

# Why Devs hate Agile and what to do about it

Allan Kelly

@allankellynet

[allan@allankelly.net](mailto:allan@allankelly.net)

[www.allankelly.net](http://www.allankelly.net)

November 2018

Why hate Agile?

It wasn't always so

# Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck  
Mike Beedle  
Arie van Bennekum  
Alistair Cockburn  
Ward Cunningham  
Martin Fowler

James Grenning  
Jim Highsmith  
Andrew Hunt  
Ron Jeffries  
Jon Kern  
Brian Marick

Robert C. Martin  
Steve Mellor  
Ken Schwaber  
Jeff Sutherland  
Dave Thomas

Programmer

Software Tester

Software engineer

Programmer

Programmer

Programmer

Software  
engineer

Software developer

Programmer

Programmer


Software engineer

Software developer

Programmer

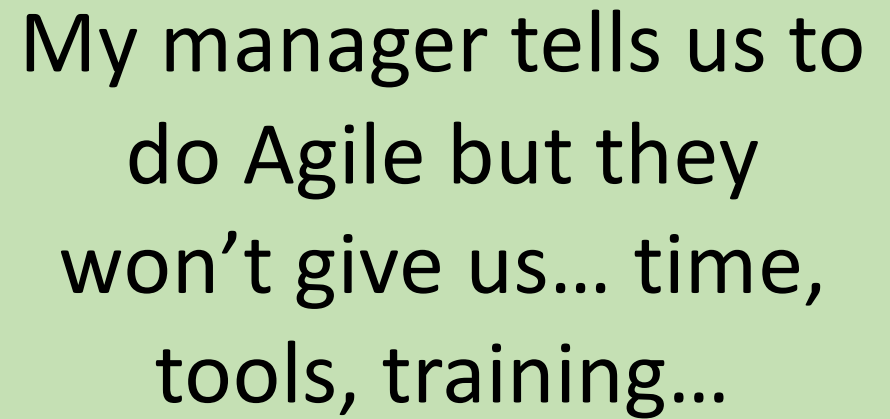
Software  
developer

# Theft

An orange speech bubble with a blue outline and a tail pointing towards the bottom left.

I would love to do  
Agile but my manager  
want let me

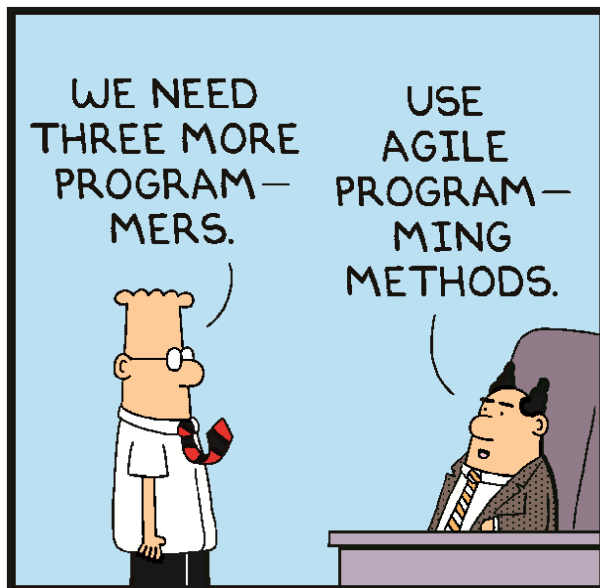
An Other Developer, 2005

A green speech bubble with a blue outline and a tail pointing towards the bottom left.

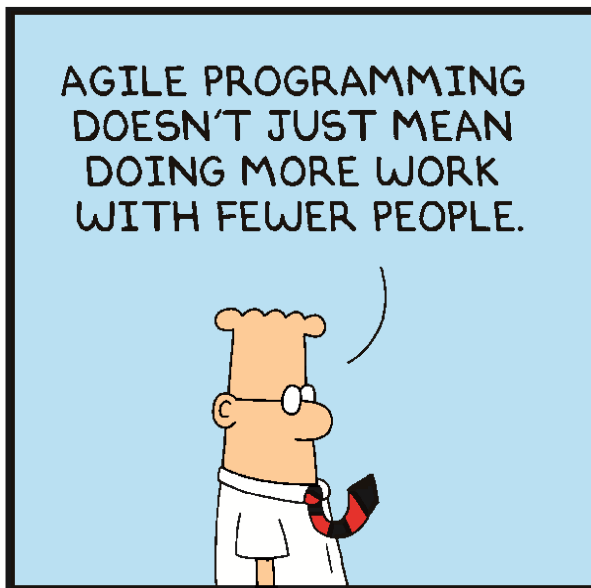
My manager tells us to  
do Agile but they  
won't give us... time,  
tools, training...

