bcs
The Chartered Institute for IT

PROMS-G
Project Management
Specialist Group

Agile Spring School
March 2010

The Future of Agile 2010
A look at where Agile is going

Allan Kelly, allan@allankelly.net

www.allankelly.net

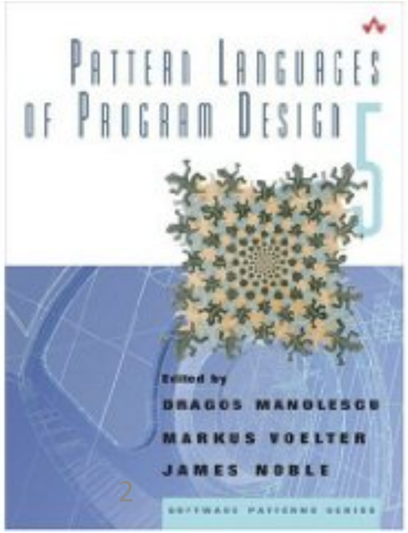
blog.allankelly.net

Allan Kelly, BSc, MBA

Agile coach & consultant

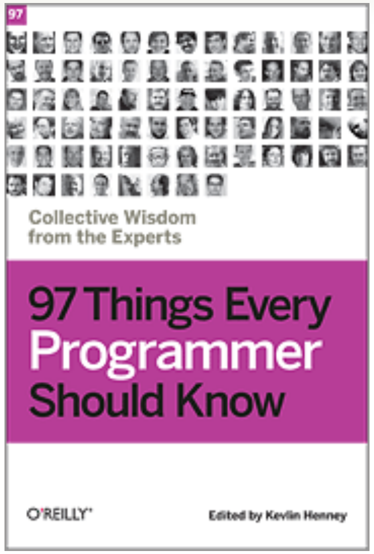
Author:

Changing Software Development: Learning to Be Agile, Wiley 2008

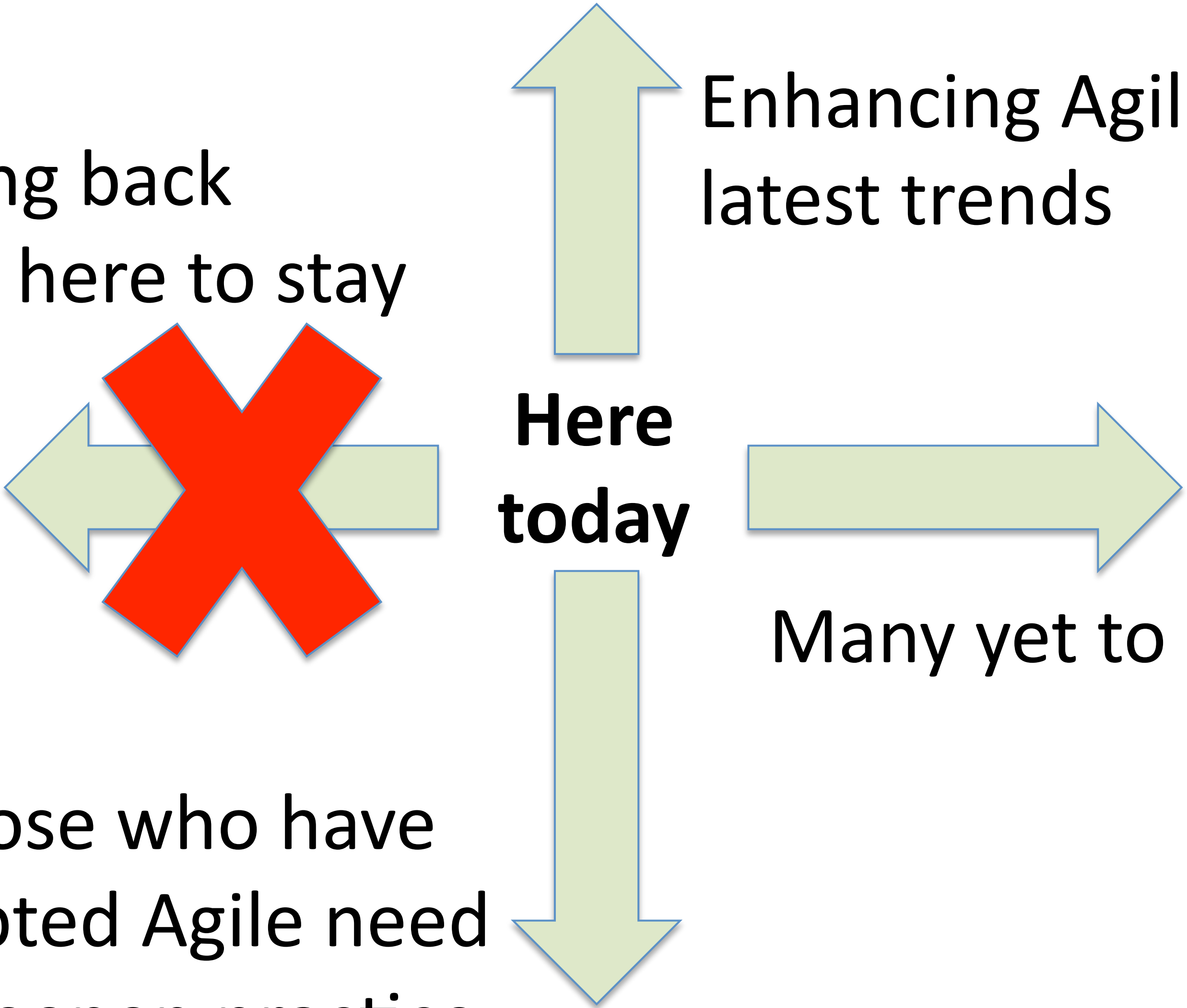


97 Things Every Programmer Should Know,
Hennery, 2010

Encapsulated Context in
Pattern Languages of Program Design 5,
Manolescu, Voelter, Noble, 2006



Which way the future?



