Organizational Design: Lean, Agile, Scrum, LeSS, & Scaling

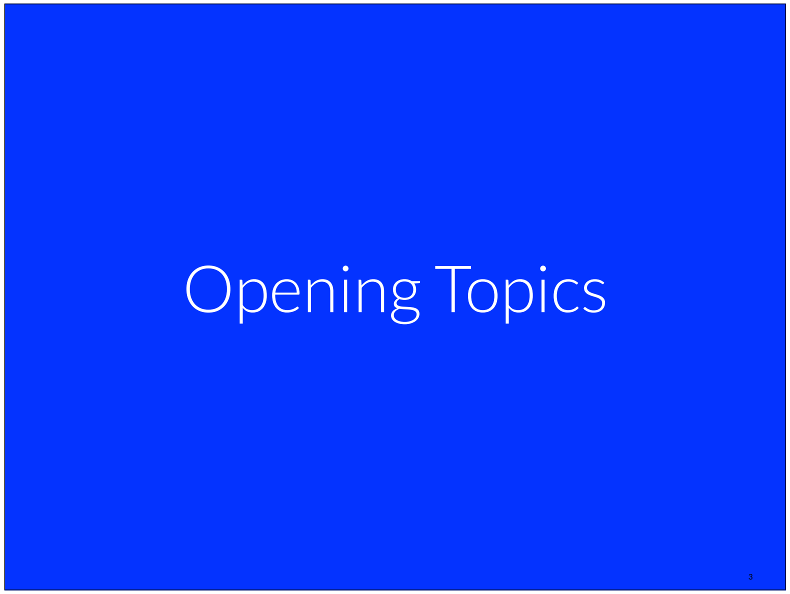
1

Craig Larman
www.craiglarman.com

Please...
Do not copy or share this material, or re-use for other education.
Exceptions require prior written consent of the author.

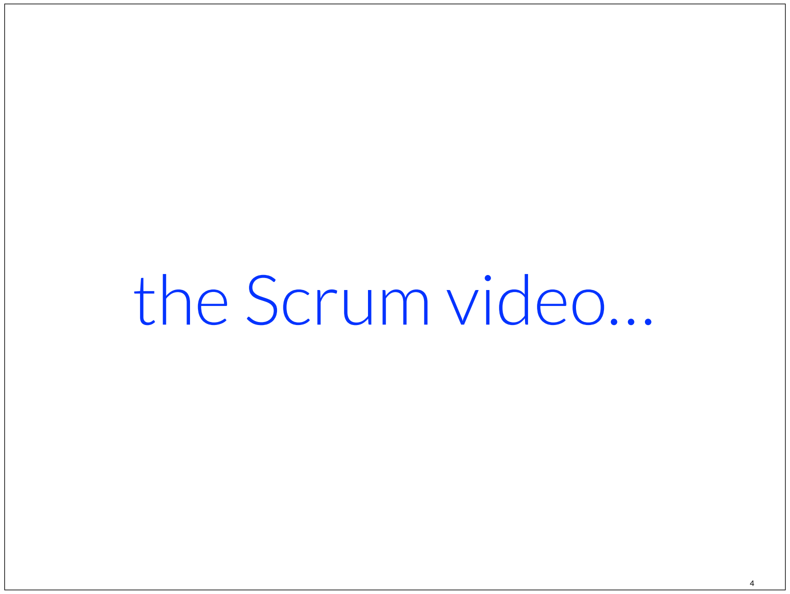
v.26

2



Opening Topics

3



the Scrum video...

4

5

Macro Schedule

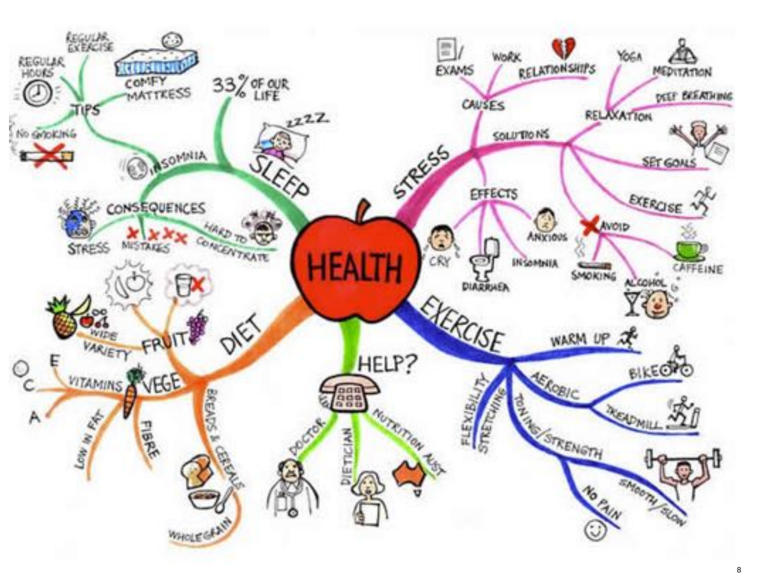
1. core topics
2. deep-dive Q&A

6

Core Topics

1. variability & complexity
2. agile
3. systems thinking, local vs system optimization
4. lean thinking
5. empirical process control
6. Scrum
7. adoption & change
8. scaling & LeSS

7



8

Preparation

9

Prerequisites

- completed the pre-readings

10

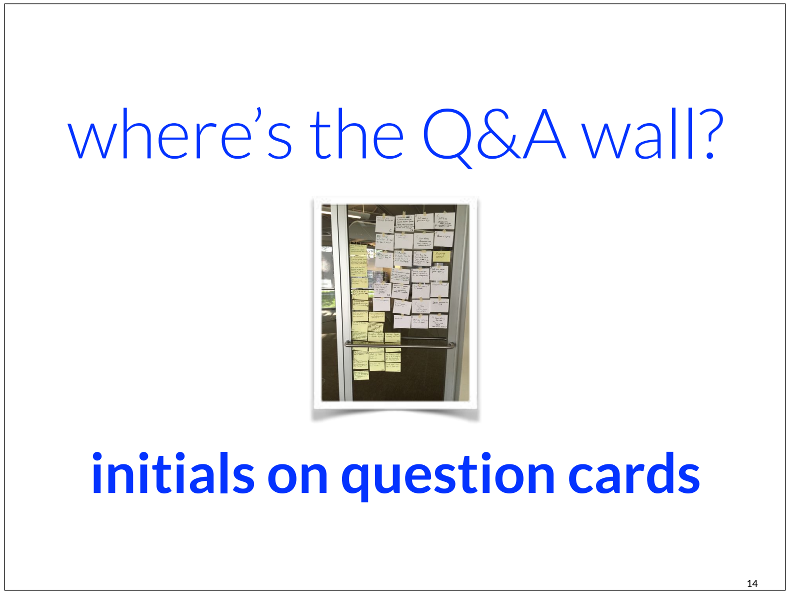
10

11

12

12

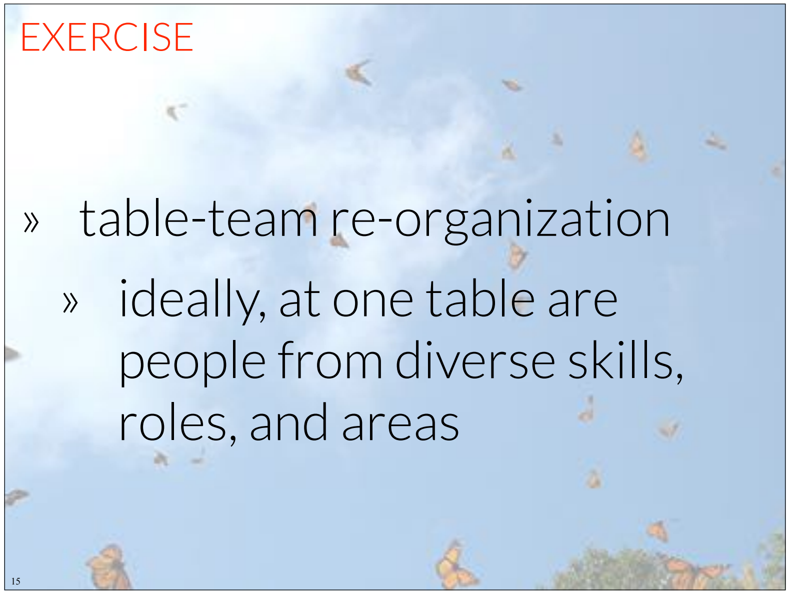
13



where’s the Q&A wall?

initials on question cards

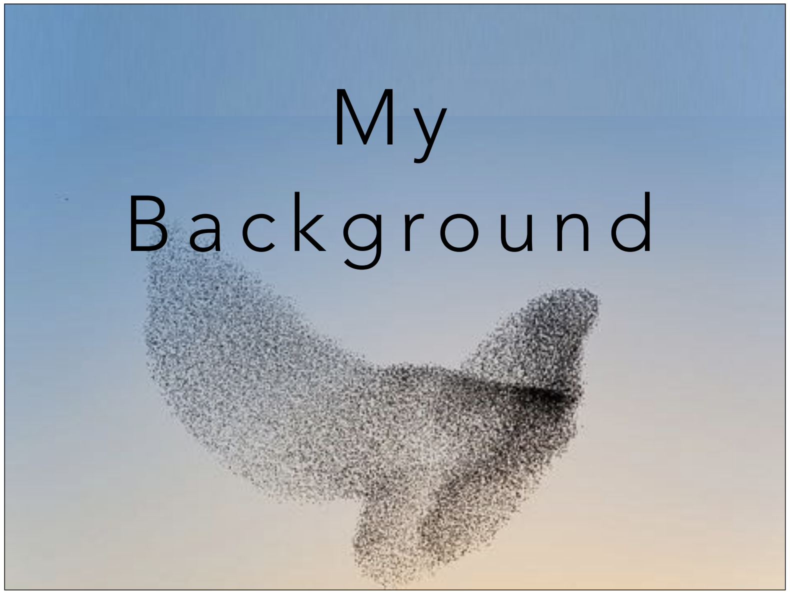
14



EXERCISE

- table-team re-organization
  - ideally, at one table are people from diverse skills, roles, and areas

15



My Background

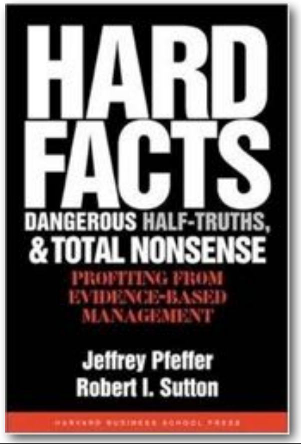
16

don’t believe anything i say

my biases…

evidence-based management

Stanford Business School



HARD FACTS

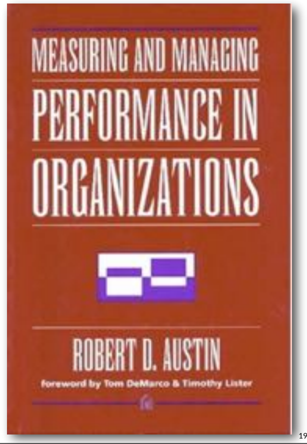
DANGEROUS HALF-TRUTHS, & TOTAL NONSENSE

PROFITING FROM EVIDENCE-BASED MANAGEMENT

Jeffrey Pfeffer
Robert I. Sutton

HARVARD BUSINESS SCHOOL PRESS

Harvard BusinessSchool



MEASURING AND MANAGING PERFORMANCE IN ORGANIZATIONS

ROBERT D. AUSTIN

foreword by Tom DeMarco & Timothy Lister

evidence-based management

Harvard
Business
School

1986

HBR

JANUARY-FEBRUARY 1986

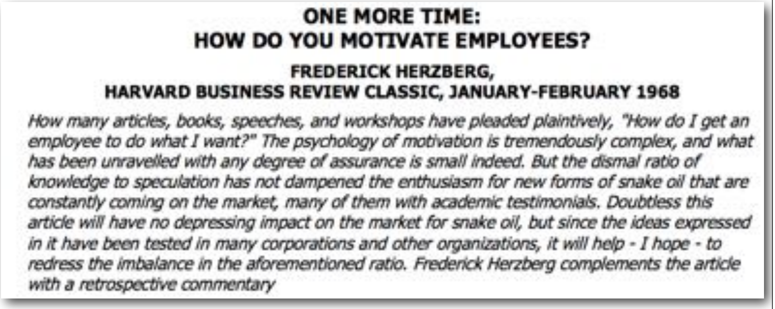
The New New Product Development Game

Hirotaka Takeuchi and Ikujiro Nonaka

Scrum

“Old School” & Evidence-Based

Harvard BusinessSchool: **1968**



ONE MORE TIME:
HOW DO YOU MOTIVATE EMPLOYEES?

FREDERICK HERZBERG,
HARVARD BUSINESS REVIEW CLASSIC, JANUARY-FEBRUARY 1968

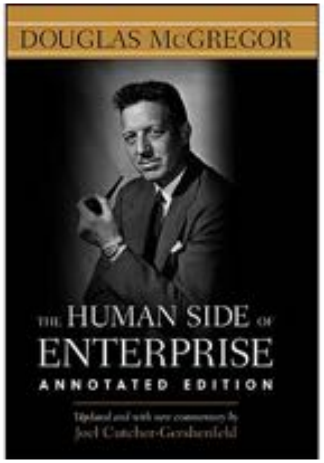
How many articles, books, speeches, and workshops have pleaded plaintively, "How do I get an employee to do what I want?" The psychology of motivation is tremendously complex, and what has been unravelled with any degree of assurance is small indeed. But the dismal ratio of knowledge to speculation has not dampened the enthusiasm for new forms of snake oil that are constantly coming on the market, many of them with academic testimonials. Doubtless this article will have no depressing impact on the market for snake oil, but since the ideas expressed in it have been tested in many corporations and other organizations, it will help - I hope - to redress the imbalance in the aforementioned ratio. Frederick Herzberg complements the article with a retrospective commentary

“Old School” & Evidence-Based

MIT Sloan School of Management

1960-present

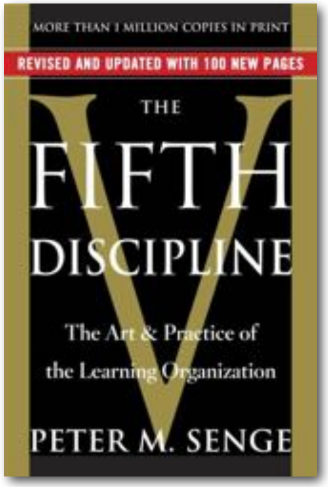
“… good management requires a Theory Y orientation.”
— MIT Sloan


DOUGLAS McGREGOR
THE HUMAN SIDE OF ENTERPRISE
ANNOTATED EDITION
Updated and with new commentary by
Joel Cutcher-Gershenfeld

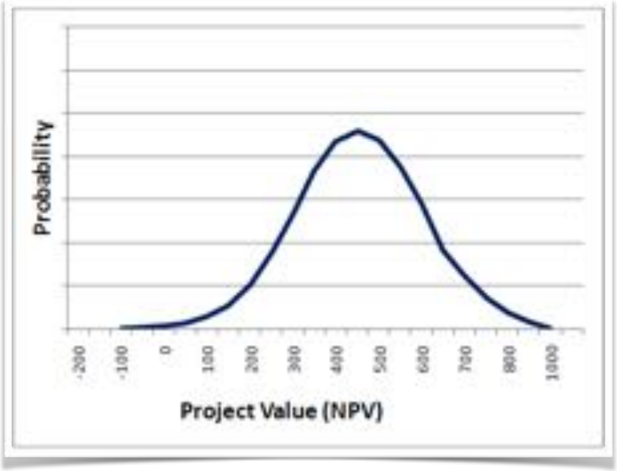
Systems Thinking & Evidence-Based

MIT Sloan School of Management

1958-present


MORE THAN 1 MILLION COPIES IN PRINT
REVISED AND UPDATED WITH 100 NEW PAGES
THE
FIFTH
DISCIPLINE
The Art & Practice of
the Learning Organization
PETER M. SENGE

Probabilistic (not deterministic) Modeling



my work…

25

25

Craig Larman

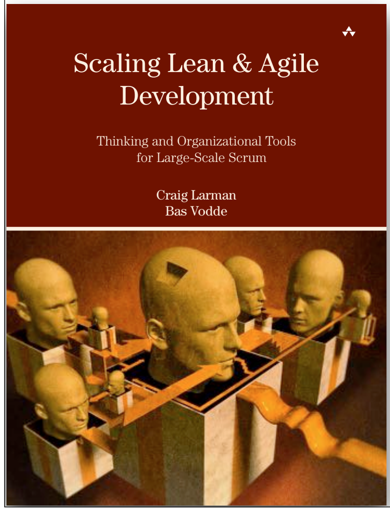
co-creator of LeSS (with Bas Vodde)

large + multisite + ‘offshore’
large-scale financial systems
large-scale embedded systems
large-scale telecom systems

26

26

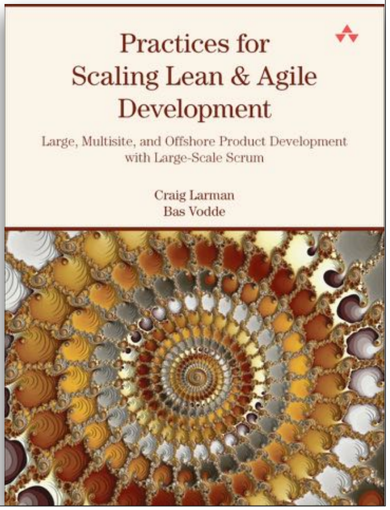
first two scaling agile books…



Scaling Lean & Agile Development

Thinking and Organizational Tools for Large-Scale Scrum

Craig Larman
Bas Vodde



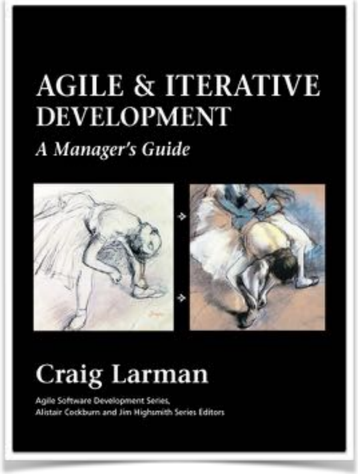
Practices for Scaling Lean & Agile Development

Large, Multisite, and Offshore Product Development with Large-Scale Scrum

Craig Larman
Bas Vodde

27

one of the first agile books…



AGILE & ITERATIVE DEVELOPMENT

A Manager's Guide

Craig Larman

Agile Software Development Series,
Alistair Cockburn and Jim Highsmith Series Editors

28

28

Highlights

- Modern 43,000 sq ft. (4,200 sq.m.) facility in Bangalore, India
- Round-the-clock operations supporting worldwide Agile development projects
- 500 full-time Valtech employees
- 24/7 physical security and access control
- Virtual project rooms with complete development & test facilities
- Cutting-edge voice, data and power distribution networks

Global Delivery Center

Founded in 2004 following the acquisition of Majoris by Valtech, the Valtech Global Delivery Center was awarded SEI-CMM Level 5 certification in July 2005.

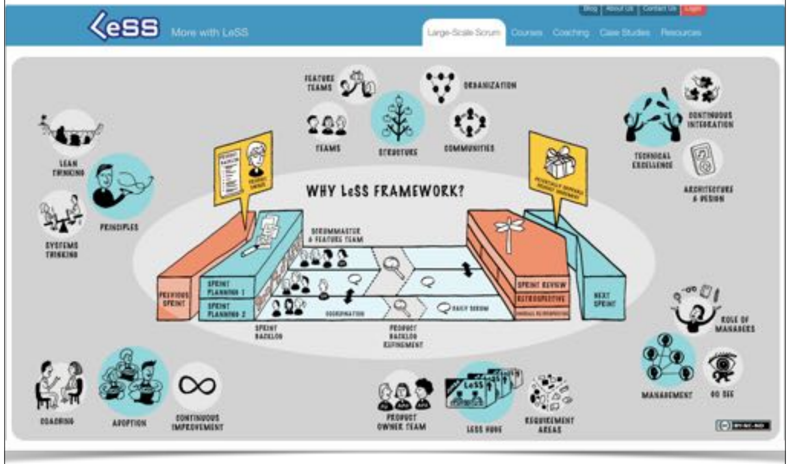
served as chief scientist @ Valtech
helped create “agile offshore” in LeSS

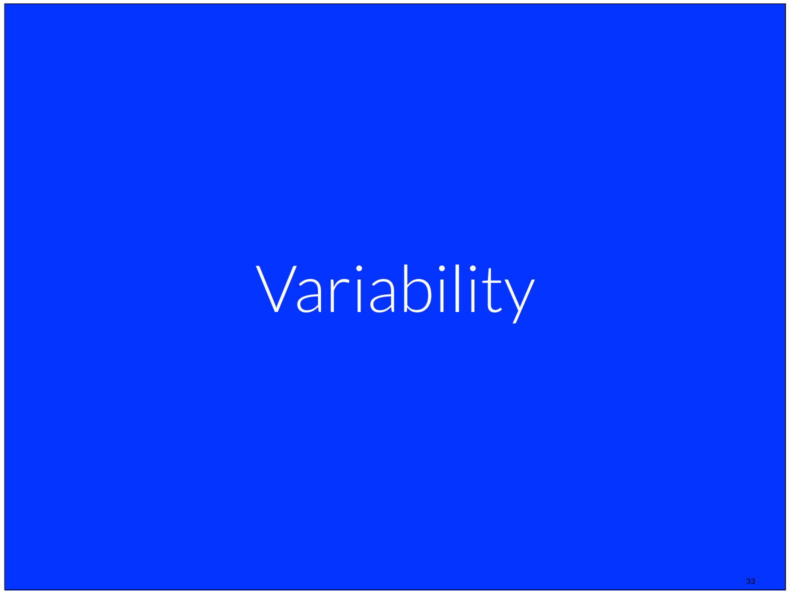
served as lead coach of
lean development @ Xerox

LeSS consultant @

- Ericsson
- JPMorgan
- Vodafone
- BAML
- Nokia Networks
- bwin.party
- UBS
- CISCO (Tandberg)
- Xerox, …

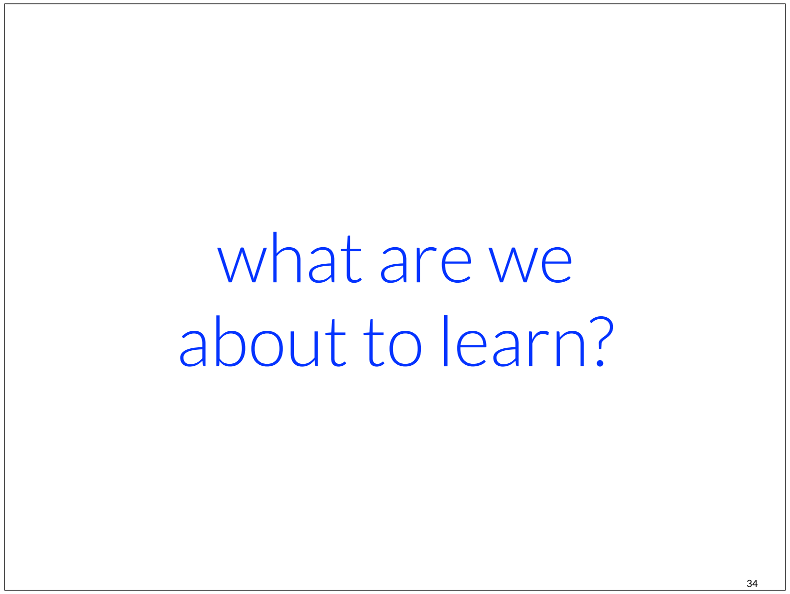
less.works





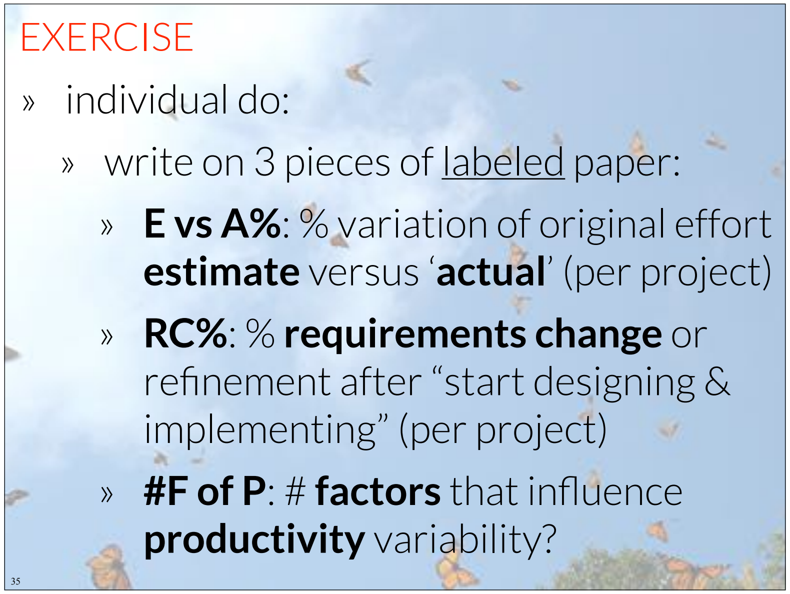
Variability

33



what are we about to learn?

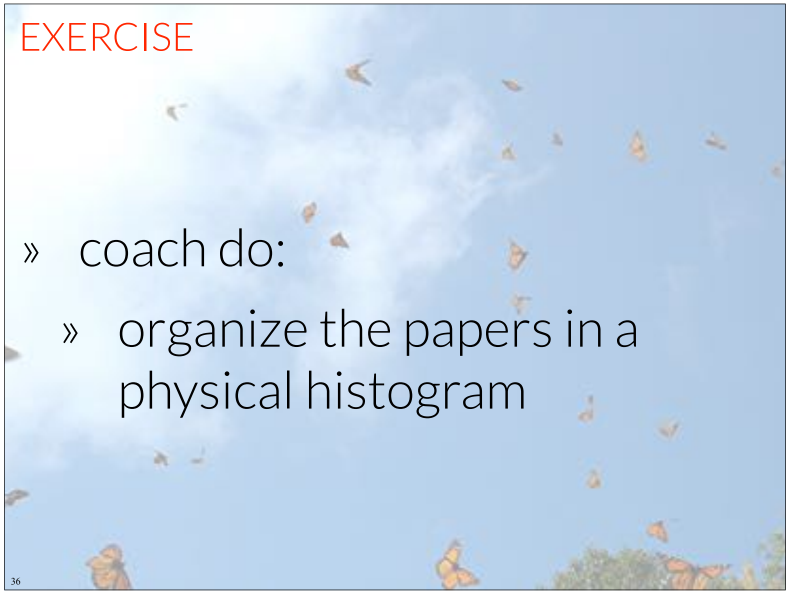
34



EXERCISE

- individual do:
  - write on 3 pieces of labeled paper:
    - **E vs A%**: % variation of original effort **estimate** versus '**actual**' (per project)
    - **RC%**: % **requirements change** or refinement after "start designing & implementing" (per project)
    - **#F of P**: # **factors** that influence **productivity** variability?

