# Agile & Lean for Systems & Software

DU Radio MT
2009-12-15

## Thomas Nilsson

Agile Developer, Coach & Mentor

Responsive
Development Technologies

# Agile & Lean for Systems & Software

## Is Agile for You?

Responsive Development Technologies

# Bad News about Agile

- It is not a silver bullet
  - There is no silver bullet!
- It is not easy
  - Requires change of culture, values, process and techniques
  - Requires more discipline and involvement
- It is not a process that can be introduced and be done
  - Continuous improvement
  - Continuous change
  - Continuous dialogue
- It shows you problems
  - It will show you early
  - It hurts!

Responsive
Development Technologies

# Good News about Agile

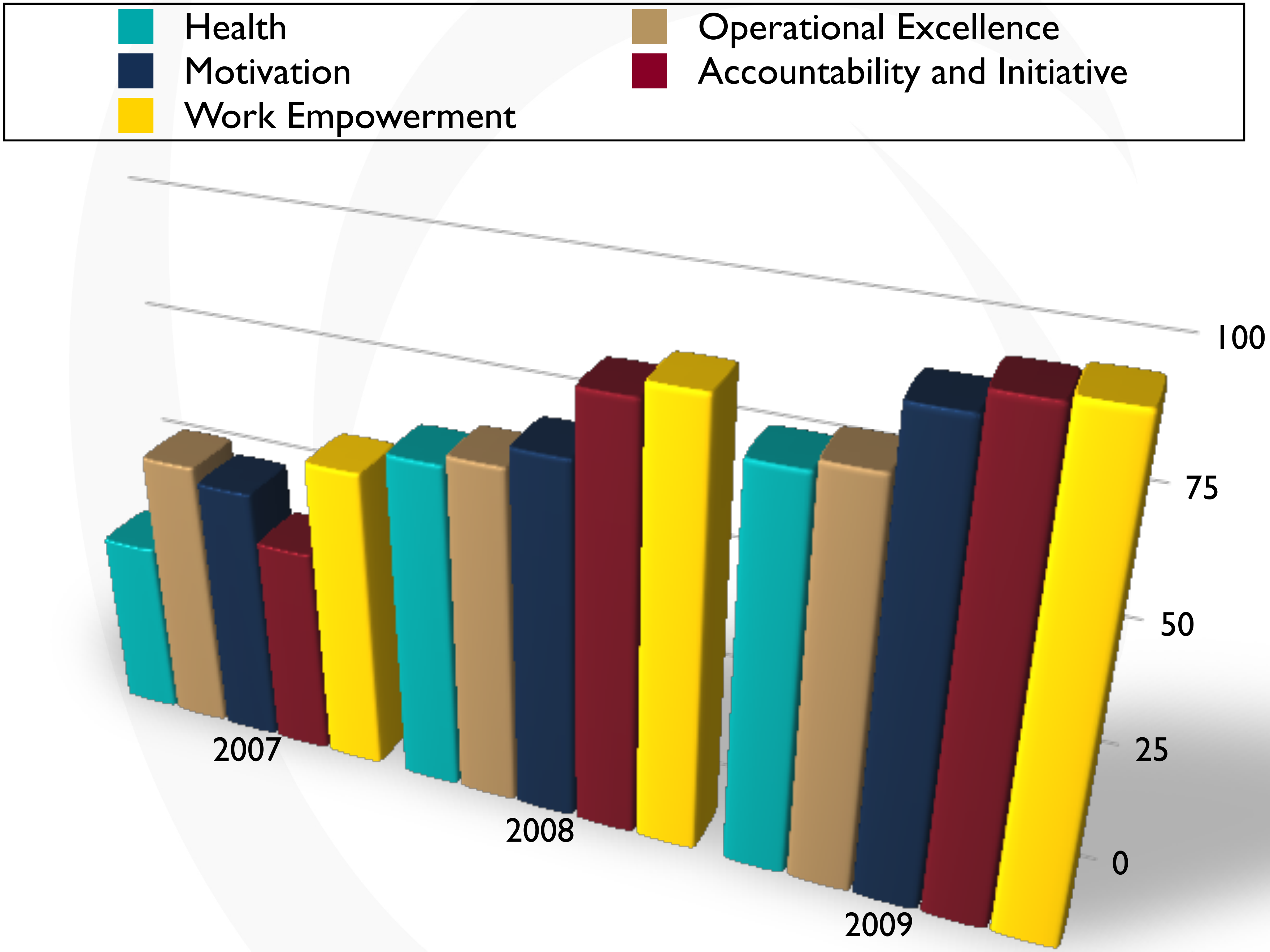
- Agile works even in large organisations
  - Small organisations work better with any process
- Agile always works and gives substantial effects
  - Higher quality
  - Higher job satisfaction
  - Higher throughput
- Agile can be applied everywhere in an organisation
  - Not only software development
  - It is not a process, it is a set of values
- Agile does not *require* implementation
  - You can always become a little more agile
  - Dig where you stand!
  - Low hanging fruits!

Responsive
Development Technologies

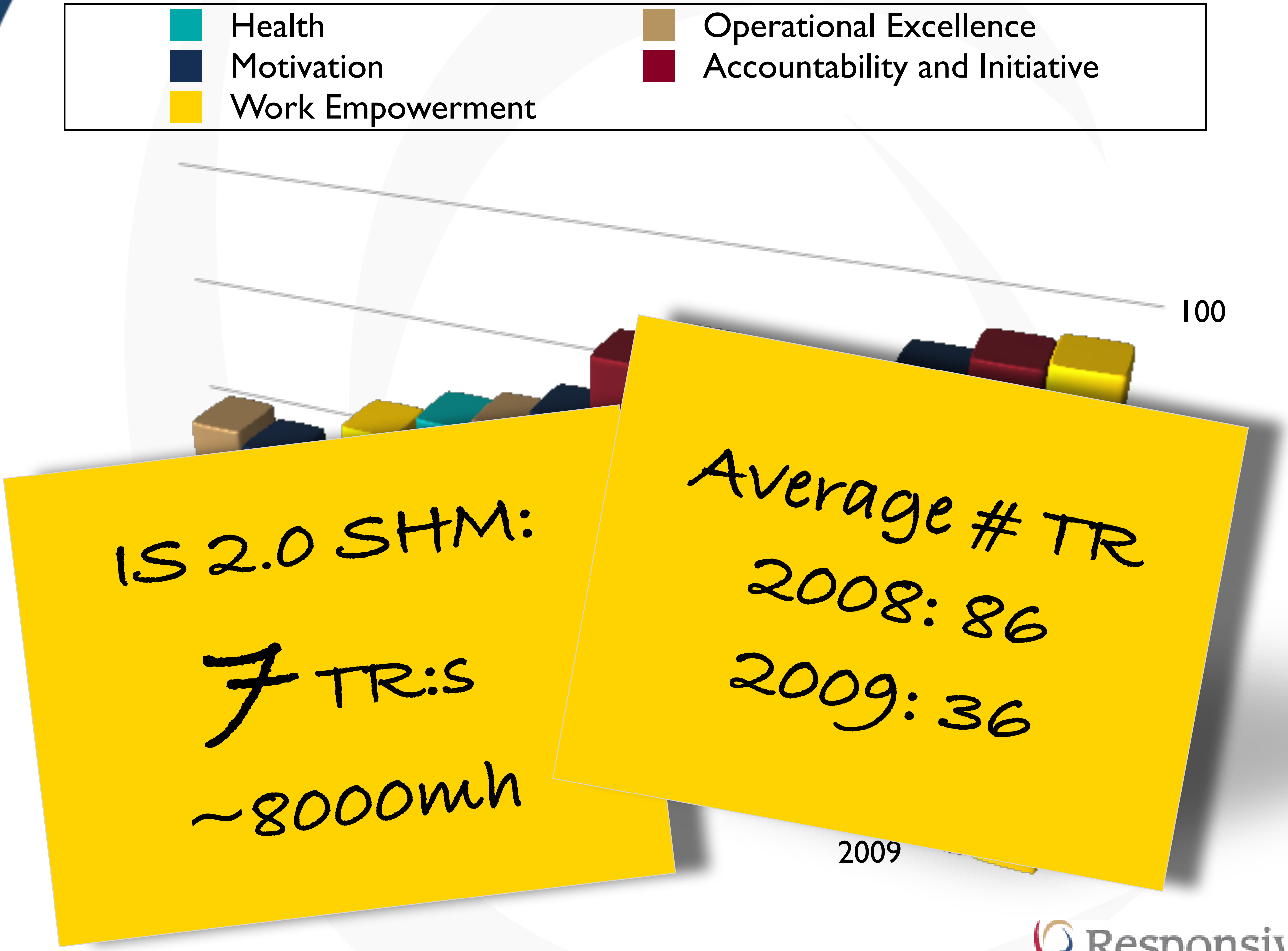
# Is Agile for You?