Enhancing Agile –
latest trends

No going back
Agile is here to stay

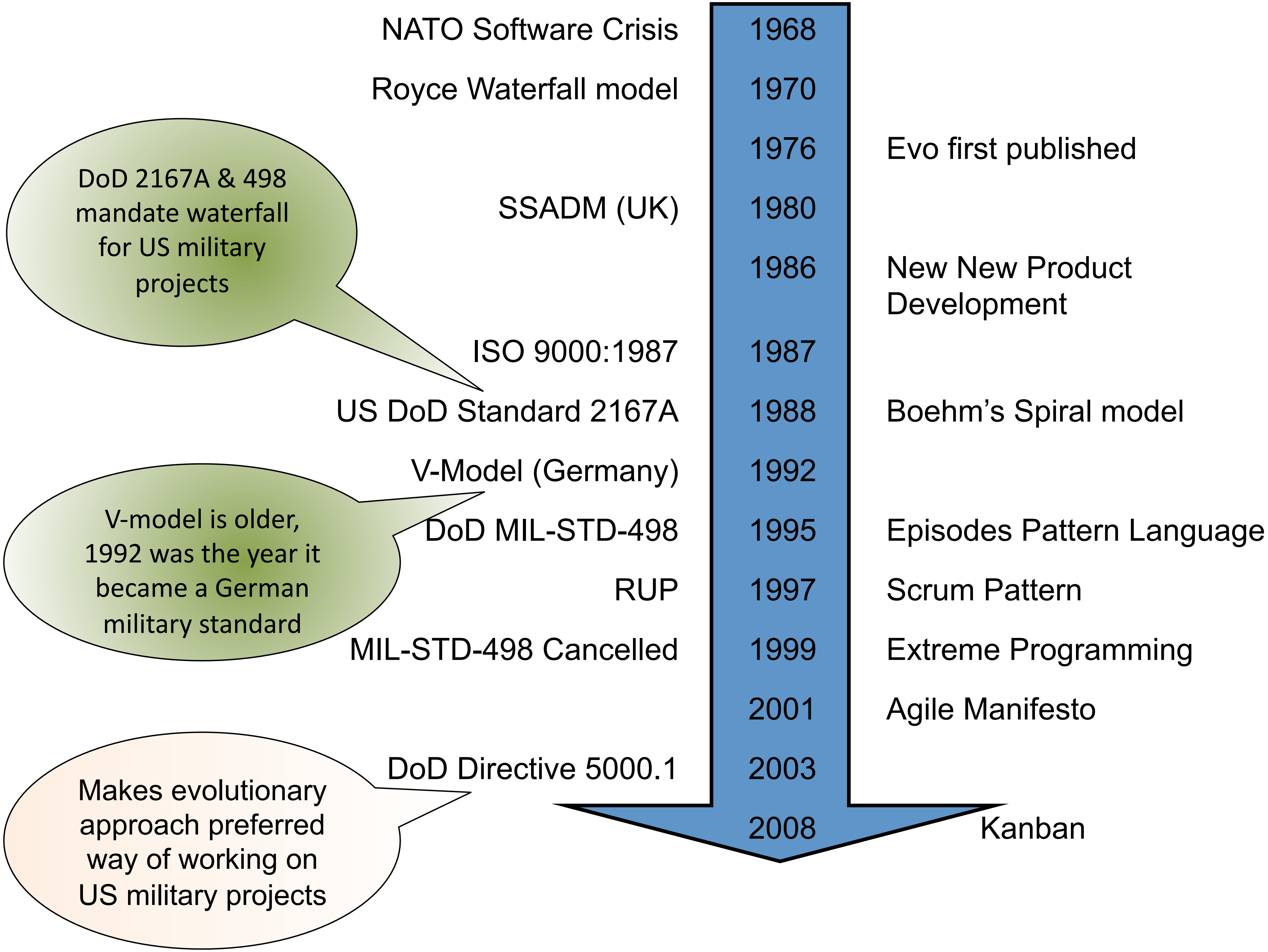
Here today

Many yet to change
How?

Those who have adopted Agile need to deepen practice

No going back

History and context



NATO Software Crisis	1968	
Royce Waterfall model	1970	
	1976	Evo first published
SSADM (UK)	1980	
	1986	New New Product Development
ISO 9000:1987	1987	
US DoD Standard 2167A	1988	Boehm’s Spiral model
V-Model (Germany)	1992	
DoD MIL-STD-498	1995	Episodes Pattern Language
RUP	1997	Scrum Pattern
MIL-STD-498 Cancelled	1999	Extreme Programming
	2001	Agile Manifesto
DoD Directive 5000.1	2003	
	2008	Kanban

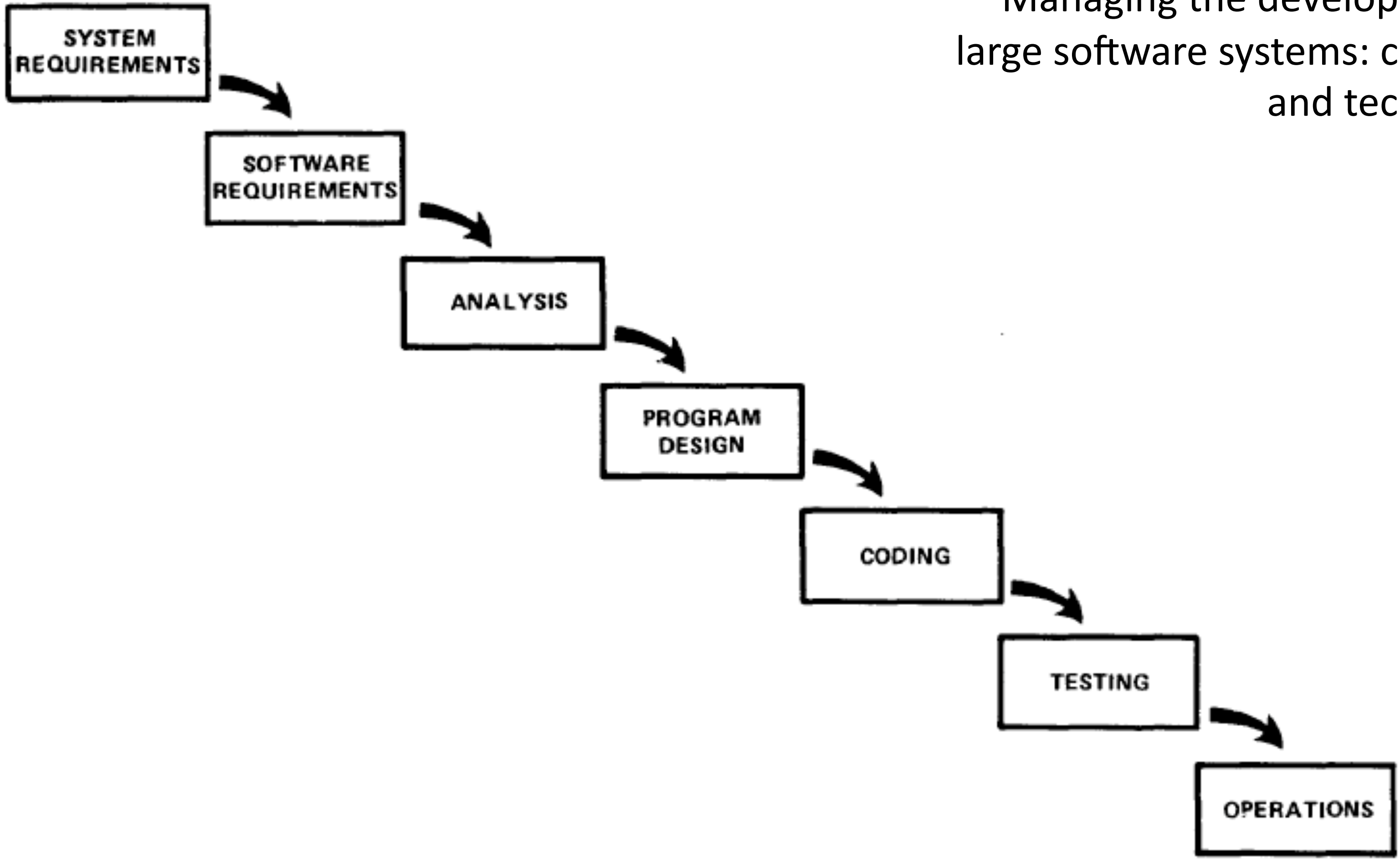
DoD 2167A & 498 mandate waterfall for US military projects