35



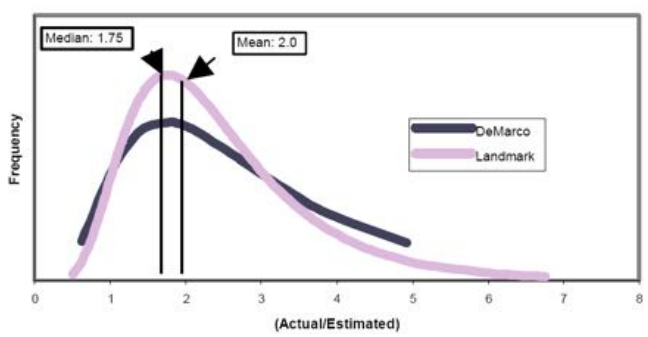
EXERCISE

- coach do:
  - organize the papers in a physical histogram

36

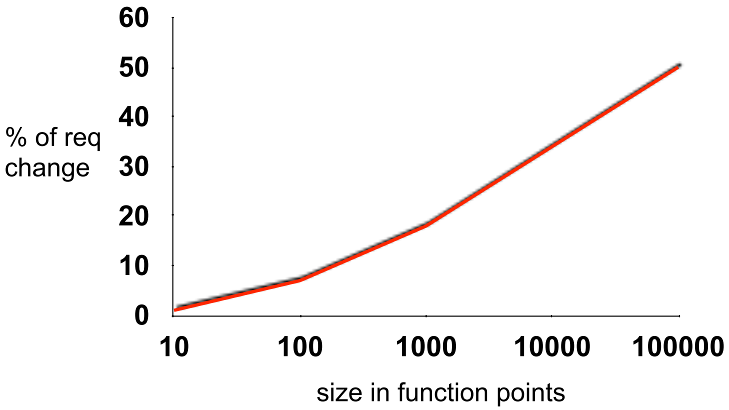
p50 of actual = 2 x estimate, p90 = 3.25 x estimate
Original estimates have a 10% chance or less of equaling actuals.

Figure 4: Probability Distribution Curve of Actual/Estimated



[DeMarco82, Little04]

37



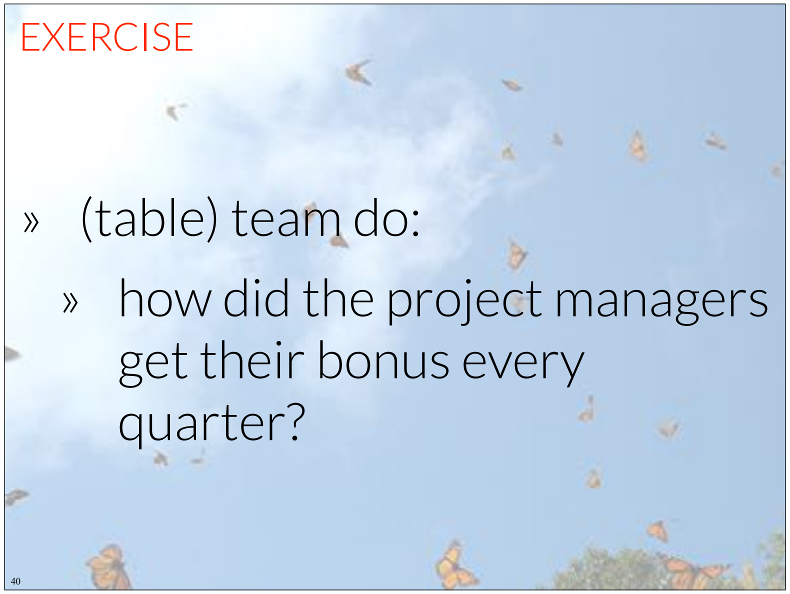
sample size: 6,700 product releases [Jones97]

38

“we’re better than that”

the story of Budapest

39

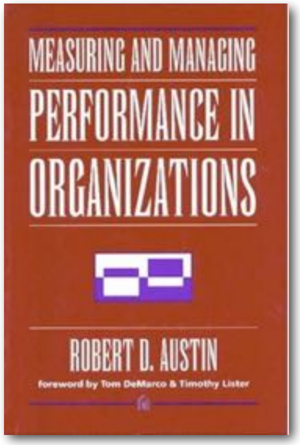


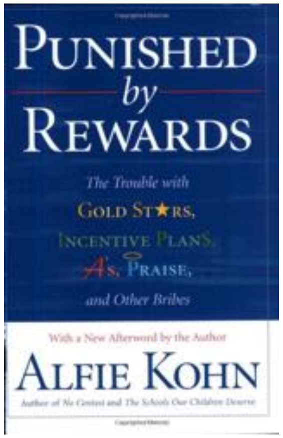
EXERCISE

- (table) team do:
  - how did the project managers get their bonus every quarter?

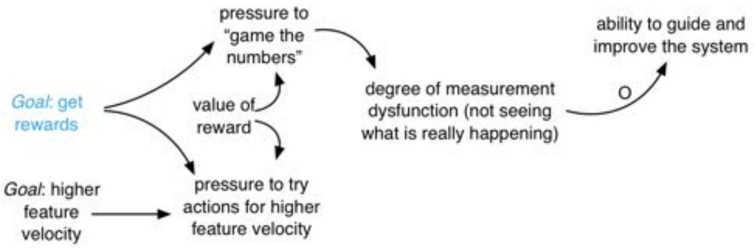
40

Punished by Rewards





Your System Dynamics?



Goal: get rewards

pressure to "game the numbers"

value of reward

degree of measurement dysfunction (not seeing what is really happening)

O

ability to guide and improve the system

Goal: higher feature velocity

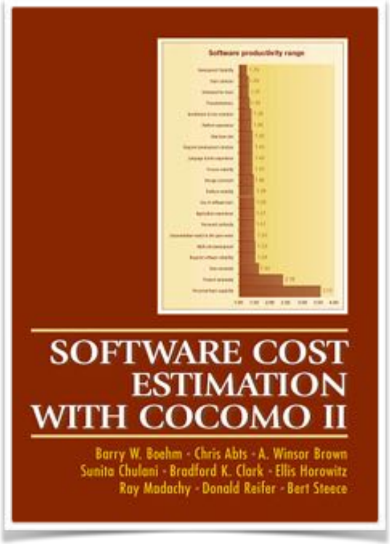
pressure to try actions for higher feature velocity

# major factors influencing overall 'productivity' variability?

-> 21

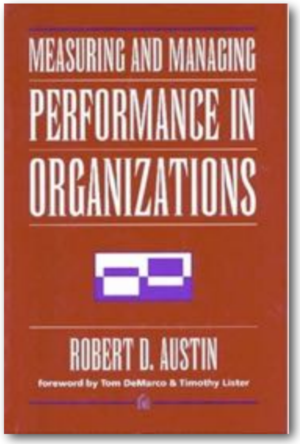
and most of the major factors have 3+ sub-factors

-> "60" factors


Software productivity range
SOFTWARE COST
ESTIMATION
WITH COCOMO II
Barry W. Boehm · Chris Abts · A. Winsor Brown
Sunita Chulani · Bradford K. Clark · Ellis Horowitz
Ray Madachy · Donald Reifer · Bert Steece

what are your variability levels? …

you have no accurate idea, and your so-called “actuals tracking” system is FAR from accurate



45

can you ascertain the source of your productivity variation from reports summarizing LOC or FPs or “feature-complete velocity”? …

46

“they’re really fast”

the story of Rochester

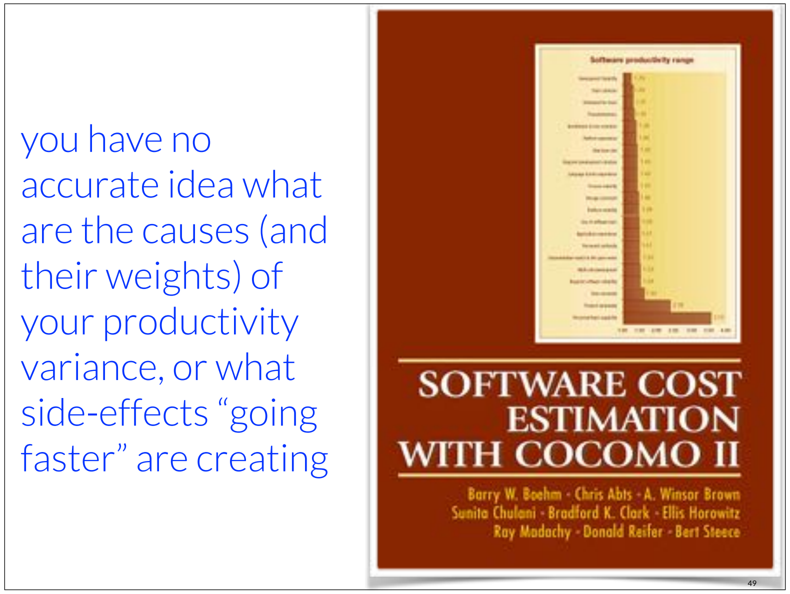
the India Outsourcer

47

“they deliver a lot of features”

the story of the 80% unused-features product

48



you have no accurate idea what are the causes (and their weights) of your productivity variance, or what side-effects “going faster” are creating

SOFTWARE COST ESTIMATION WITH COCOMO II

Barry W. Boehm · Chris Abts · A. Winsor Brown
Sunita Chulani · Bradford K. Clark · Ellis Horowitz
Ray Madachy · Donald Reifer · Bert Steece

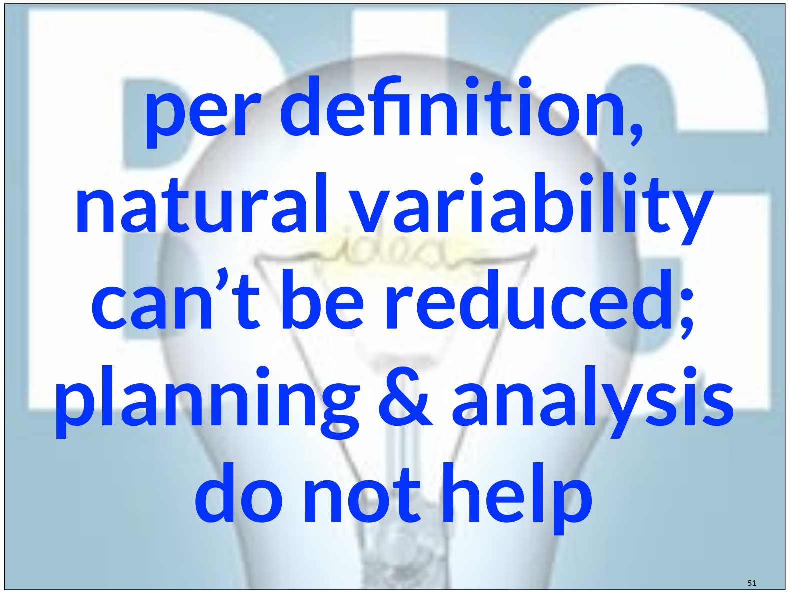
49

Shewert & Deming:

natural variability

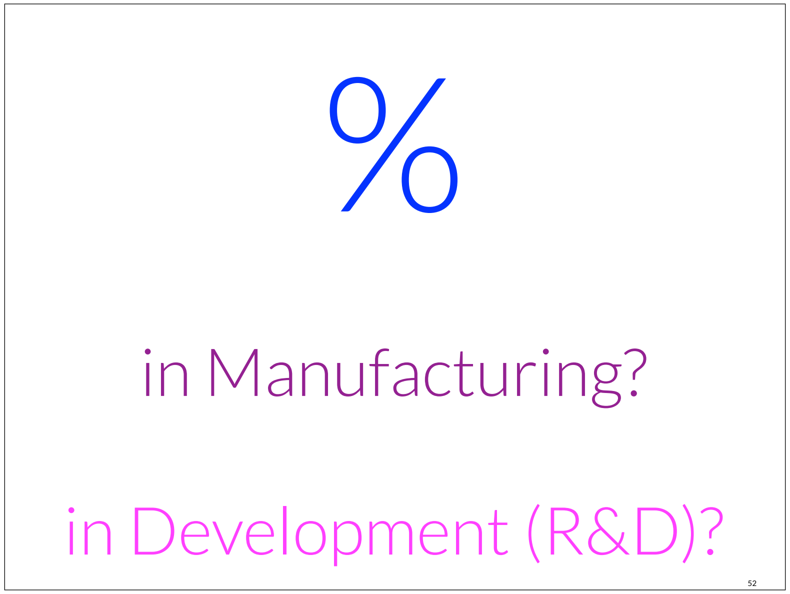
manipulatable variability

50



per definition, natural variability can’t be reduced; planning & analysis do not help

51

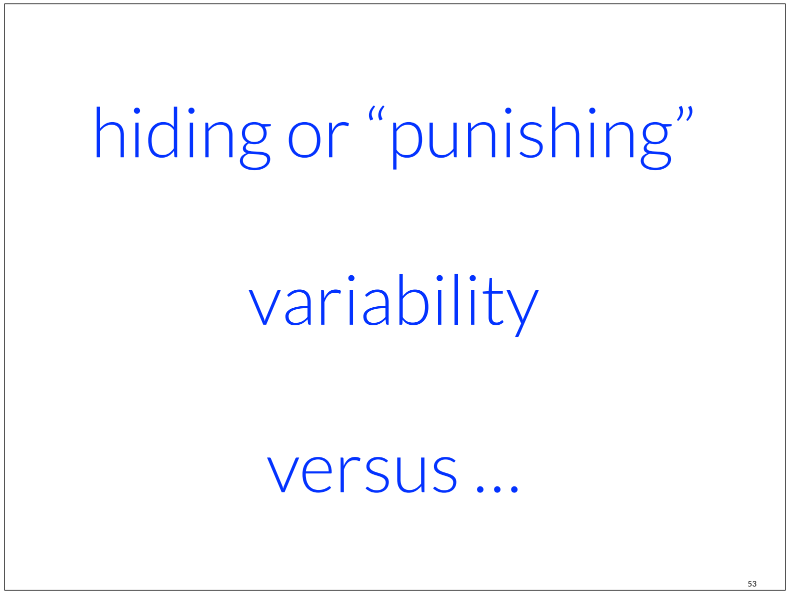


%

in Manufacturing?

in Development (R&D)?

52



hiding or "punishing"

variability

versus ...

53

transparency & manage variability

54

Business School planning...

55

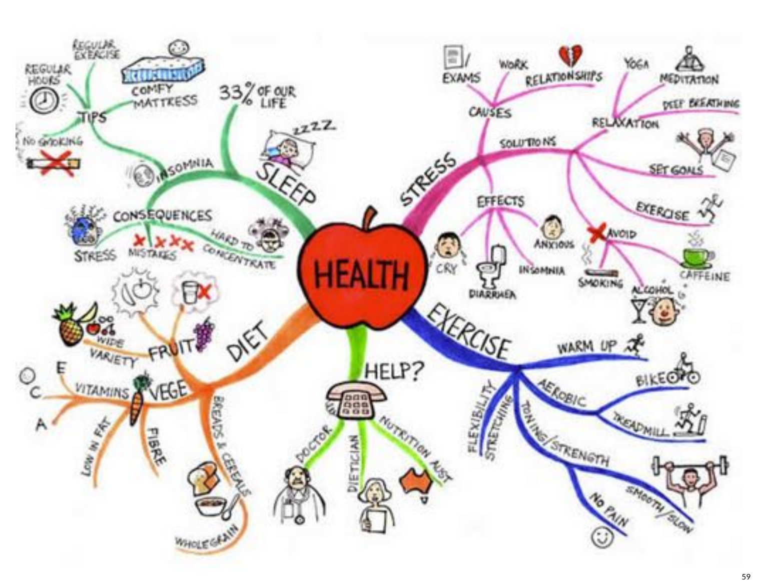
56

learn & educate

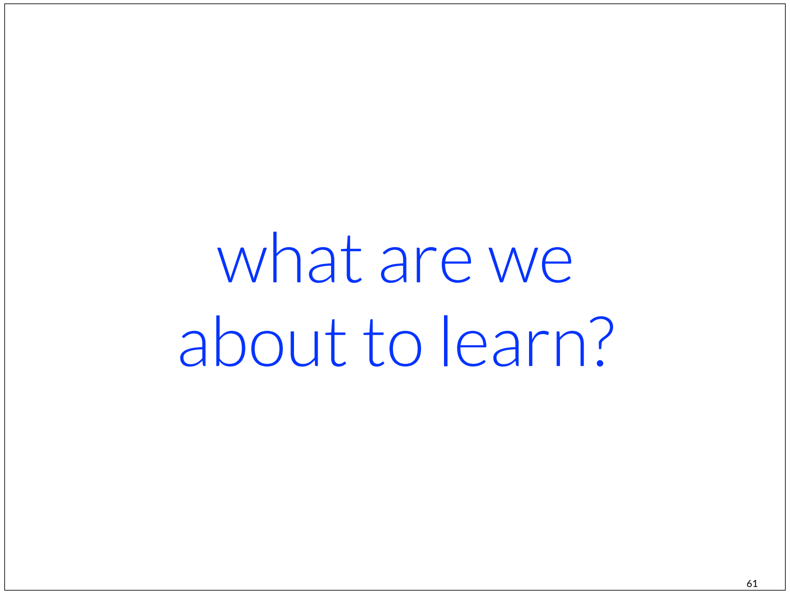
why

EXERCISE

- team: standing: round robin
  - share aspects of the “variability story”

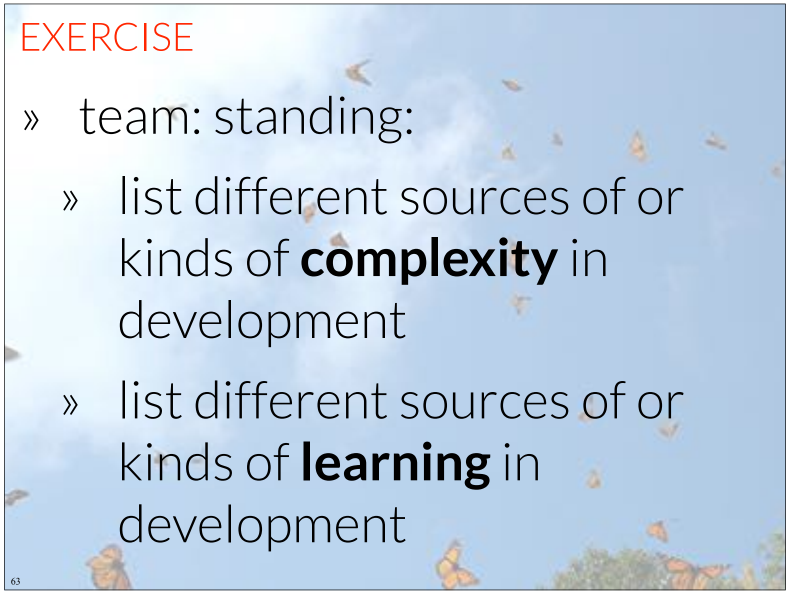


Complexity & Learning



what are we
about to learn?

where does natural
variability come from in
development? …



EXERCISE

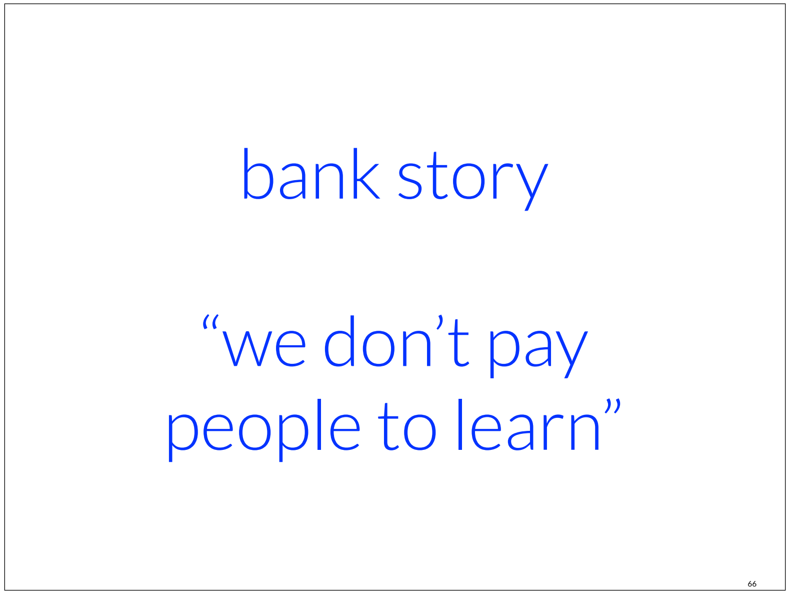
- team: standing:
  - list different sources of or kinds of **complexity** in development
  - list different sources of or kinds of **learning** in development

EXERCISE

- coach:
  - collect counts of kinds of **complexity** & **learning**

significant complexity & learning in development

65



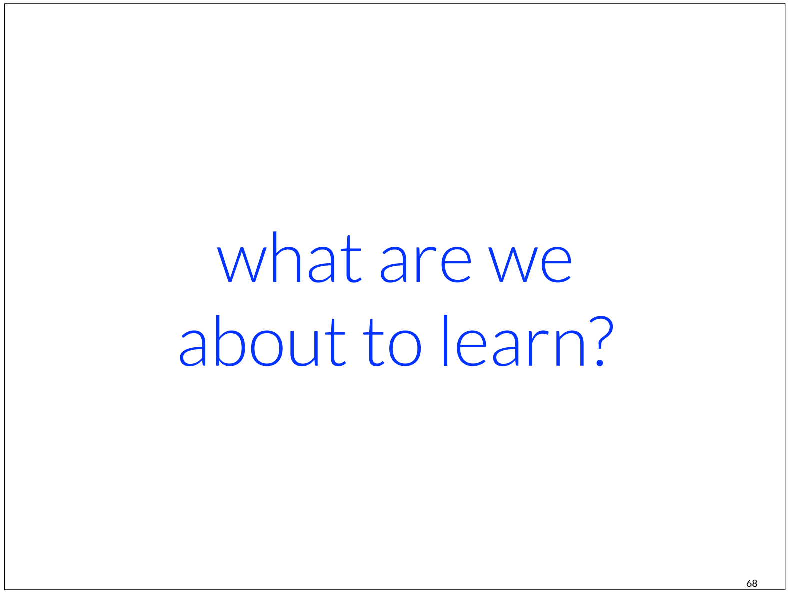
bank story

“we don’t pay people to learn”

66

Managing Variability, Complexity, & Learning

67



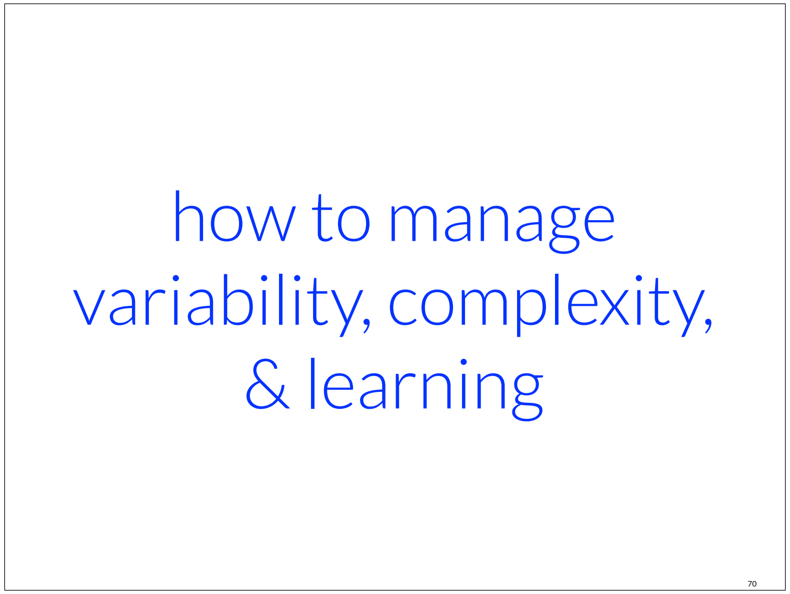
what are we about to learn?

68

the assumptions of traditional management regarding **predictability**, **control**, & **variability**?