J Random Developer, 2018

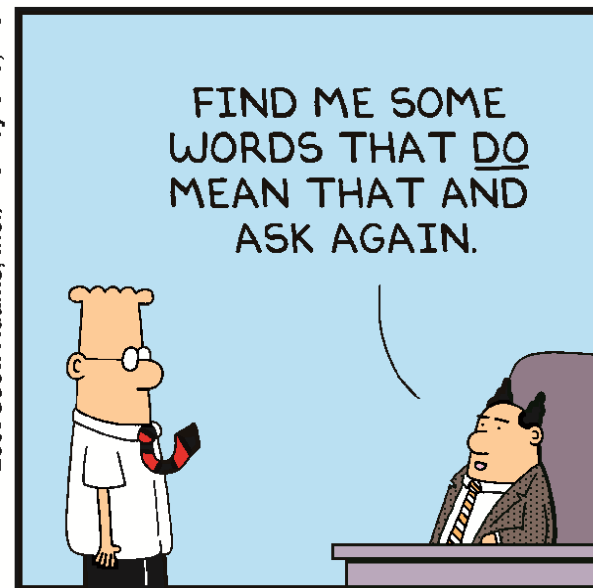




www.dilbert.com scottadams@aol.com



11-16-05 © 2005 Scott Adams, Inc./Dist. by UFS, Inc.



www.dilbert.com scottadams@aol.com



11-26-07 © 2007 Scott Adams, Inc./Dist. by UFS, Inc.



My manager says the team must work agile

**BUT**

He doesn't understand it

The business still demand everything

The deadline doesn't change

We have so much technical debt... there is no corner left to cut... we only make things worse...

You can't create anything useful in two weeks

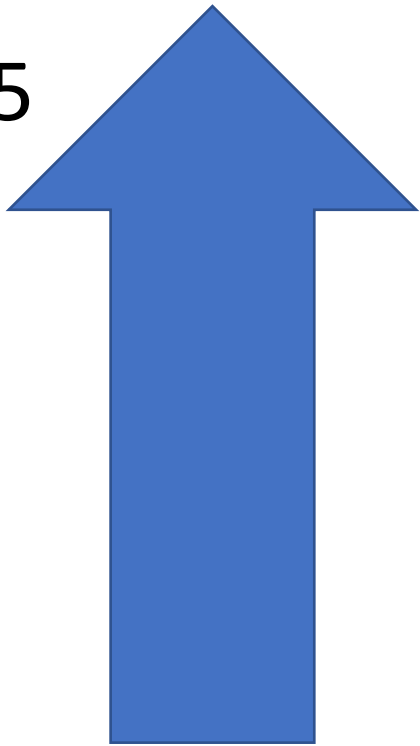
Agile community has only itself to blame

Agile community has spent over 10 years making Agile  
management friendly

In the process agile has often been neutered

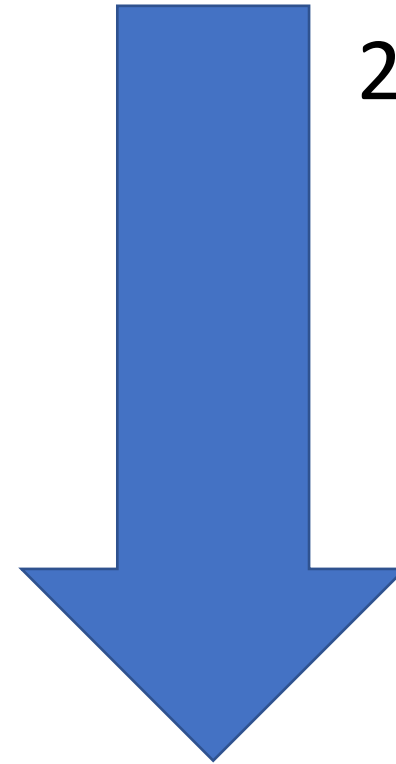
# #1: Agile is now an imposed change

2005



Bottom-up agile

2018



Top-down agile

You!

You have to be involved

You have to be part of it

Educate yourself: books, talks, blogs, ...



# Managers

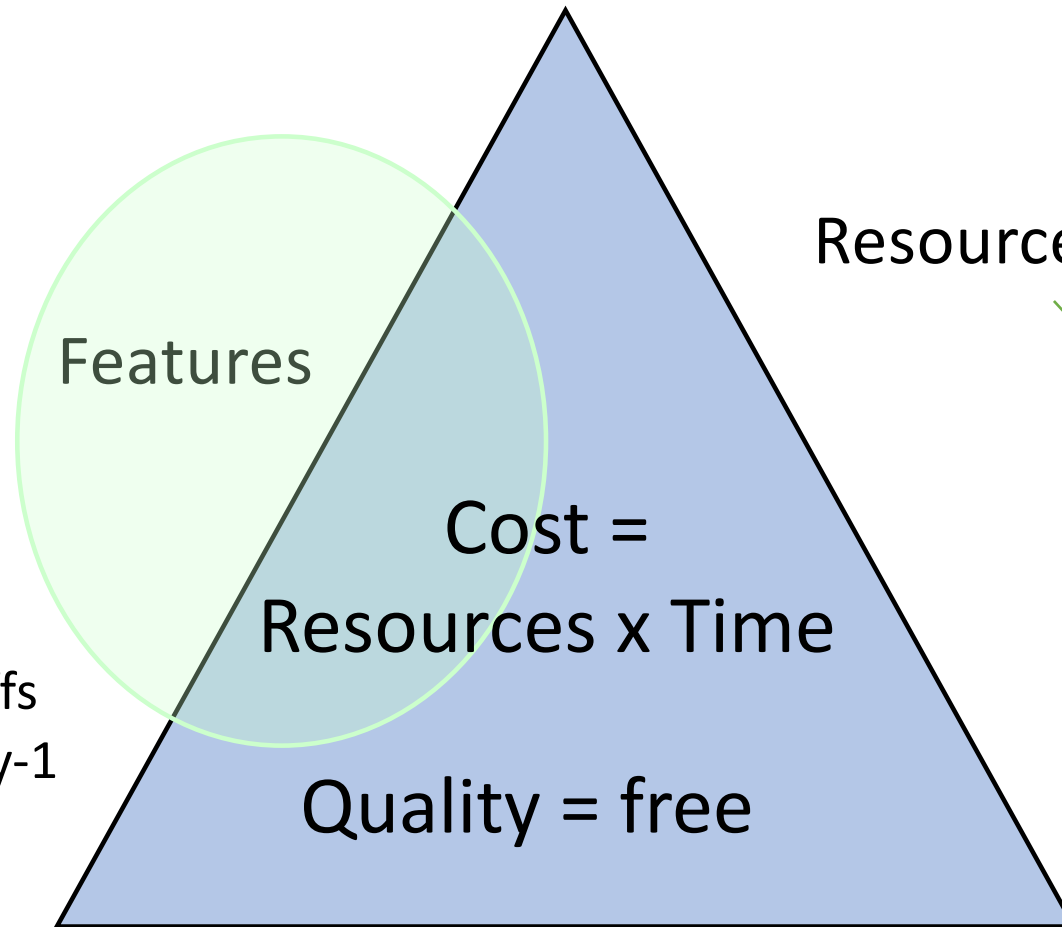
1. Agile should make life better for your staff
2. Involve them – ask them what they want & listen
3. Devolve authority
4. Provide resources: training, time, coaching, books, ...