- Job Satisfaction and Engagement in Work
  - higher quality
  - improvements
- Development Work is Problem Solving
  - adaptive
  - non-tailoristic
- Quality, Leadtime & Efficiency
  - comes from within
  - by the people doing the work
- Leadership
  - removing obstacles
  - enabling growth in people and organisation
- People
  - are intelligent
  - can solve problems
  - want to do a good job

Responsive
Development Technologies



Responsive
Development Technologies



IS 2.0 SHM:

7 TR:s

~8000mh

Average # TR

2008: 86

2009: 36

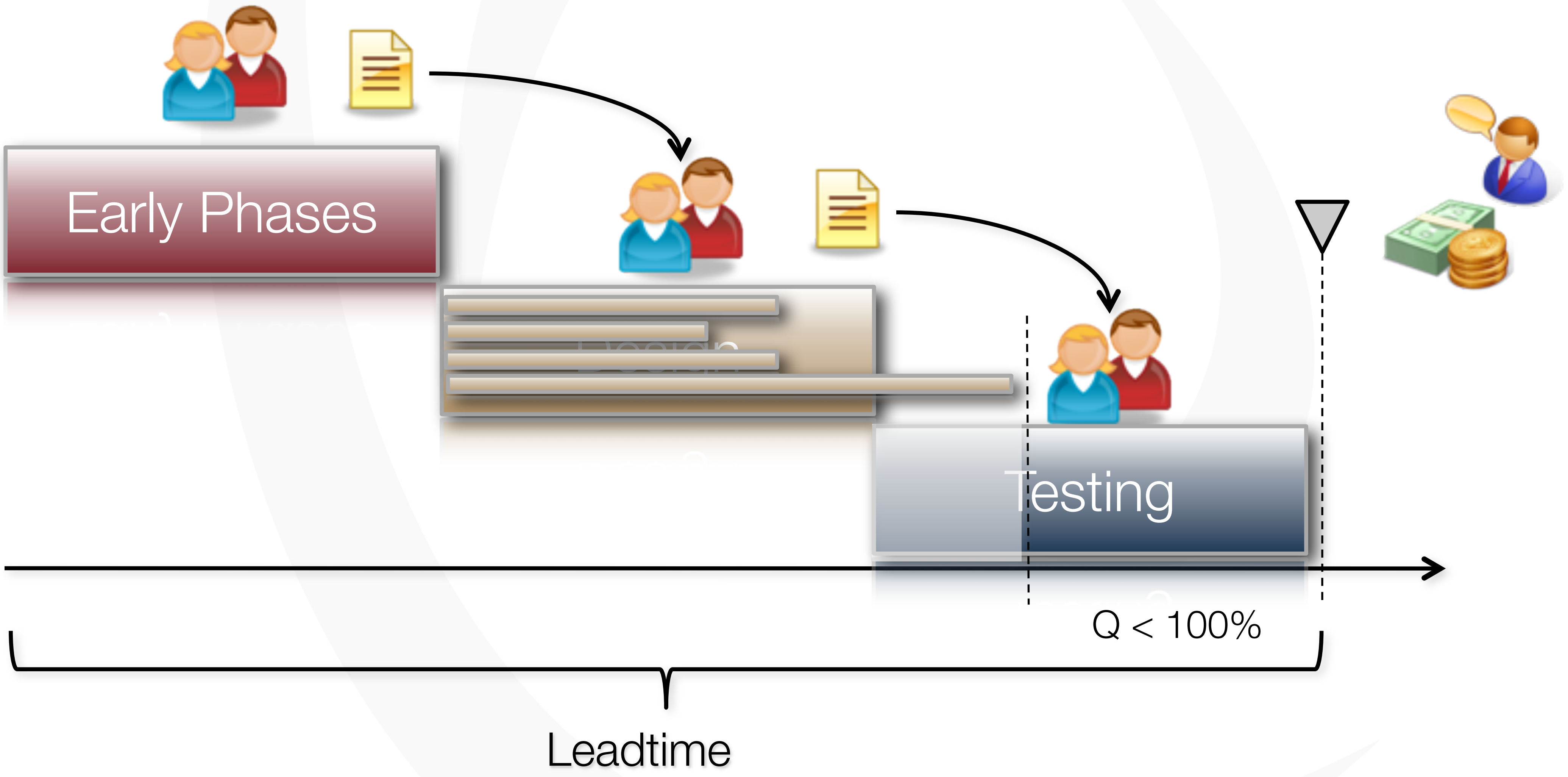
# Agile & Lean
# for Systems & Software

## Why Agile?

Responsive
Development Technologies

# Drivers for Development Cost

- Wrong Functionality
  - Including bugs, misfeatures, misunderstandings and extra non-prioritised, functions
  - Large part of developed software is never used!
- Length of Feedback Loops
  - How long does it take before we know if we are on the wrong track or did the wrong thing?
  - Fail early!
- Batches & Queues
  - Are everywhere in a prescriptive, document driven organisation
  - Example: Requirements Specs
- Repetitive Manual Labor
  - Activities that can be automated should be automated
  - Requiring them to be performed more often drives automation
  - Example: Testing
- Technical Debt
  - Hard to understand, change, improve, maintain
  - Neglecting to care for your product(s) and development environment



Early Phases

Design

Testing

Q < 100%

Leadtime

# Development of Software

- Development - into the unknown
  - is problem solving
  - inherently unpredictable
  - always new
- Software - it’s soft
  - it’s design, not manufacturing
  - possibility to make small changes, often
  - does not require manufacturing
  - software can be used to build software
- Possible to apply adaptive methods
  - Adaptive methods should be used for anything unpredictable
  - Agile would have been called Adaptive