V-model is older, 1992 was the year it became a German military standard

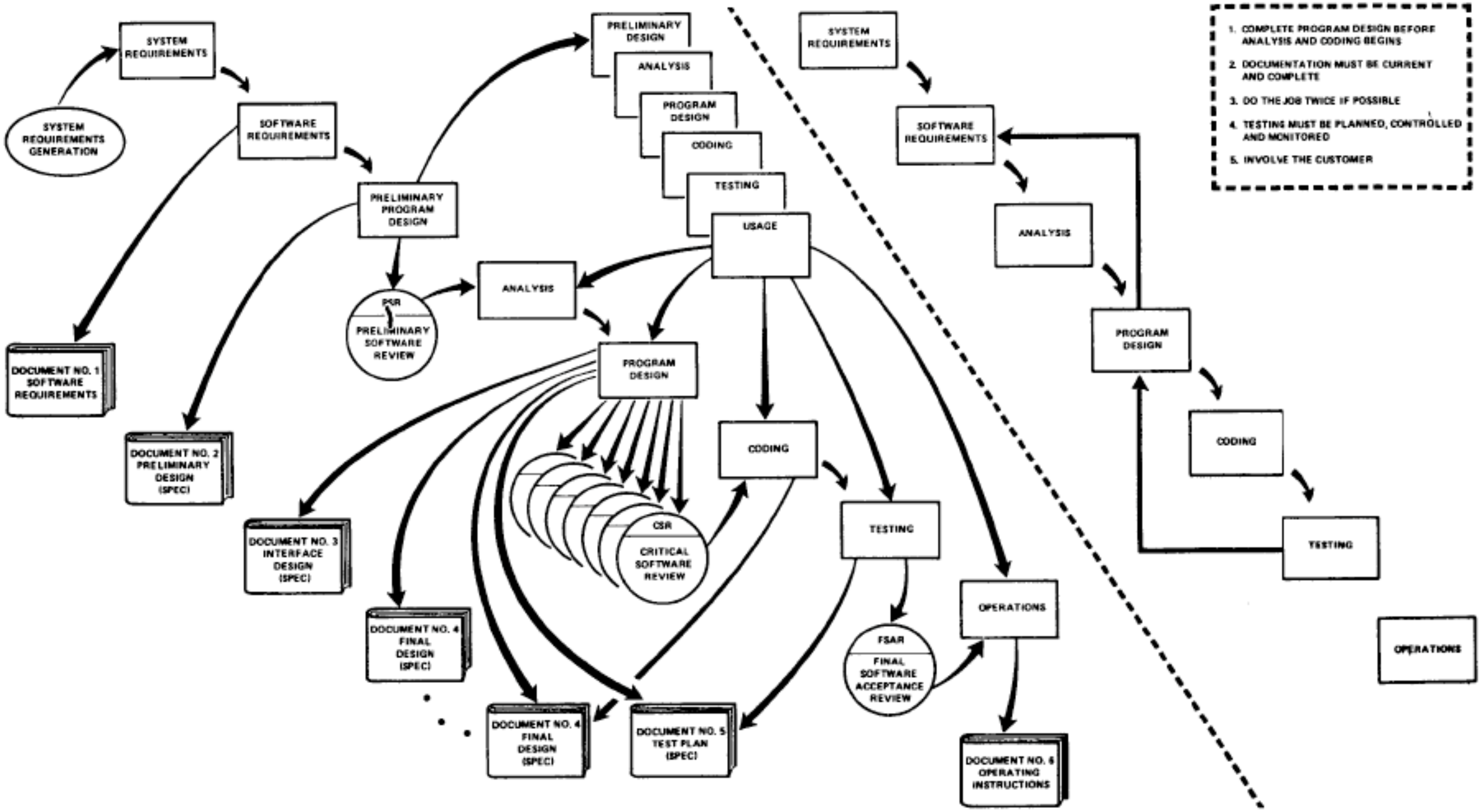
Makes evolutionary approach preferred way of working on US military projects

Royce, 1970

Managing the development of large software systems: concepts and techniques



Royce, 1970


SYSTEM REQUIREMENTS GENERATION
SYSTEM REQUIREMENTS
SOFTWARE REQUIREMENTS
PRELIMINARY PROGRAM DESIGN
PRELIMINARY DESIGN
ANALYSIS
PROGRAM DESIGN
CODING
TESTING
USAGE
PSR
PRELIMINARY SOFTWARE REVIEW
ANALYSIS
PROGRAM DESIGN
CODING
TESTING
OPERATIONS
CSR
CRITICAL SOFTWARE REVIEW
FSAR
FINAL SOFTWARE ACCEPTANCE REVIEW
DOCUMENT NO. 1 SOFTWARE REQUIREMENTS
DOCUMENT NO. 2 PRELIMINARY DESIGN (SPEC)
DOCUMENT NO. 3 INTERFACE DESIGN (SPEC)
DOCUMENT NO. 4 FINAL DESIGN (SPEC)
DOCUMENT NO. 4 FINAL DESIGN (SPEC)
DOCUMENT NO. 5 TEST PLAN (SPEC)
DOCUMENT NO. 6 OPERATING INSTRUCTIONS
SYSTEM REQUIREMENTS
SOFTWARE REQUIREMENTS
ANALYSIS
PROGRAM DESIGN
CODING
TESTING
OPERATIONS

1. COMPLETE PROGRAM DESIGN BEFORE ANALYSIS AND CODING BEGINS
2. DOCUMENTATION MUST BE CURRENT AND COMPLETE
3. DO THE JOB TWICE IF POSSIBLE
4. TESTING MUST BE PLANNED, CONTROLLED AND MONITORED
5. INVOLVE THE CUSTOMER