# Agile

4 realities  
1 change

Trade-offs  
from day-1



Resources (People)

Fixed over  
short run  
(Brooks Law)

Time

Time boxed



#2: Agile without technical quality makes developers lives worse

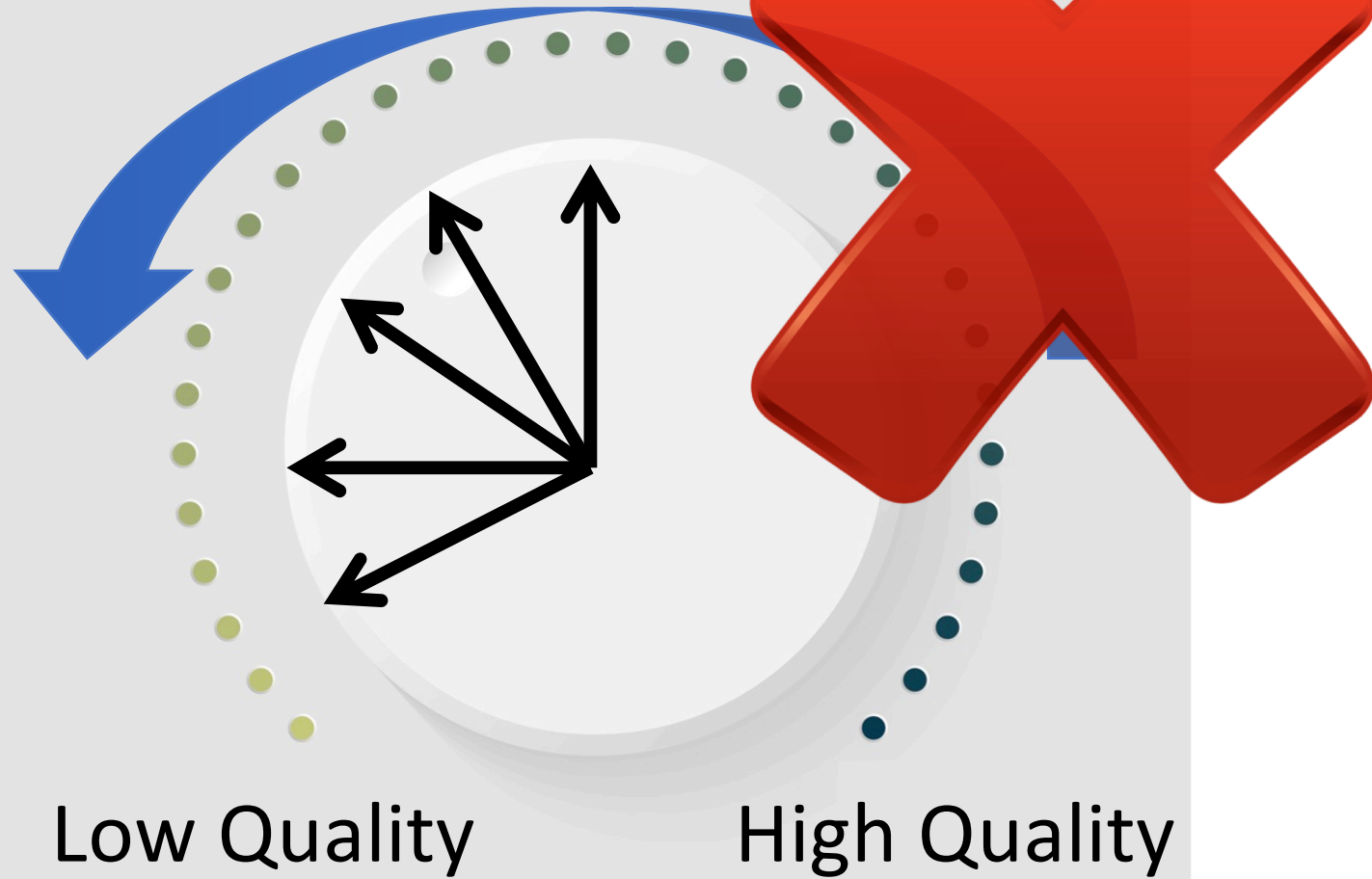
Continuous attention to technical  
excellence and good design  
enhances agility