Continuous Integration

Pair Programming

SetBasedDesign

XP

REFACTORING

# Agile & Lean for Systems & Software

## What's Agile?

DSDM

Iteration Planning

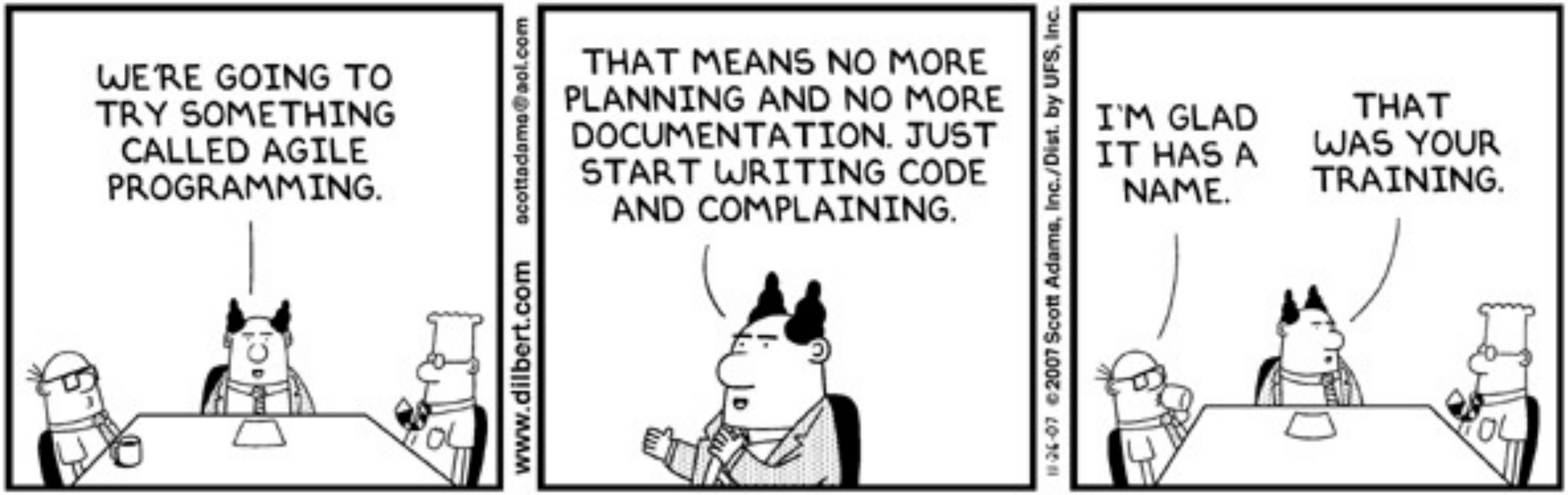
Test Driven Development

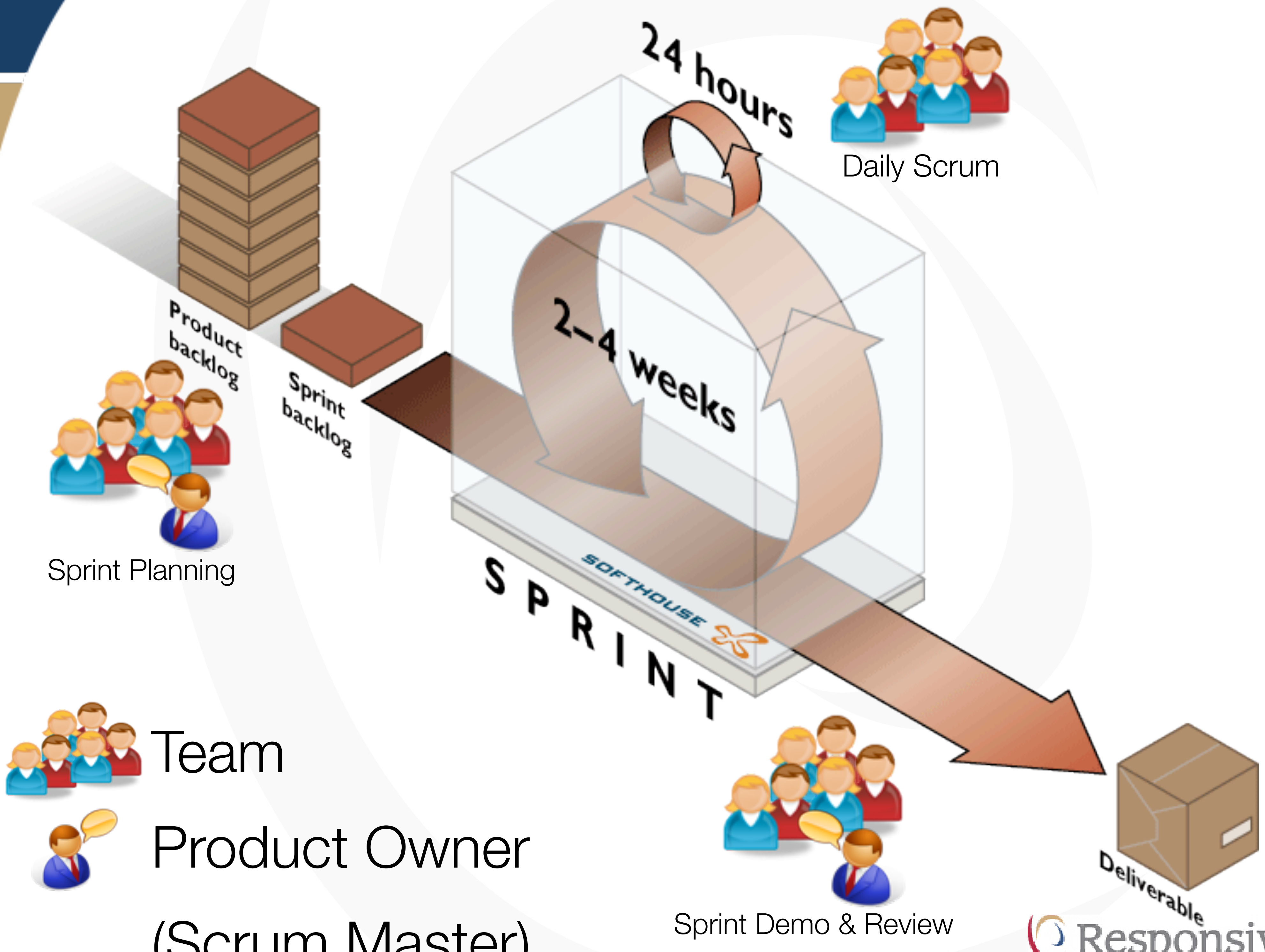
Product Owner

Scrum

Sprint

Responsive
Development Technologies


WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING.
scottadams@aol.com
www.dilbert.com
THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING.
11-26-07 ©2007 Scott Adams, Inc./Dist. by UFS, Inc.
I'M GLAD IT HAS A NAME.
THAT WAS YOUR TRAINING.



Team

Product Owner

(Scrum Master)

Responsive
Development Technologies

# 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 | | comprehensive documentation |
| Customer collaboration | | contract negotiation |
| Responding to change | | 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

# "Continuously increasing the amount and frequency of successfully delivered business value."

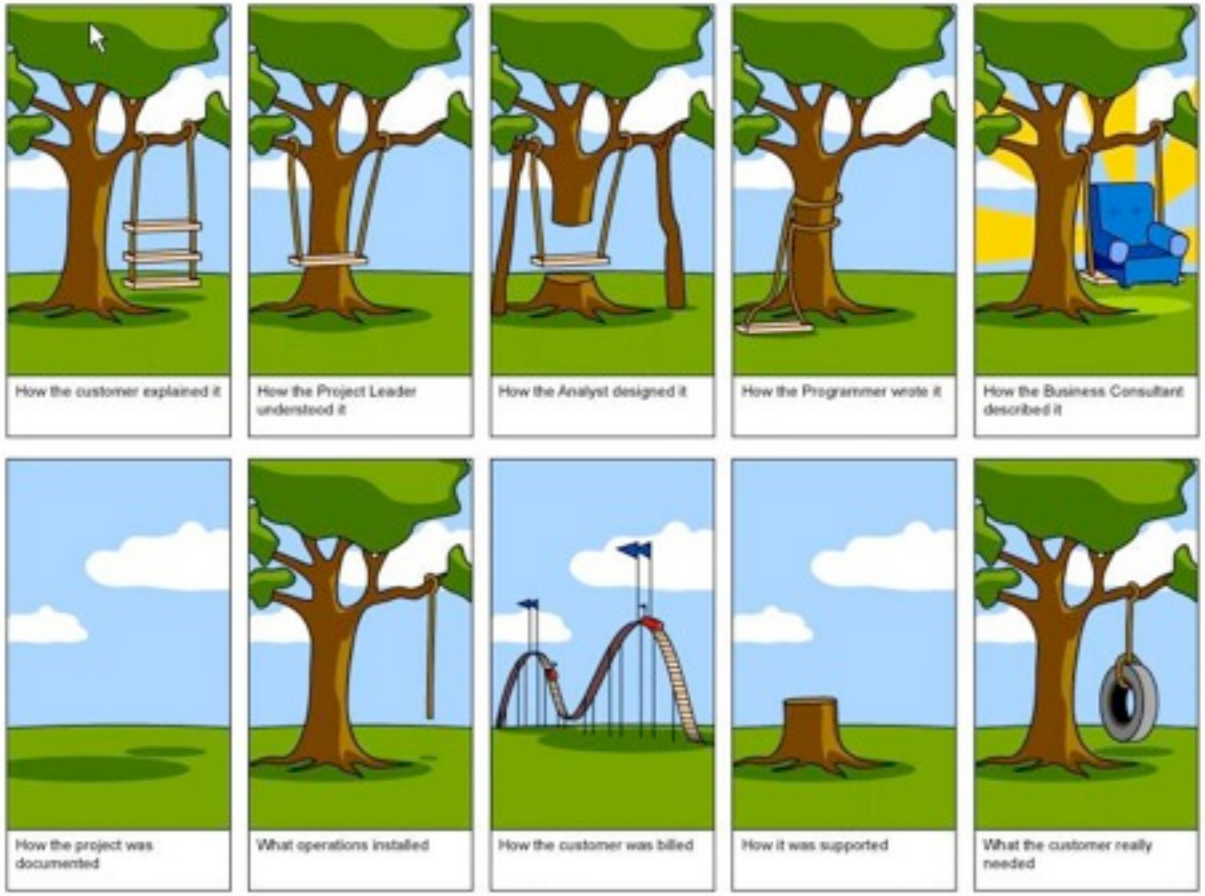
- ”Continuously increasing ...”
  - Continuous Improvement
  - Processes and WoW need to change all the time
  - Quality and Measurements cannot be based on processes
- ”... the amount and frequency ...”
  - Increasing flow and finding more valuable things
  - Cycle Times are important, think shortest possible lead time
- ”... successfully delivered ...”
  - Only actually delivered matters!
- ”... business value.”
  - Not software, not applications, but *Solutions to some stakeholders actual Need!*