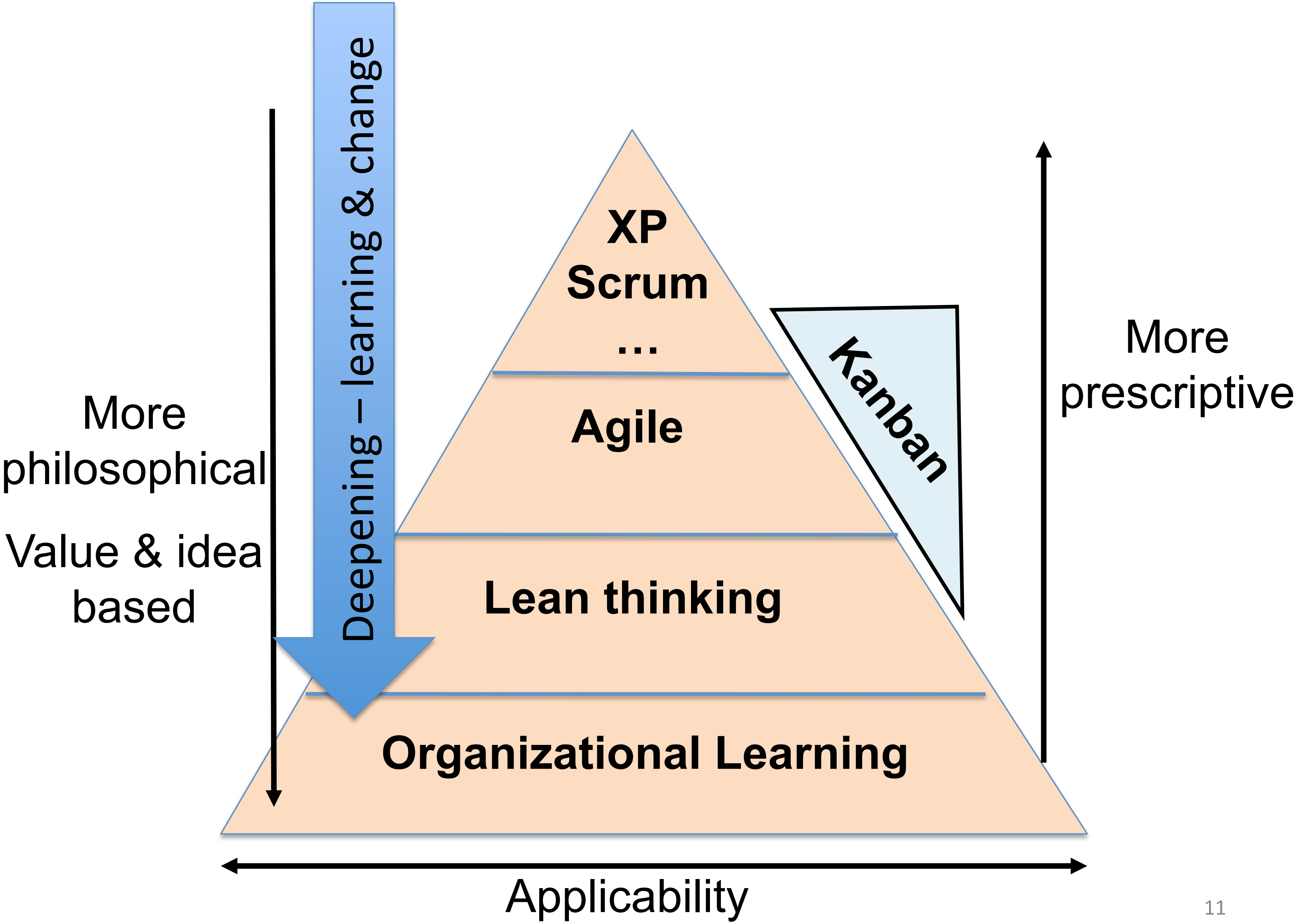
Deepening Agile

Nokia Agile test from Bas Voode

1. Are you doing Iterative Development?
   - Iterations timeboxed to less than 4 weeks
   - Features tested and working at the end of each iteration
   - Iteration starts before specification is complete
2. Are you are doing Scrum?
   - You know who the product owner is
   - There is a product backlog
     - Prioritized by business value
     - With estimates created by the team
   - Team generates burndown charts and knows velocity
   - No project managers (or anyone else) disrupting the work of the team

Three tests of Agile

1. Is the team delivering business value to the customer (market) ?
2. Is the team learning, changing and improving?
3. When the consultants, trainers and coaches who introduced Agile leave, are you still Agile?



Deepening – learning & change

More philosophical

Value & idea based

XP

Scrum

…

Agile

Lean thinking

Organizational Learning

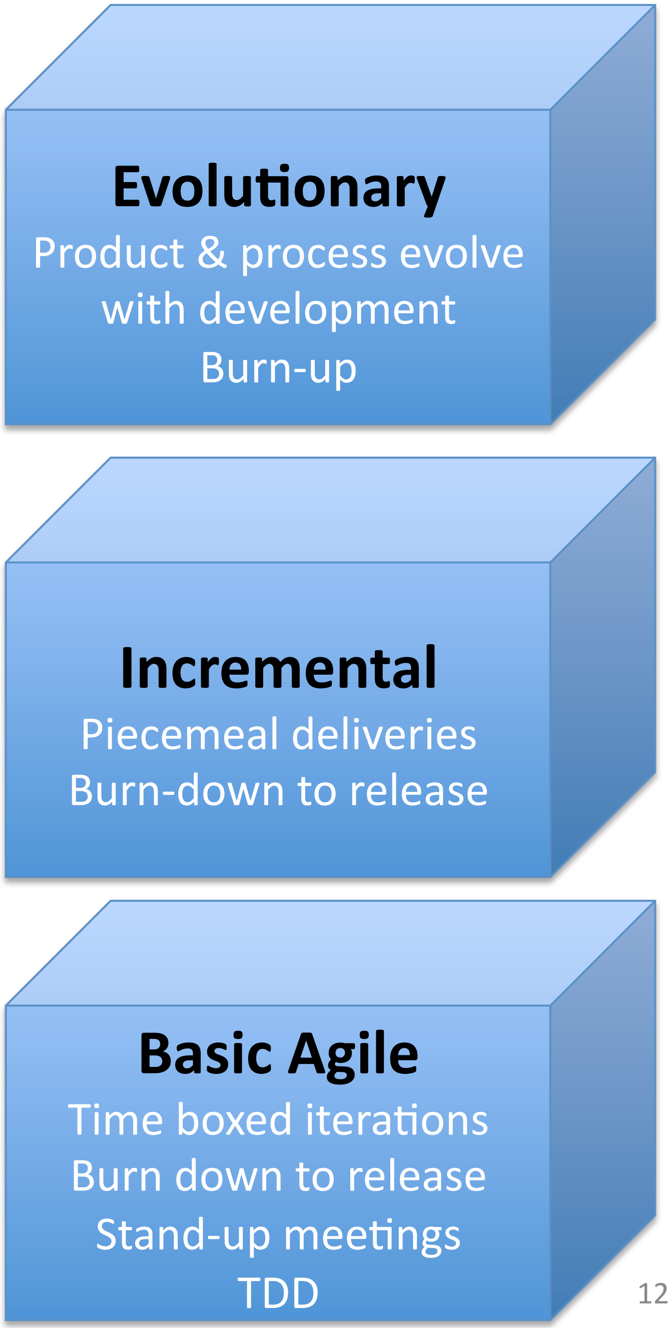
Kanban

More prescriptive

Applicability

Deepening Agile

- Adopt more practices
- Organization structure supports Agile
- Enhance learning
- Expand boundaries
- Embrace change