Poor quality bites you fast

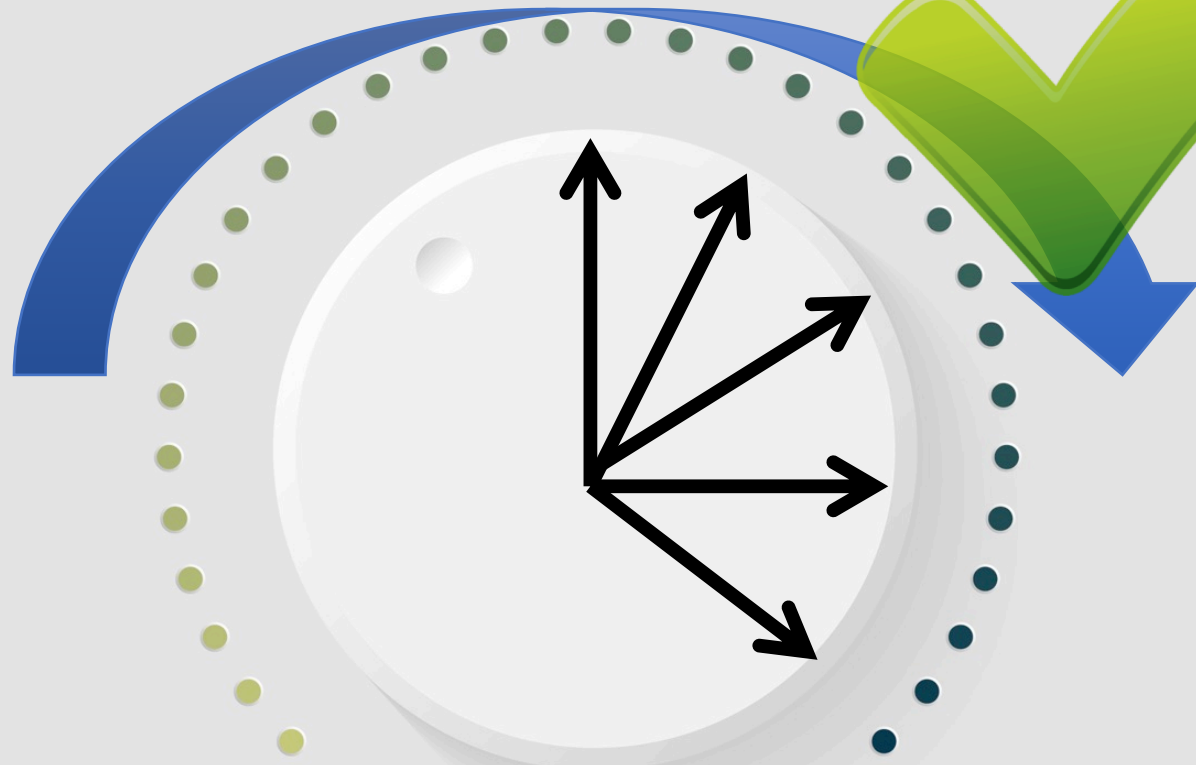
No such thing as quick & dirty

Only slow and dirty

Faster!

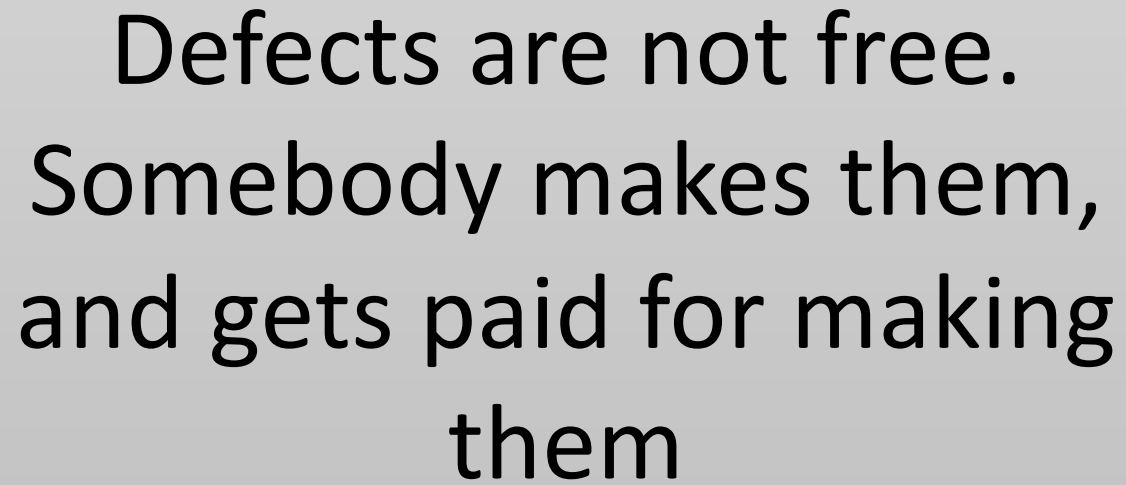


# Faster!



Low Quality

High Quality



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

John Cage  
composer

```
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();  
        }  
    }  
}
```