69

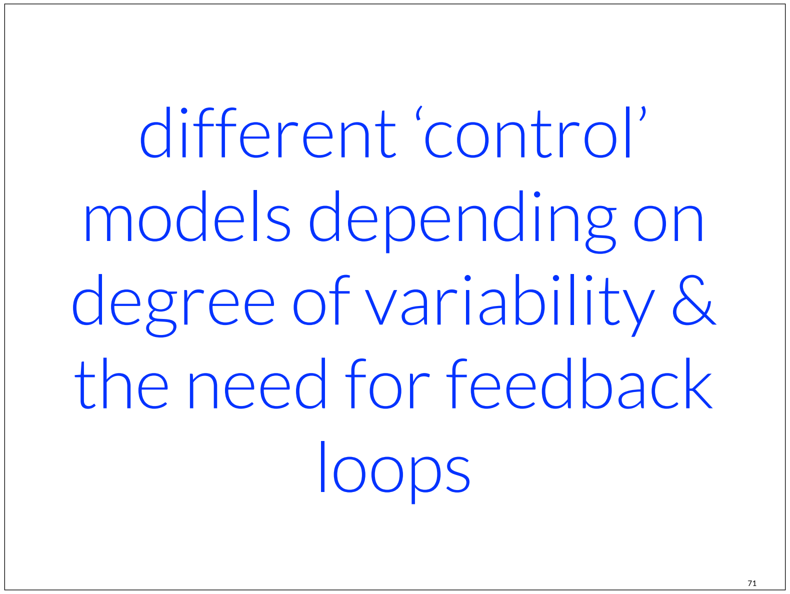
69



how to manage variability, complexity, & learning

70

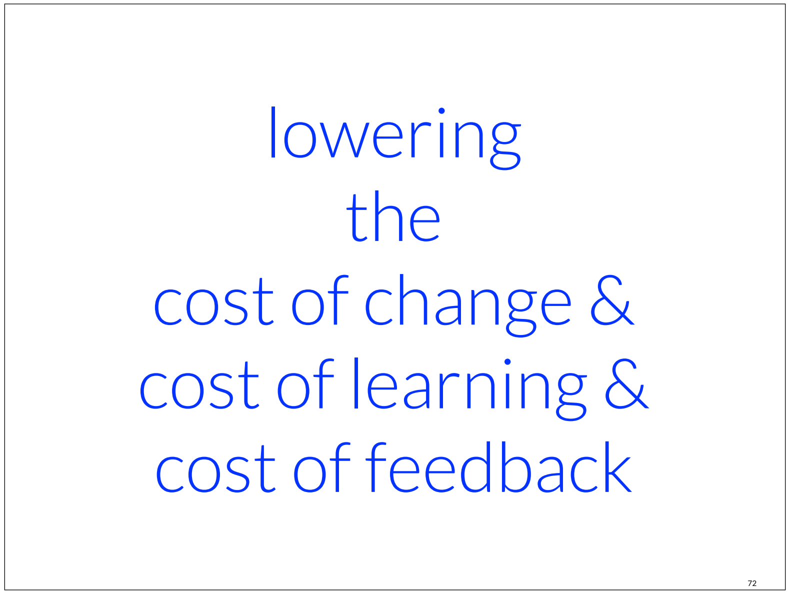
70



different ‘control’ models depending on degree of variability & the need for feedback loops

71

71



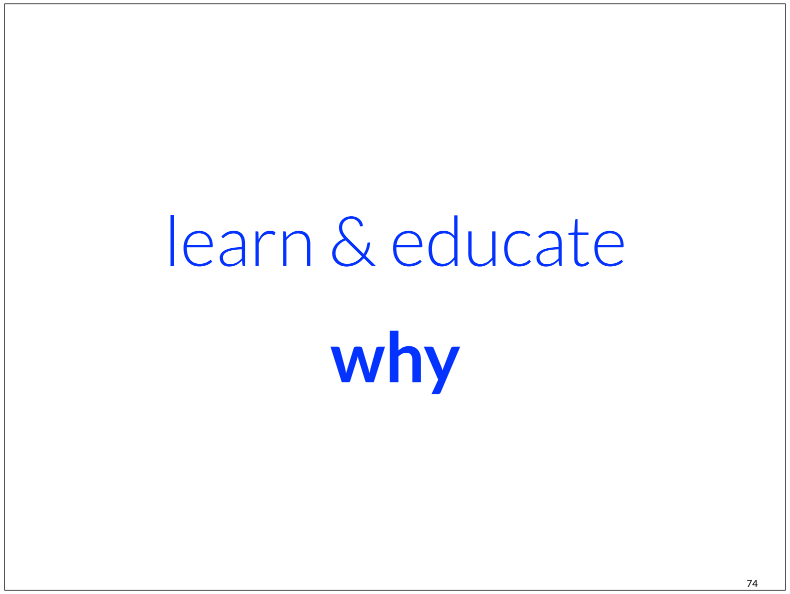
lowering
the
cost of change &
cost of learning &
cost of feedback

72

72

transparency
& manage
variability

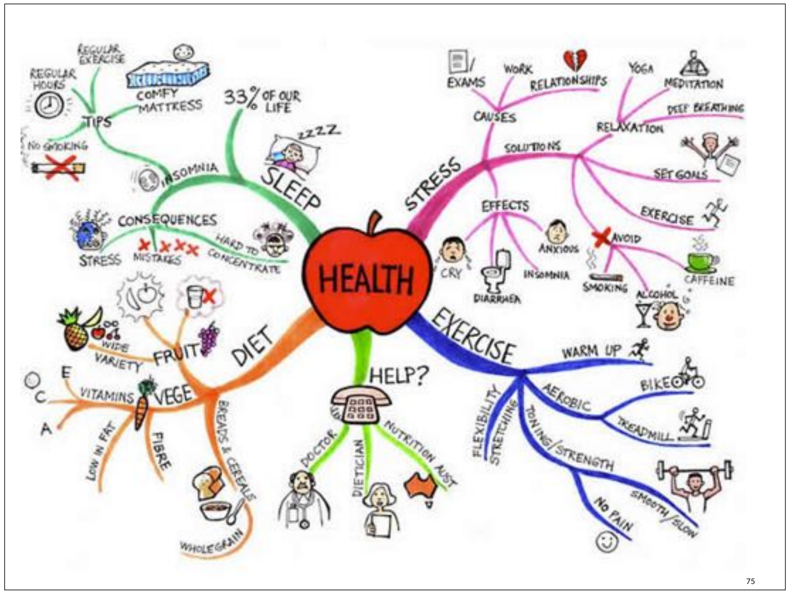
73



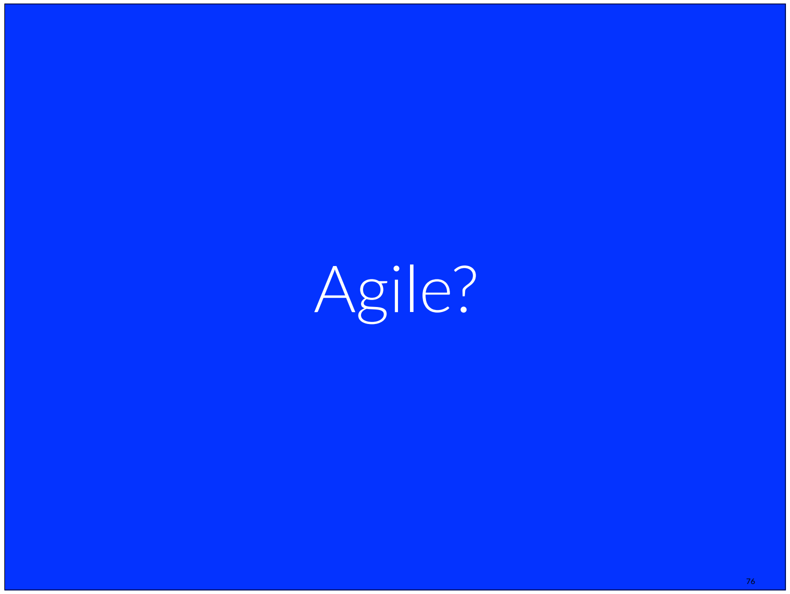
learn & educate

why

74



75

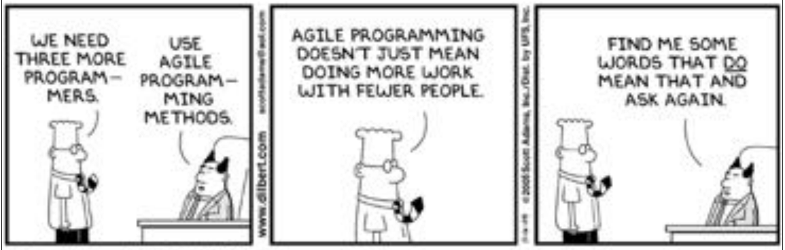


Agile?

76

what are we about to learn?

77

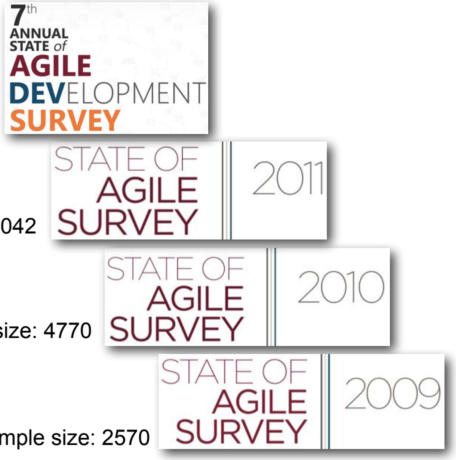

WE NEED THREE MORE PROGRAMMERS.
USE AGILE PROGRAMMING METHODS.
AGILE PROGRAMMING DOESN'T JUST MEAN DOING MORE WORK WITH FEWER PEOPLE.
FIND ME SOME WORDS THAT DO MEAN THAT AND ASK AGAIN.
www.dilbert.com

© Scott Adams, Inc./Dist. by UFS, Inc.

78

the implications of “agile” management?…

79


7th ANNUAL STATE of AGILE DEVELOPMENT SURVEY
STATE OF AGILE SURVEY 2011
STATE OF AGILE SURVEY 2010
STATE OF AGILE SURVEY 2009

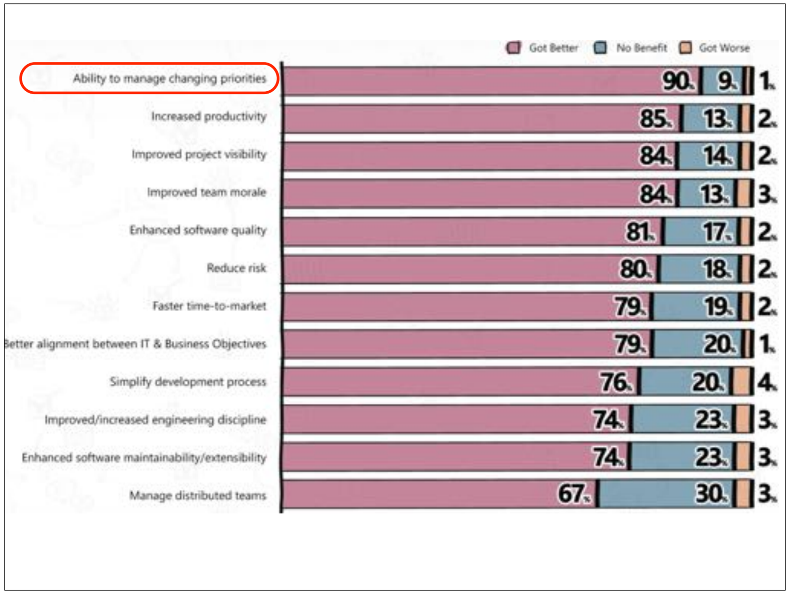
sample size: 4048

sample size: 6042

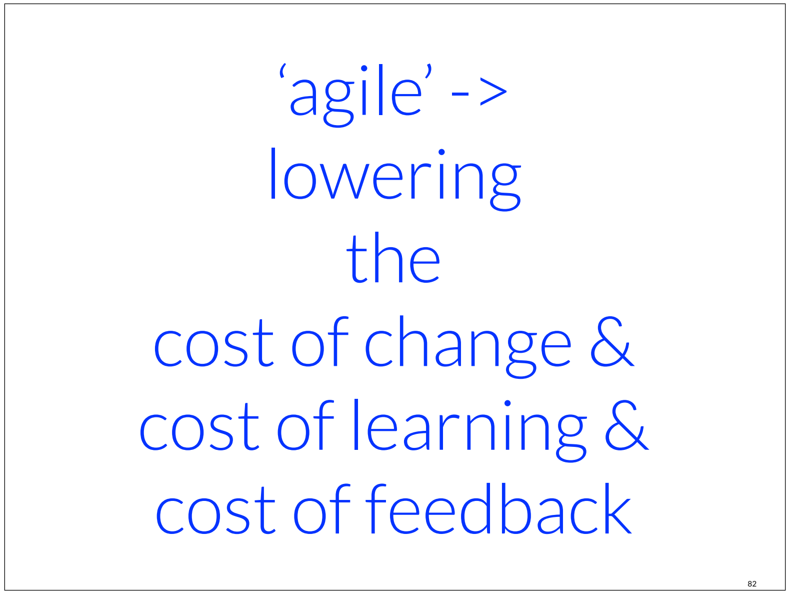
sample size: 4770

sample size: 2570

80



81


‘agile’ ->
lowering
the
cost of change &
cost of learning &
cost of feedback

82

agile
is for
agility

83

agile
is for
managing
variability

84

… and implies

‘agile’ is for

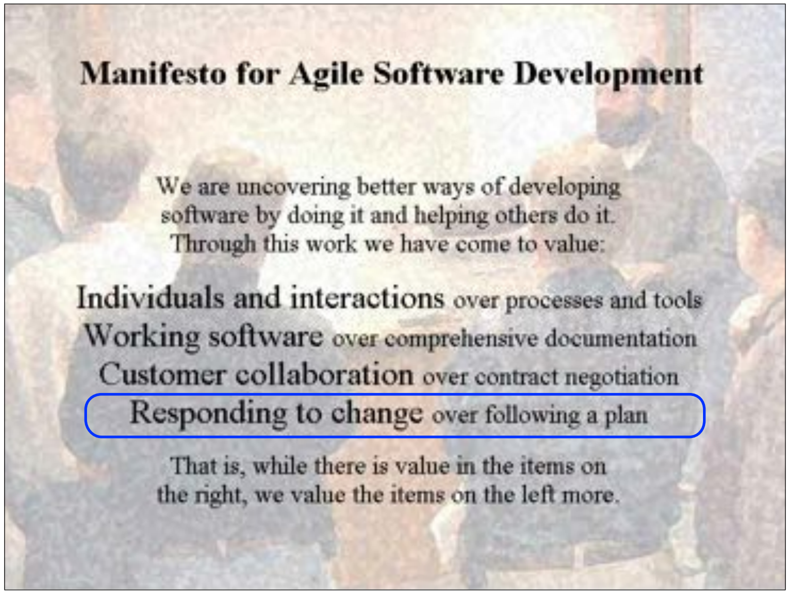
learning & adapting

85

the

Agile Values …

86



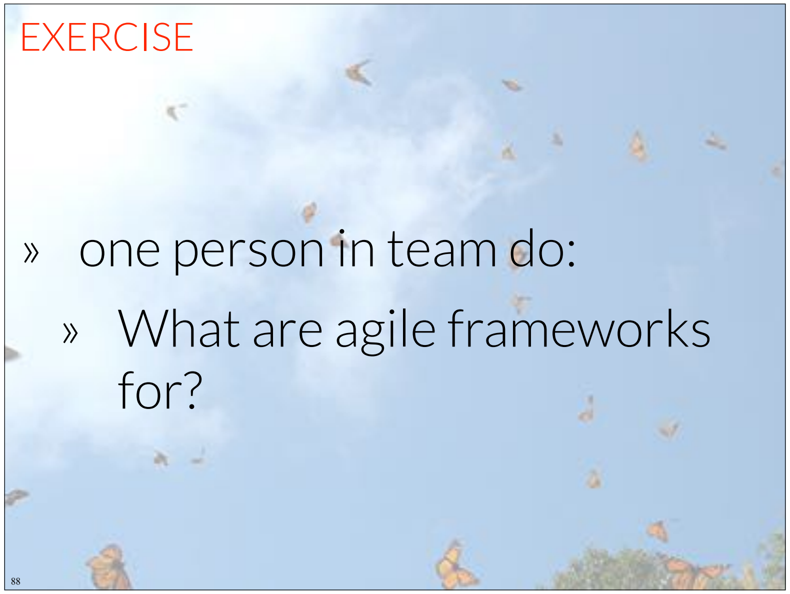
Manifesto for Agile Software Development

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.

87



EXERCISE

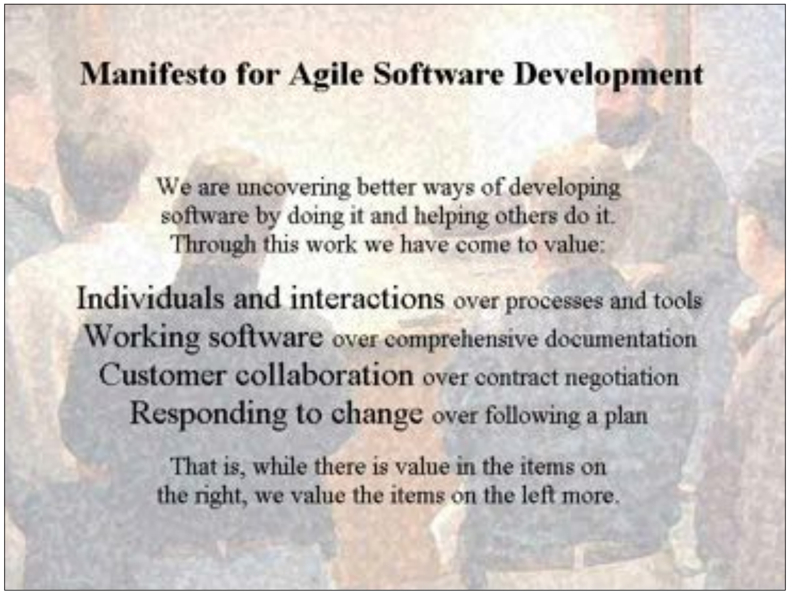
- one person in team do:
  - What are agile frameworks for?

88

notice that 'agile'
is **not a practice**

it is a set of **4 values**...

89



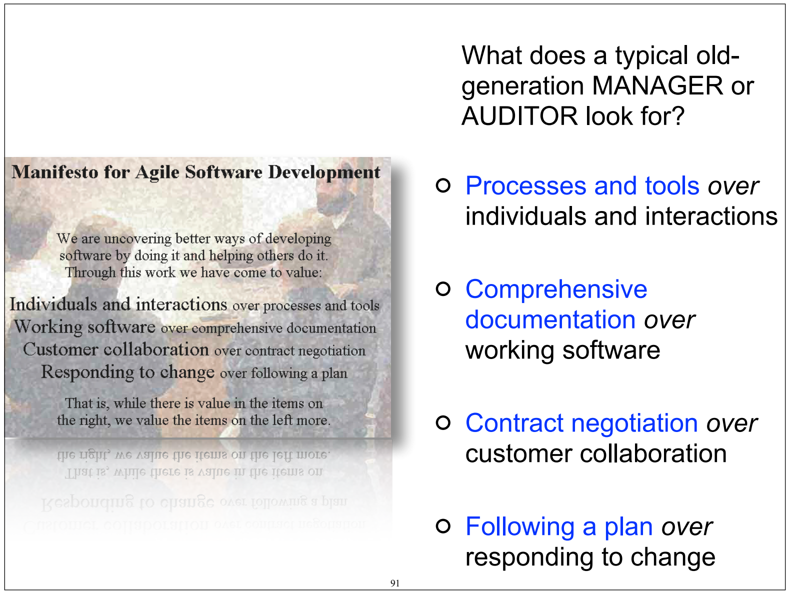
Manifesto for Agile Software Development

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.

90



What does a typical old-generation MANAGER or AUDITOR look for?

- Processes and tools *over* individuals and interactions
- Comprehensive documentation *over* working software
- Contract negotiation *over* customer collaboration
- Following a plan *over* responding to change

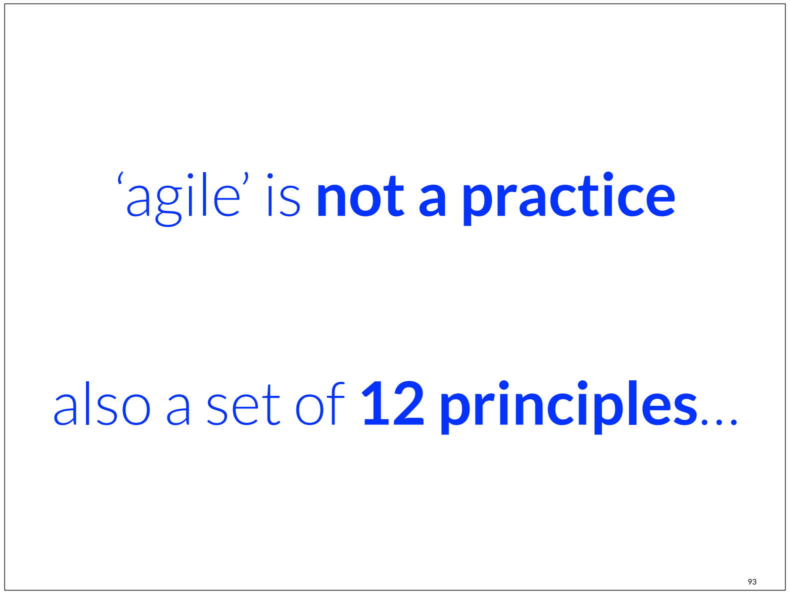
91

EXCISE

EXERCISE

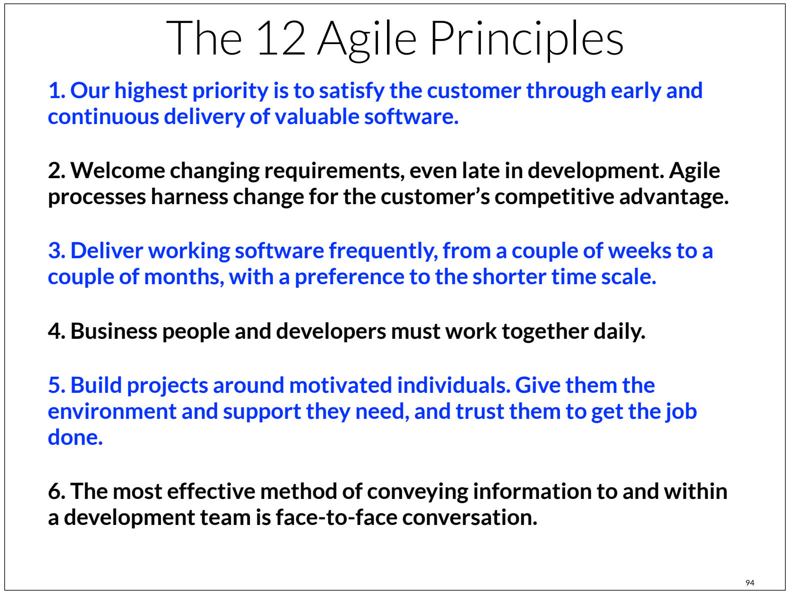
- team: standing: round robin
  - a concrete change in your organization to support each of the agile values?

92



‘agile’ is **not a practice**

also a set of **12 principles**…



The 12 Agile Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

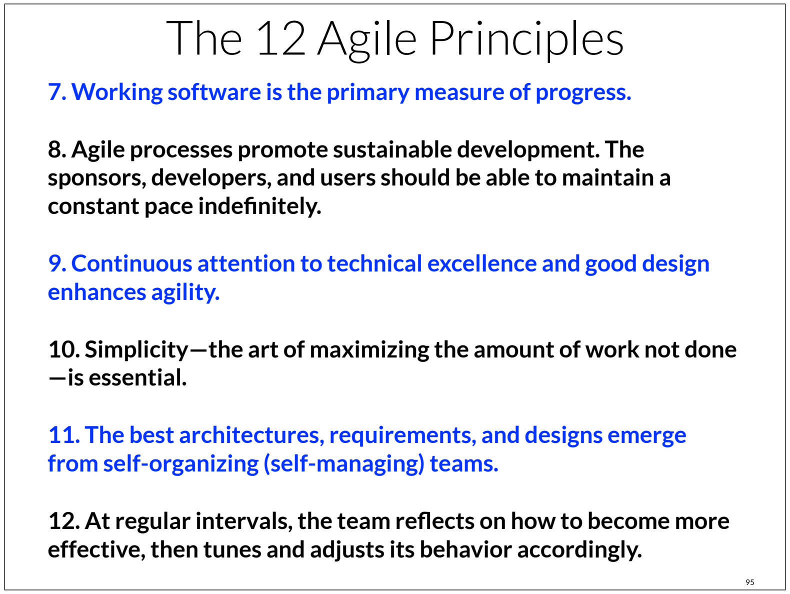
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter time scale.

4. Business people and developers must work together daily.

5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most effective method of conveying information to and within a development team is face-to-face conversation.



The 12 Agile Principles

7. Working software is the primary measure of progress.

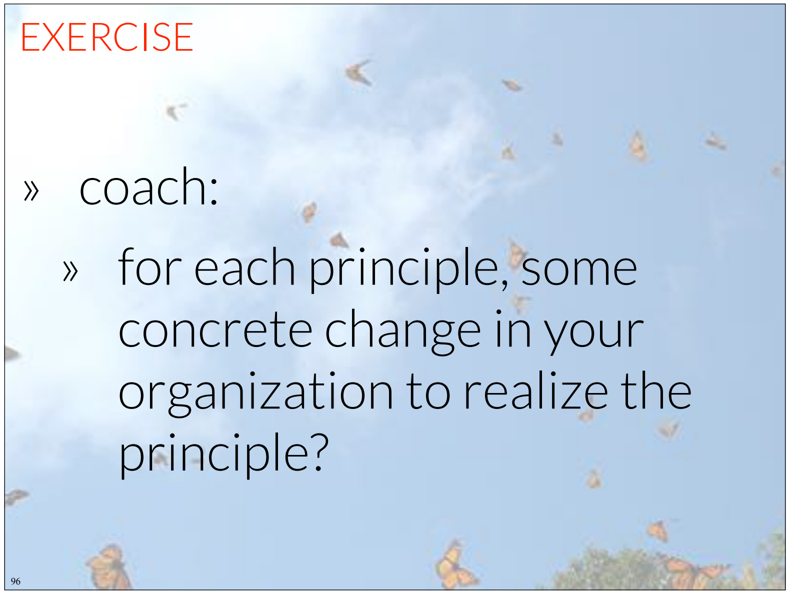
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

9. Continuous attention to technical excellence and good design enhances agility.

10. Simplicity—the art of maximizing the amount of work not done—is essential.

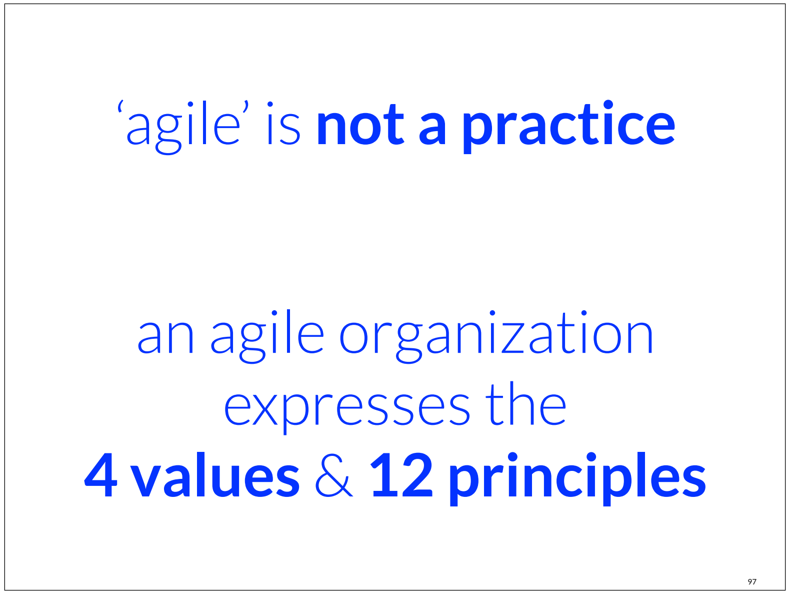
11. The best architectures, requirements, and designs emerge from self-organizing (self-managing) teams.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



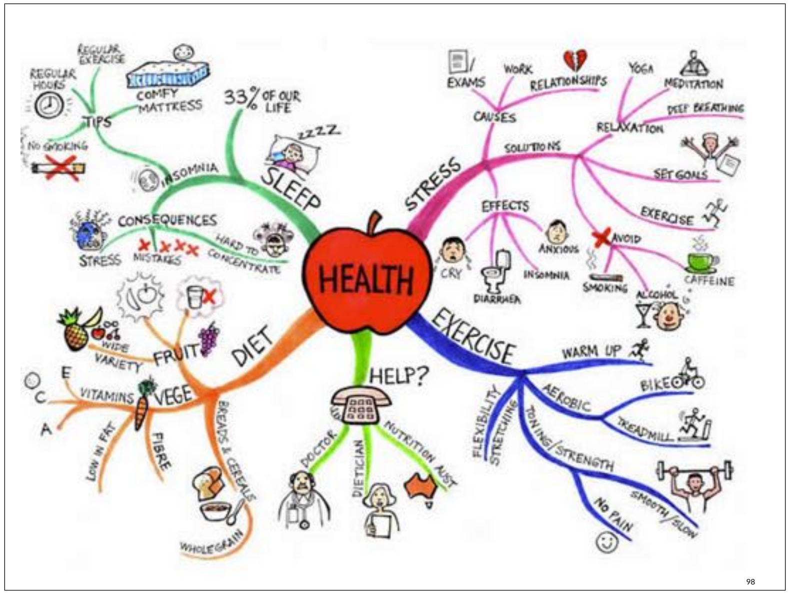
EXERCISE

- coach:
  - for each principle, some concrete change in your organization to realize the principle?

