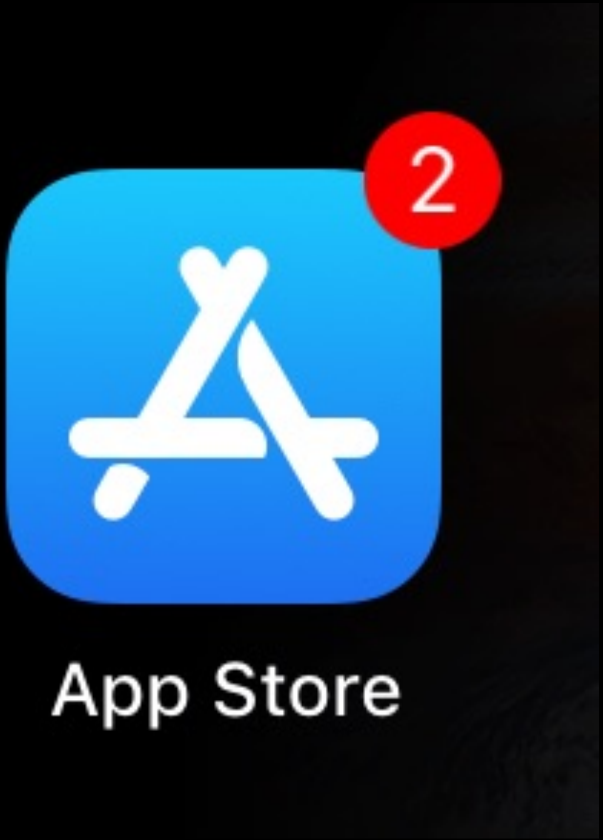
The solution defines the problem



Workbook1
100%
Home
Layout
Tables
Charts
SmartArt
Formulas
Data
Review
Edit
Fill
Clear
Paste
Font
Calibri (Body)
12
B
I
U
Alignment
abc
Wrap Text
Merge
Number
General
%
Conc
Forn
A1
fx
A
B
C
D
E
F
G
H
I
J
K
L

What problem did iOS 11 solve?