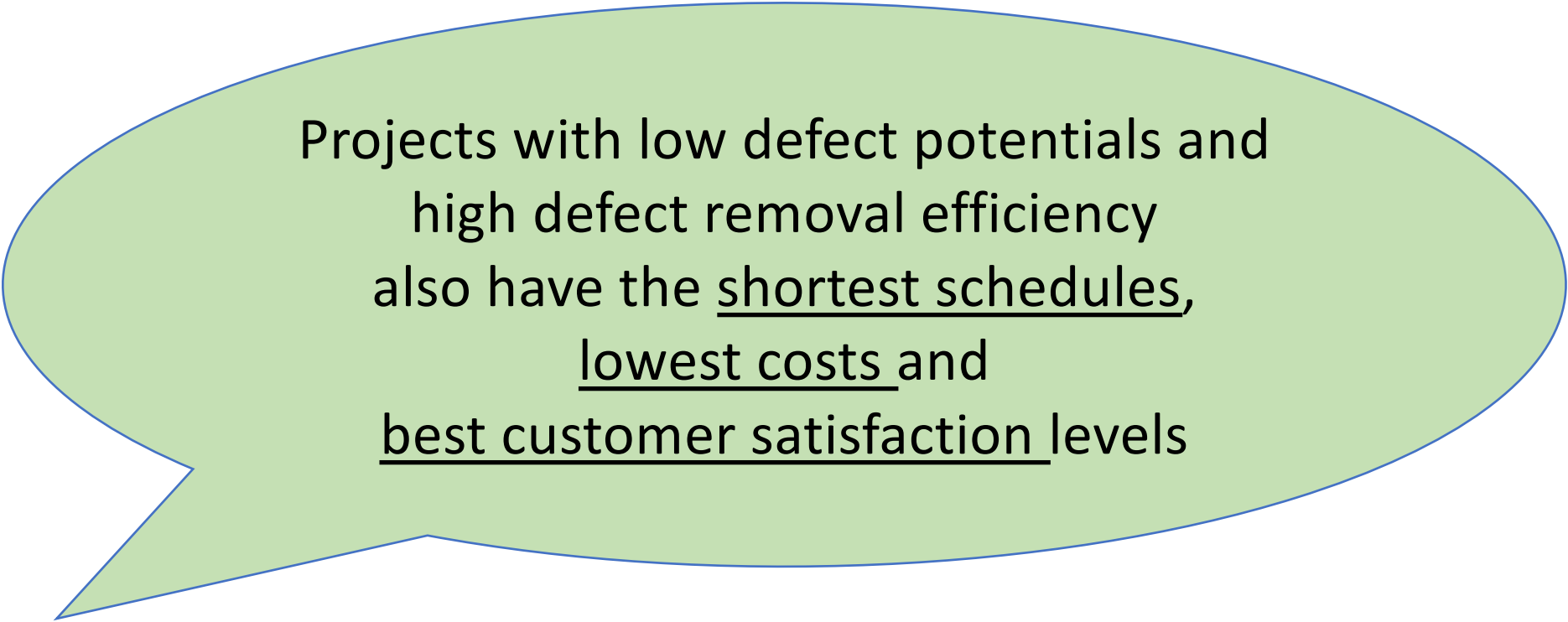
```
    public int Count {  
        get {  
            int size = list.Count;  
            return size; } }  
    public void Add(string newItem) {  
        if (list.Contains(newItem)) {  
            int position =  
                list.IndexOf(newItem);  
            string existingItem = list[position];  
            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](http://www.curbralan.com)



Quality -> Quicker



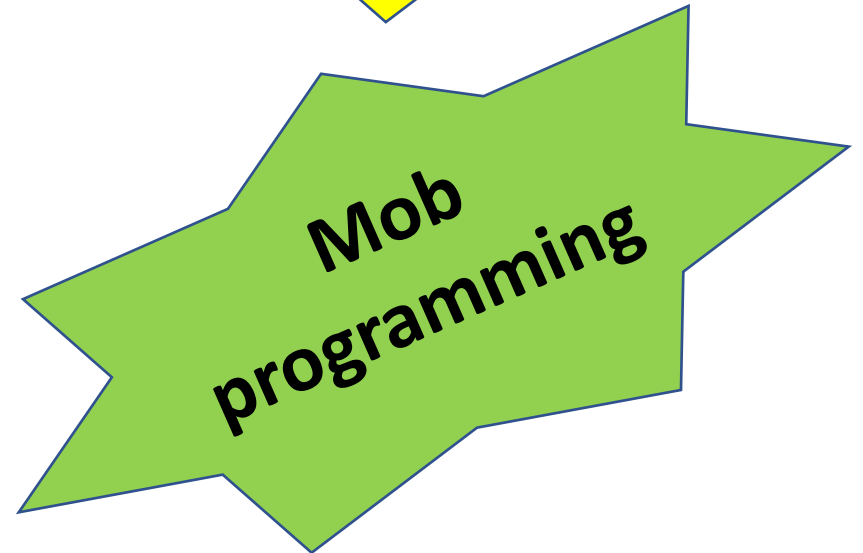
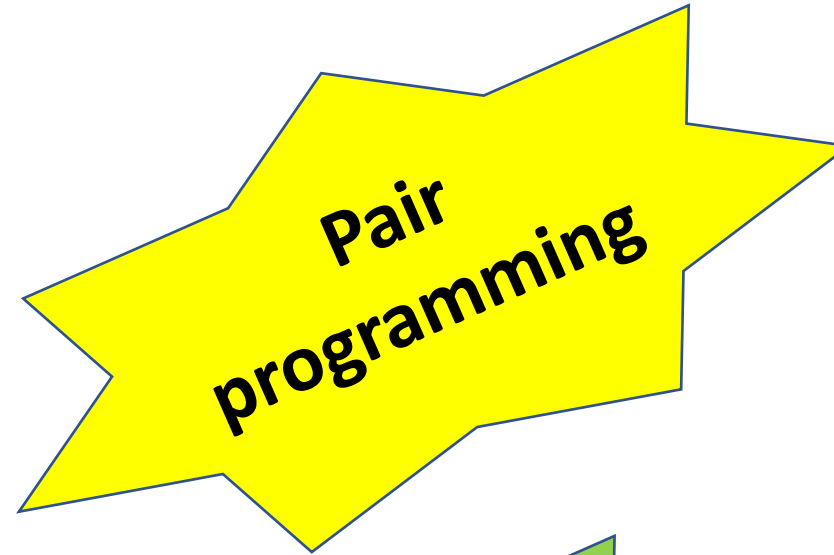
Projects with low defect potentials and  
high defect removal efficiency  
also have the shortest schedules,  
lowest costs and  
best customer satisfaction levels

Capers Jones, 2008

*Applied Software Measurement*

How do you improve quality?

T D D  
A T D D  
B D D



Many teams do Scrum

The best teams do XP

# Testing



You don't have the time  
**not** to test

# You



- You are a professional
- Insist on quality
- Just Do It

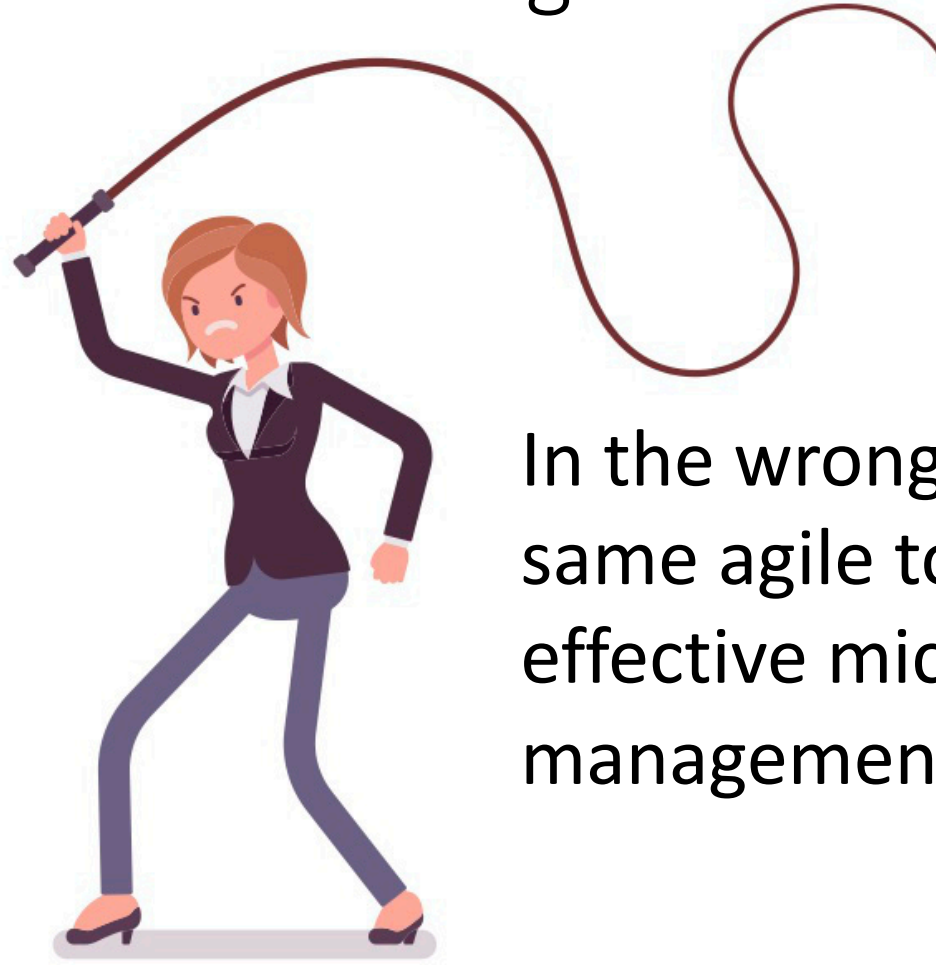


# You

- Learn Test Driven Development
- Practice Test Driven Development
- Learn Behaviour Driven Development
- Encourage Product & Test to learn BDD
- Practice BDD together
- Try pair programming for 2 weeks
- Try mob programming



### #3: Agile can be used for good or evil



In the wrong hands the same agile tools are very effective micro-management tools



# Agile tools alone will make you better

- Stand-up meetings
- Planning meetings
- Visual tracking
- TDD, Refactoring
- Regular releases
- Customer demos
- Etc.