Evolutionary

Product & process evolve with development

Burn-up

Incremental

Piecemeal deliveries

Burn-down to release

Basic Agile

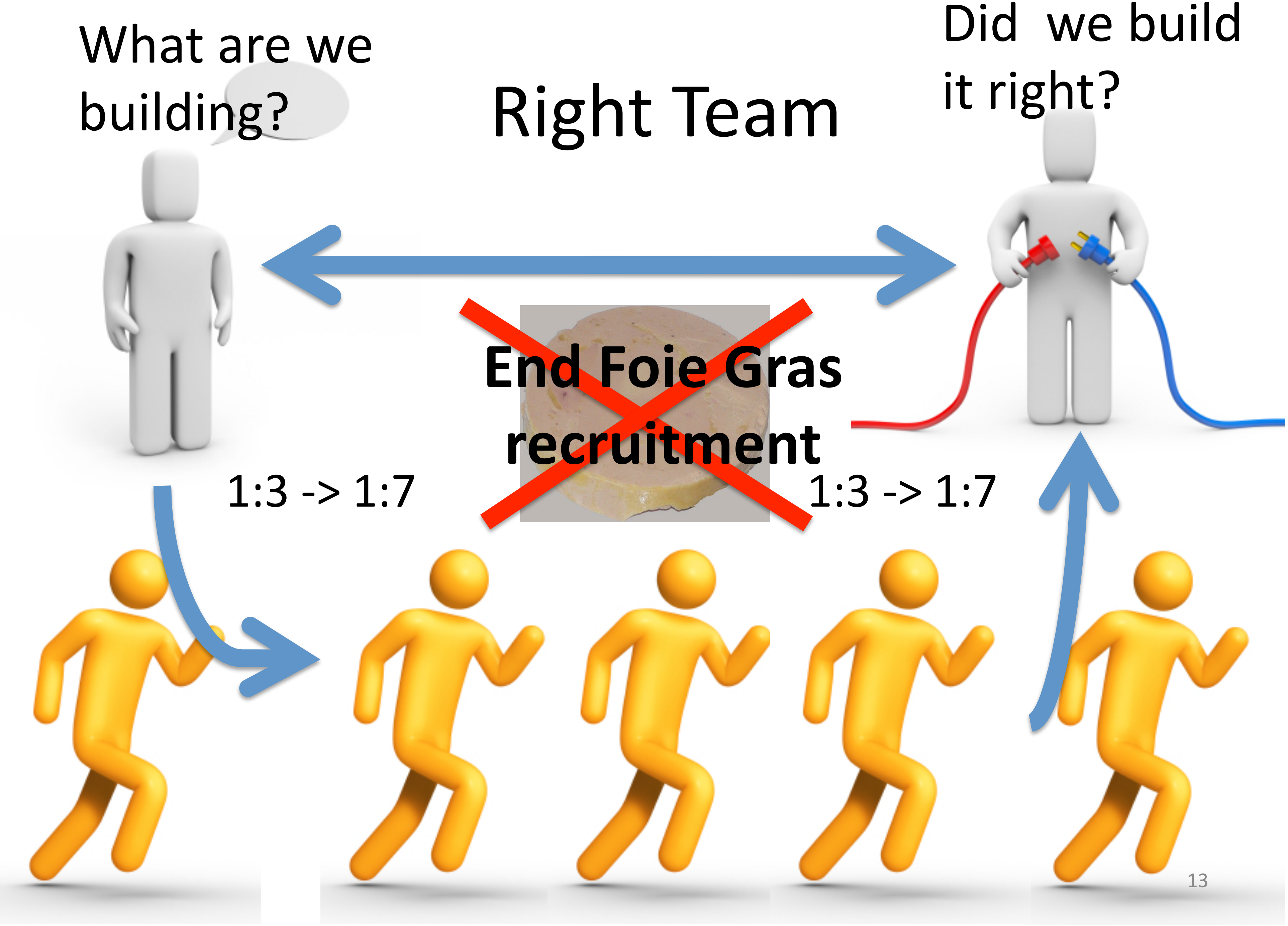
Time boxed iterations

Burn down to release

Stand-up meetings

TDD

Right Team



What are we building?

Did we build it right?

~~End Foie Gras recruitment~~

1:3 -> 1:7

1:3 -> 1:7

Back to the Future

What not *How*

and

The Return of the …

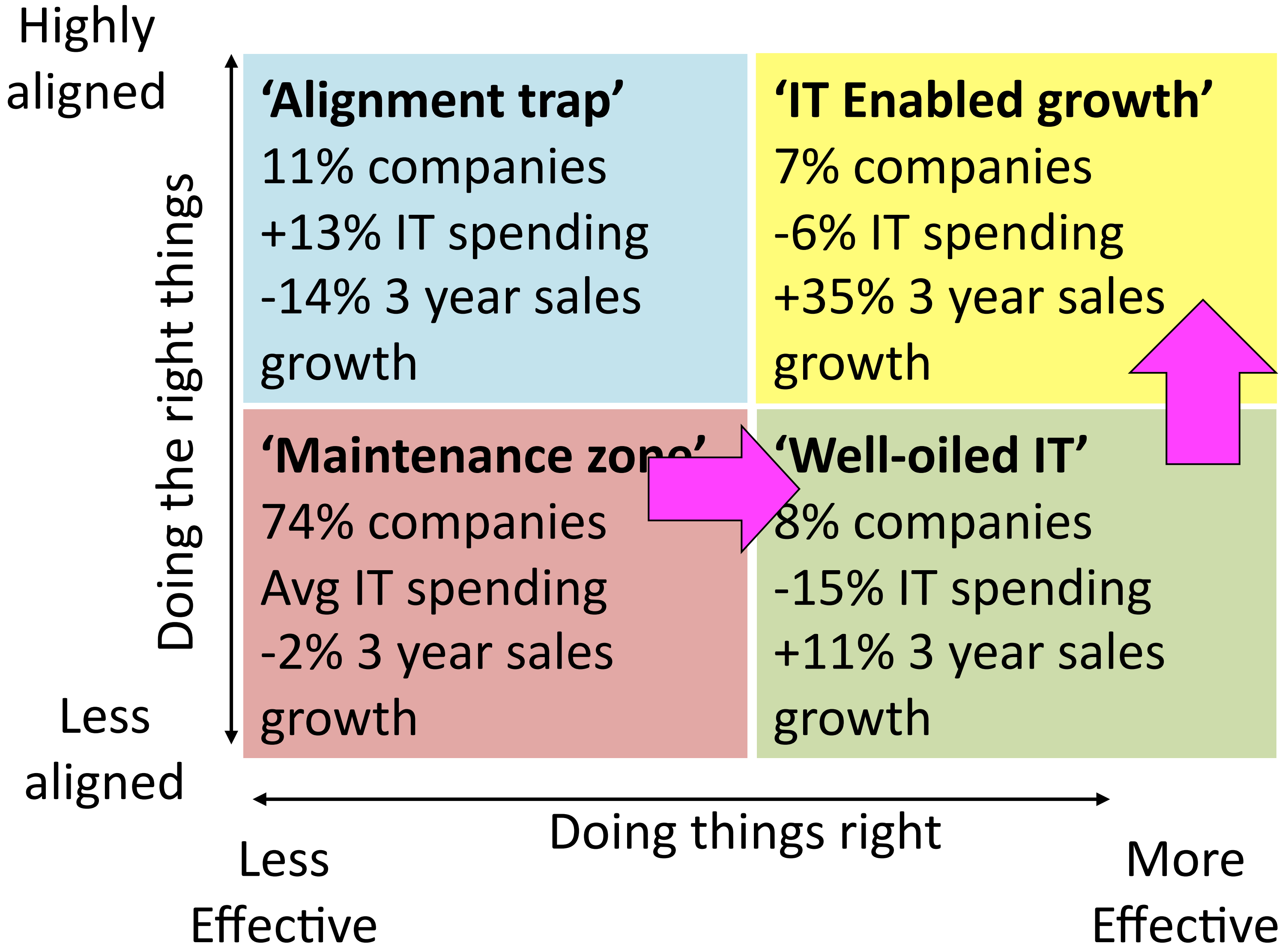
~~Product Manager~~

~~Business Analyst~~

~~Customer~~

Product Owner

The Alignment Trap



'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

'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

Return of the Product Owner (BA / Product Manager)

- Agile currently fixes *How*
  - Time to look again at the *What*
  - A job for the Product Owner
- User Stories are not enough
  - How do you know what to put on them?
- Features are not the only fruit
  - "Non functional" requirements
- Are you delivering business value?