iOS 10

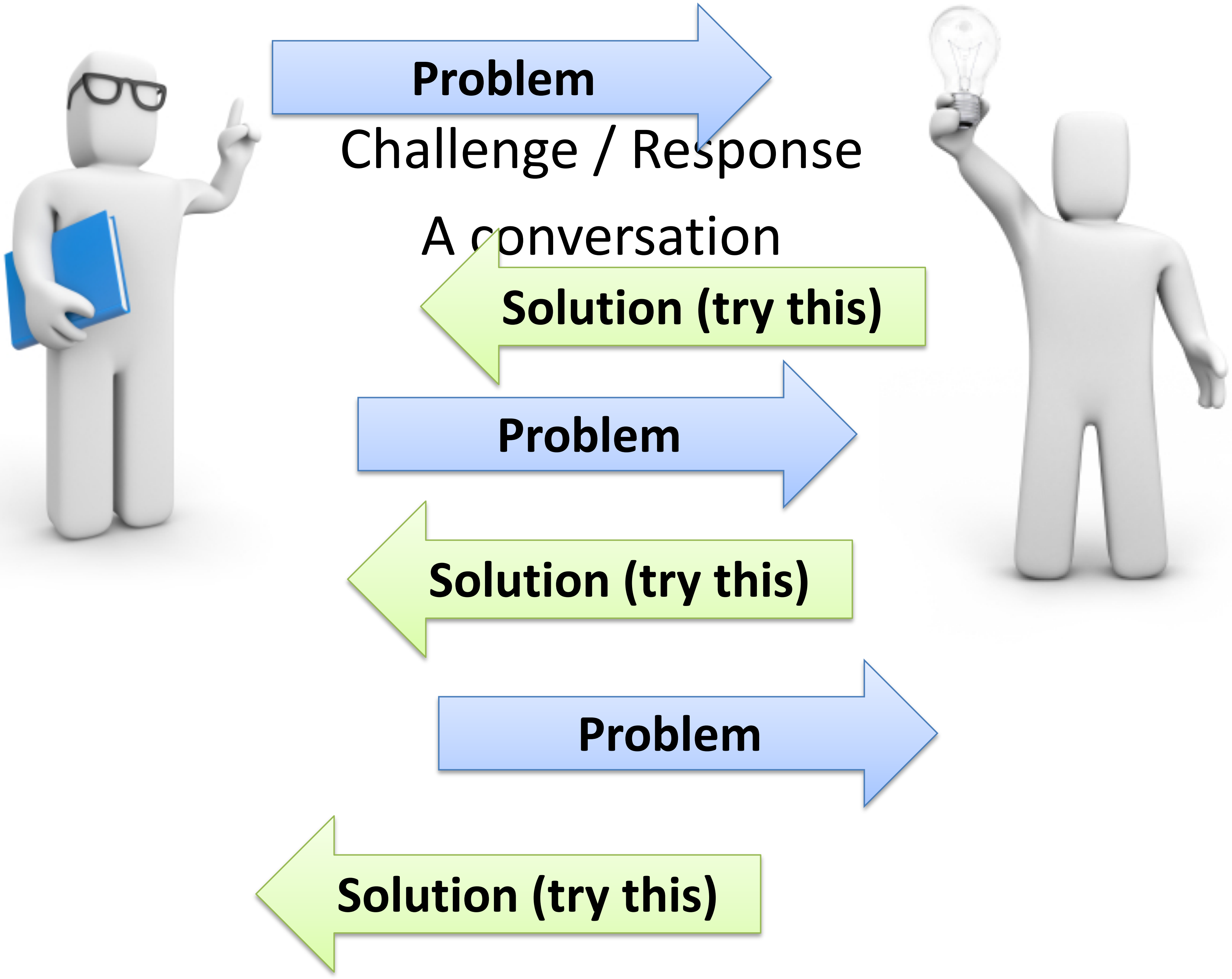
iOS 11

You cannot define what is wanted at the start

Problem understanding & solution co-evolve

Challenge / Response

A conversation



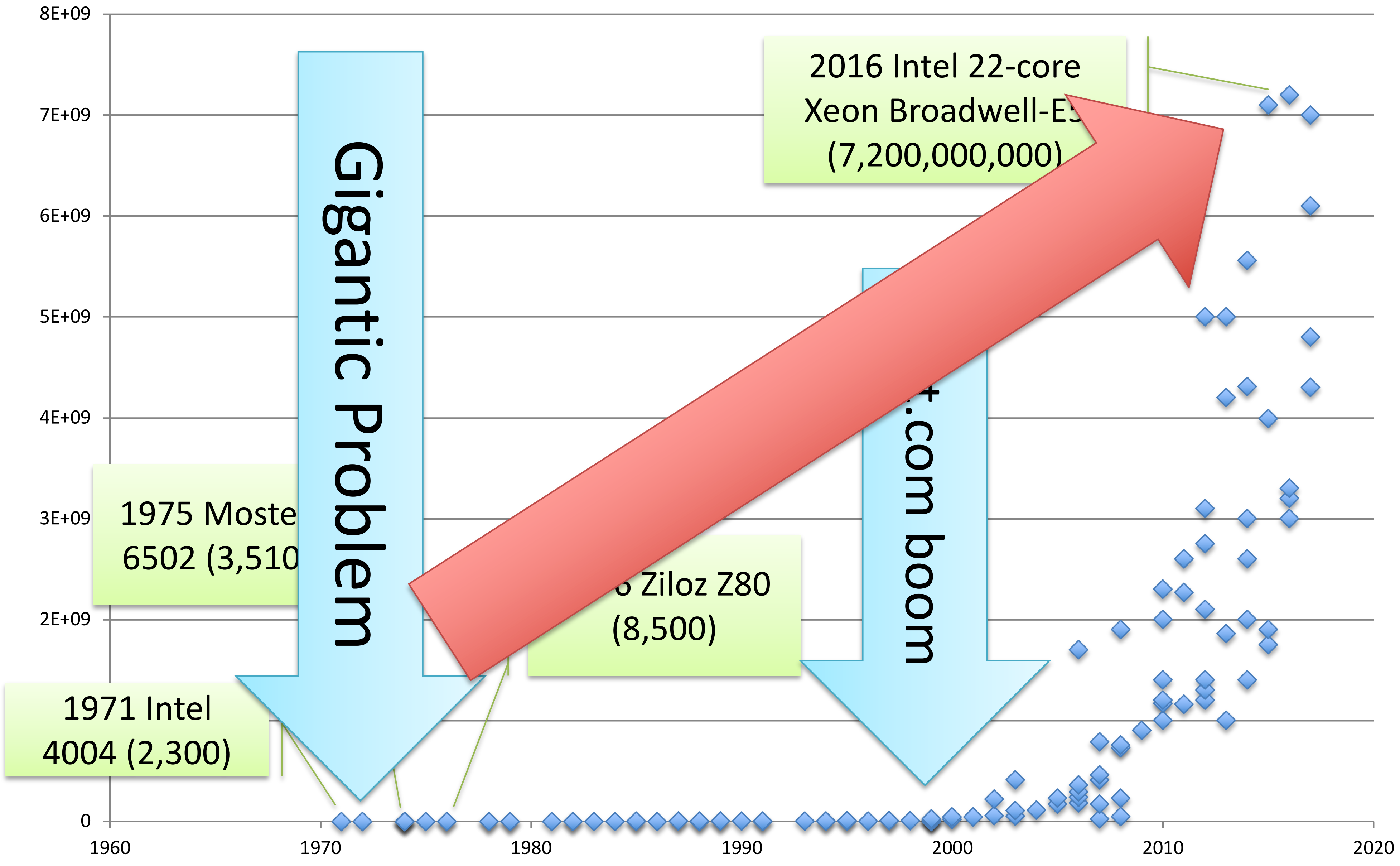
Embrace uncertainty & ambiguity

Solve problems by redefining them

“To put it quite bluntly: as long as there were no machines, programming was no problem at all;
when we had a few weak computers, programming became a mild problem, and now we have gigantic computers, programming has become an equally gigantic problem."