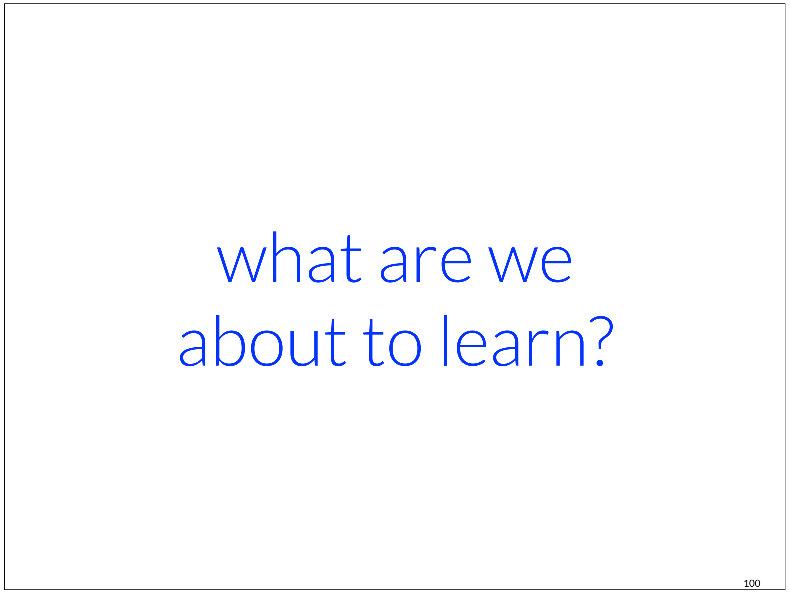


‘agile’ is **not a practice**

an agile organization
expresses the
4 values & **12 principles**



Systems Thinking:
Local Optimization
&
System Optimization



what are we
about to learn?

in traditional large-scale organizational design, the overarching and repeating theme is **local optimization**

Local Optimization, Local Efficiency

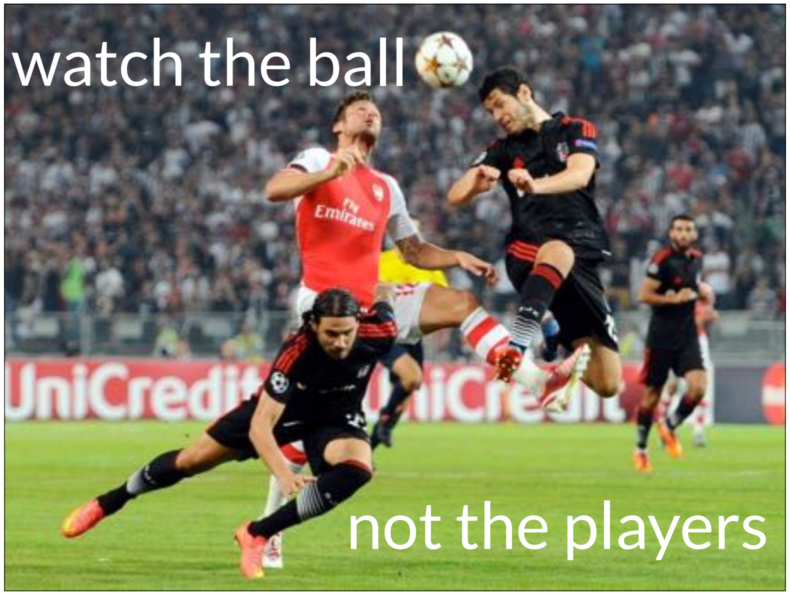
"it's more **efficient/productive** when a person/group does one thing"

Local Optimization, Local Efficiency

"everyone is busy and doing their best on their task, yet the system is delivering slow and not delighting the user"

system optimization? …

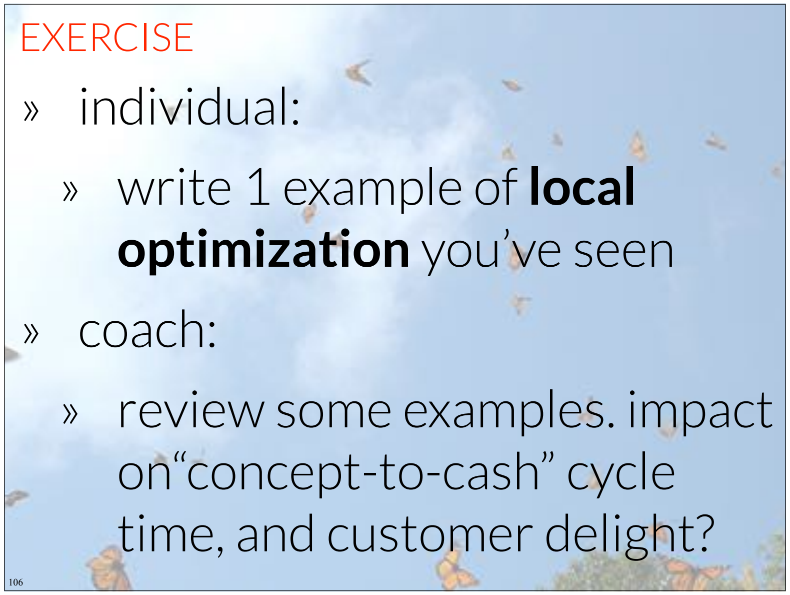
(a part of systems thinking)



watch the ball

not the players

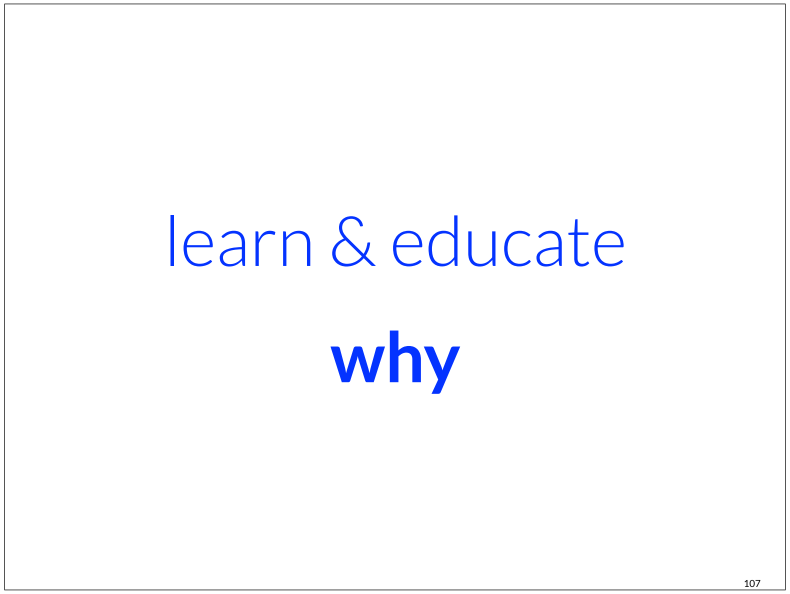
105



EXERCISE

- individual:
  - write 1 example of **local optimization** you've seen
- coach:
  - review some examples. impact on"concept-to-cash" cycle time, and customer delight?

106



learn & educate

why

107

Local Optimization in Planning: The Contract Game

108

what are we
about to learn?

109

109

the

Contract Game

110

110

EXCERCISE

- team: standing:
  - list negative consequences of the Contract Game?

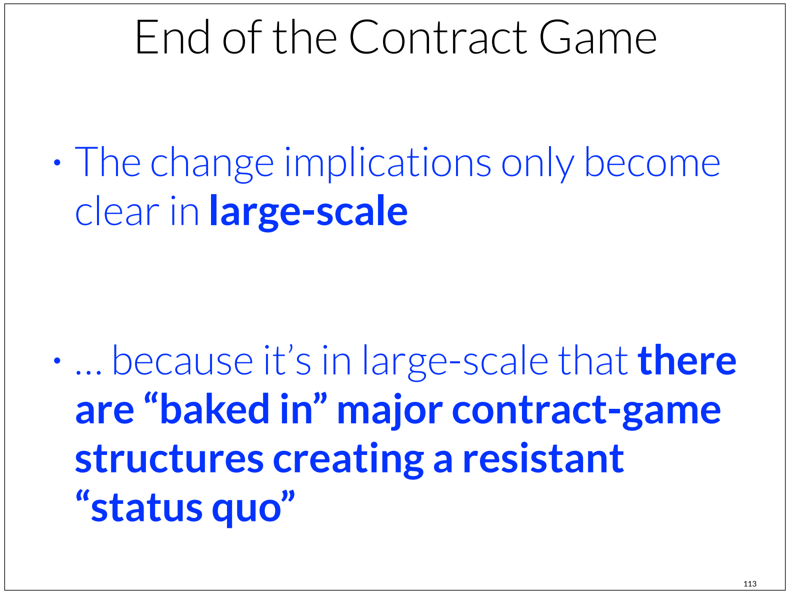
111

111

the Contract Game
hides variability
rather than
manages variability

112

112

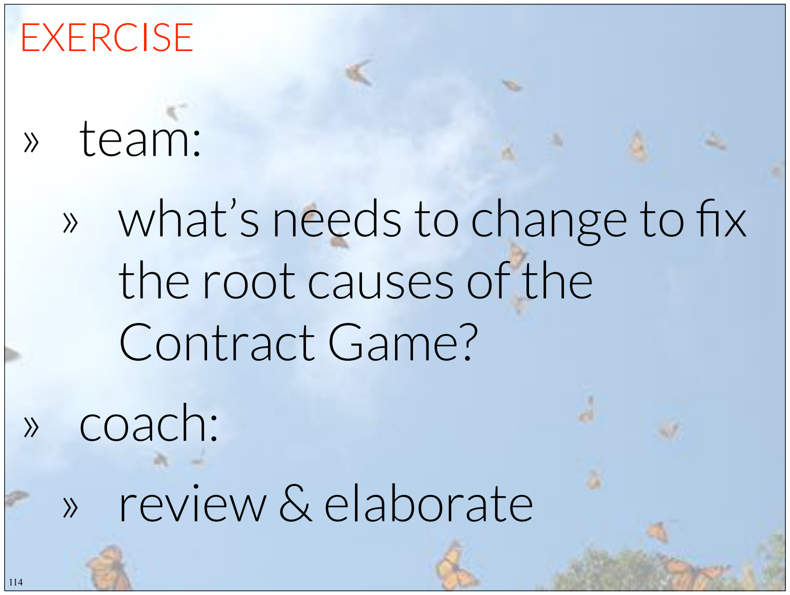


End of the Contract Game

- The change implications only become clear in **large-scale**
- … because it's in large-scale that **there are "baked in" major contract-game structures creating a resistant "status quo"**

113

113



EXERCISE

» team:
  » what's needs to change to fix the root causes of the Contract Game?

» coach:
  » review & elaborate

114

114

the

Agile Values …

115

115

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.

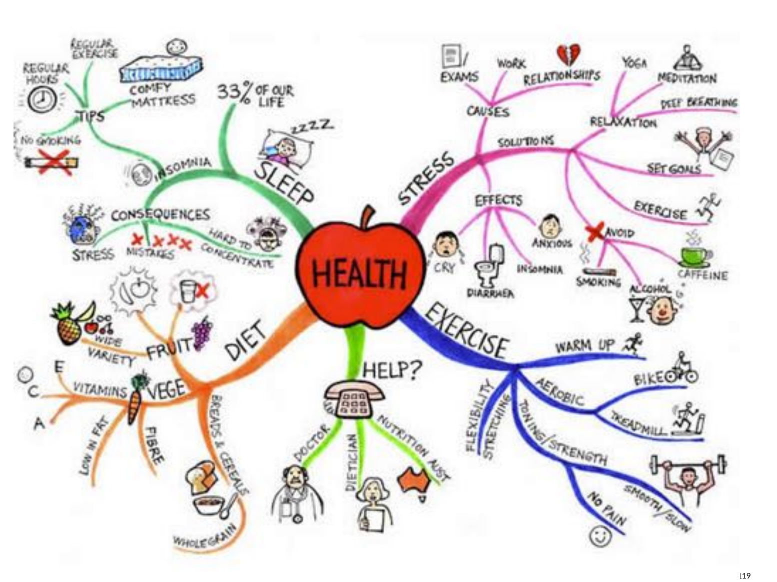
116

remove local optimization of planning with…

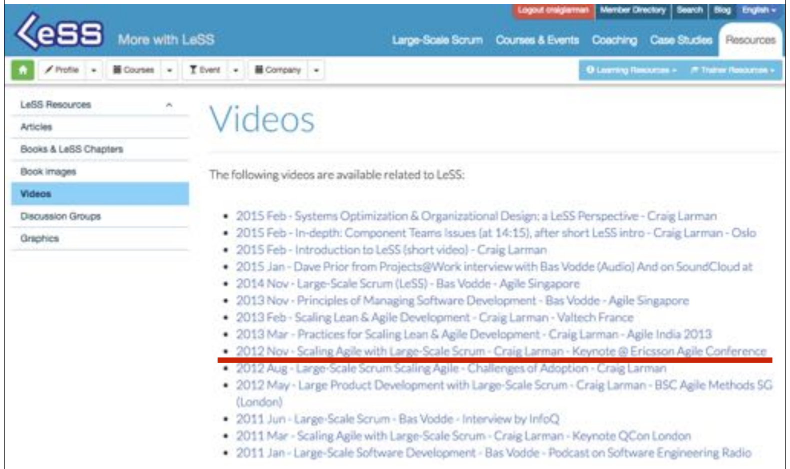
ship every Sprint, with adaptive planning steered by a business-side Product Owner

learn & educate

why



want to see the explanation again?



Videos

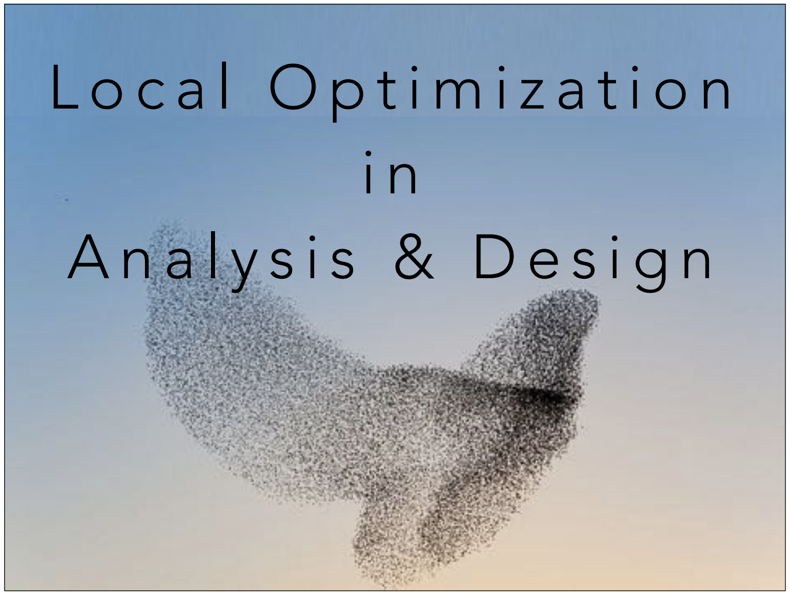
The following videos are available related to LeSS:

- 2015 Feb - Systems Optimization & Organizational Design: a LeSS Perspective - Craig Larman
- 2015 Feb - In-depth: Component Teams Issues (at 14:15), after short LeSS intro - Craig Larman - Oslo
- 2015 Feb - Introduction to LeSS (short video) - Craig Larman
- 2015 Jan - Dave Prior from Projects@Work interview with Bas Vodde (Audio) And on SoundCloud at
- 2014 Nov - Large-Scale Scrum (LeSS) - Bas Vodde - Agile Singapore
- 2013 Nov - Principles of Managing Software Development - Bas Vodde - Agile Singapore
- 2013 Feb - Scaling Lean & Agile Development - Craig Larman - Valtech France
- 2013 Mar - Practices for Scaling Lean & Agile Development - Craig Larman - Agile India 2013
- 2012 Nov - Scaling Agile with Large-Scale Scrum - Craig Larman - Keynote @ Ericsson Agile Conference
- 2012 Aug - Large-Scale Scrum Scaling Agile - Challenges of Adoption - Craig Larman
- 2012 May - Large Product Development with Large-Scale Scrum - Craig Larman - BSC Agile Methods SG (London)
- 2011 Jun - Large-Scale Scrum - Bas Vodde - Interview by InfoQ
- 2011 Mar - Scaling Agile with Large-Scale Scrum - Craig Larman - Keynote QCon London
- 2011 Jan - Large-Scale Software Development - Bas Vodde - Podcast on Software Engineering Radio

video

Contract Game & Experts

121



Local Optimization in Analysis & Design

122

what are we about to learn?

123

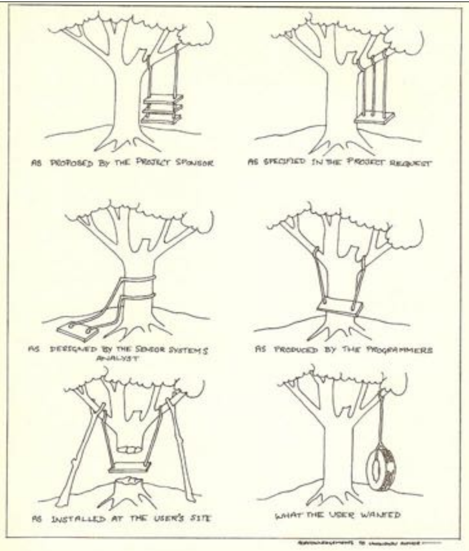
Lean Wastes

1. **Over-production**—of intermediate, WIP, or finished things; sooner, faster, greater than demand
2. **Inventory**—intermediate, WIP, or finished things
3. **Over-processing**—& extra processes, rediscovery
4. **Conveyance**—& handoff
5. **Motion**—& task switching
6. **Waiting**—& delay
7. **Defects & finding/correcting**—tasks to find & correct: test, inspect, review, modify
8. **Not using people's full potential**—working to title, not multi-skilling
9. **Knowledge/information scatter/loss**—& connection to handoff & inventory & rediscovery; communication barriers: indirection, 1-way flows
10. **Wishful thinking**—[design/spec/estimate] is correct, learning & feedback is only mildly important, what said=what heard, …

124

EXERCISE

- team: standing: list
  - **why** are these waste in the eyes of paying customers or investors?
    - Over-production
    - Inventory
    - Over-processing
    - Conveyance—& handoff
    - Motion—& task switching
    - Waiting—& delay
    - Defects & finding/correcting
    - Not using people's full potential
    - Knowledge/information scatter/loss
    - Wishful thinking
  - coach: review answers


AS PROPOSED BY THE PROJECT SPONSOR
AS SPECIFIED IN THE PROJECT REQUEST
AS DESIGNED BY THE SENIOR SYSTEMS ANALYST
AS PRODUCED BY THE PROGRAMMERS
AS INSTALLED AT THE USER'S SITE
WHAT THE USER WANTED

From the *University of London Computer Centre Newsletter* No. 53, March 1973

most of you have seen this cartoon

notice the date

what does this say about what was widely understood even by that time?

have you seen this mistake even today? why does it persist?

Lean Wastes

1. **Over-production**—of intermediate, WIP, or finished things; sooner, faster, greater than demand
2. **Inventory**—intermediate, WIP, or finished things
3. **Over-processing**—& extra processes, rediscovery
4. **Conveyance**—& handoff
5. **Motion**—& task switching
6. **Waiting**—& delay
7. **Defects & correction**—tasks to find & correct: test, inspect, review, modify
8. **Not using people's full potential**—working to title, not multi-skilling
9. **Knowledge/information scatter/loss**—& connection to handoff & inventory & rediscovery; communication barriers: indirection, 1-way flows
10. **Wishful thinking**—[design/spec/estimate] is correct, learning & feedback is only mildly important, what said=what heard, …

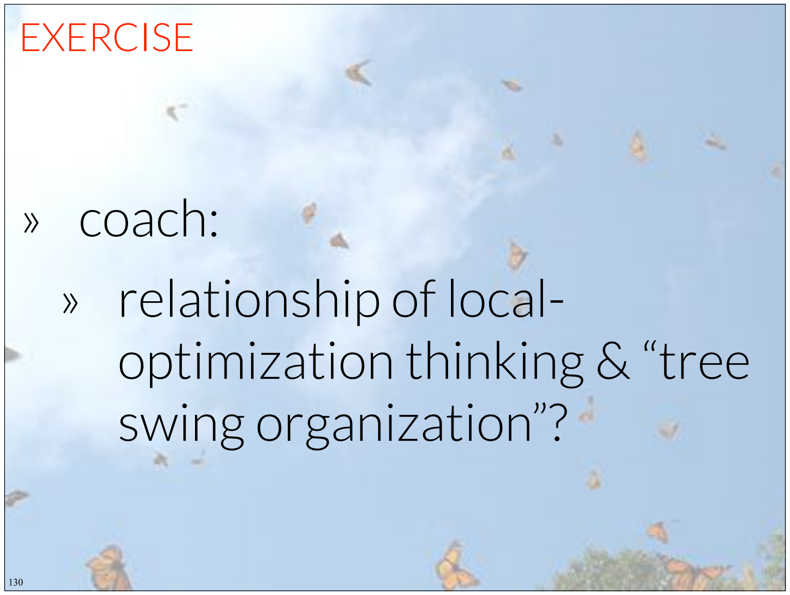
EXERCISE

- team: at wall/flipchart
  - write: what lean wastes are implied by the cartoon?

EXERCISE

- » coach:
  - » relationship of local-optimization thinking & wastes?

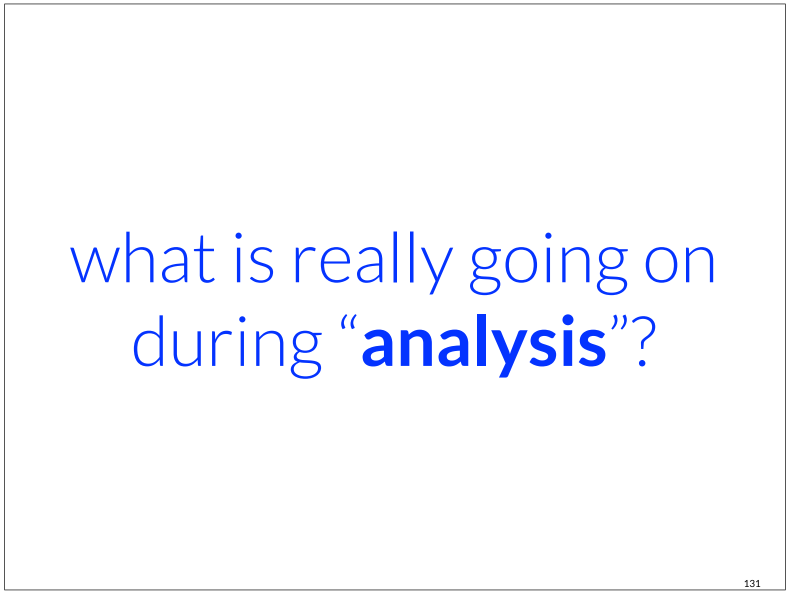
129



EXERCISE

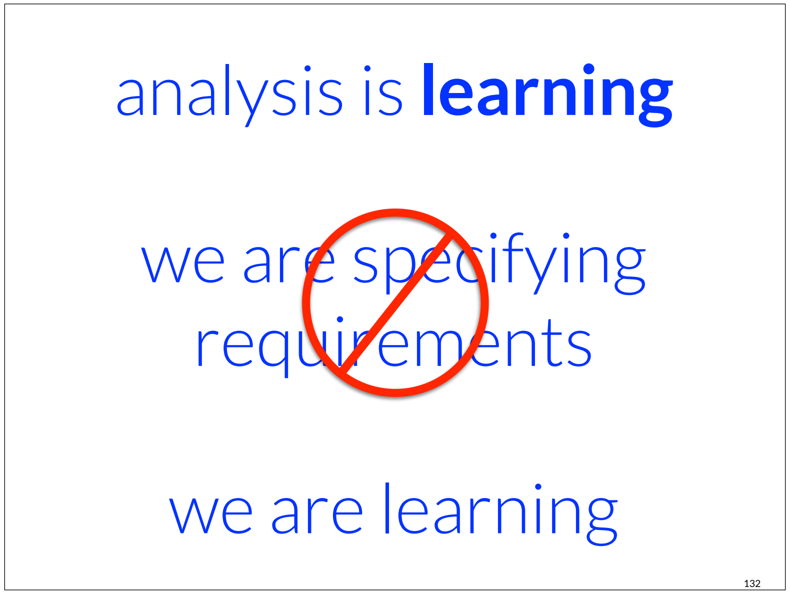
- » coach:
  - » relationship of local-optimization thinking & “tree swing organization”?

130



what is really going on during “**analysis**”?

131



analysis is **learning**

~~we are specifying requirements~~

we are learning

132

EXERCISE

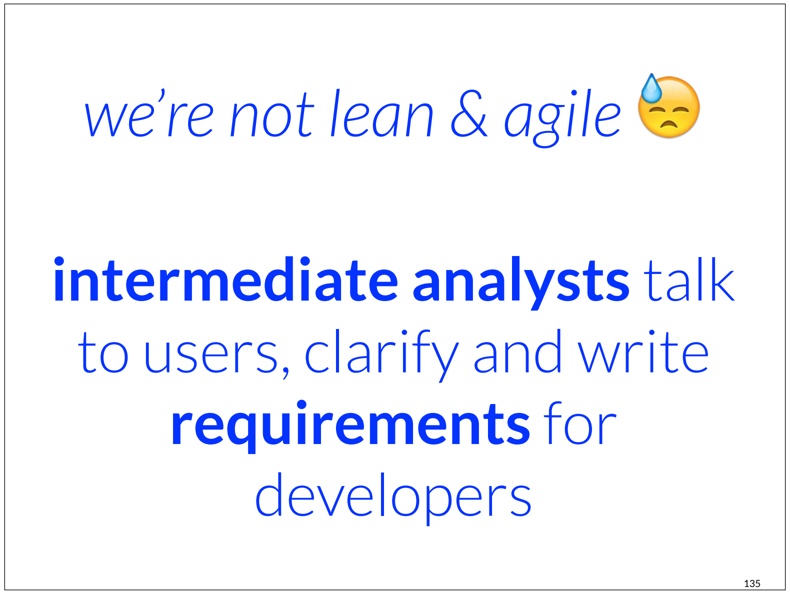
- coach:
  - what **role** ultimately hands-on **applies** the learning to **create running software**?