> Only about 20% of features & functions in typical custom software are used

> We often encounter requirements churn of 30% to 50%

Mary & Tom Poppendieck

Implementing Lean Software Development 2007

> From the point of view of understanding "competitiveness", "levels of achievement" and "associated risk," the <u>performance requirements are by far the most interesting requirements.</u> Yet, traditionally, too much attention has been given to specification of functional requirements and resource requirements.

Tom Gilb
Competitive
Engineering,
2005

Moving to Agile

How do I get from here to there?

History

- Agile has been a bottom-up change initiative
  - Developers just started doing it
  - Little or no management instruction
  - Managers seen as a problem
- Top-down change initiatives have poor record
  - ISO-9000, CMM(I), BPR, ...
- Largely ignored business need (requirements)
  - Focus on development effectiveness

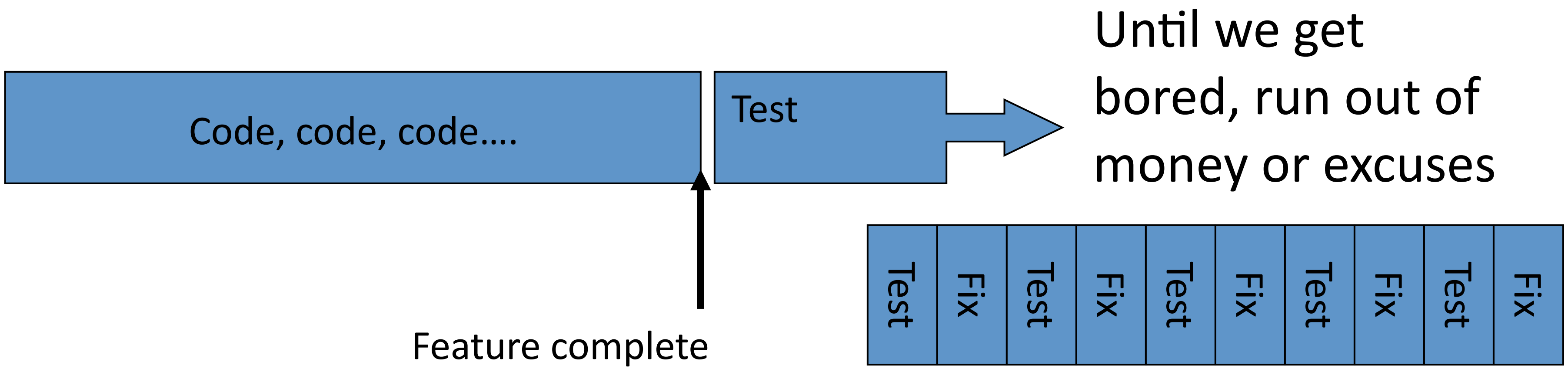
Quality underpins Agile

- Quality makes all thing possible

Quality is free

Philip Crosby, 1980

Traditionally Test at the end



Code, code, code….

Test

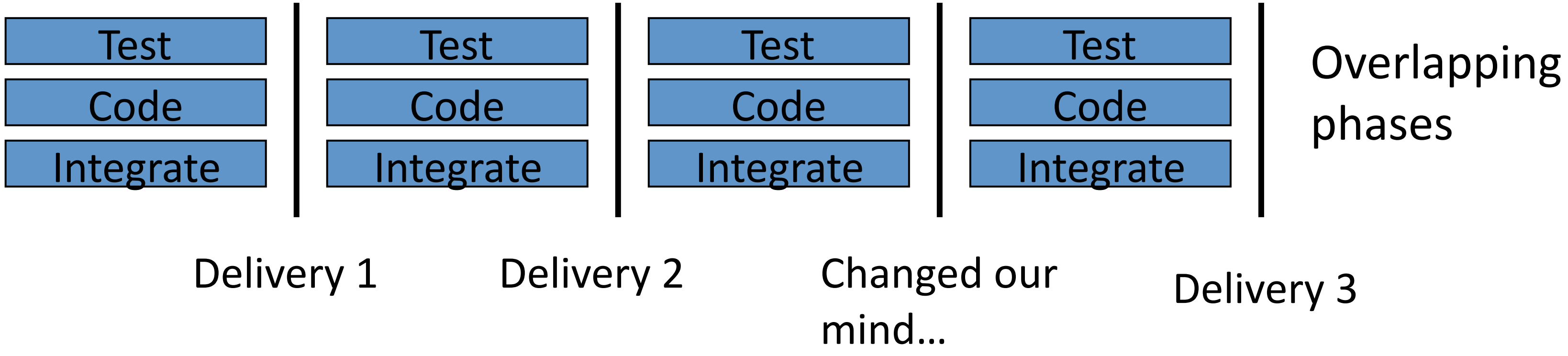
Until we get bored, run out of money or excuses

Feature complete

Test | Fix | Test | Fix | Test | Fix | Test | Fix | Test | Fix

- Depend on test because quality low
- Try inspect quality into the product
- Can’t release until we get through test
- More test -> more bugs found

Quality makes all things possible



Test

Code

Integrate

Test

Code

Integrate

Test

Code

Integrate

Test

Code

Integrate

Overlapping phases

Delivery 1

Delivery 2

Changed our mind…

Delivery 3

- Little and often
  - Deliver anytime
  - Business reaction?
- Requirements can change
- Predictable deliveries

Ceremonies are the easy bit

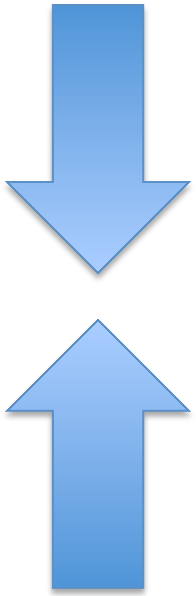
- Stand ups, planning, retrospectives
  - Effective ceremonies are more difficult
- Big up front requirements doc if you want
  - Don’t get hung up on scope, it changes anyway
- Don’t need coaching and training
  - But it’s a lot easier and less risky if you do
- Fixed length iterations are FIXED
- Use velocity from previous iterations to determine work in next (don’t target)

As a….