# Agile & Lean
# for Systems & Software

## Agile Remedies

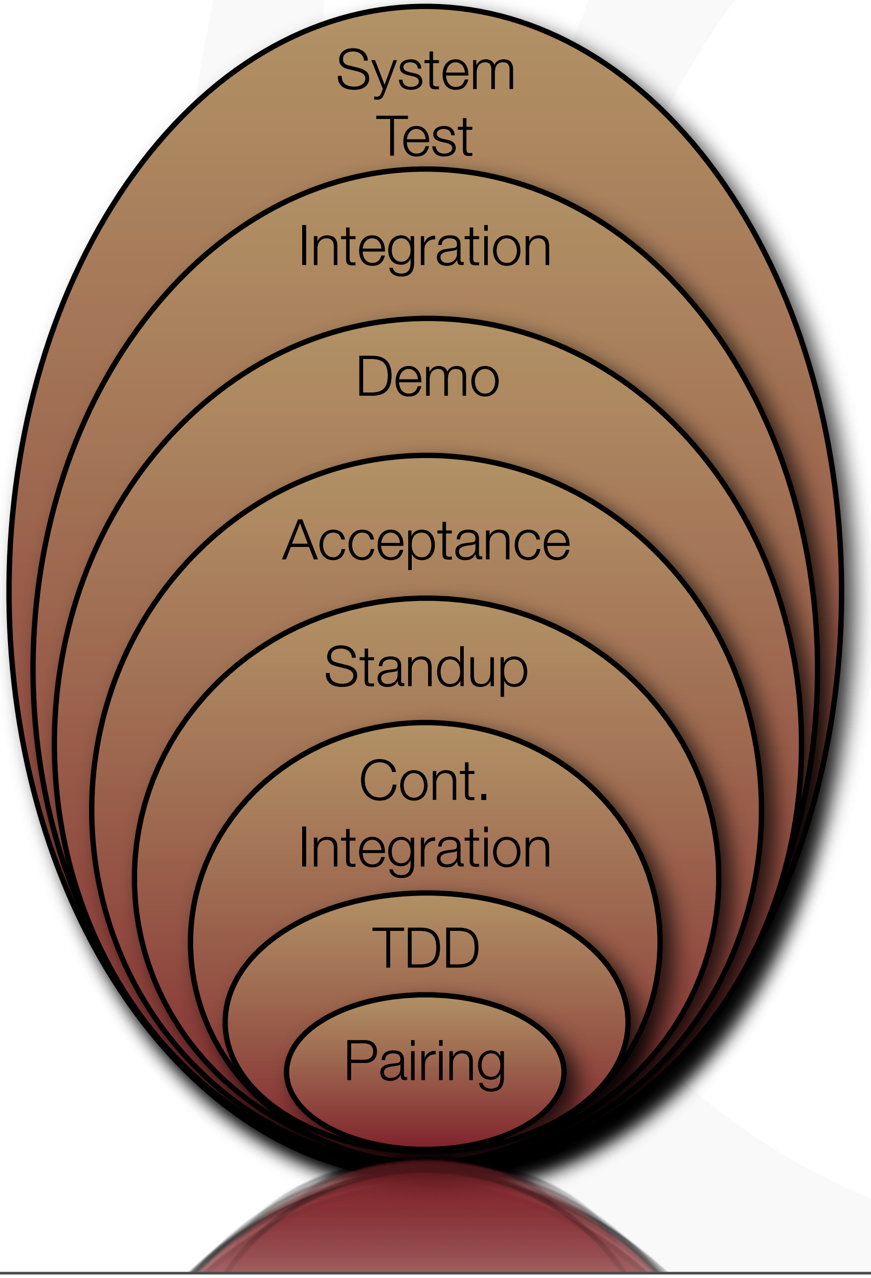
Responsive
Development Technologies

# Wrong Functionality


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

- Why?
  - IKIWISI
  - Changing environment
  - New realisations
  - New possibilities
- Agile
  - ”Customer on site”
  - Direct customer collaboration and communication
  - Demos
  - Acceptance Tests
  - Unit Tests
  - Extensive Regression testing

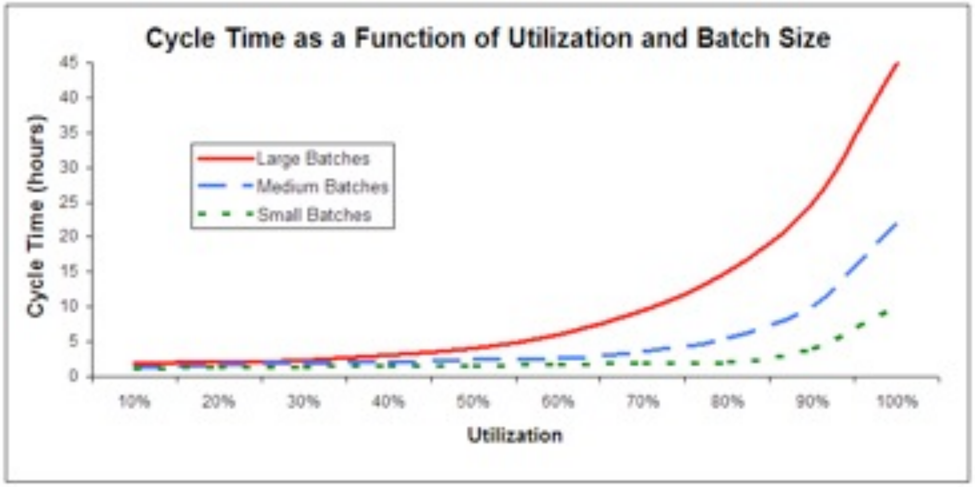
# Feedback Loops



- unit testing
- automated functional tests
- product owner ("customer") on site
- stories
- teams
- pair work
- test driven

Responsive Development Technologies

# Batch & Queue



- Short iterations
  - 2-3 weeks
- One functional change at the time
  - Concrete, easy to manage, easy to test
  - Regression testing becomes important
- Less Specification Driven
  - Specifications are batches
  - Formal Approval forces batching
- Everything should be included
  - Testing, cleaning up, planning, quality, etc. is always done
- Consider flow rather than cost
  - Lean thinking
  - Queuing Theory

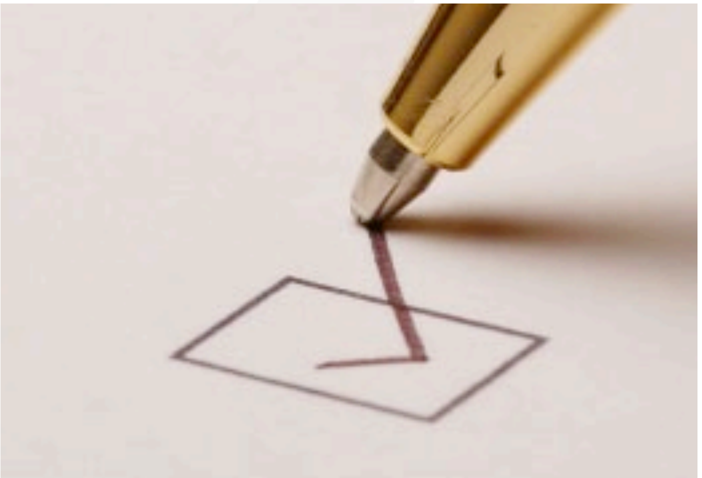
# Repetitive Manual Labour

- Manual work is
  - costly
  - error-prone
- Automation!
  - Software can be used to automate!
- What hurts? Do it more often!
  - Building
  - Integration
  - Regression Testing
  - ...
- Until it
  - can be used to Fail Early!
  - costs almost nothing!

Responsive
Development Technologies

# Technical Debt

- Cleaning up never happens!
- ”Everything is included”
  - Test Driven
  - Automation
  - Refactoring
  - Test- & build scripts
  - Manuals  & Documentation
  - ...
- For every Story!
- Culture and Way of Working
- You always have the time!
- And You are always Done!



Responsive Development Technologies

# Agile & Lean for Systems & Software

## Opposing Forces

Responsive
Development Technologies

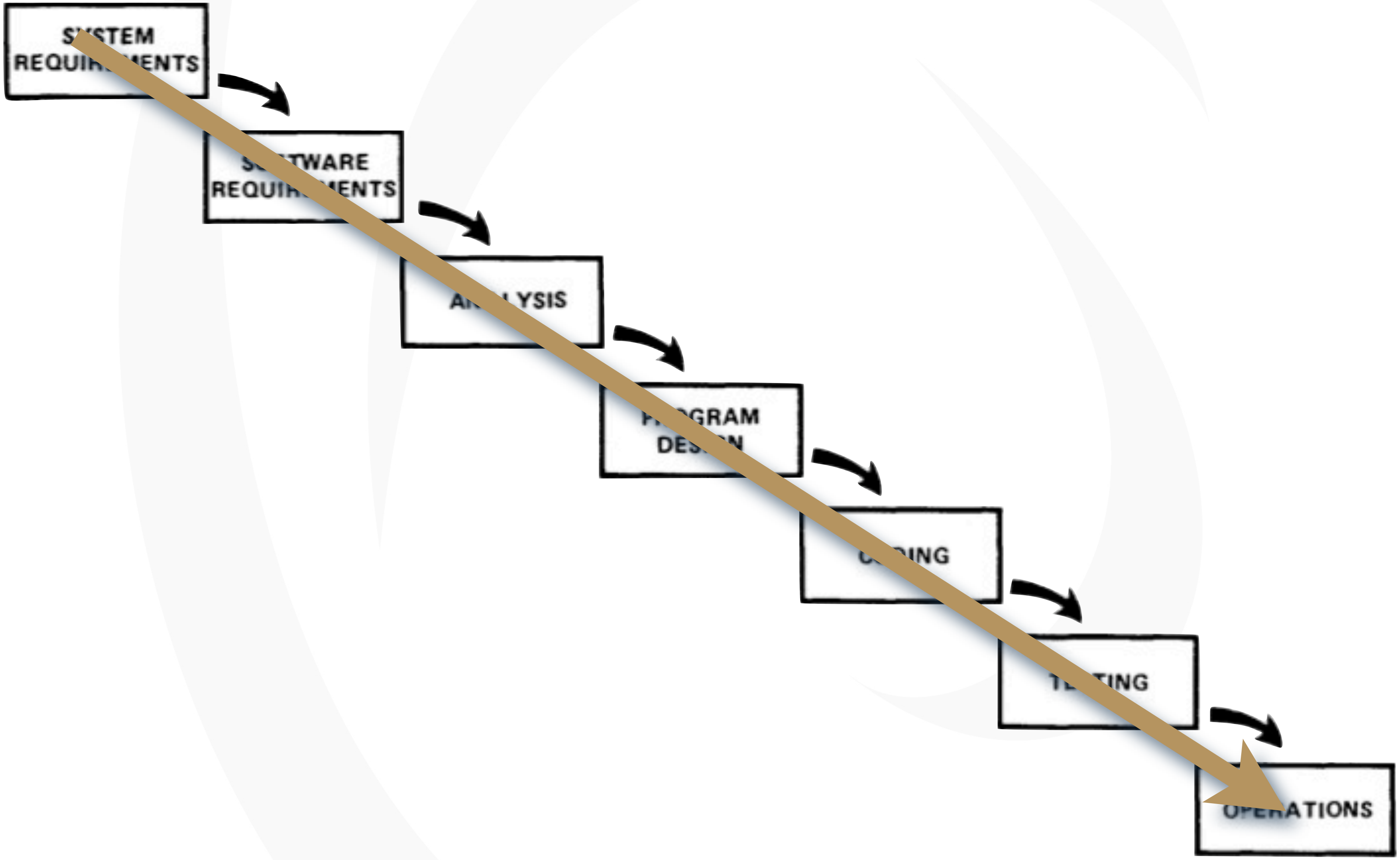


Figure 2. Implementation steps to develop a large computer program for delivery to a customer