133

“intermediate analysts talk to users, clarify and write requirements for developers”

local optimization
many wastes

134



we’re not lean & agile 😓

intermediate analysts talk to users, clarify and write **requirements** for developers

135

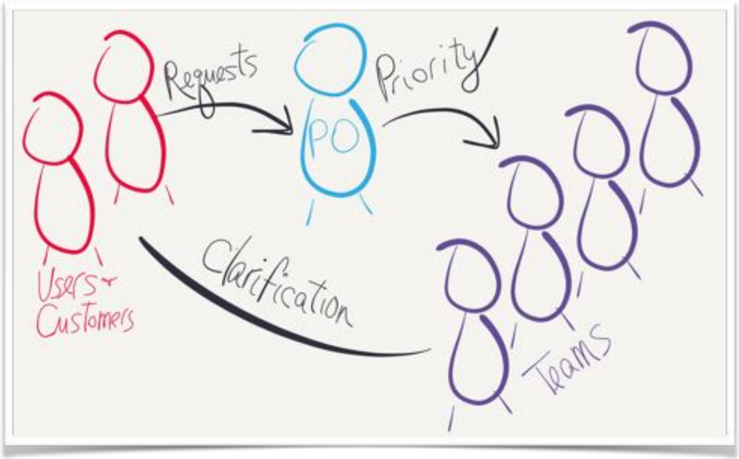
now we’re lean & agile! 😅

intermediate ~~analysts~~ **Product Owners** talk to users, clarify and write ~~requirements~~ **stories** for developers

136

therefore…

Clarification vs Prioritization



remove local optimization of analysis with…

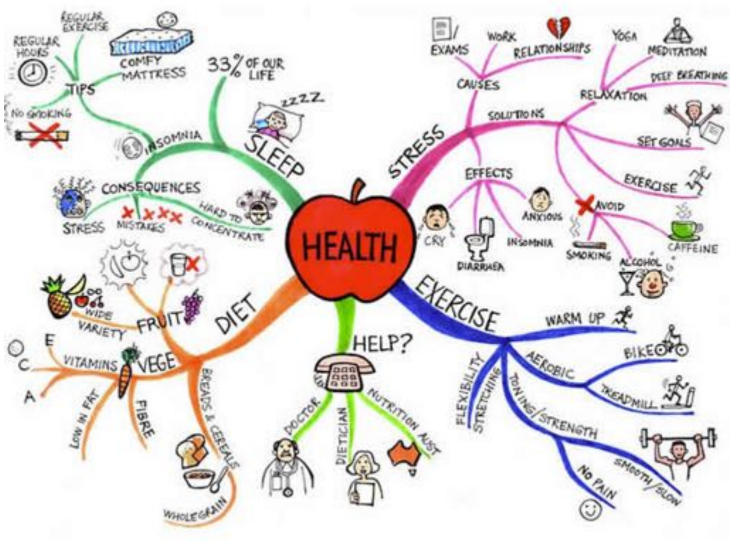
hands-on developers learning/clarifying with hands-on users

remove local optimization of architecture/design with…

hands-on developers doing architecture/design

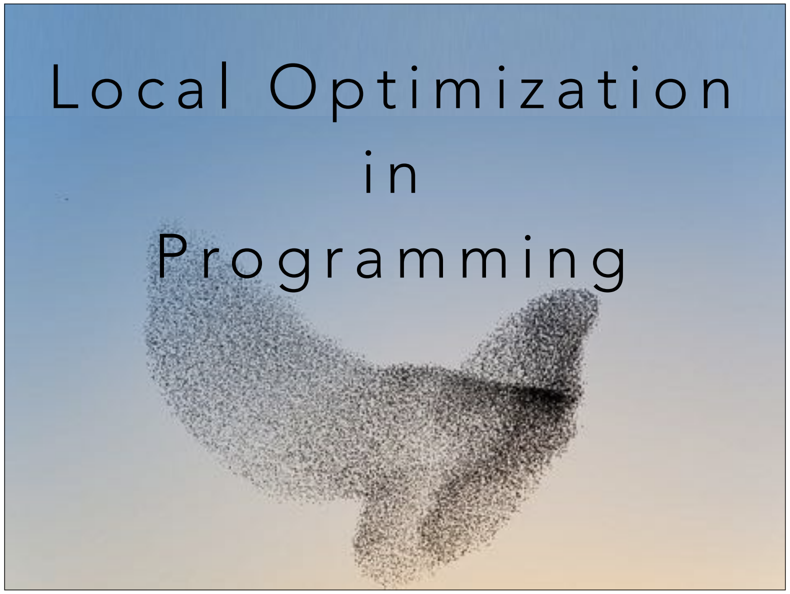
learn & educate

why



video

[the analyst manger]



Local Optimization in Programming

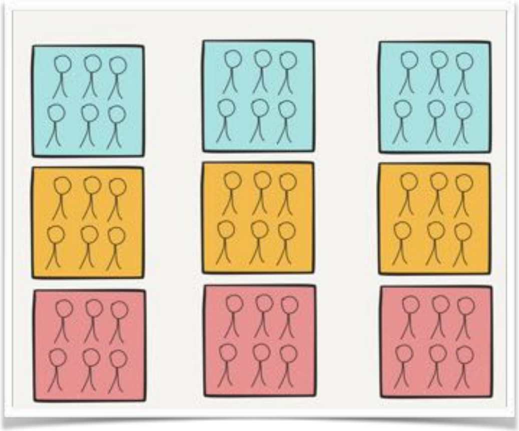
what are we
about to learn?

145

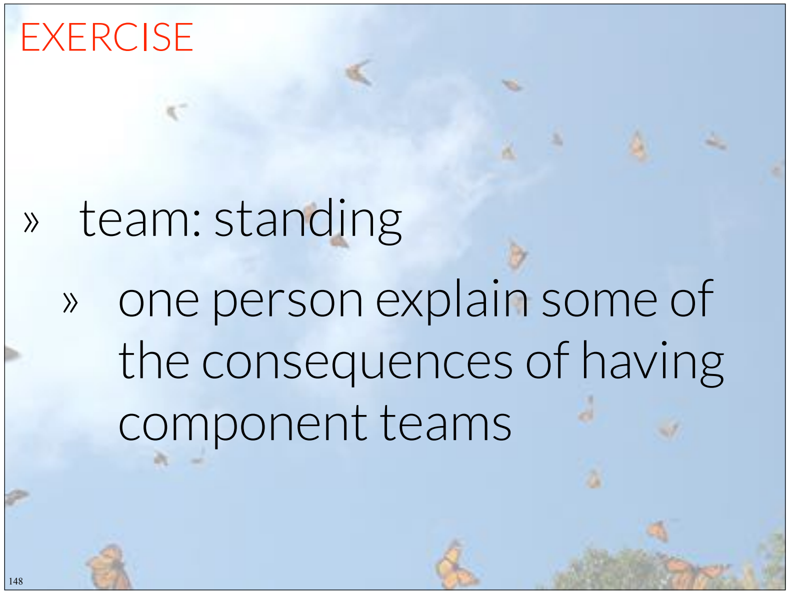
the One True
system optimizing goal?

146

Component Teams



147

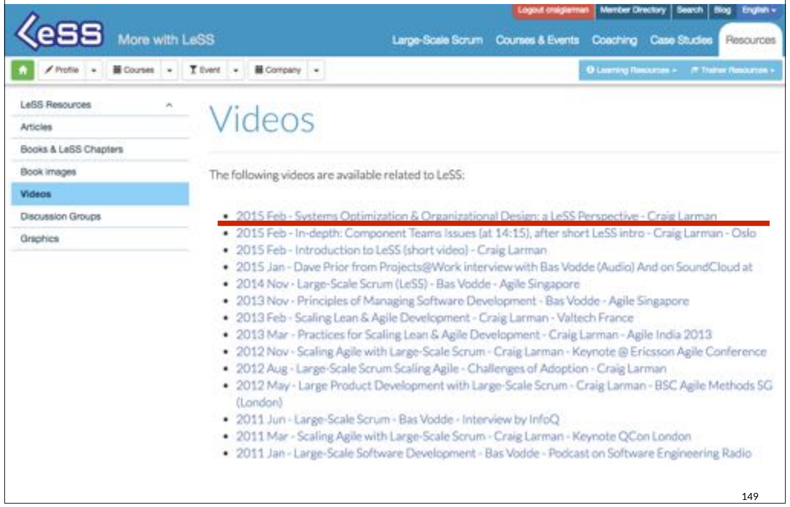


EXERCISE

- team: standing
  - one person explain some of the consequences of having component teams

148

want to see the explanation again?



Videos

The following videos are available related to LeSS:

- 2015 Feb - Systems Optimization & Organizational Design: a LeSS Perspective - Craig Larman
- 2015 Feb - In-depth: Component Teams Issues (at 14:15), after short LeSS intro - Craig Larman - Oslo
- 2015 Feb - Introduction to LeSS (short video) - Craig Larman
- 2015 Jan - Dave Prior from Projects@Work interview with Bas Vodde (Audio) And on SoundCloud at
- 2014 Nov - Large-Scale Scrum (LeSS) - Bas Vodde - Agile Singapore
- 2013 Nov - Principles of Managing Software Development - Bas Vodde - Agile Singapore
- 2013 Feb - Scaling Lean & Agile Development - Craig Larman - Valtech France
- 2013 Mar - Practices for Scaling Lean & Agile Development - Craig Larman - Agile India 2013
- 2012 Nov - Scaling Agile with Large-Scale Scrum - Craig Larman - Keynote @ Ericsson Agile Conference
- 2012 Aug - Large-Scale Scrum Scaling Agile - Challenges of Adoption - Craig Larman
- 2012 May - Large Product Development with Large-Scale Scrum - Craig Larman - BSC Agile Methods SG (London)
- 2011 Jun - Large-Scale Scrum - Bas Vodde - Interview by InfoQ
- 2011 Mar - Scaling Agile with Large-Scale Scrum - Craig Larman - Keynote QCon London
- 2011 Jan - Large-Scale Software Development - Bas Vodde - Podcast on Software Engineering Radio

149

149

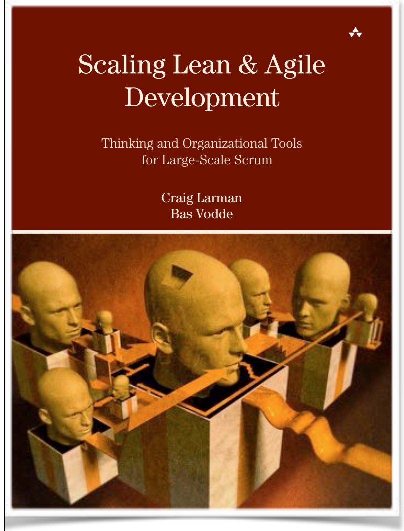

Scaling Lean & Agile Development
Thinking and Organizational Tools for Large-Scale Scrum
Craig Larman
Bas Vodde

Table of Contents

1. Introduction

Thinking Tools

2. Systems Thinking
3. Lean Thinking
4. Queueing Theory
5. False Dichotomies
6. Be Agile

Organizational Tools

7. Feature Teams
8. Teams
9. Requirement Areas
10. Organization
11. Large-Scale Scrum

150

remove local optimizations of programming with...

feature teams coding cross-components

151

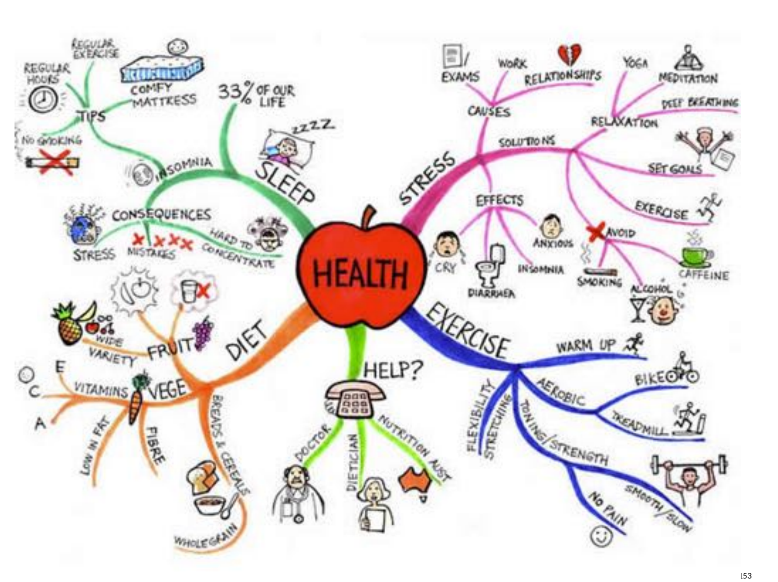
151

learn & educate

why

152

152



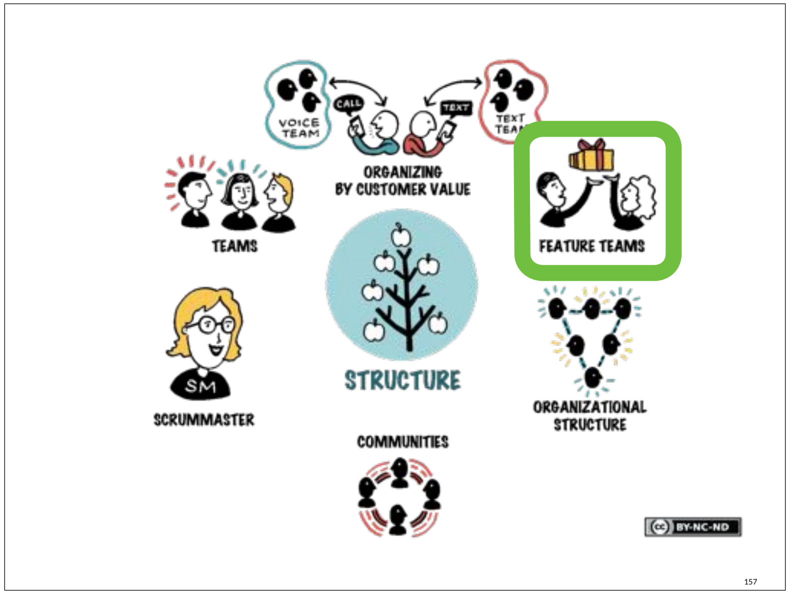
Organizing by Customer Value: Feature Teams

what are we
about to learn?

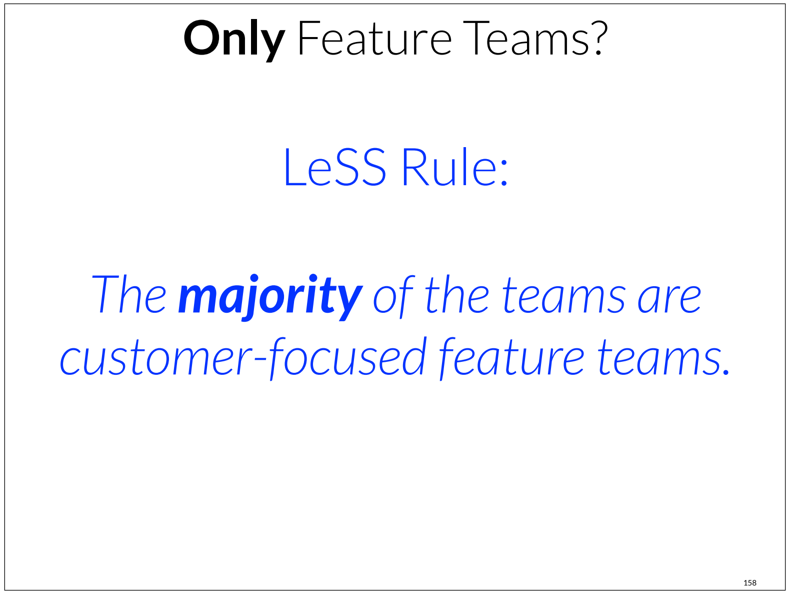
Descaling with LeSS

remove local optimizations of single-specialist groups with

feature teams



157

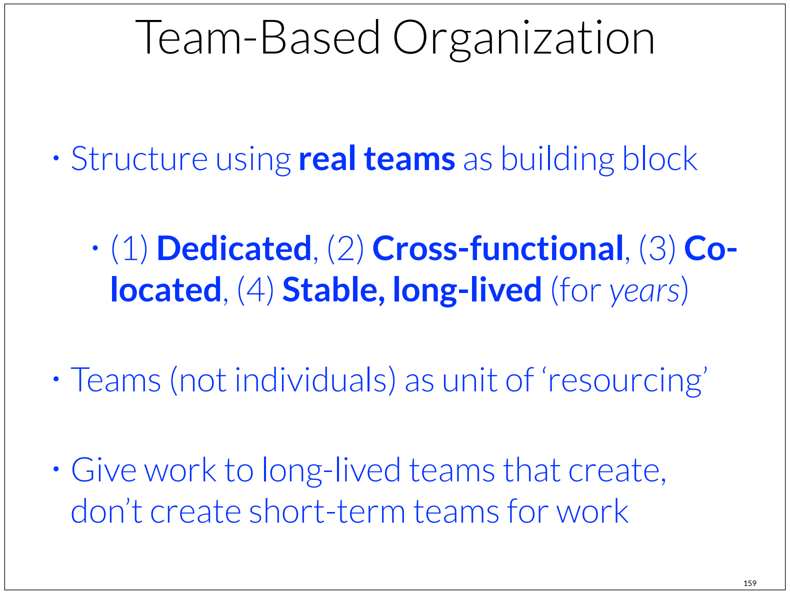


Only Feature Teams?

LeSS Rule:

*The **majority** of the teams are customer-focused feature teams.*

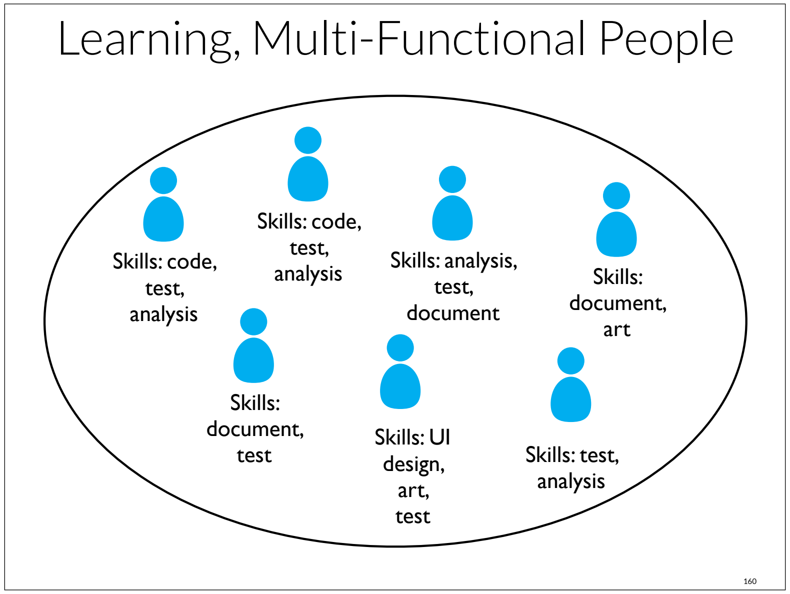
158



Team-Based Organization

- Structure using **real teams** as building block
  - (1) **Dedicated**, (2) **Cross-functional**, (3) **Co-located**, (4) **Stable, long-lived** (for *years*)
- Teams (not individuals) as unit of ‘resourcing’
- Give work to long-lived teams that create, don’t create short-term teams for work

159



Learning, Multi-Functional People

Skills: code, test, analysis

Skills: code, test, analysis

Skills: analysis, test, document

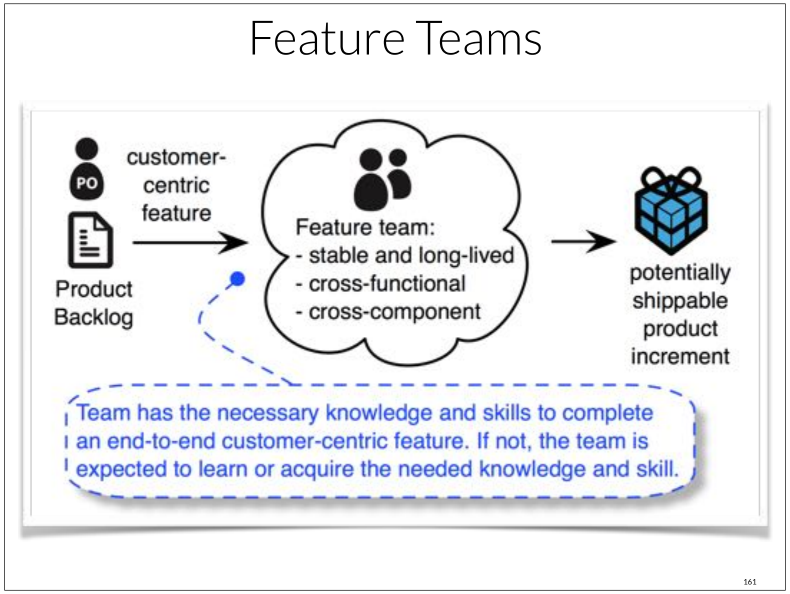
Skills: document, art

Skills: document, test

Skills: UI design, art, test

Skills: test, analysis

160



Feature Teams

customer-centric feature

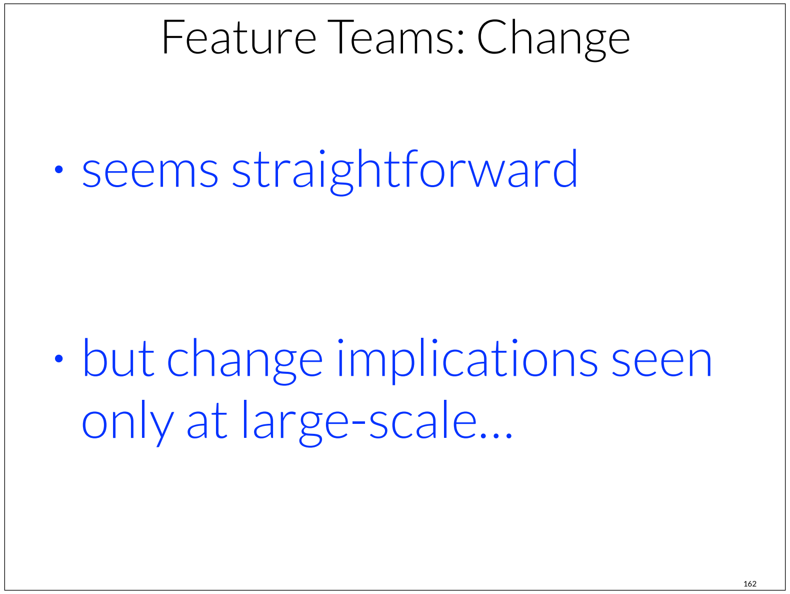
PO

Product Backlog

Feature team:
- stable and long-lived
- cross-functional
- cross-component

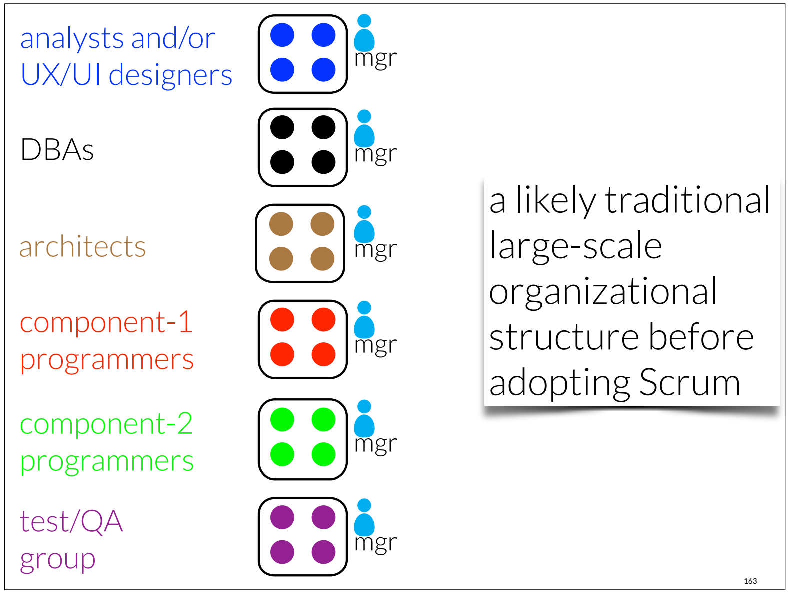
potentially shippable product increment

Team has the necessary knowledge and skills to complete an end-to-end customer-centric feature. If not, the team is expected to learn or acquire the needed knowledge and skill.



Feature Teams: Change

- seems straightforward
- but change implications seen only at large-scale...



analysts and/or UX/UI designers — mgr

DBAs — mgr

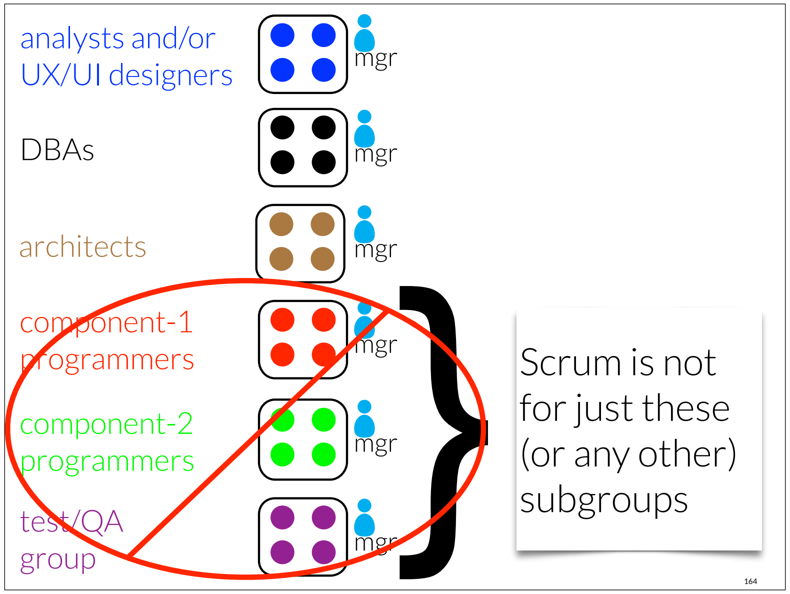
architects — mgr

component-1 programmers — mgr

component-2 programmers — mgr

test/QA group — mgr

a likely traditional large-scale organizational structure before adopting Scrum



analysts and/or UX/UI designers — mgr

DBAs — mgr

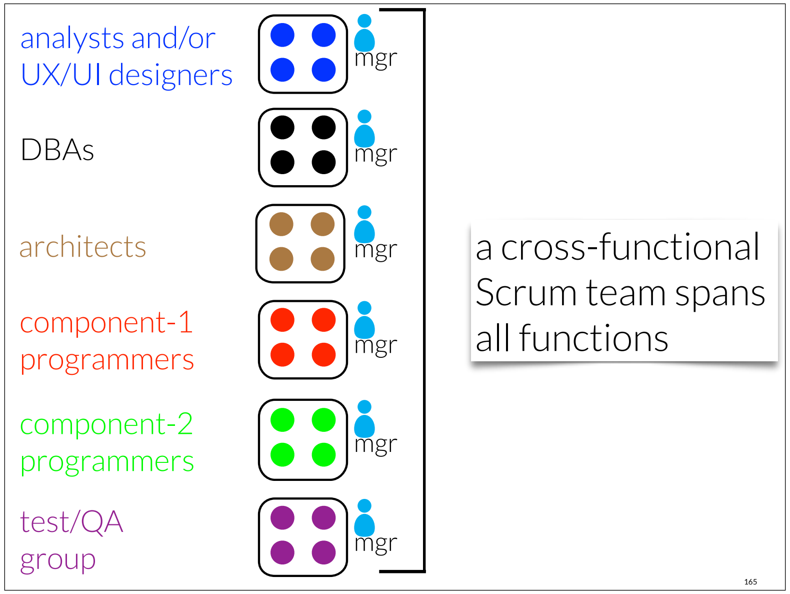
architects — mgr

component-1 programmers — mgr

component-2 programmers — mgr

test/QA group — mgr

Scrum is not for just these (or any other) subgroups



analysts and/or UX/UI designers

DBAs

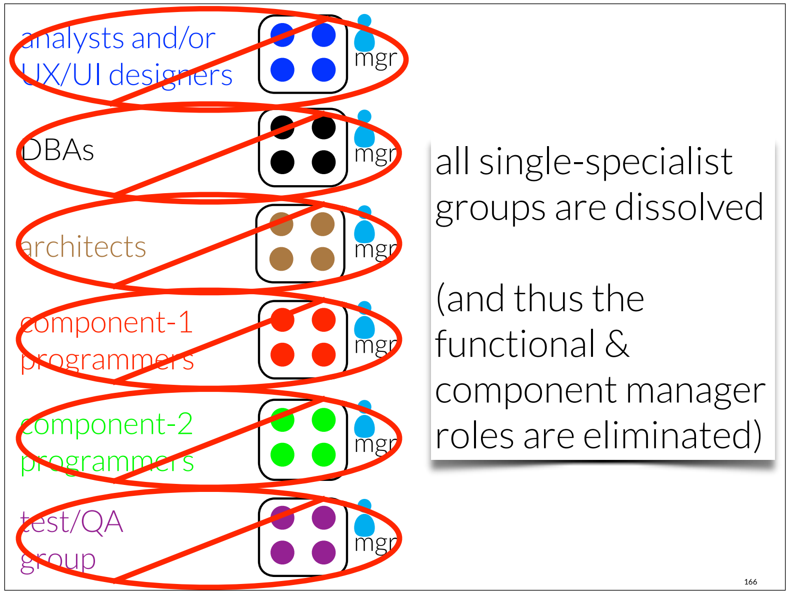
architects

component-1 programmers

component-2 programmers

test/QA group

a cross-functional Scrum team spans all functions



analysts and/or UX/UI designers

DBAs

architects

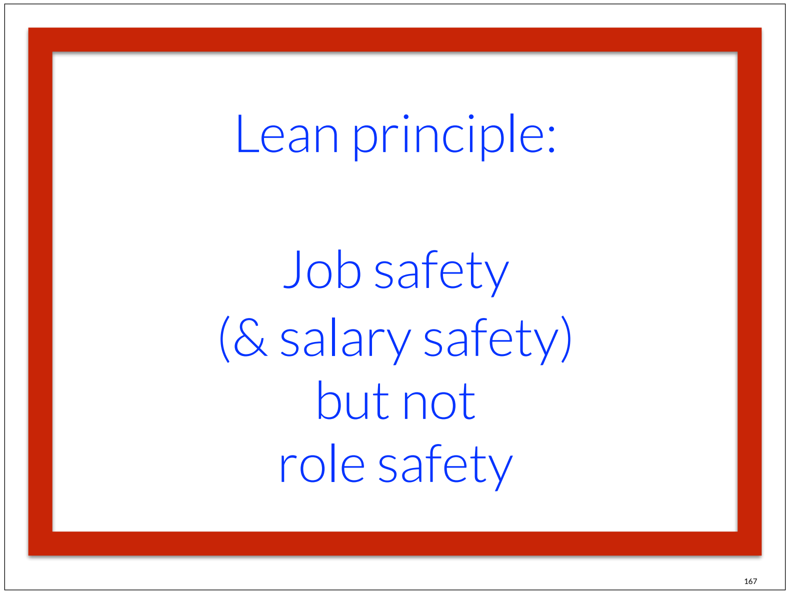
component-1 programmers

component-2 programmers

test/QA group

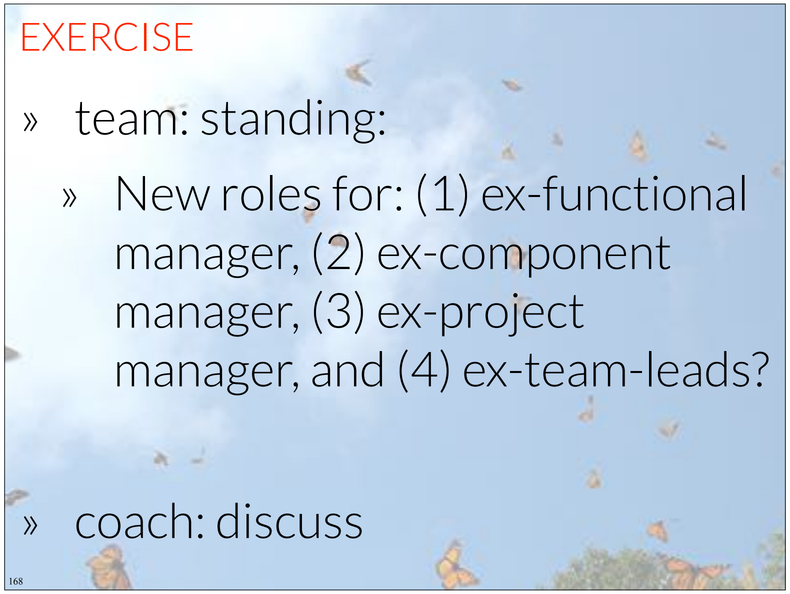
all single-specialist groups are dissolved

(and thus the functional & component manager roles are eliminated)



Lean principle:

Job safety
(& salary safety)
but not
role safety



EXERCISE

- team: standing:
  - New roles for: (1) ex-functional manager, (2) ex-component manager, (3) ex-project manager, and (4) ex-team-leads?
- coach: discuss

Only Title: (Product) Developer

Scrum Guide:

“Scrum recognizes no titles for Development Team members other than Developer, regardless of the work being performed by the person; **there are no exceptions to this rule**.”