- **Developer**: Just do it
  - Get started on TDD
- As a **Project Manager**: work empirically
  - Use data to forecast: burn down/up charts and statistics
  - Stop projecting end dates until you have data
  - Shorten horizon: the next iteration; ask for working features
  - Encourage & support TDD
  - Involve customers in scope/time decisions

As a…

- **BA** get closer to your customers
  - Only the next feature is important
  - Specify acceptance criteria
- **Development Manager**
  - Move Heaven & Earth for TDD
  - Get people learning
  - Loosen procedures
  - Pincer movement
    - Stir up interest below
    - Provide support from above

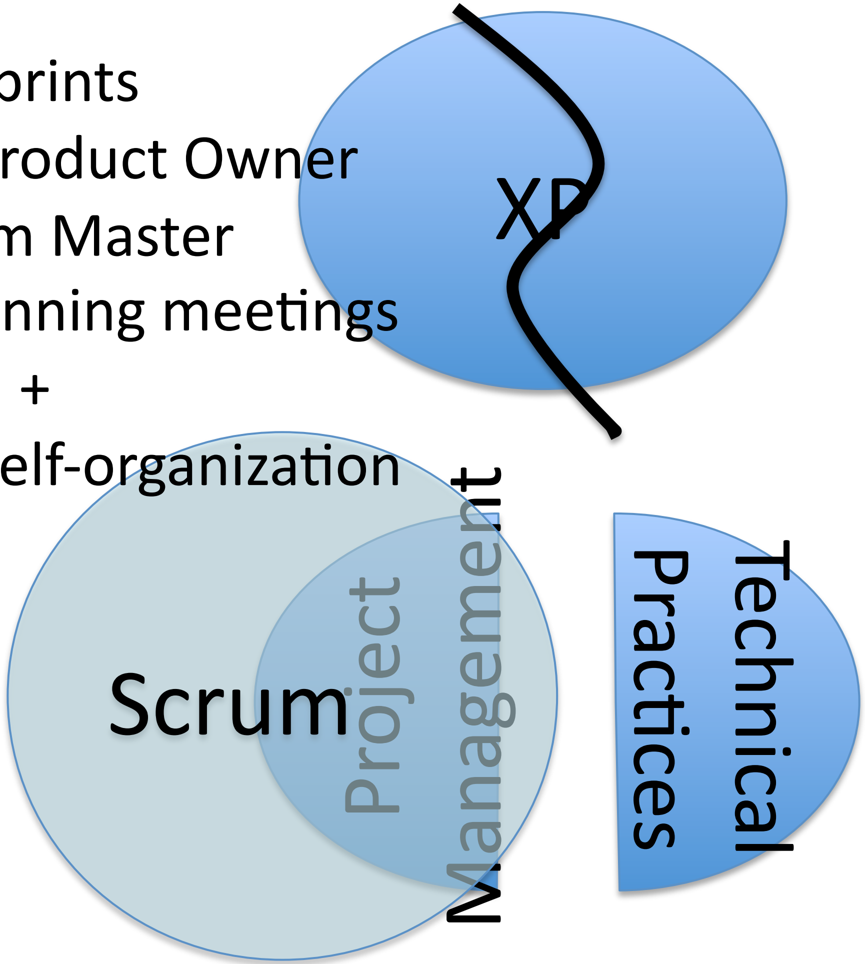


Whence Agile?

The history & future of Agile

Pre-Agile period	< 1995	Royce, Spiral model, Evo, Episodes Prototype Scrum, XP, etc.

Iterations -> Sprints
Customer -> Product Owner
Coach -> Scrum Master
Stand-ups, Planning meetings
+
Burn downs, self-organization



Successful Scrum teams use XP technical practices

Others hit Scrum-Wall

Scrum succeeded were XP couldn't
Credibility: Framework, Certifications, Management

Software Craftsmanship movement returns technical practices to focus

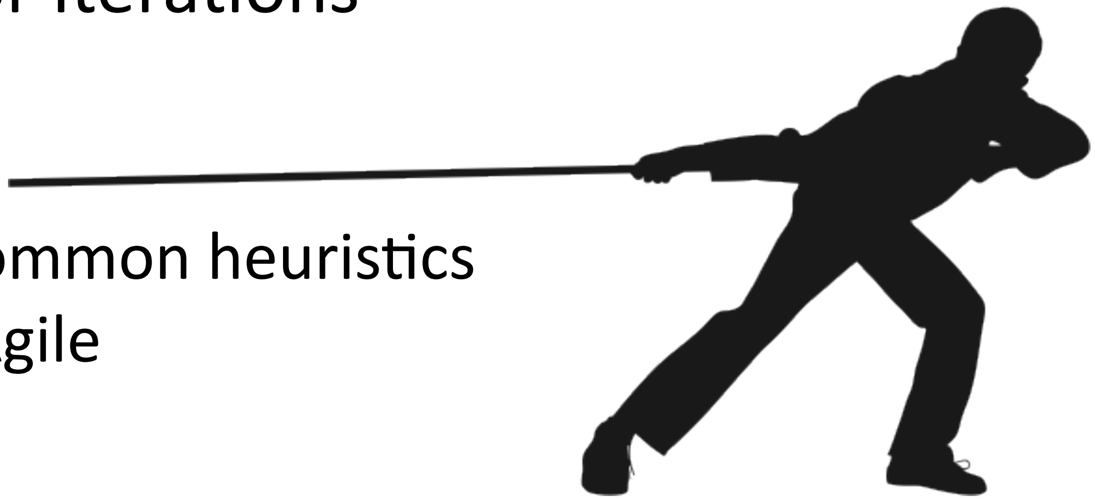
The Future is Lean

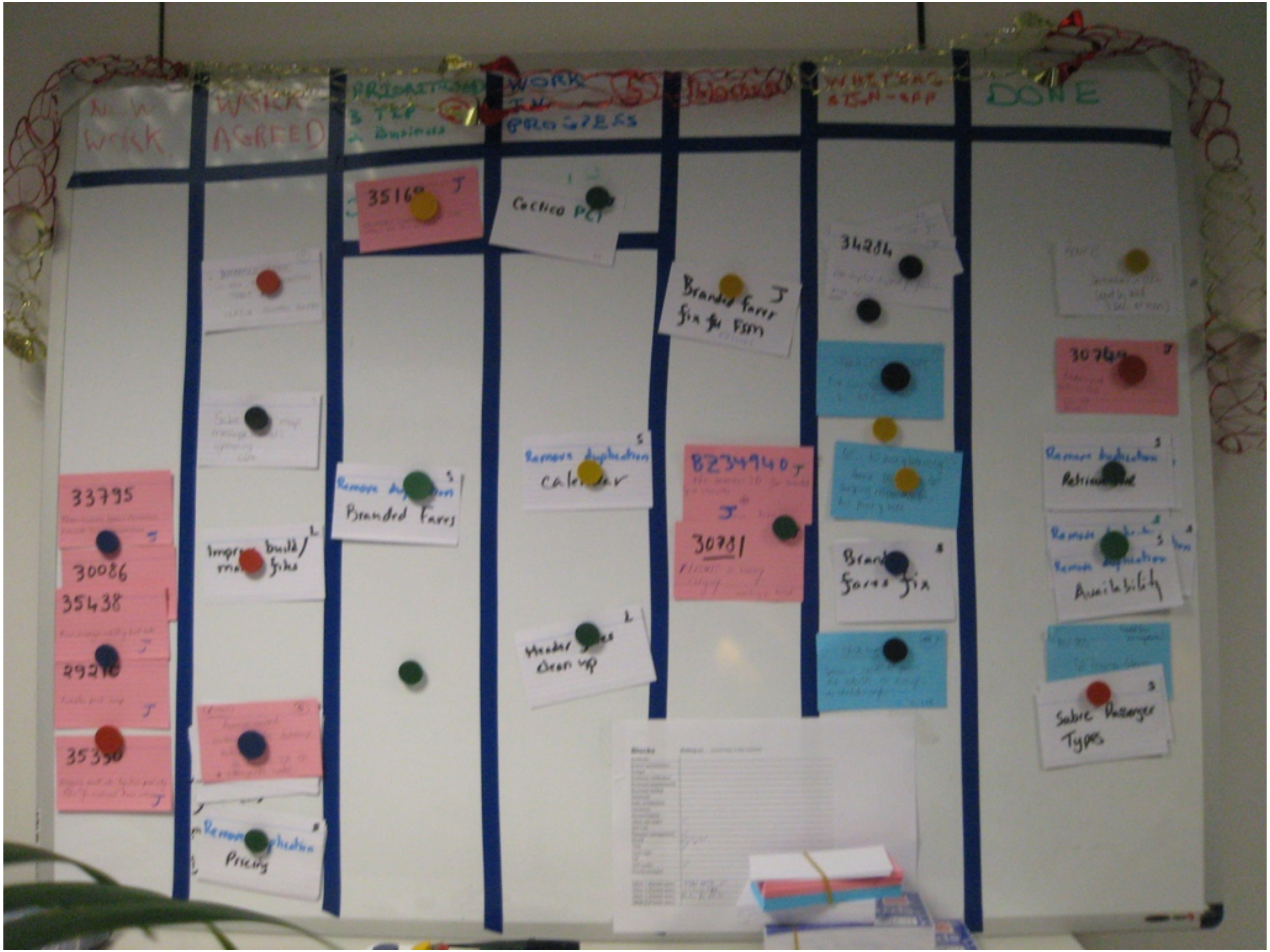
- Agile Software Development is Lean Product Development
- Lean provides tools to help you resolve problems more than ready made solution
  - Useful for deepening and refining practices