# Scientific Management

- Full Resource Utilisation
- Specialisation
- Functional Organisation
- Cost Reduction
- Cheap Un-Educated Labor
- Simple Tasks

# Non-Tayloristic

- People are not resources
  - Factor 10
- Economics of scale
  - Does not apply!
  - Single ”production”
  - New design every time
- Thinking hard and long doesn’t help!
- The devil is in the details
  - Pareto principle, 80% effect with 20% of effort
  - Empirical and statistical
- Prioritisation
- Portfolio Management

| Sep-09 | Oct-09 | Nov-09 | Dec-09 | Jan-10 | Feb-10 | Mar-10 | Apr-10 | May-10 | Jun-10 |
|---|---|---|---|---|---|---|---|---|---|
| SMX | | MSC 14.1 extended IS 2.0 support | | O11 features | | | | | |
| Job Handling | BSS licensing | | | | O11 features | | | | |
| Minor features (CRs, Q-rank etc.) | | | MSC 14.1 extended IS 2.0 support | | O11 Minor features(CRs, Q-rank etc.) | | | | |

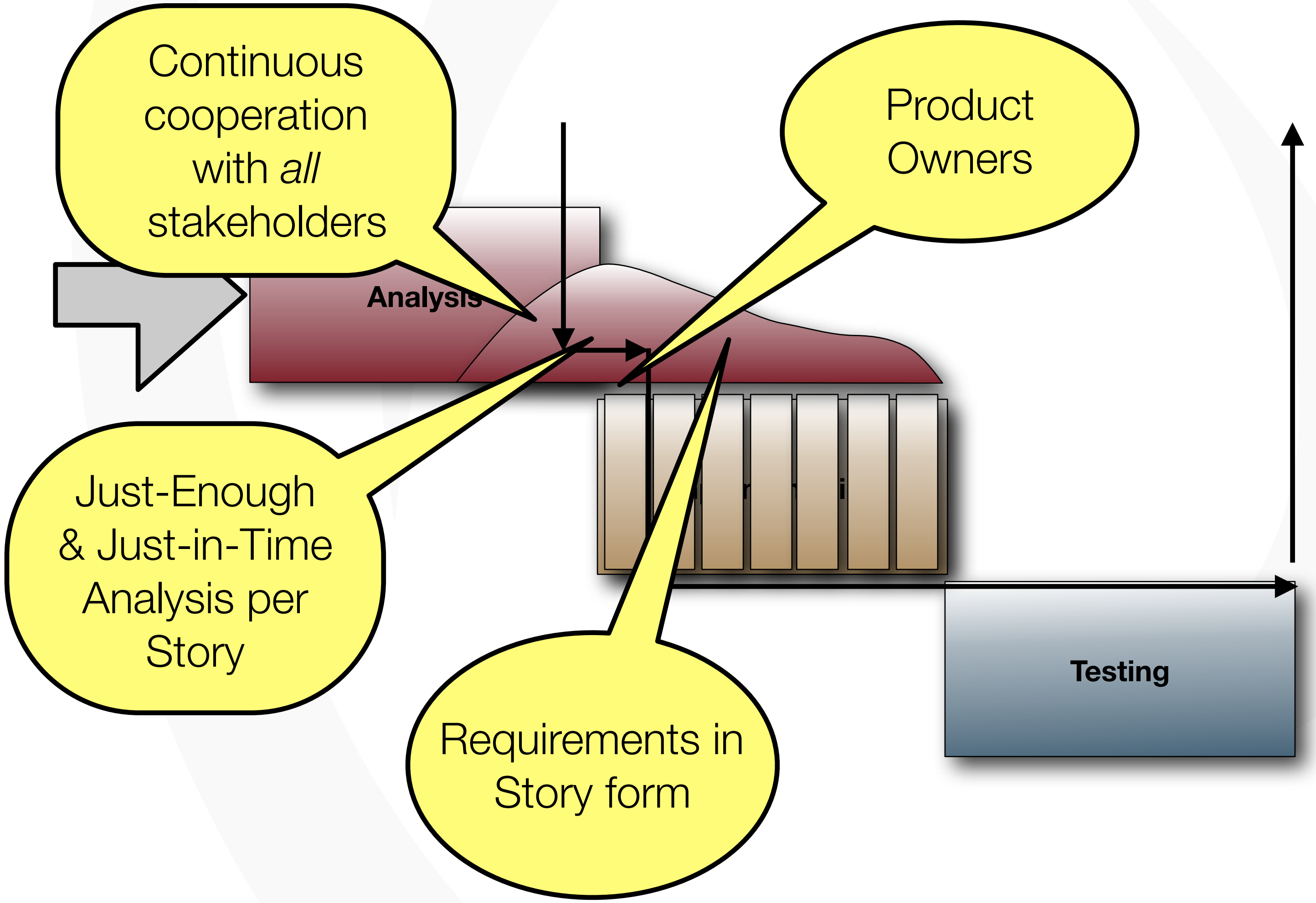
Responsive
Development Technologies

# Agile & Lean for Systems & Software

## A New Model

Responsive
Development Technologies

# Agile Implementation & Analysis



Continuous cooperation with *all* stakeholders

Product Owners

Just-Enough & Just-in-Time Analysis per Story

Requirements in Story form

# The Product Owner Responsibility

User

Customer

CEO

Business Analysts

SPM

Test Coordinator

Project Management

Systems

TSPOC

OPM

Sub-PM

Architect

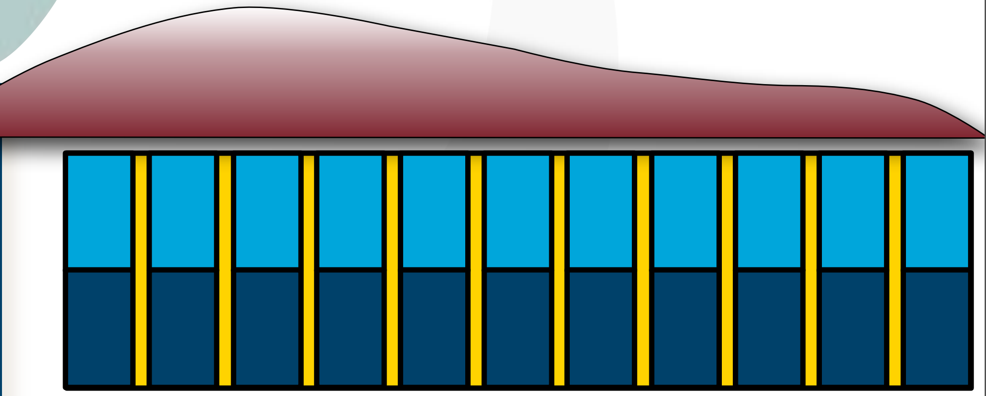
IP-writer

IP-writer

| | | |
|---|---|---|
| 1 day | Tue 10/9/01 | Tue 10/9/01 |
| 25 days? | Mon 10/1/01 | Thu 10/25/01 |
| **28 days?** | **Thu 9/20/01** | **Wed 10/17/01** |
| 7 days | Thu 9/20/01 | Wed 9/26/01 |
| 9 days | Wed 9/26/01 | Thu 10/4/01 |
| **20 days?** | **Fri 9/28/01** | **Wed 10/17/01** |
| 8 days | Fri 9/28/01 | Fri 10/5/01 |
| 8 days | Fri 9/28/01 | Fri 10/5/01 |
| 3 days | Sat 10/6/01 | Mon 10/8/01 |

10/9
Whole Group
Whole Group
Whole Group
Carla Pienknagura,Patrick Willoughby
Hrishikesh Deo,Aaron Dollar
Patrick Willoughby

Story 1
Story 2
Story 3
....
Story 11
Story 12
Story 13
....
Story 21
Story 22
Story 23
....



# The Product Owner Responsibility

User

Customer

CEO

SPM

Pr

Business Analysts

Systems

Architect

IP-writer

Focus on What & Why, not How!

Story

Story 2

Story 3

....