169

169

EXCEPTION

» coach: Why?

170

170

don’t wait for
the org chart

171

171

Not a Team of Single-Specialists

Scrum Guide:

“Team does not contain sub-teams dedicated to particular domains such as testing or analysis”

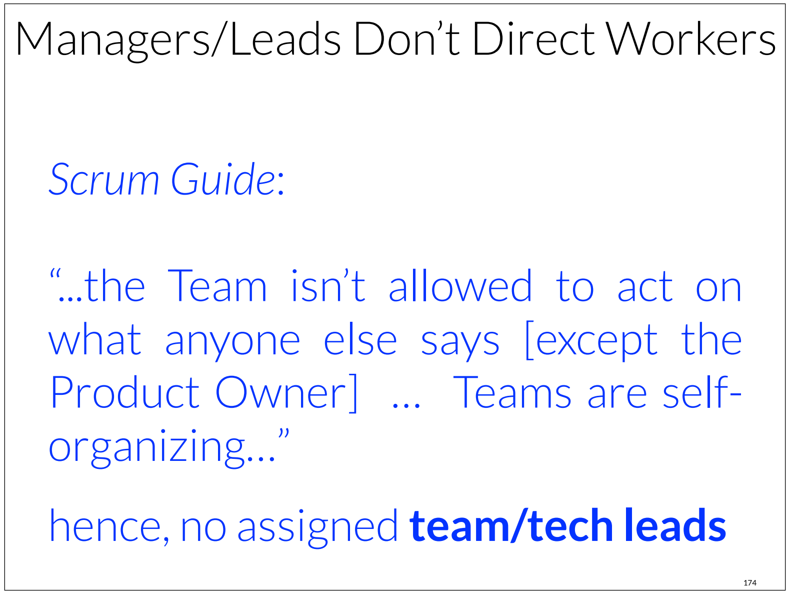
172

172

EXERCISE

» coach: Why?

173



Managers/Leads Don't Direct Workers

Scrum Guide:

"...the Team isn't allowed to act on what anyone else says [except the Product Owner] ... Teams are self-organizing..."

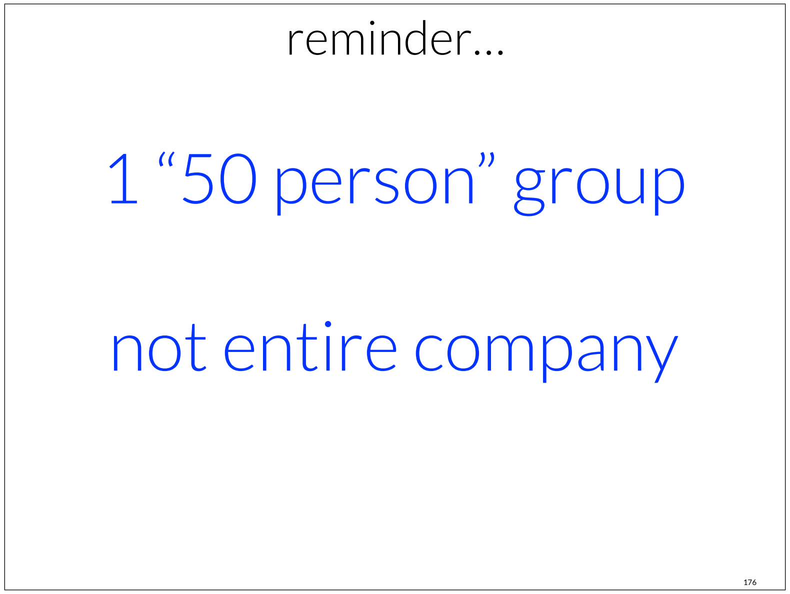
hence, no assigned **team/tech leads**

174

EXERCISE

» coach: Why?

175

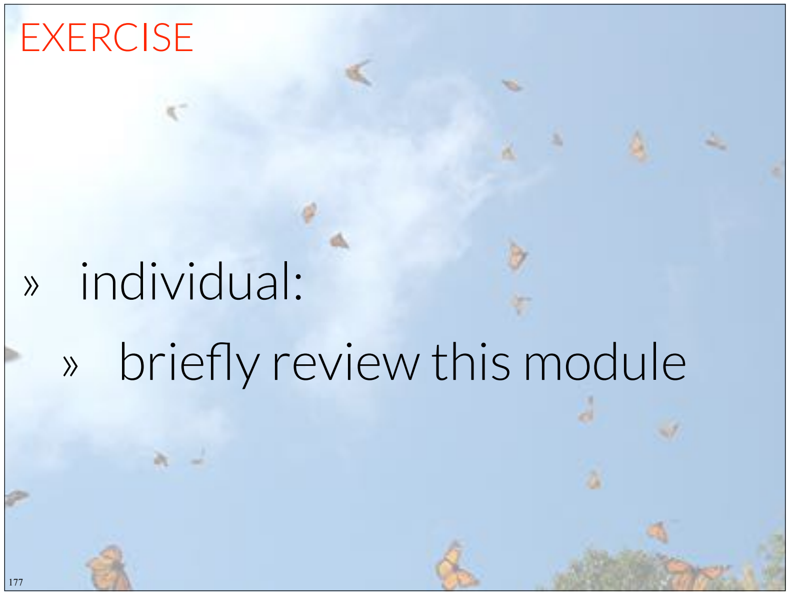


reminder...

1 "50 person" group

not entire company

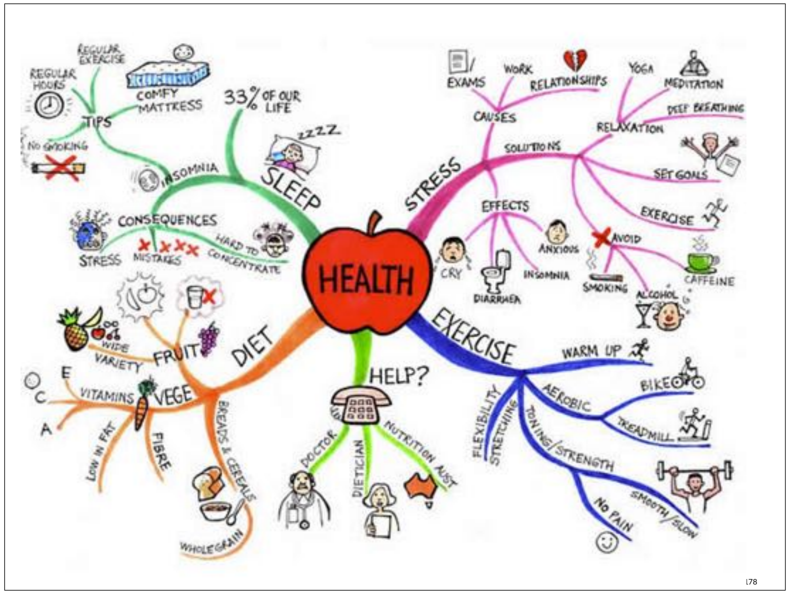
176

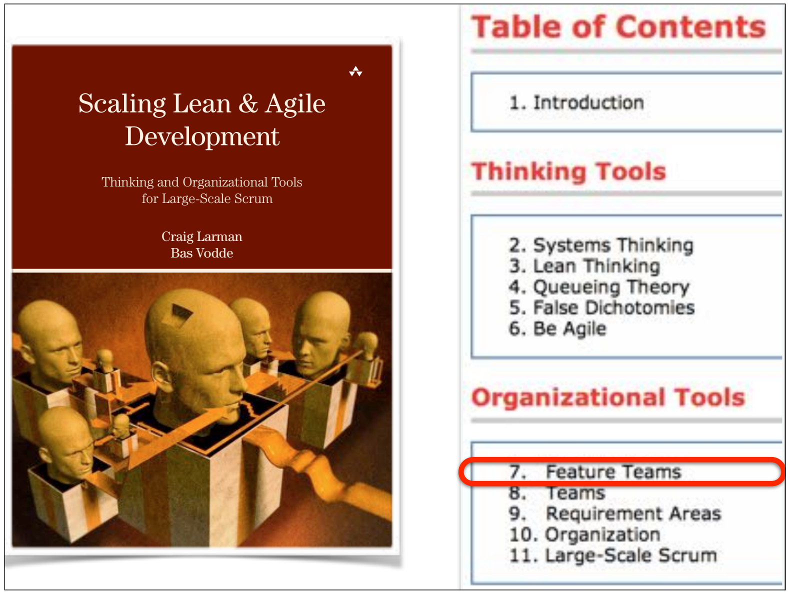


EXERCISE

» individual:

» briefly review this module





Scaling Lean & Agile Development

Thinking and Organizational Tools for Large-Scale Scrum

Craig Larman
Bas Vodde

Table of Contents

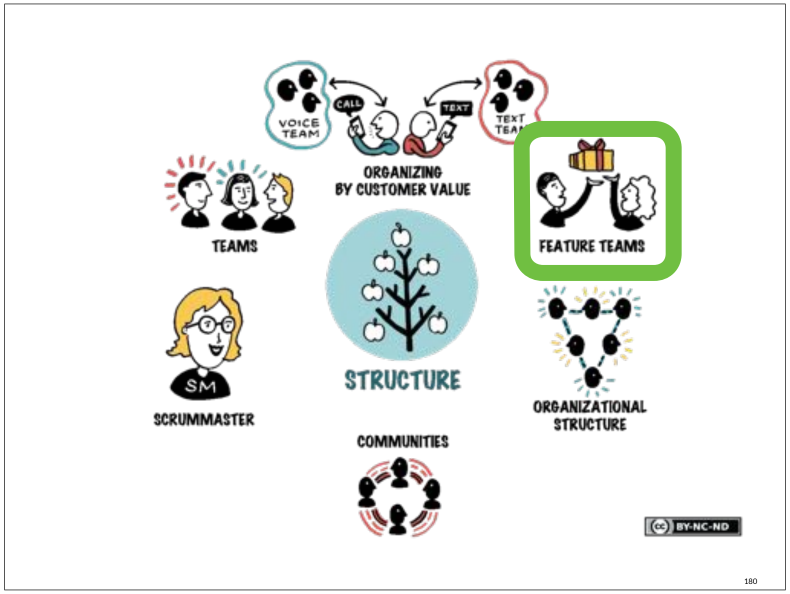
1. Introduction

Thinking Tools

2. Systems Thinking
3. Lean Thinking
4. Queueing Theory
5. False Dichotomies
6. Be Agile

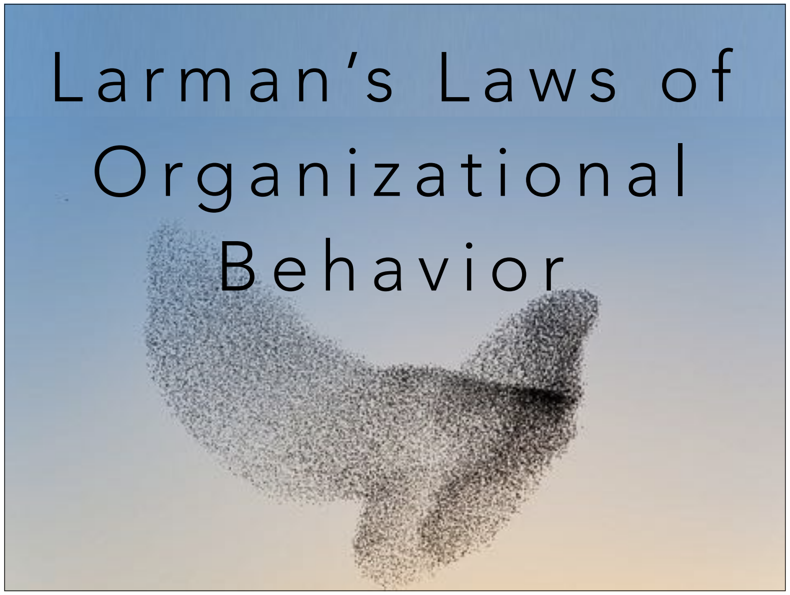
Organizational Tools

7. Feature Teams
8. Teams
9. Requirement Areas
10. Organization
11. Large-Scale Scrum



video

team work



Larman's Laws of Organizational Behavior

why so much?...

fake **Lean**

fake **Scrum**

fake **Kanban**

fake **DevOps**

fake **AnyChangeIdea**

Larman's 4 Laws of Organizational Behavior

1. Organizations are implicitly optimized to avoid changing the status quo middle- and first-level manager and "specialist" positions & power structures.
2. As a corollary to (1), any change initiative will be reduced to overloading or redefining the new terminology to mean basically the same as status quo.
3. As a corollary to (1), any change initiative will be derided as "purist", "theoretical", and "needing pragmatic customization for local concerns" — which deflects from addressing weaknesses and manager/specialist status quo.
4. Culture follows structure (or culture follows system)

Culture follows Structure

- has change implications
- most applicable in large, established organizations
- conversely, in a start-up with 5 people, "**structure follows culture**" is a strong pattern

185

Lean Thinking

186

what are we about to learn?

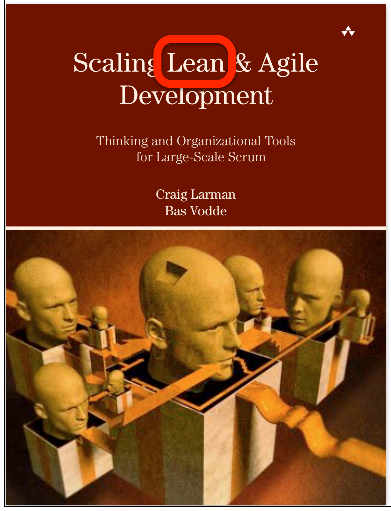
187

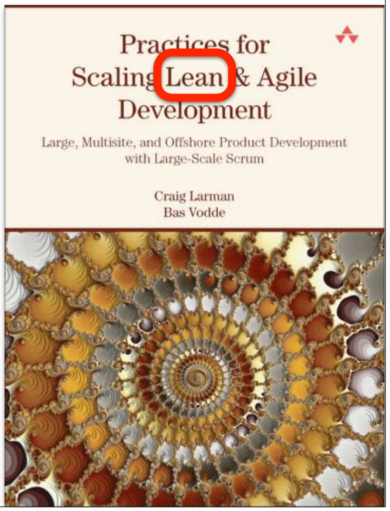
TOYOTA

188

served as lead coach of
lean development @ Xerox

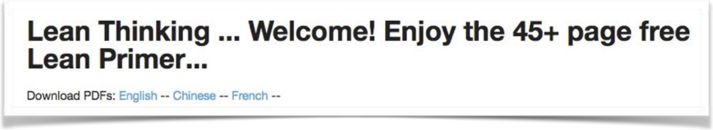
books…


Scaling Lean & Agile Development
Thinking and Organizational Tools for Large-Scale Scrum
Craig Larman
Bas Vodde


Practices for Scaling Lean & Agile Development
Large, Multisite, and Offshore Product Development with Large-Scale Scrum
Craig Larman
Bas Vodde

Co-author of…

leanprimer.com



Lean Thinking … Welcome! Enjoy the 45+ page free Lean Primer…

Download PDFs: English -- Chinese -- French --

essence? …

The essence of [the Toyota Way] is that each individual employee is given the opportunity to find problems in his own way of working, to solve them and to make improvements.
— Wakamatsu & Kondo

The root of the Toyota Way is to be dissatisfied with the status quo; you have to ask constantly, “Why are we doing this?”
— Watanabe (Toyota CEO)

Standards are not developed and then communicated from headquarters. ...We must let individuals decide what they will do to fix their problems and close gaps. We cannot have someone from corporate saying you need to do X, Y, Z, because this is completely contrary to Toyota problem solving.
— Toyota manager [LH08]

people/teams own
their own processes &
continuously
experiment with
improving them

~~conform to~~
~~centralized standards~~
~~& “best practices”~~

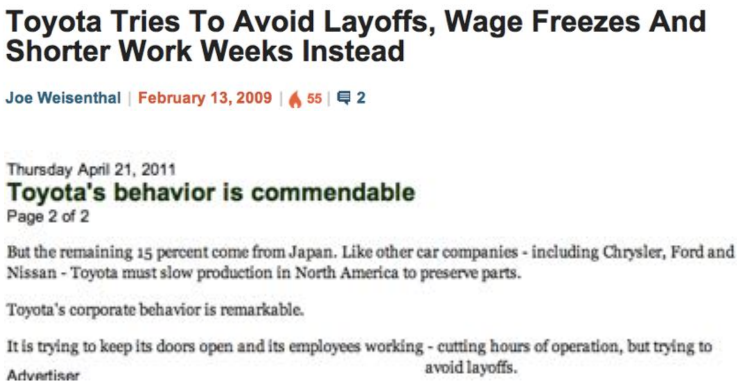
~~external “lean consultants”~~
~~who identify problems and~~
~~plan the solutions~~

essence? …

“Build people, then build products”

— classic Toyota quotes

“job safety, but not role safety”



Toyota Tries To Avoid Layoffs, Wage Freezes And Shorter Work Weeks Instead

Joe Weisenthal | February 13, 2009 | 55 | 2

Thursday April 21, 2011

Toyota's behavior is commendable

Page 2 of 2

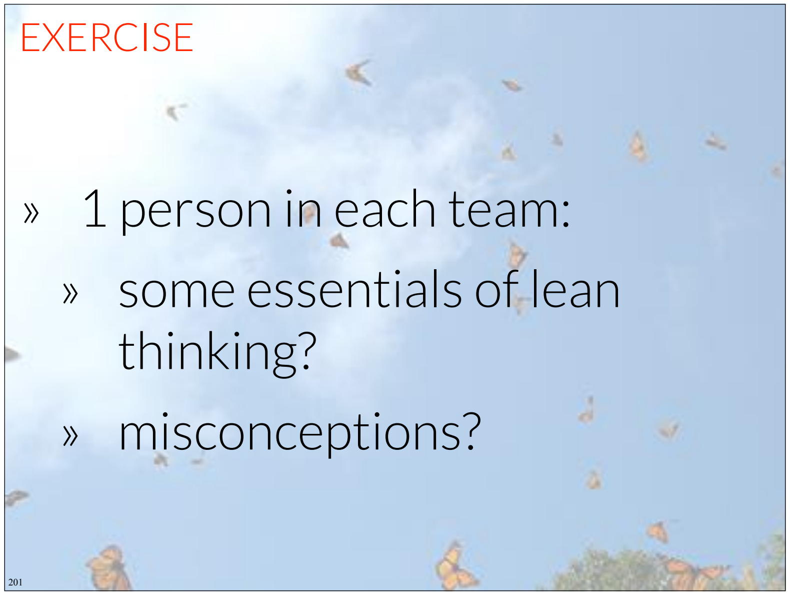
But the remaining 15 percent come from Japan. Like other car companies - including Chrysler, Ford and Nissan - Toyota must slow production in North America to preserve parts.

Toyota's corporate behavior is remarkable.

It is trying to keep its doors open and its employees working - cutting hours of operation, but trying to avoid layoffs.

Advertiser

~~laying people off~~



EXERCISE

- 1 person in each team:
  - some essentials of lean thinking?
  - misconceptions?

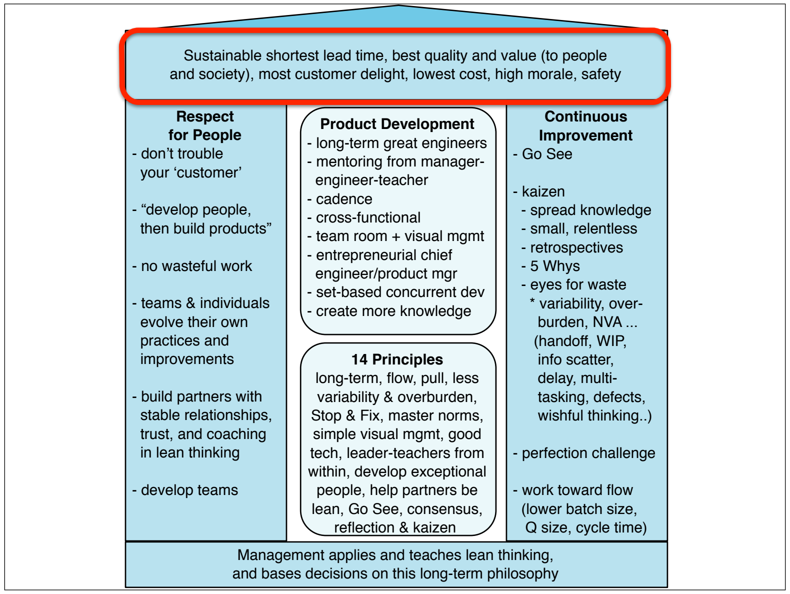
201

201

more? …

202

202



Sustainable shortest lead time, best quality and value (to people and society), most customer delight, lowest cost, high morale, safety

Respect for People
- don't trouble your 'customer'
- "develop people, then build products"
- no wasteful work
- teams & individuals evolve their own practices and improvements
- build partners with stable relationships, trust, and coaching in lean thinking
- develop teams

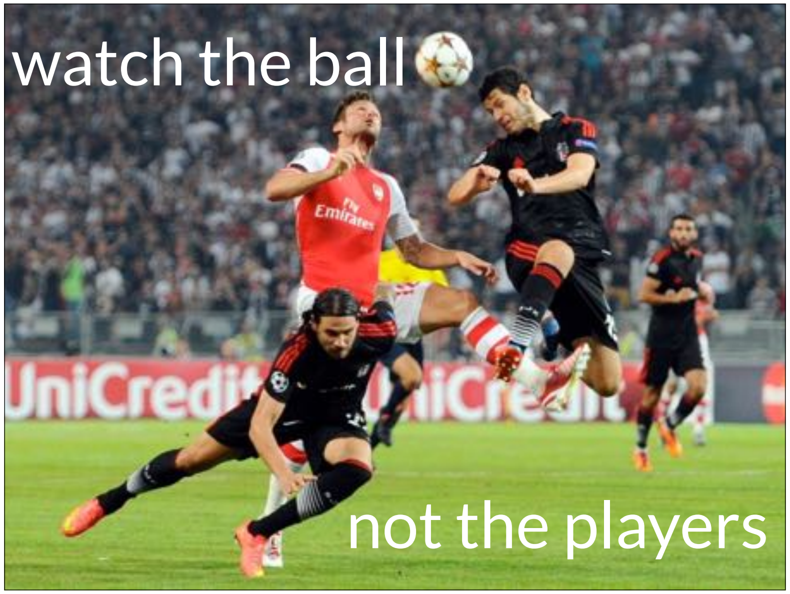
Product Development
- long-term great engineers
- mentoring from manager-engineer-teacher
- cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief engineer/product mgr
- set-based concurrent dev
- create more knowledge

14 Principles
long-term, flow, pull, less variability & overburden, Stop & Fix, master norms, simple visual mgmt, good tech, leader-teachers from within, develop exceptional people, help partners be lean, Go See, consensus, reflection & kaizen

Continuous Improvement
- Go See
- kaizen
  - spread knowledge
  - small, relentless
  - retrospectives
  - 5 Whys
  - eyes for waste
    * variability, over-burden, NVA ... (handoff, WIP, info scatter, delay, multi-tasking, defects, wishful thinking..)
- perfection challenge
- work toward flow (lower batch size, Q size, cycle time)

Management applies and teaches lean thinking, and bases decisions on this long-term philosophy