- Etc.

**But....**

Without motivated, involved,  
passionate team members who  
share authority

You will not see the real benefits

Sex without love Is an  
empty experience

Agile without  
involvement is an  
empty experience

Yes, but as empty experiences  
go, it's one of the best.

Yes, but as empty experiences go,  
it's one of the best.



Don't hate agile

Own it

Take back agile

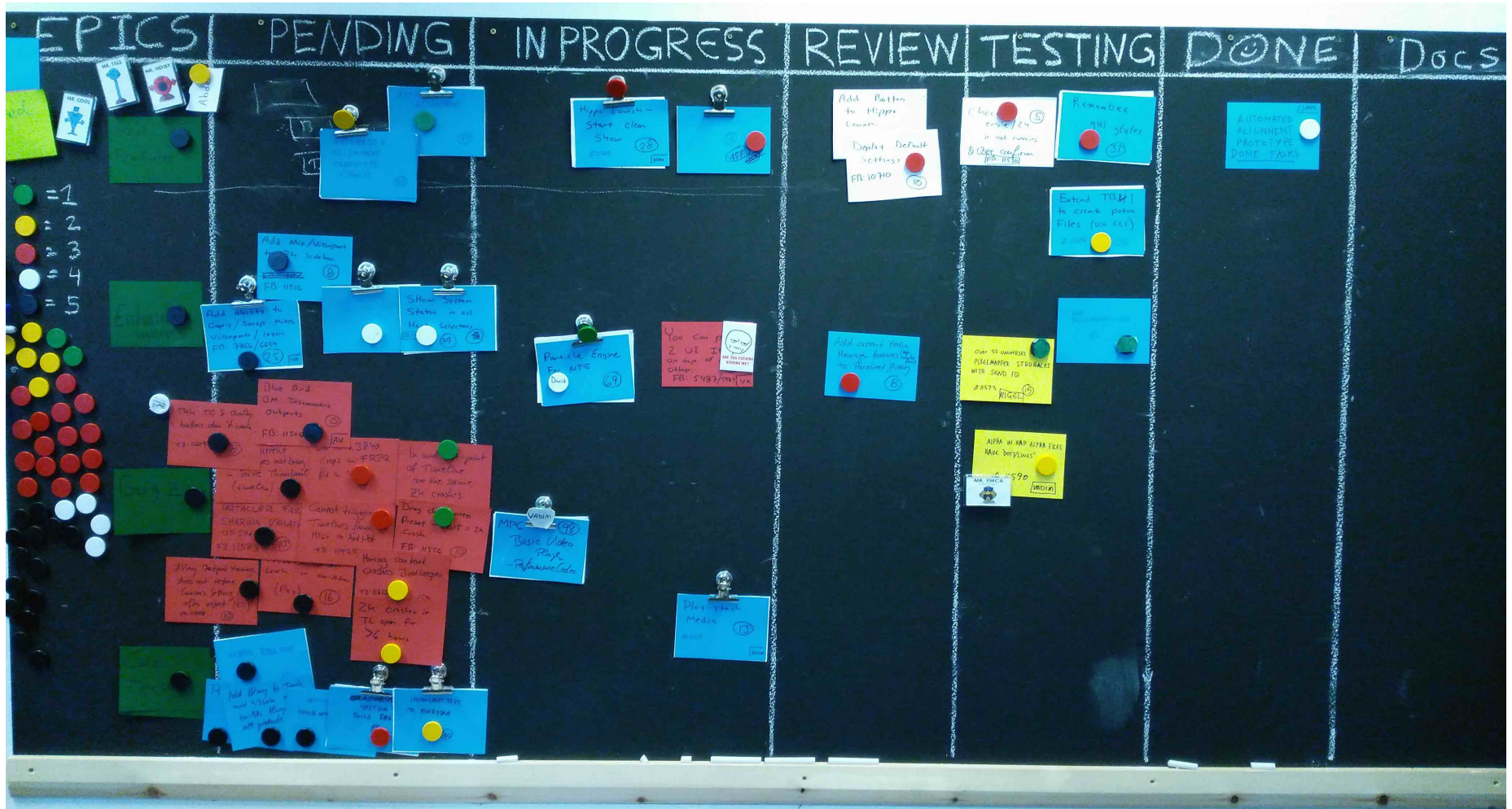
You have to want change

Agile software development is about software

Agile software development is by programmers for  
programmers

Some more ideas

# Visualise your work



Do less work

Limit Work In Progress

Move less work to done faster



# Quality comes first

- Do less
- Do it better

# Hold Retrospectives

<http://www.dialoguesheets.com>

**Retrospective Dialogue Sheet (Sprint V5 A1 - General)**

www.dialoguesheets.com

Use this space however you like notes, ideas, comments and suggestions - you can write **anywhere** you like on this sheet!

**1. Start here**  
This is a dialogue sheet, it is designed to promote good conversation. Team members should seat themselves equally around the sheet so each question can be read by at least one person easily. Take one question at a time, **skip questions if you like**. The person closest to the question should read out the question and take notes of the discussion. Each person should get a chance to read and note at least one question.

**2. Set up**  
Agree how long you will spend working on this sheet and write it in this box:  