Story 11

Story 12

Story 13

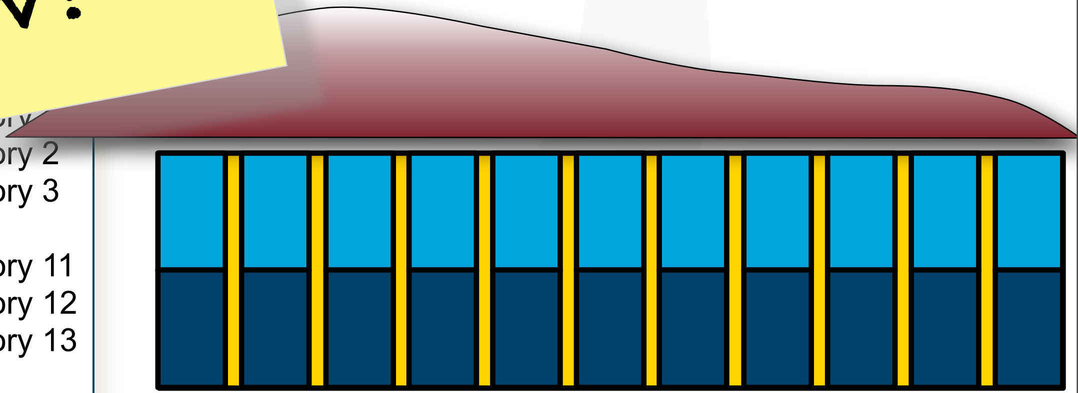
....

Story 21

Story 22

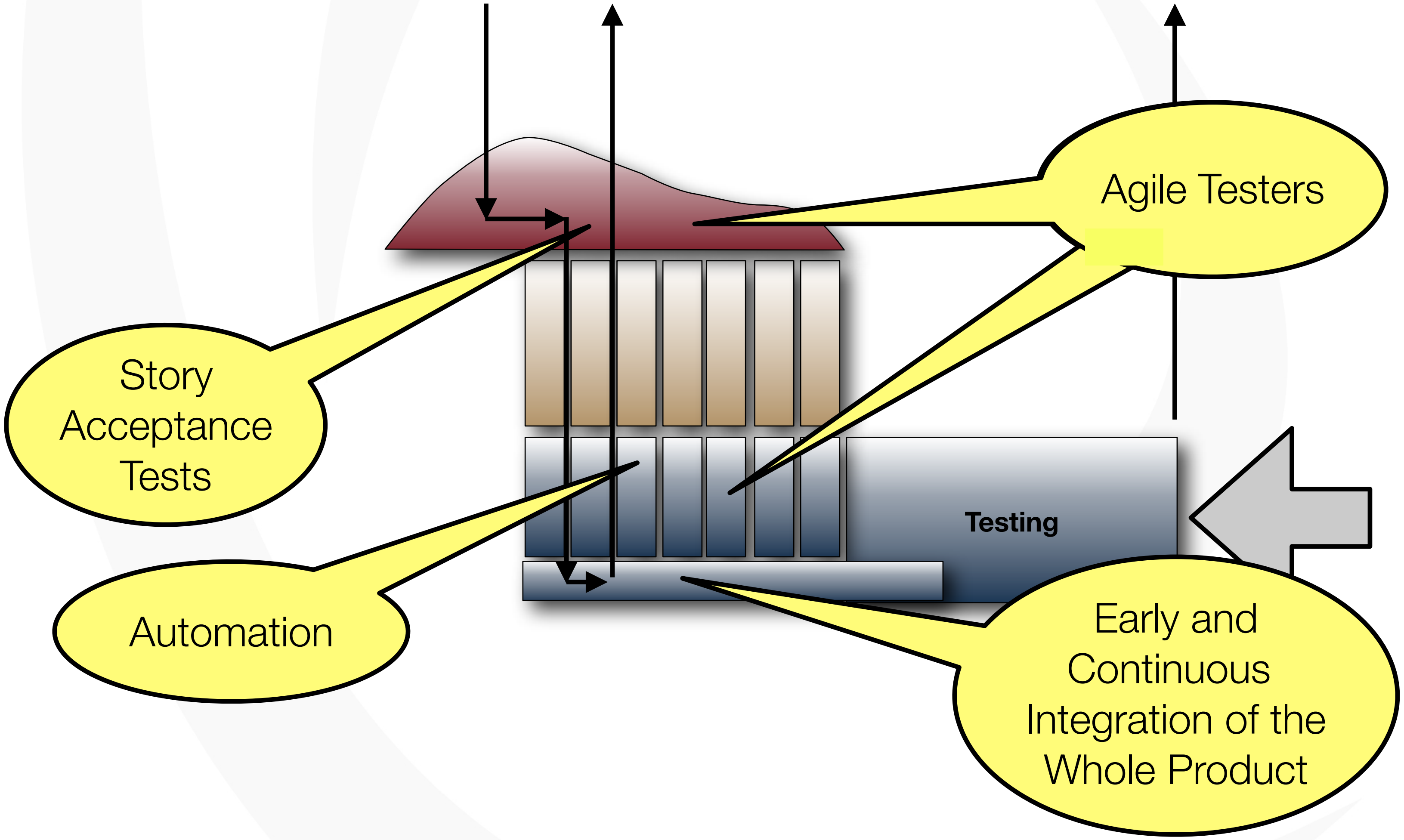
Story 23

....