203



watch the ball

not the players

204

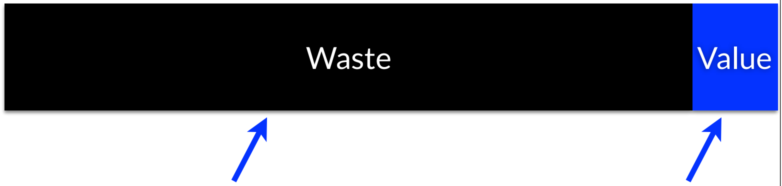
~~local optimization/ efficiency~~

value stream

perfection goal:
value flow to customer
without pause or impediment

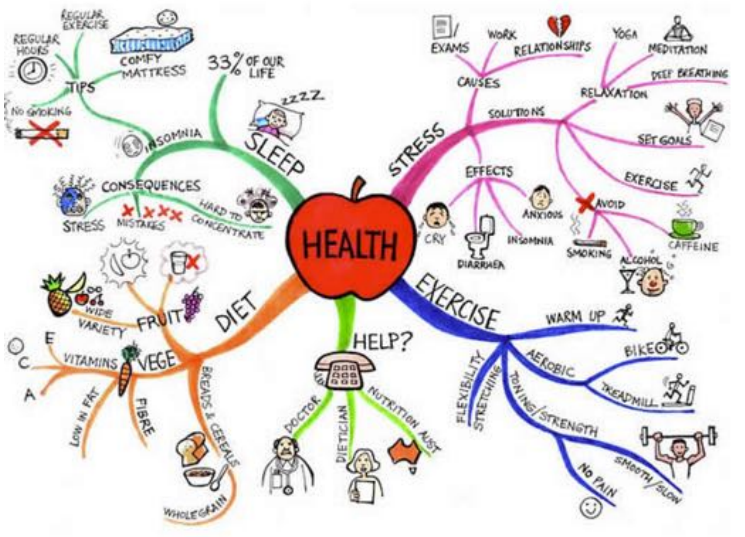
Better Value Stream Ratio? **Subtractive**

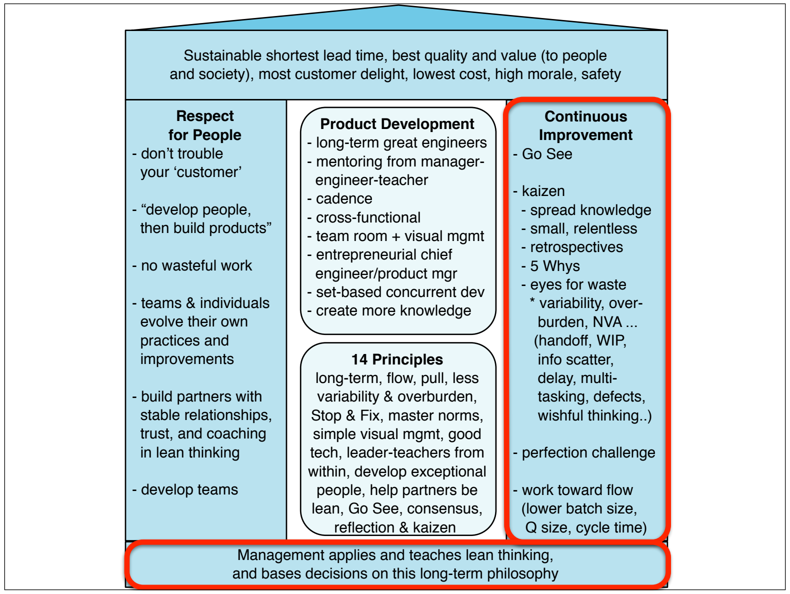
Total task time of worker



Opportunity:
REMOVE
Wastes

Traditional
Improvement
Effort





Sustainable shortest lead time, best quality and value (to people and society), most customer delight, lowest cost, high morale, safety

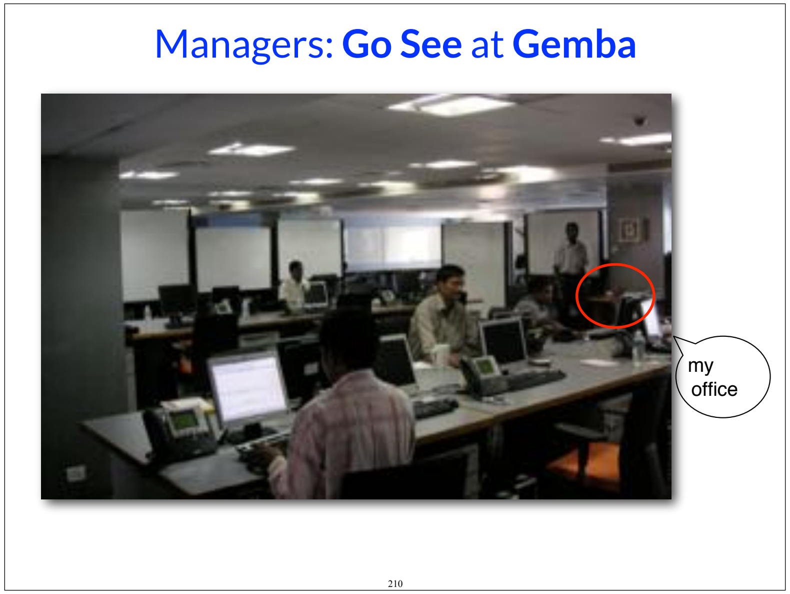
Respect for People
- don't trouble your 'customer'
- "develop people, then build products"
- no wasteful work
- teams & individuals evolve their own practices and improvements
- build partners with stable relationships, trust, and coaching in lean thinking
- develop teams

Product Development
- long-term great engineers
- mentoring from manager-engineer-teacher
- cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief engineer/product mgr
- set-based concurrent dev
- create more knowledge

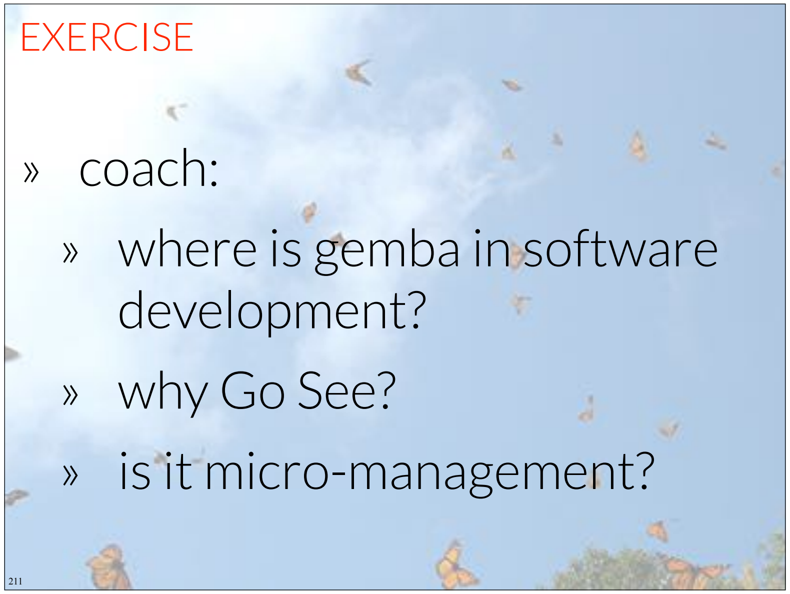
14 Principles
long-term, flow, pull, less variability & overburden, Stop & Fix, master norms, simple visual mgmt, good tech, leader-teachers from within, develop exceptional people, help partners be lean, Go See, consensus, reflection & kaizen

Continuous Improvement
- Go See
- kaizen
  - spread knowledge
  - small, relentless
  - retrospectives
  - 5 Whys
  - eyes for waste
    * variability, over-burden, NVA ... (handoff, WIP, info scatter, delay, multi-tasking, defects, wishful thinking..)
- perfection challenge
- work toward flow (lower batch size, Q size, cycle time)

Management applies and teaches lean thinking, and bases decisions on this long-term philosophy

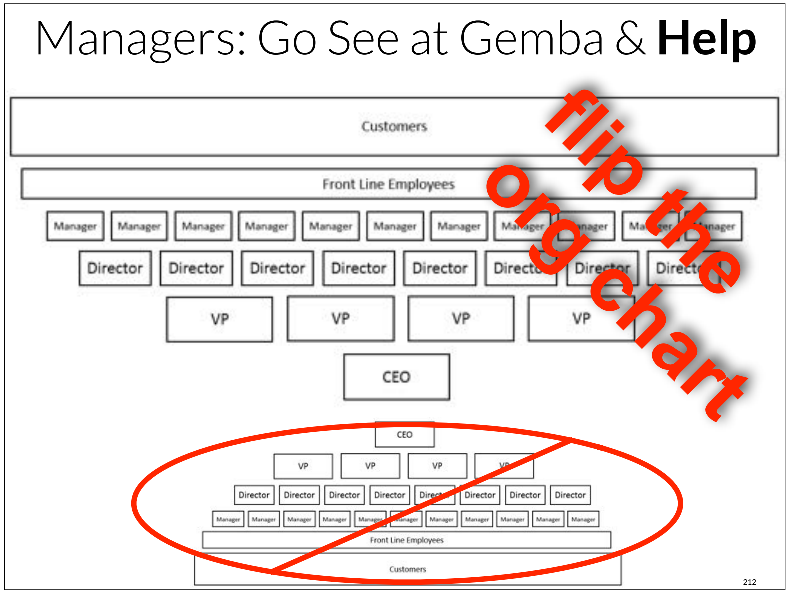


Managers: **Go See** at **Gemba**



EXERCISE

» coach:
  » where is gemba in software development?
  » why Go See?
  » is it micro-management?



Managers: Go See at Gemba & **Help**

Customers

Front Line Employees

Manager Manager Manager Manager Manager Manager Manager Manager Manager Manager Manager

Director Director Director Director Director Director Director Director

VP VP VP VP

CEO

flip the org chart

CEO

VP VP VP VP

Director Director Director Director Director Director Director Director

Manager Manager Manager Manager Manager Manager Manager Manager Manager Manager Manager

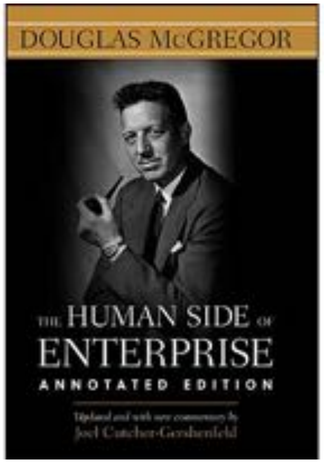
Front Line Employees

Customers

Theory Y Management Culture

"… good management requires a Theory Y orientation."

— MIT Sloan School of Management


DOUGLAS McGREGOR
THE HUMAN SIDE OF ENTERPRISE
ANNOTATED EDITION
Updated and with new commentary by
Joel Cutcher-Gershenfeld

Managers: Go See at Gemba & **Help**

Decoding the DNA of Toyota

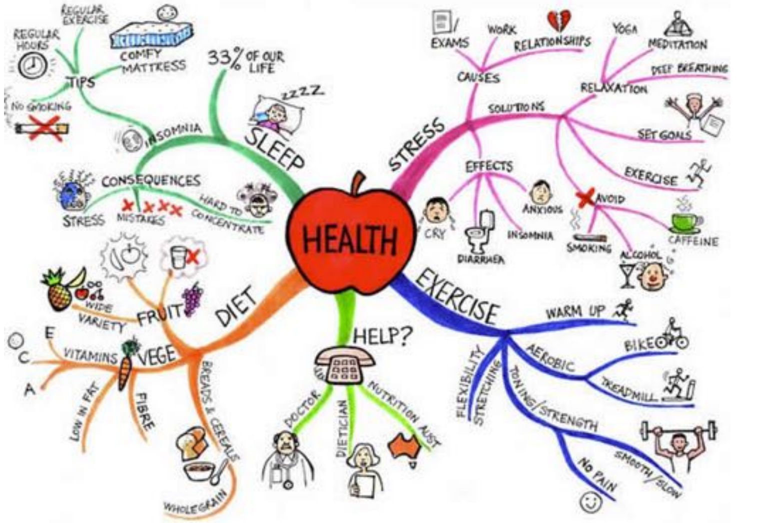
Harvard Business Review

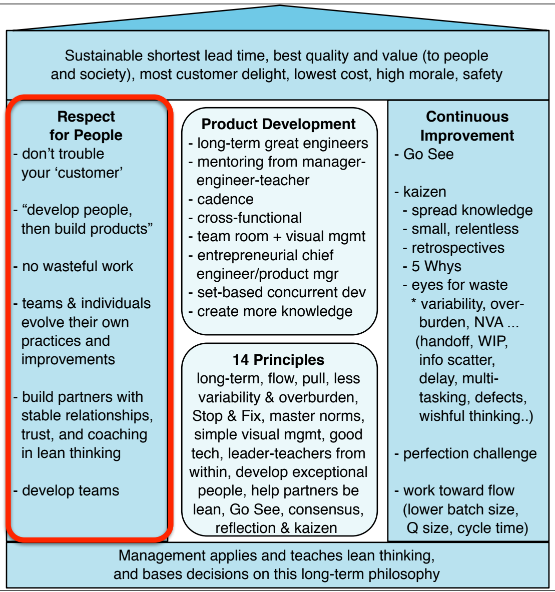
Sept. 1999

"Frontline workers make the improvements to their own jobs, and their supervisors provide assistance and teaching."

"That's why at these organizations all managers are expected to be able to do the jobs of everyone they supervise."

"My manager can do my job better than me."





Sustainable shortest lead time, best quality and value (to people and society), most customer delight, lowest cost, high morale, safety

Respect for People
- don't trouble your 'customer'
- "develop people, then build products"
- no wasteful work
- teams & individuals evolve their own practices and improvements
- build partners with stable relationships, trust, and coaching in lean thinking
- develop teams

Product Development
- long-term great engineers
- mentoring from manager-engineer-teacher
- cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief engineer/product mgr
- set-based concurrent dev
- create more knowledge

14 Principles
long-term, flow, pull, less variability & overburden, Stop & Fix, master norms, simple visual mgmt, good tech, leader-teachers from within, develop exceptional people, help partners be lean, Go See, consensus, reflection & kaizen

Continuous Improvement
- Go See
- kaizen
  - spread knowledge
  - small, relentless
  - retrospectives
  - 5 Whys
  - eyes for waste
    * variability, over-burden, NVA ... (handoff, WIP, info scatter, delay, multi-tasking, defects, wishful thinking..)
- perfection challenge
- work toward flow (lower batch size, Q size, cycle time)

Management applies and teaches lean thinking, and bases decisions on this long-term philosophy

People: Experiment, Learn, Many Skills; not “Resources”

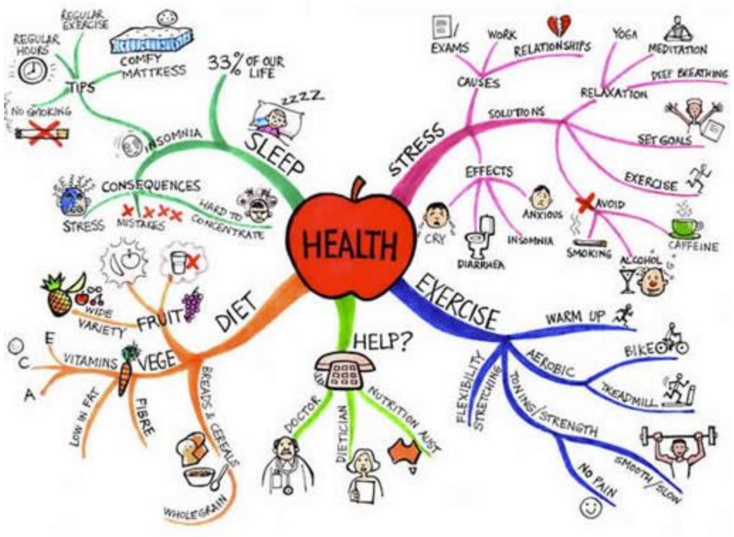
vs

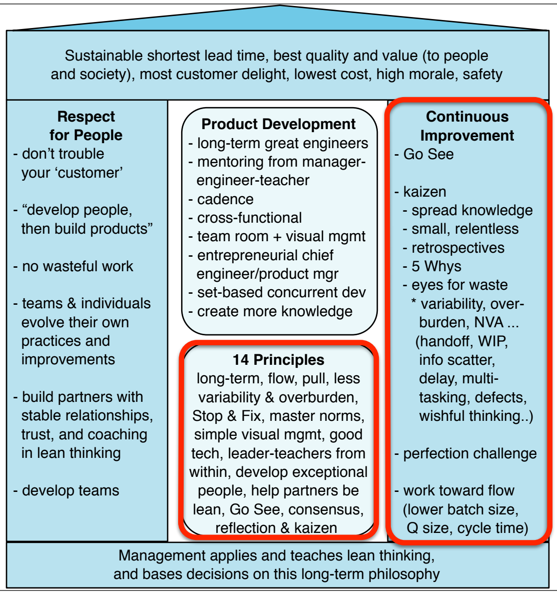
“Resources”

Respect for People

Good Thinking, Good Products
品質と効率
Quality and Efficiency
品质与效率

Job Safety
(but not role safety)





Sustainable shortest lead time, best quality and value (to people and society), most customer delight, lowest cost, high morale, safety

Respect for People
- don't trouble your 'customer'

- “develop people, then build products”

- no wasteful work

- teams & individuals evolve their own practices and improvements

- build partners with stable relationships, trust, and coaching in lean thinking

- develop teams

Product Development
- long-term great engineers
- mentoring from manager-engineer-teacher
- cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief engineer/product mgr
- set-based concurrent dev
- create more knowledge

14 Principles
long-term, flow, pull, less variability & overburden, Stop & Fix, master norms, simple visual mgmt, good tech, leader-teachers from within, develop exceptional people, help partners be lean, Go See, consensus, reflection & kaizen

Continuous Improvement
- Go See

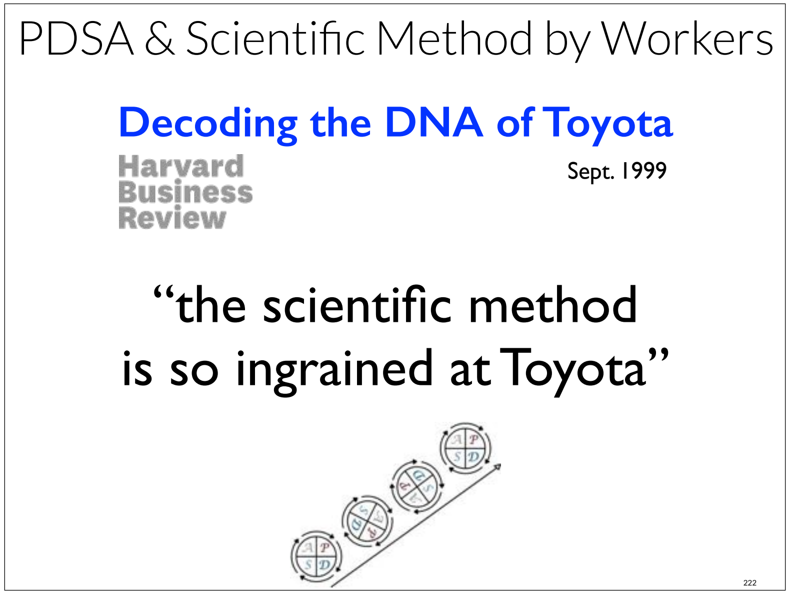
- kaizen
  - spread knowledge
  - small, relentless
  - retrospectives
  - 5 Whys
  - eyes for waste
    * variability, over-burden, NVA ... (handoff, WIP, info scatter, delay, multi-tasking, defects, wishful thinking..)

- perfection challenge

- work toward flow (lower batch size, Q size, cycle time)

Management applies and teaches lean thinking, and bases decisions on this long-term philosophy

Continuous Improvement towards Perfection



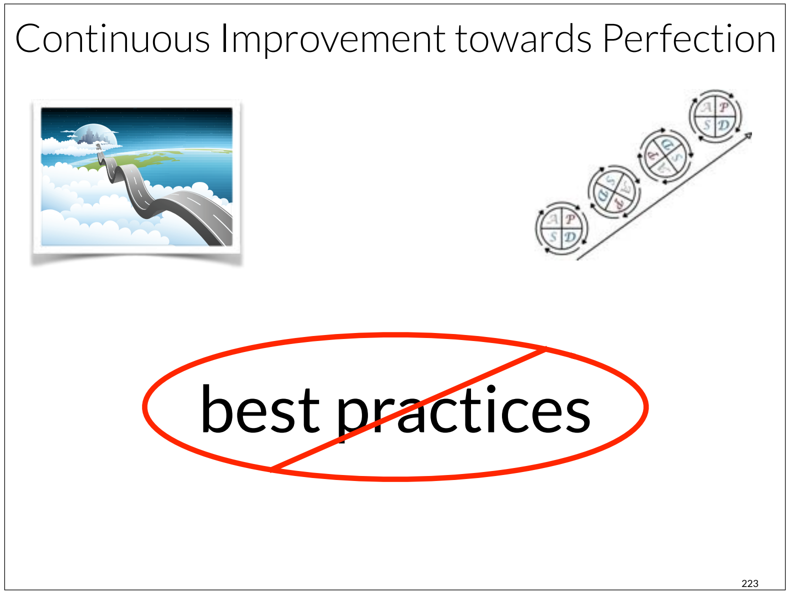
PDSA & Scientific Method by Workers

Decoding the DNA of Toyota

Harvard Business Review

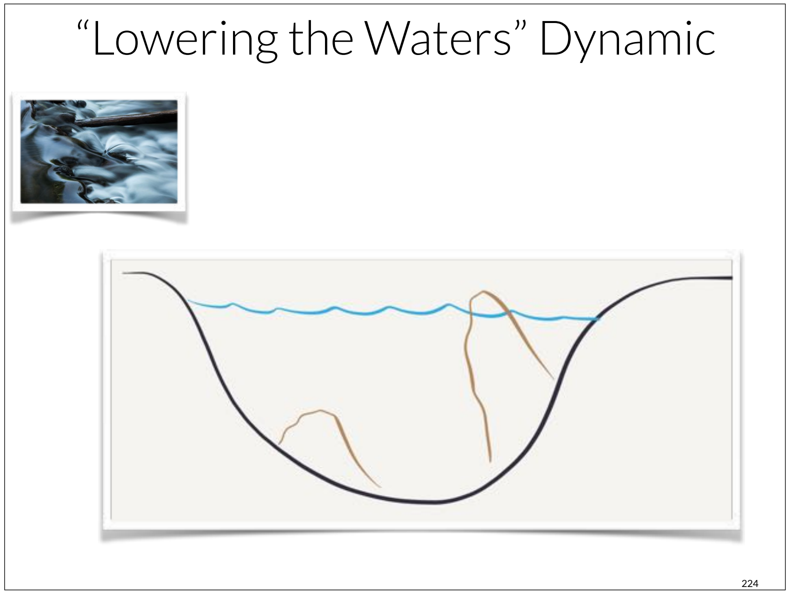
Sept. 1999

"the scientific method is so ingrained at Toyota"

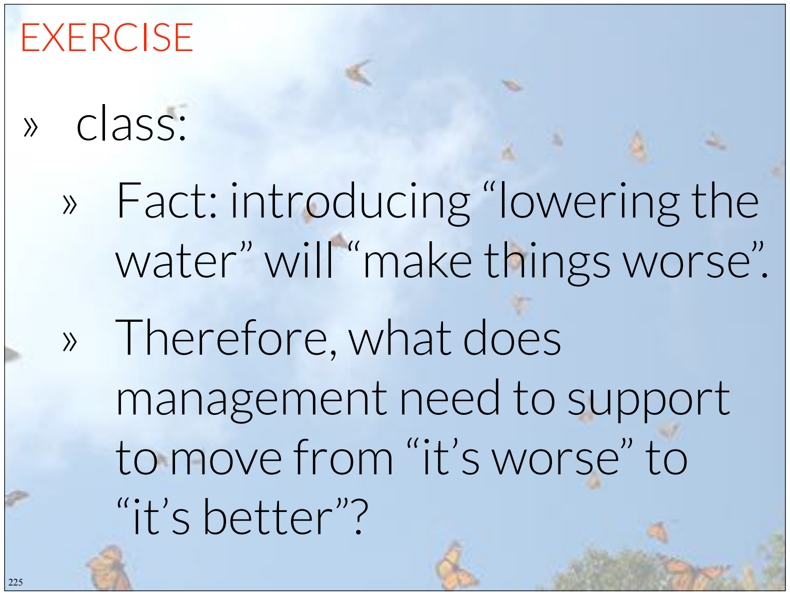


Continuous Improvement towards Perfection

~~best practices~~



"Lowering the Waters" Dynamic



EXERCISE

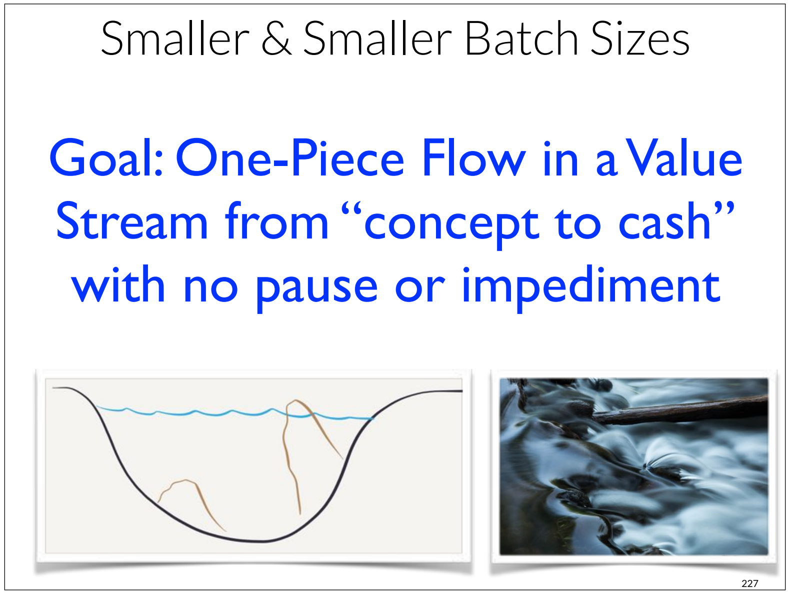
- class:
  - Fact: introducing "lowering the water" will "make things worse".
  - Therefore, what does management need to support to move from "it's worse" to "it's better"?