If time is short then, in the box next to each question, write the number of minutes you will spend on the question.

**3. Set up**  
Does everyone agree to follow Kerth's Prime Directive (left) while working on this sheet?  
The aim of this sheet is to find better way of working, not to apportion blame. Focus on the cause of problems and not the individuals involved.

**4. Create a timeline**  
Create a timeline for the sprint you are considering in the space above.  
Mark the start and end of the sprint (iteration) then mark significant and memorable events.

**5. Successes**  
What do you consider to be the greatest successes of this sprint?  
Highlight successes on the timeline or write it on the sheet.

**6. Difficulties**  
What difficulties did you encounter during this sprint?  
Record the difficulties on the timeline or elsewhere sheet.

**7. Keep**  
Collectively make a list of all the things you did which you want to keep doing.  
Make a long list, include all ideas then quickly count (and record) how many of you agree with each idea.

**8. Stop**  
Is there anything you should stop doing?

**9. Do different**  
Make a list of things you could do differently next sprint to work better. Include as many ideas as possible - at least 4!

**10. Action plan**  
From the lists in #8 & #9 choose 3 things you will do, or not do, to make the next piece of work better?

**11. Who?**  
Note the names of everyone who took part in this exercise here

**Before you begin**  
The goal of the sheet is to help you work better together. The sheet will take at least one hour to complete and maybe over two hours.