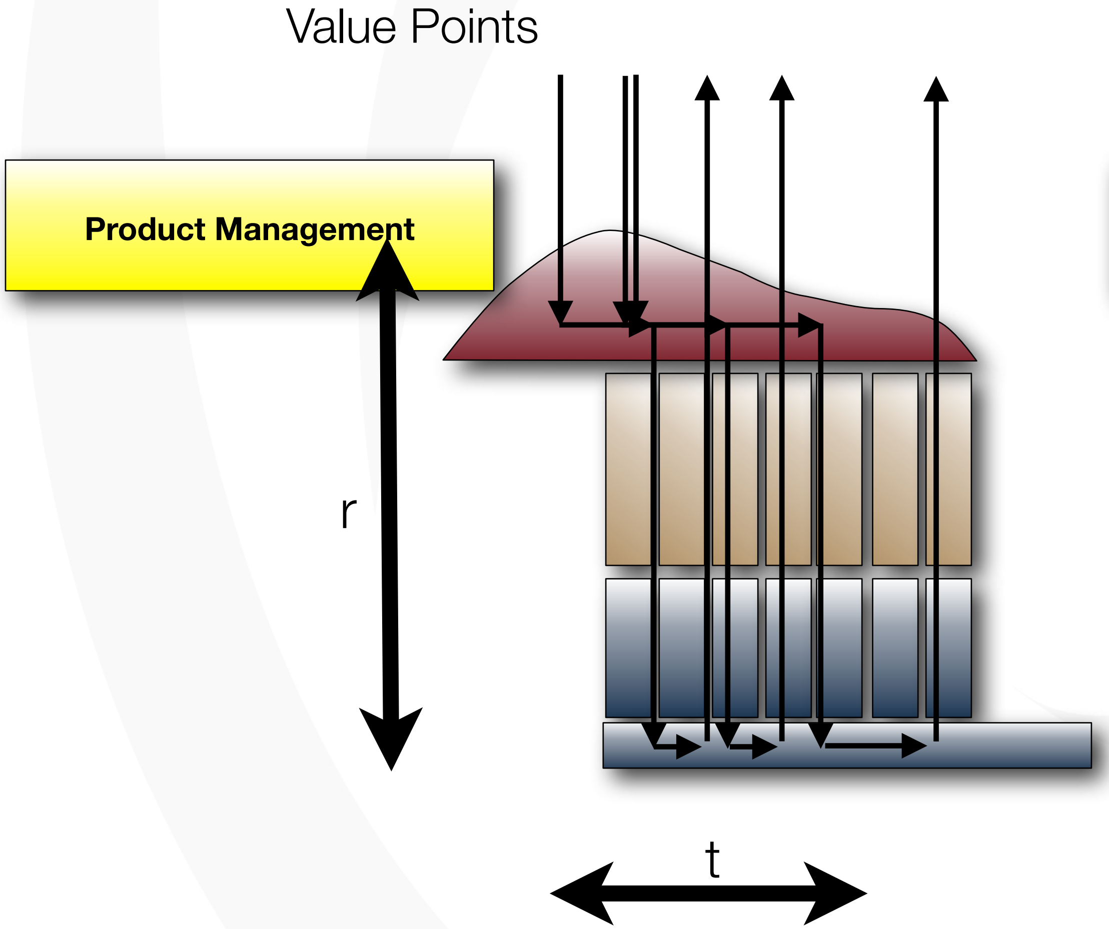
Responsive
Development Technologies

# Agile Implementation, Analysis & Testing

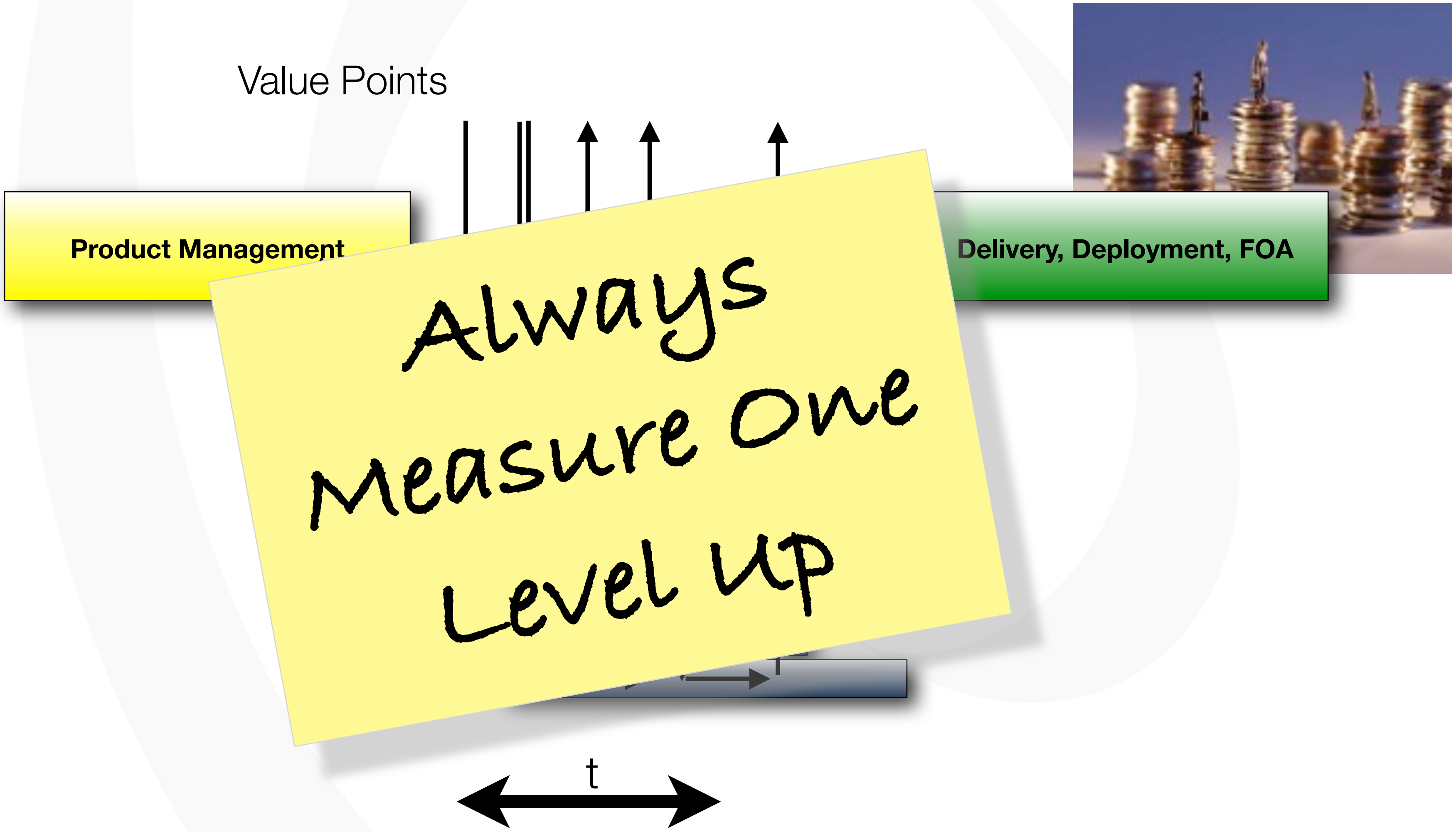


Responsive Development Technologies

# Efficiency: Measure Flow and Lead Time of Value



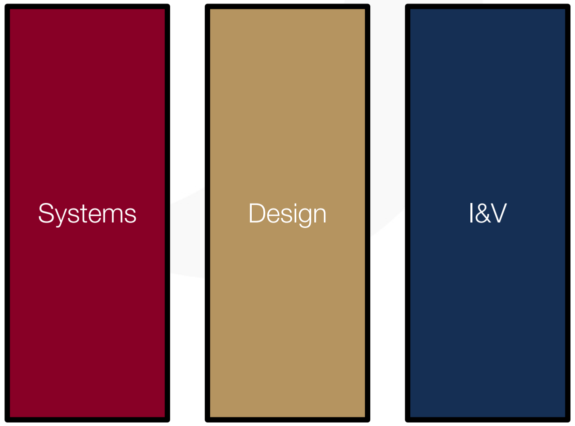
# Efficiency: Measure Flow and Lead Time of Value



Always Measure One Level Up

# Organisation

- Traditional
  - Competence or Process step Stovepipes/Silos
  - ”Over the wall” batching
  - Command and Report/Follow-up
  - Push
  - Measurements on process steps



Responsive Development Technologies

# Organisation

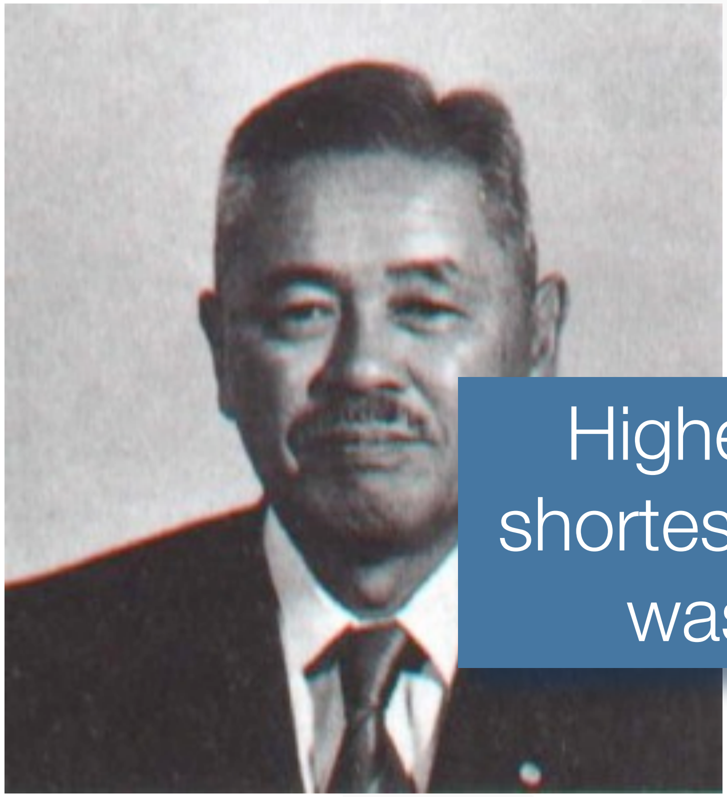
- Traditional
  - Competence or Process step Stovepipes/Silos
  - ”Over the wall” batching
  - Command and Report/Follow-up
  - Push
  - Measurements on process steps
- Modern
  - Product Focus
  - Team-centric
  - Pull
  - Increased Communication
  - Collaboration of Interests
  - Measurements on Flow
  - Balancing resources

# Agile & Lean
# for Systems & Software

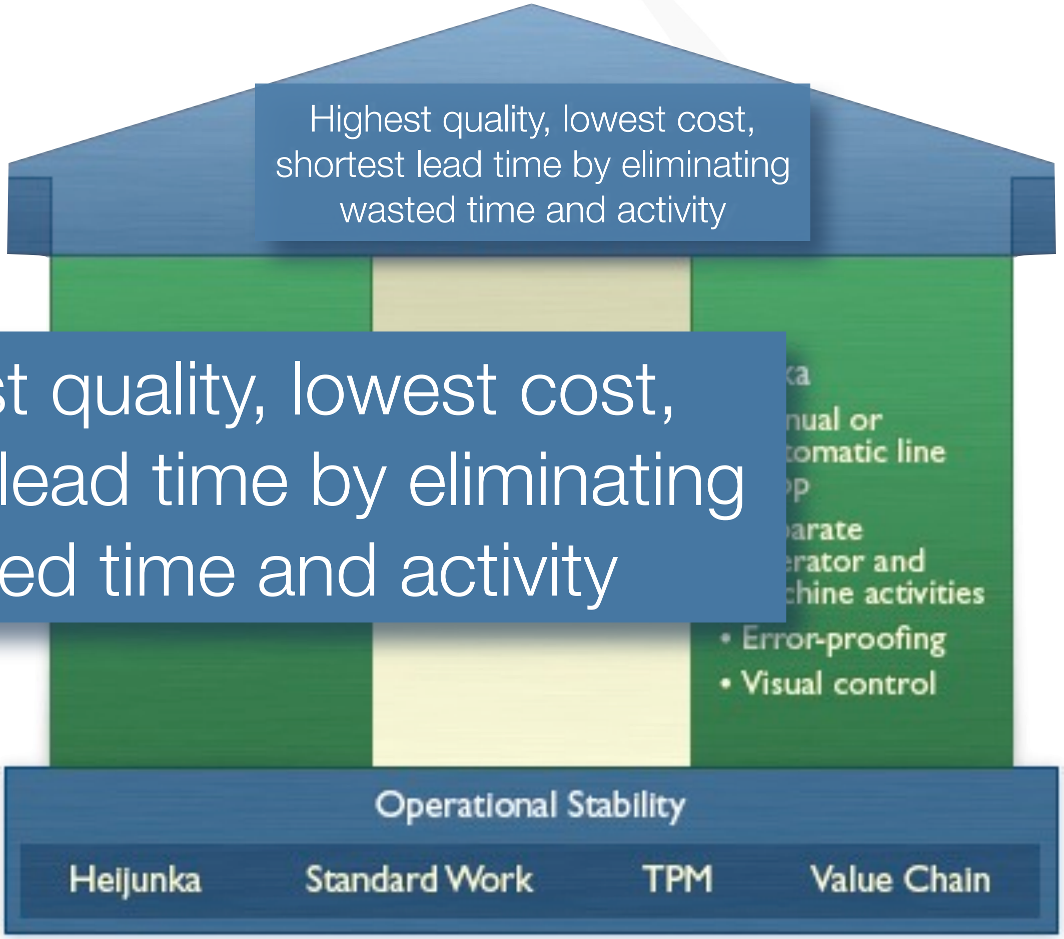
## Becoming Agile

Responsive
Development Technologies

# Toyota Production System



Highest quality, lowest cost, shortest lead time by eliminating wasted time and activity