Kanban – new kid on the block

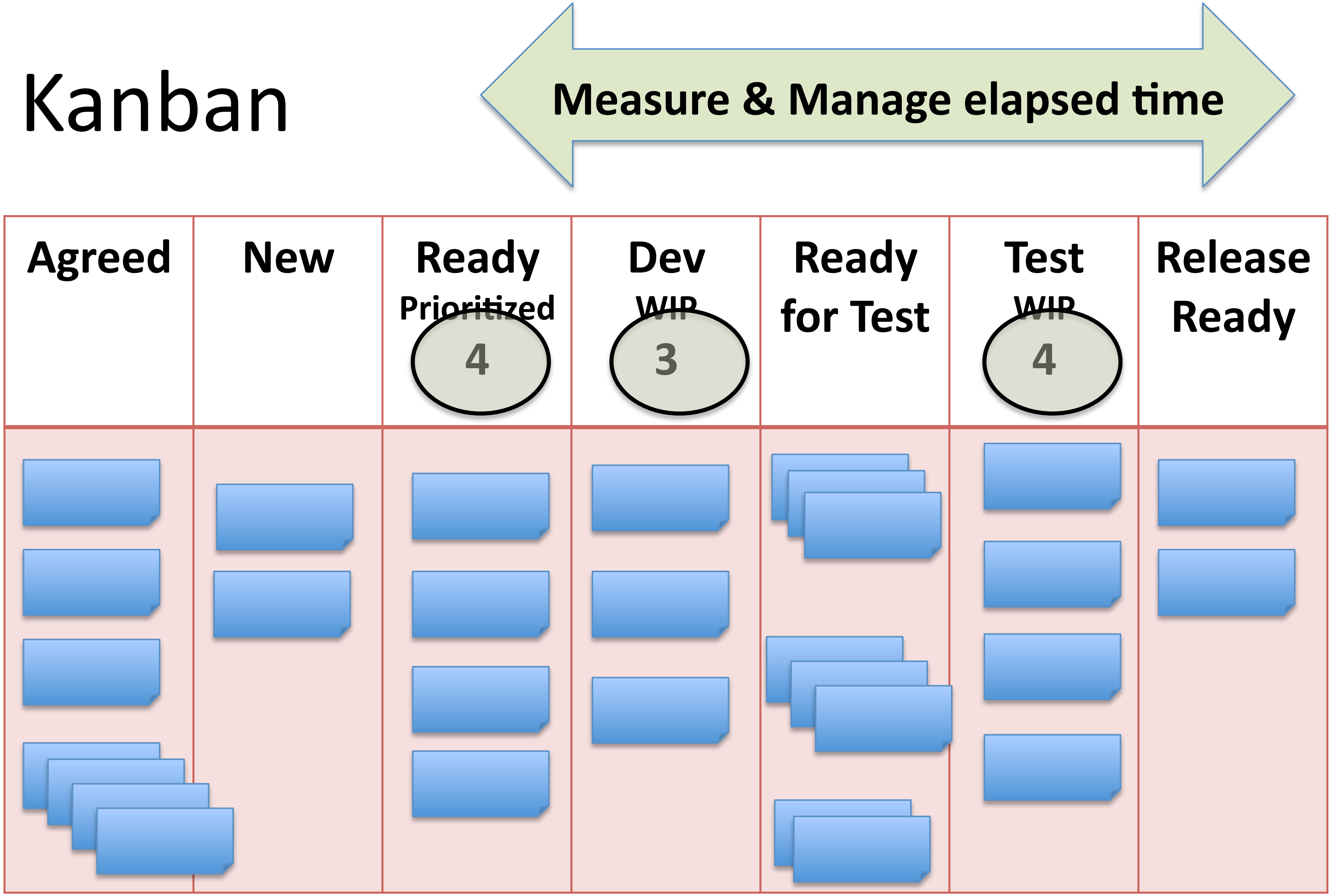
- The Kanban innovation
  - Explicitly limit Work in Progress
  - Break flow down into more stages
- Work is pulled from business
  - MMF - Minimally Marketable Feature
- No estimation or iterations
- Kanban is
  - Post-Agile
  - A collection of common heuristics
  - Dangerously unAgile

From David Anderson




NEW WORK
WORK AGREED
PRIORITISED
3 TFP
2 Business
WORK IN PROGRESS
BLOCKED
WAITING SIGN-OFF
DONE
33795
30086
35438
29210
35330
Improve build/ make files
Remove duplication
Pricing
35169
Remove duplication
Branded Fares
Coclico PCI
Remove duplication
Calendar
Header files clean up
Branded Fares fix for FSM
BZ349407
30781
34284
Branded Fares Fix
30769
Remove duplication
Retrieve
Remove duplication
Availability
Sabre Passenger Types

Kanban



Measure & Manage elapsed time

Agreed	New	Ready Prioritized 4	Dev WIP 3	Ready for Test	Test WIP 4	Release Ready

Estimation? Iteration?

“a subtle difference between kanban and typical agile processes such as Scrum. Scrum focuses on being agile which may (and should) lead to improving. Kanban focuses on improving, which may lead to being agile. However, being agile itself is not important - it just happens to be the best way we (or at least I) know at the moment.”

Karl Scotland, Early Kanban adopter, December 2008

Experience with Kanban

- Easier to introduce
- More responsive than Scrum/XP
- Models current process then you change
- Keep management by routine to start with
  - iterations
  - planning
  - work breakdown
  - estimates
- Greater need to emphasis learning & change
  - No safety net of

Summary

- Agile adoption continues
- Adoptees deepen practices
- Next for Agile
  - End of Scrum hegemony
  - The Future is Lean
  - Rise of Kanban
  - Return of “requirements”
  - Focus on business value

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