**Kerth's Prime Directive**  
Regardless of what we discover, we must understand and truly believe that everyone did the best given what was known at the time, resources available, and the situation at hand.

**Fix the problem, not the blame.**  
"Inquiry during the same thing over and over again and expecting different results." Albert Einstein

**Nothing in life is to be feared, it is only to be understood. Now is the time to understand more, so that we may fear less.** Marie Curie

**Restrictions aren't limitations, they're just what you happen to be working with!** Steve Paston

**If you want to achieve excellence, you can get there today. As of this second, quit doing less than excellent work.** Thomas J. Watson Sr.

**A change in perspective is worth 80 IQ points.** Alan Kay

**None we can do as little, together we can do so much.** Helen Keller

**Copyright © 2013 Alan Kelly Associates. All rights reserved. This document is the property of Alan Kelly Associates. It is not to be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without the prior written permission of Alan Kelly Associates.**

**Alan Kelly Associates**

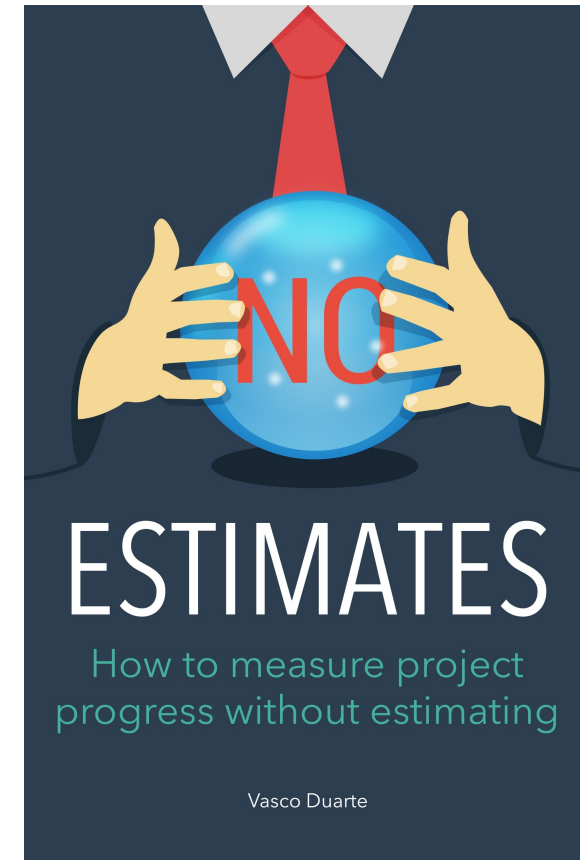
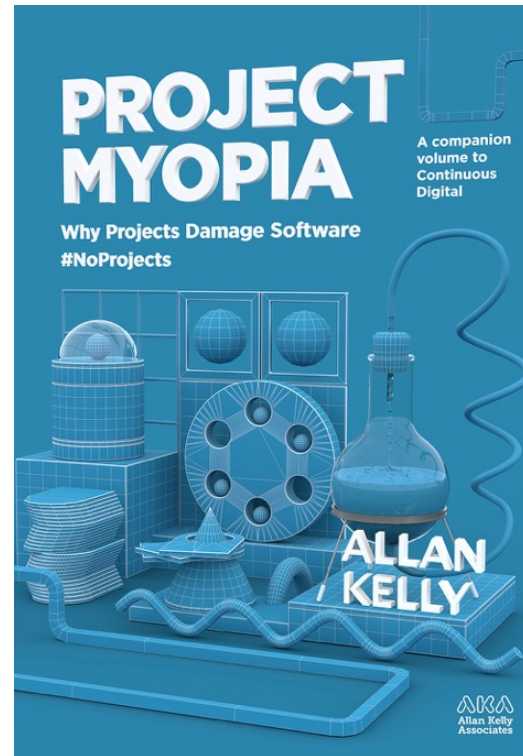
Are you Agile?

What have you changed in the last 3 months?

Are you as good as you could be?

How often do you release? (to live)

Get radical



<http://www.learningconnexions.com/>

#NoEstimates  
Vasco Duarte

# Agile: don't hate it. Own it.

Allan Kelly

allan@allankelly.net

@allankelly.net