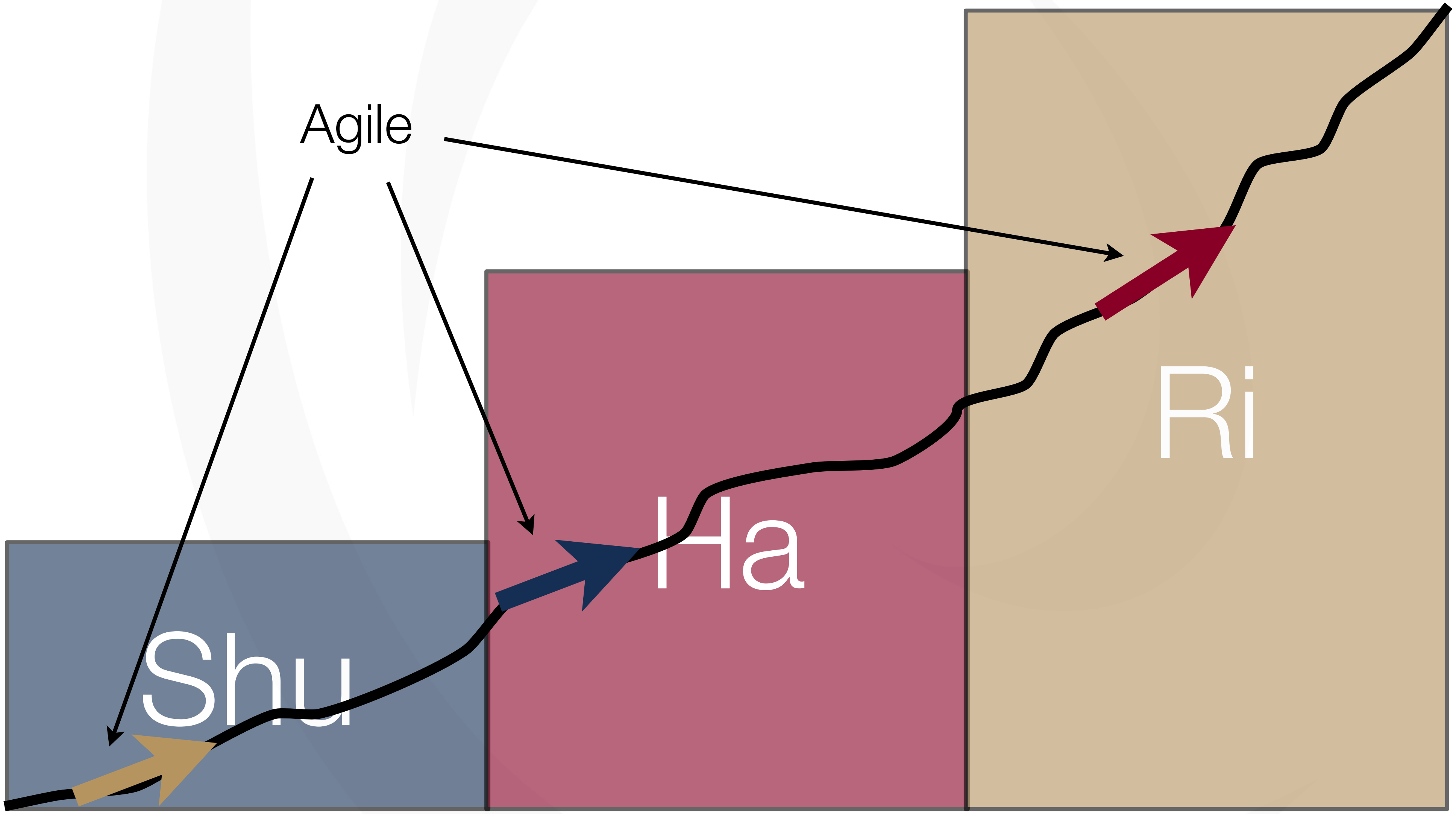
Highest quality, lowest cost, shortest lead time by eliminating wasted time and activity

ka
nual or
tomatic line
PP
parate
erator and
chine activities
- Error-proofing
- Visual control

Operational Stability

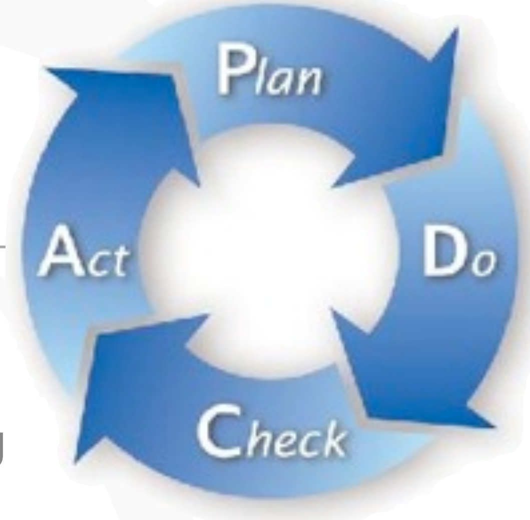
Heijunka Standard Work TPM Value Chain

# Change = Learning = Improvement = Agile!

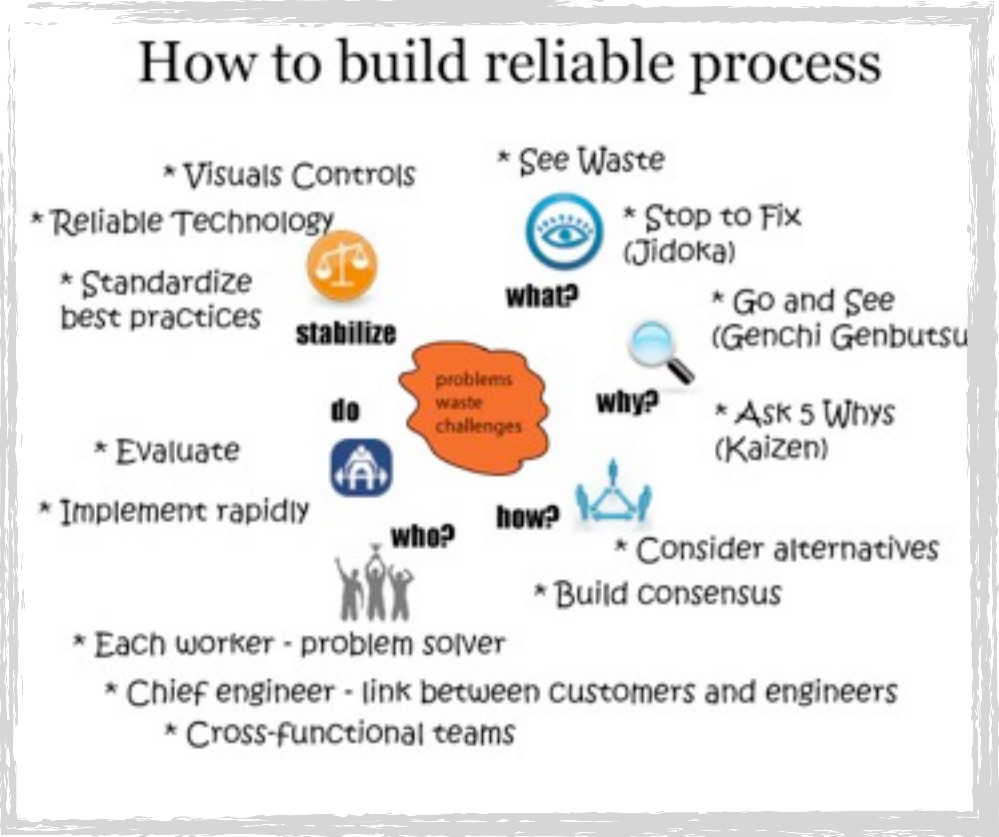


Responsive Development Technologies

# Change is a Continuous Process



- Old Style
  - Let Experts Develop and Document a new Way of Working
  - Pilot
  - Implement by pushing
- New Style
  - Let the People Working (in any capacity) reflect on their Way of Working
  - Find Things the Team can do:
    - implement immediately
    - repeat
  - Find Bigger Things
    - propagate up one level in the line
    - prioritise and implement
  - Frameworks and Visions



# Agile & Lean
# for Systems & Software

## A little bit more Agile today!

Thomas Nilsson
thomas.nilsson@responsive.se
http://www.responsive.se/thomas

Responsive
Development Technologies