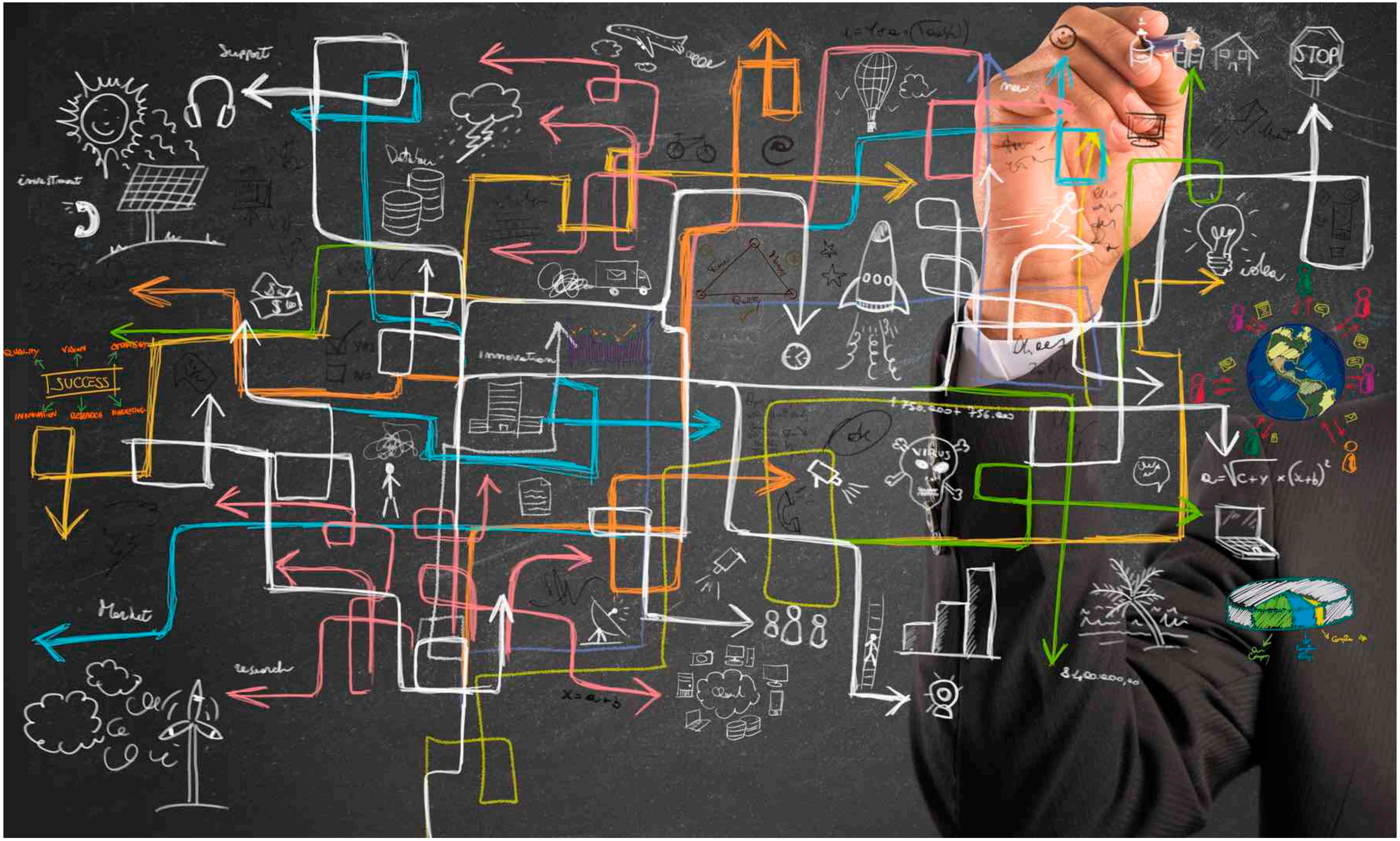
Edsger W. Dijkstra 1972

Transistors per CPU: 1970->2016



Data from https://en.wikipedia.org/wiki/Transistor_count

Complexity


Support
investment
Database
innovation
SUCCESS
QUALITY
VISION
STRATEGY
INNOVATION
RESEARCH
MARKETING
YES
NO
Market
research
VIRUS
idea
STOP
x = a + b

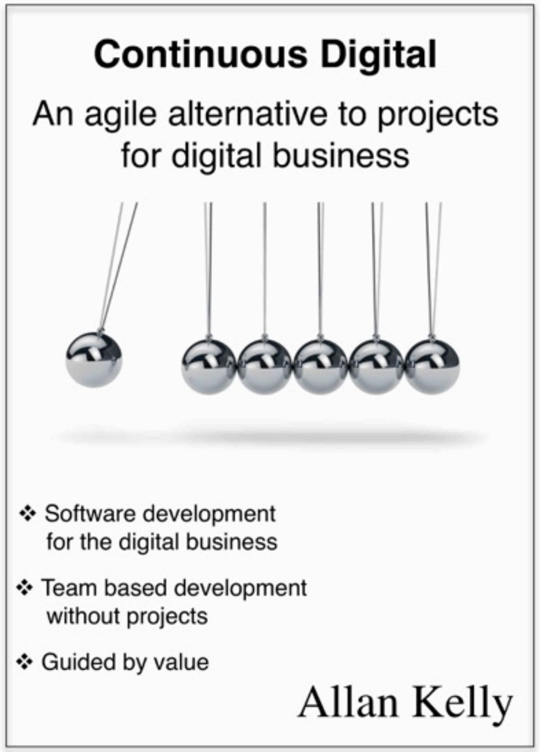
Upside down thinking makes it all more complex

Upside down thinking makes it all more complicated

AllanKelly
www.allankelly.net

1. Diseconomies of Scale
2. Higher quality is faster
3. Quickest way to learn is to do
4. Do it right,

   then do the right thing
5. Solutions defines problem


Continuous Digital

An agile alternative to projects for digital business

❖ Software development for the digital business

❖ Team based development without projects

❖ Guided by value

Allan Kelly

LeanPub
https://leanpub.com/cdigital