225

Stop & Fix

(all forms of imperfection)

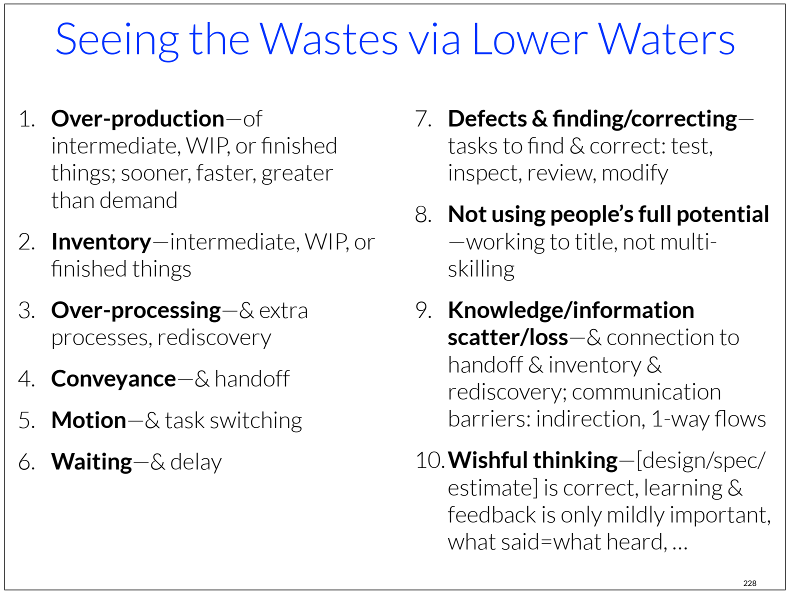
226



Smaller & Smaller Batch Sizes

Goal: One-Piece Flow in a Value Stream from "concept to cash" with no pause or impediment

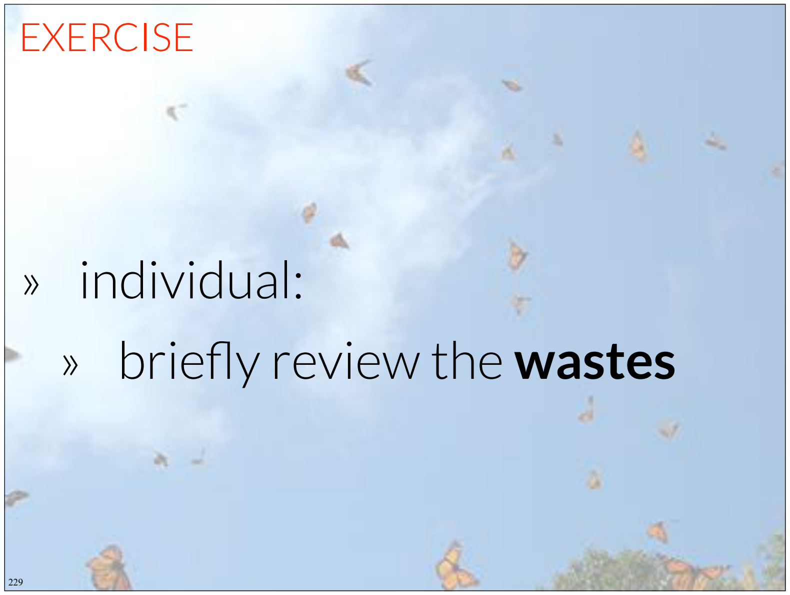
227



Seeing the Wastes via Lower Waters

1. **Over-production**—of intermediate, WIP, or finished things; sooner, faster, greater than demand
2. **Inventory**—intermediate, WIP, or finished things
3. **Over-processing**—& extra processes, rediscovery
4. **Conveyance**—& handoff
5. **Motion**—& task switching
6. **Waiting**—& delay
7. **Defects & finding/correcting**—tasks to find & correct: test, inspect, review, modify
8. **Not using people's full potential**—working to title, not multi-skilling
9. **Knowledge/information scatter/loss**—& connection to handoff & inventory & rediscovery; communication barriers: indirection, 1-way flows
10. **Wishful thinking**—[design/spec/estimate] is correct, learning & feedback is only mildly important, what said=what heard, ...

228

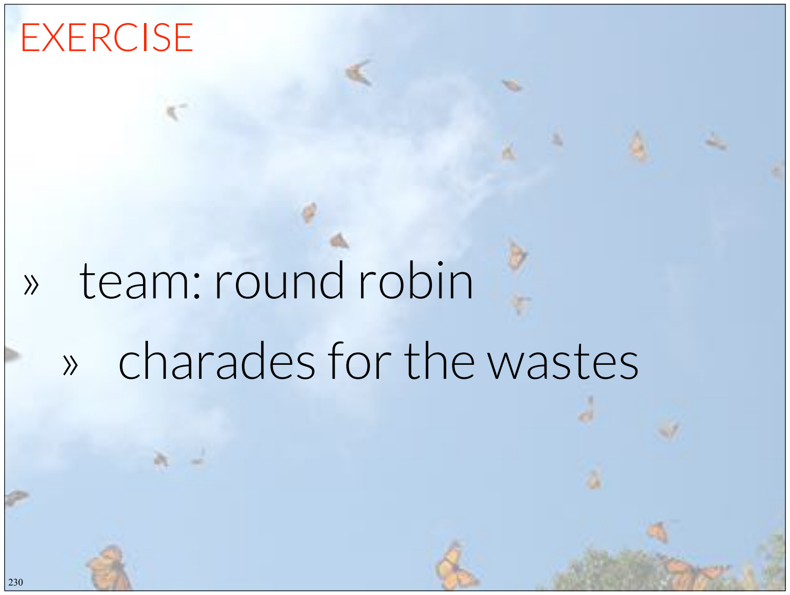


EXERCISE

- individual:
  - briefly review the **wastes**

229

229

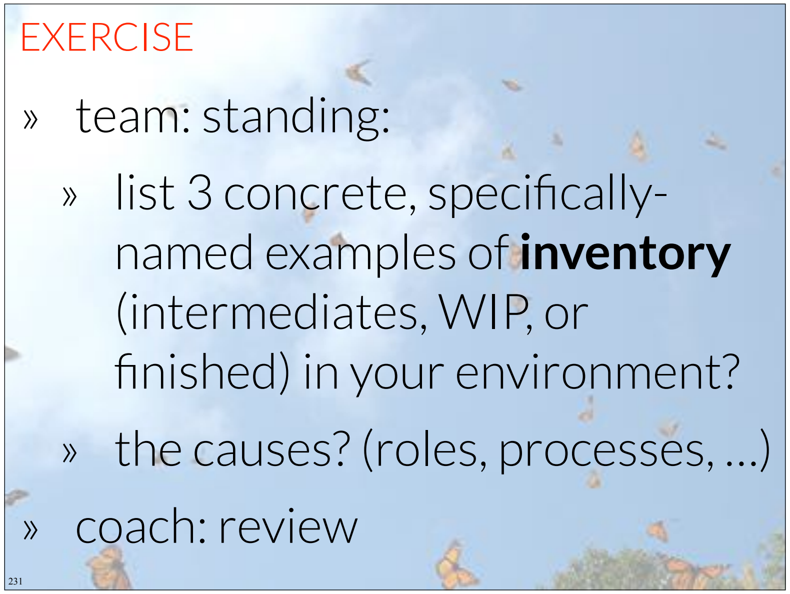


EXERCISE

- team: round robin
  - charades for the wastes

230

230

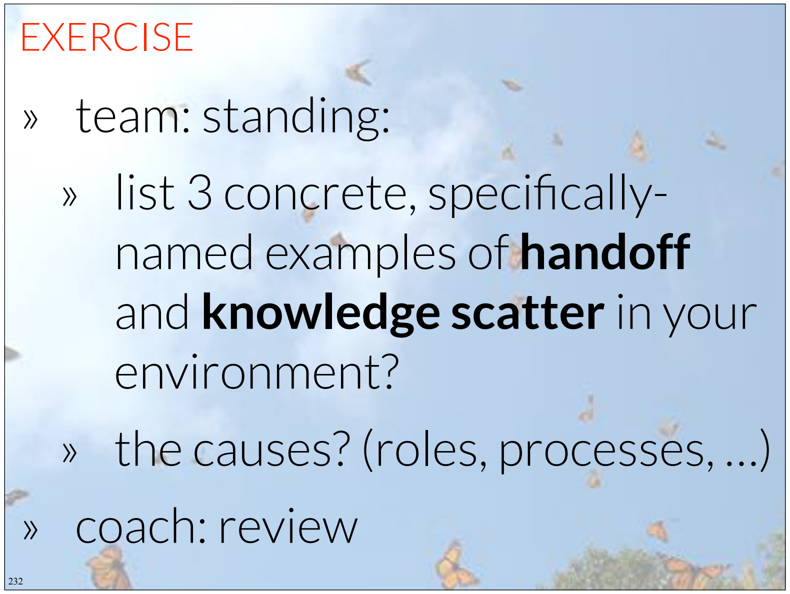


EXERCISE

- team: standing:
  - list 3 concrete, specifically-named examples of **inventory** (intermediates, WIP, or finished) in your environment?
  - the causes? (roles, processes, …)
- coach: review

231

231



EXERCISE

- team: standing:
  - list 3 concrete, specifically-named examples of **handoff** and **knowledge scatter** in your environment?
  - the causes? (roles, processes, …)
- coach: review

232

232

EXERCISE

- team:
  - How do single-function groups that can do one step “efficiently” and “fast” lead to more over-processing and more inventory?
  - coach: review

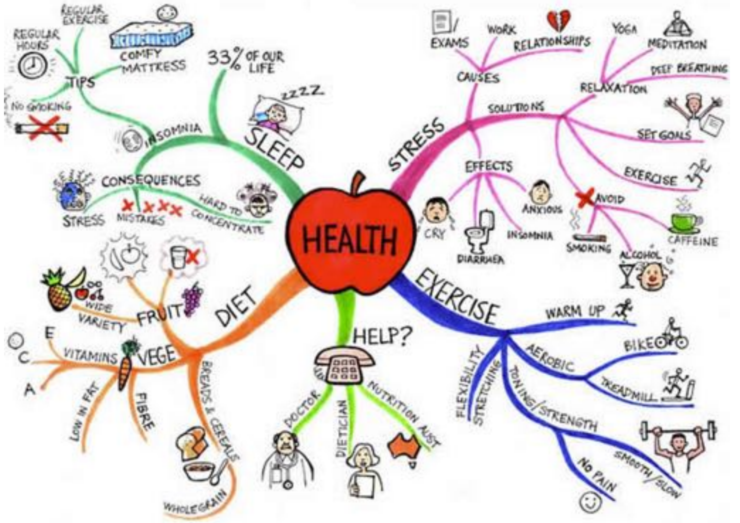
reduced wastes by …

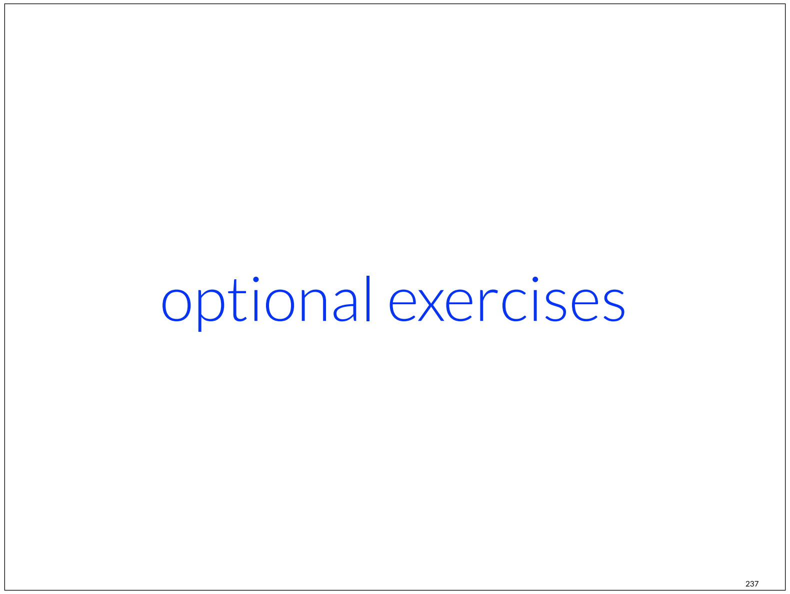
intensive task/job rotation

(multi-functional workers)

Lean Thinking is NOT a focus on…

24.37%

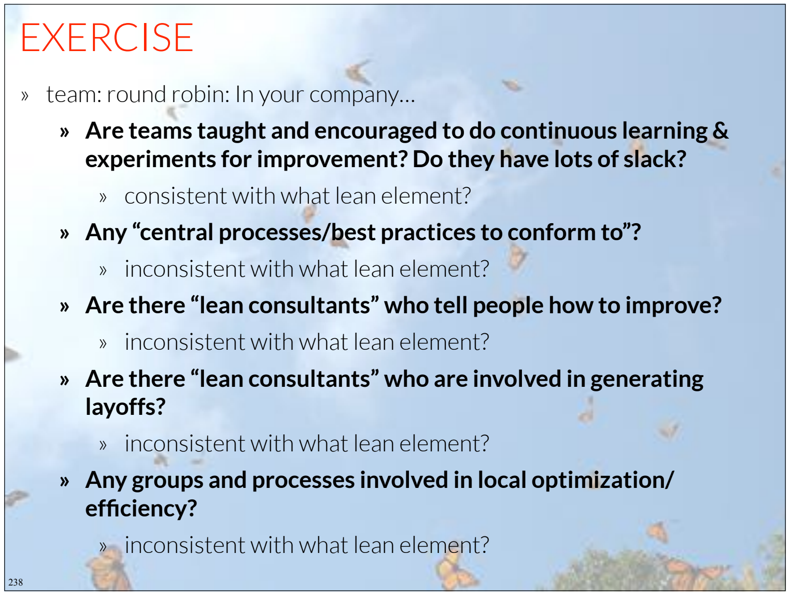




optional exercises

237

237

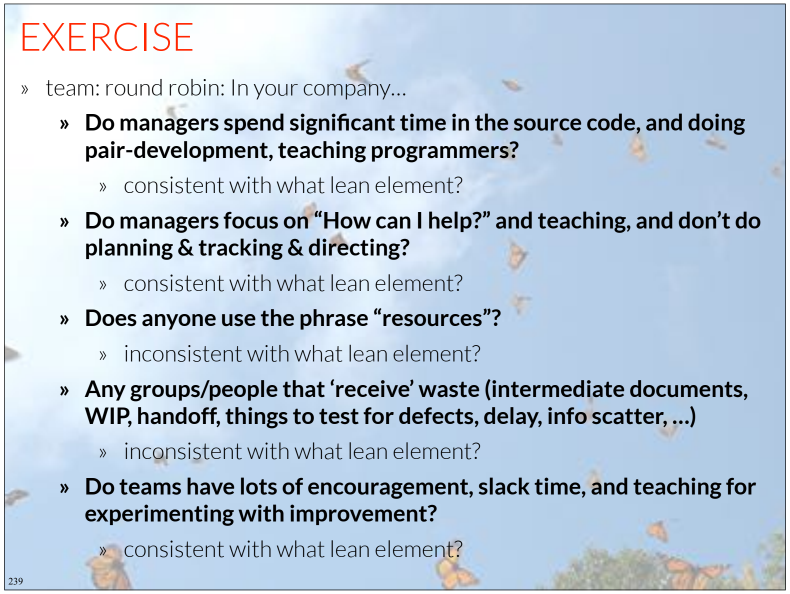


EXERCISE

- team: round robin: In your company…
  - **Are teams taught and encouraged to do continuous learning & experiments for improvement? Do they have lots of slack?**
    - consistent with what lean element?
  - **Any “central processes/best practices to conform to”?**
    - inconsistent with what lean element?
  - **Are there “lean consultants” who tell people how to improve?**
    - inconsistent with what lean element?
  - **Are there “lean consultants” who are involved in generating layoffs?**
    - inconsistent with what lean element?
  - **Any groups and processes involved in local optimization/ efficiency?**
    - inconsistent with what lean element?

238

238

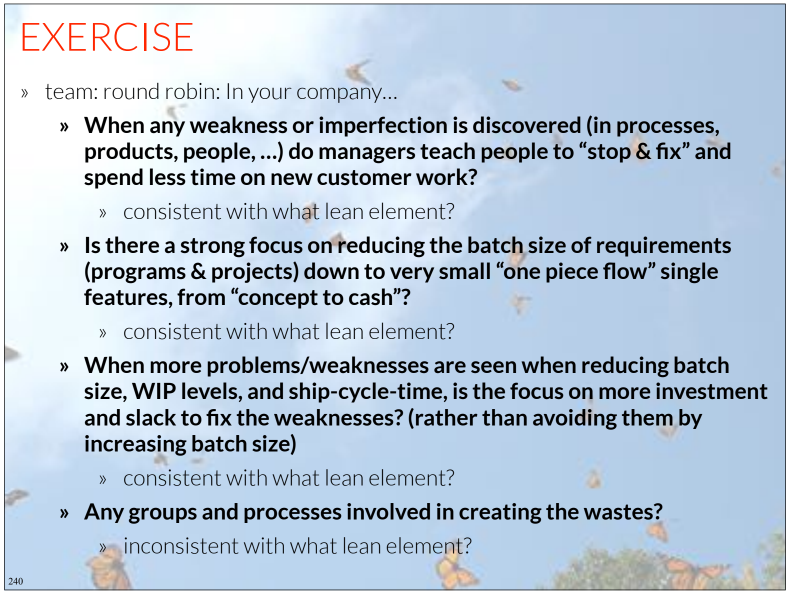


EXERCISE

- team: round robin: In your company…
  - **Do managers spend significant time in the source code, and doing pair-development, teaching programmers?**
    - consistent with what lean element?
  - **Do managers focus on “How can I help?” and teaching, and don’t do planning & tracking & directing?**
    - consistent with what lean element?
  - **Does anyone use the phrase “resources”?**
    - inconsistent with what lean element?
  - **Any groups/people that ‘receive’ waste (intermediate documents, WIP, handoff, things to test for defects, delay, info scatter, …)**
    - inconsistent with what lean element?
  - **Do teams have lots of encouragement, slack time, and teaching for experimenting with improvement?**
    - consistent with what lean element?

239

239



EXERCISE

- team: round robin: In your company…
  - **When any weakness or imperfection is discovered (in processes, products, people, …) do managers teach people to “stop & fix” and spend less time on new customer work?**
    - consistent with what lean element?
  - **Is there a strong focus on reducing the batch size of requirements (programs & projects) down to very small “one piece flow” single features, from “concept to cash”?**
    - consistent with what lean element?
  - **When more problems/weaknesses are seen when reducing batch size, WIP levels, and ship-cycle-time, is the focus on more investment and slack to fix the weaknesses? (rather than avoiding them by increasing batch size)**
    - consistent with what lean element?
  - **Any groups and processes involved in creating the wastes?**
    - inconsistent with what lean element?

240

240

EXERCISE

- team: round robin: In your company…
  - **Any groups involved in the wastes (creating inventory of intermediate documents, WIP for downstream groups, handoff, delay, over-processing, information scatter, things to be inspected or tested for defects, …) that incorrectly classify their activity as 'value'?**
    - inconsistent with what lean element?
  - **Are people in one role with one specialized task?**
    - inconsistent with what lean element?
  - **Any "lean experts" that teach it means tools & metrics?**

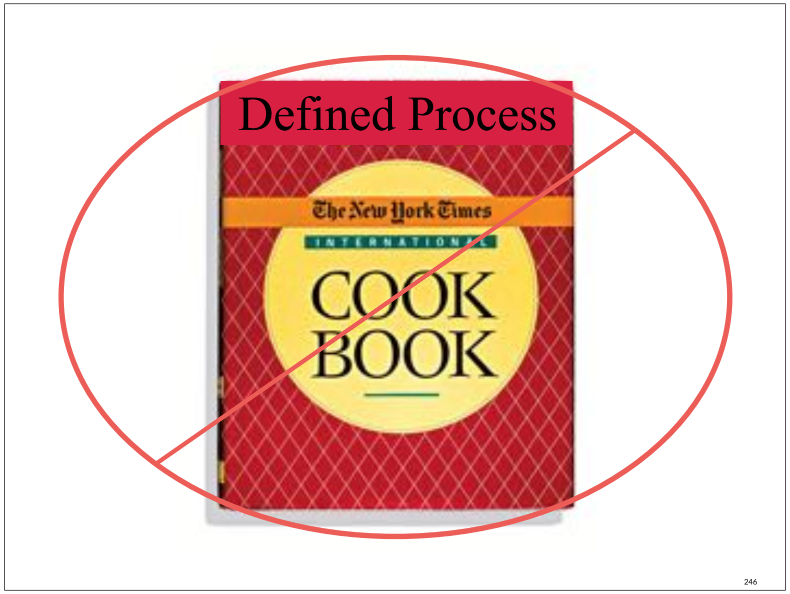
Empirical Process Control

OUR PROJECT PLAN WILL FOLLOW THE USUAL ARC.
PHASE ONE WILL BE UNWARRANTED OPTIMISM SUPPORTED BY DELUSIONS OF COMPETENCE.
IN PHASE TWO, THE OBSTRUCTIONISTS WILL SLITHER OUT OF THEIR LAIRS AND TRY TO SMOTHER OUR DREAMS.
IGNORANCE AND ENVY WILL FUEL RUMORS THAT GET REPEATED UNTIL THEY MORPH INTO COMMON KNOWLEDGE.
RESOURCES WILL BE ALLOCATED BASED ON MISINFORMATION AND FAVORITISM.
AND REQUIREMENTS WILL DRIFT UNTIL THE PROJECT IS BOTH UNDESIRABLE AND IMPOSSIBLE.
THAT BRINGS US TO THE SECOND WEEK.
I WANT MY UNWARRANTED OPTIMISM BACK.

what DEFINED process recipe has consistently worked really well for large-scale development over the last 50 years?

high levels of inherent complexity, variability, learning, context in R&D

245



246

247

and that's scary or uncomfortable for...

248

WEIGHT GAUGE
DADDY, WHAT'S A CONTROL FREAK?
CHRIS MADDEN

249

“we don’t pay you people to learn here”

250

all we need is…

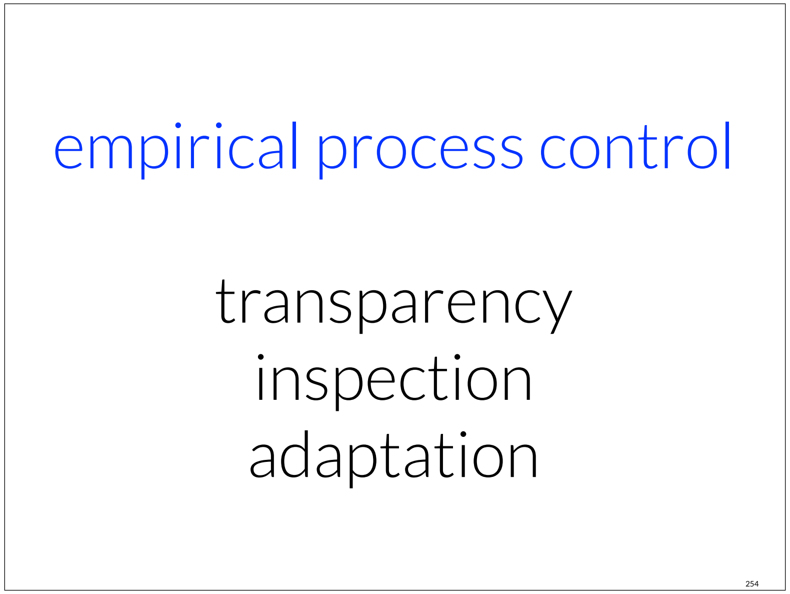
more defined steps & techniques
more specialization & managers
more clear requirements
more careful plans
more control
and then! ...

251

252

once we have finally given up the belief we will eventually find The Recipe, we are left with...

253



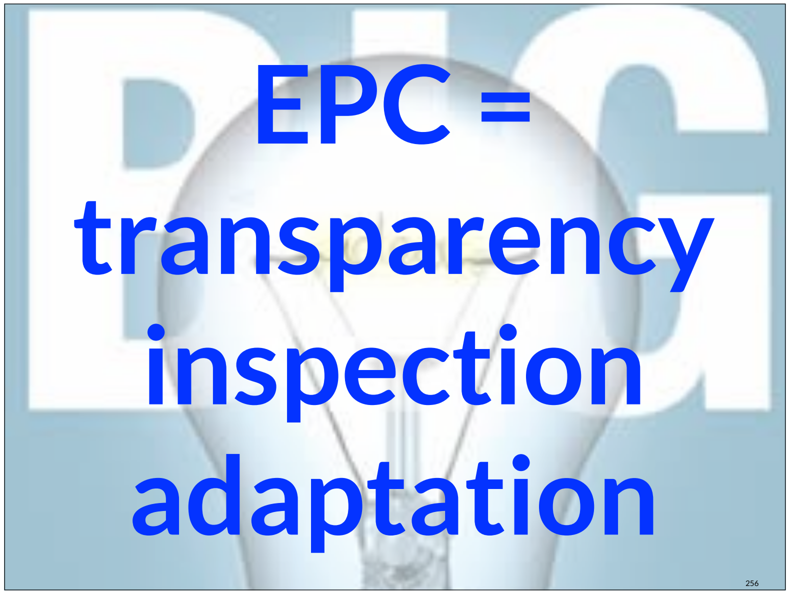
empirical process control

transparency
inspection
adaptation

254

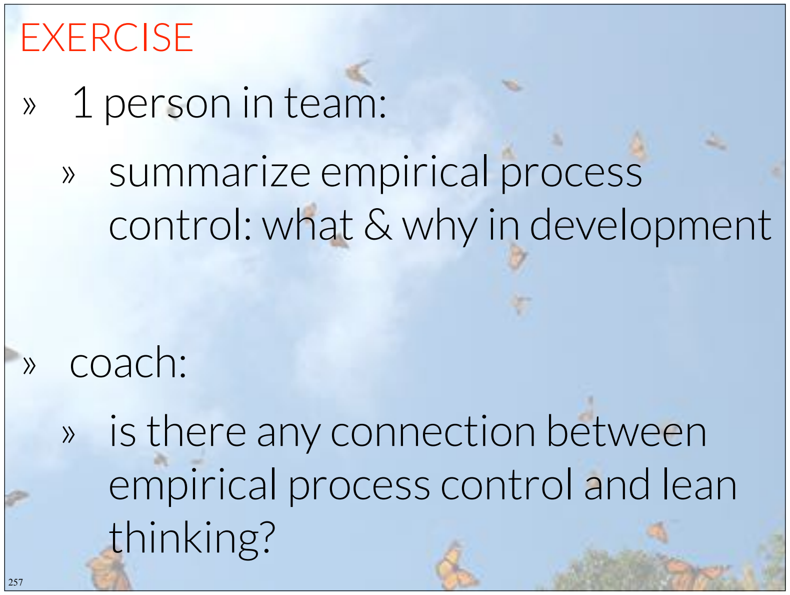
empirical process control

255



EPC =
transparency
inspection
adaptation

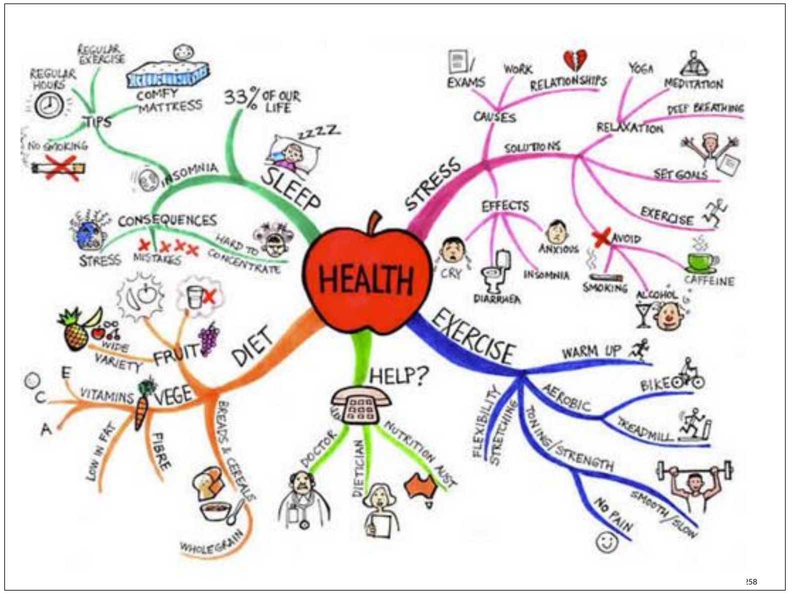
256



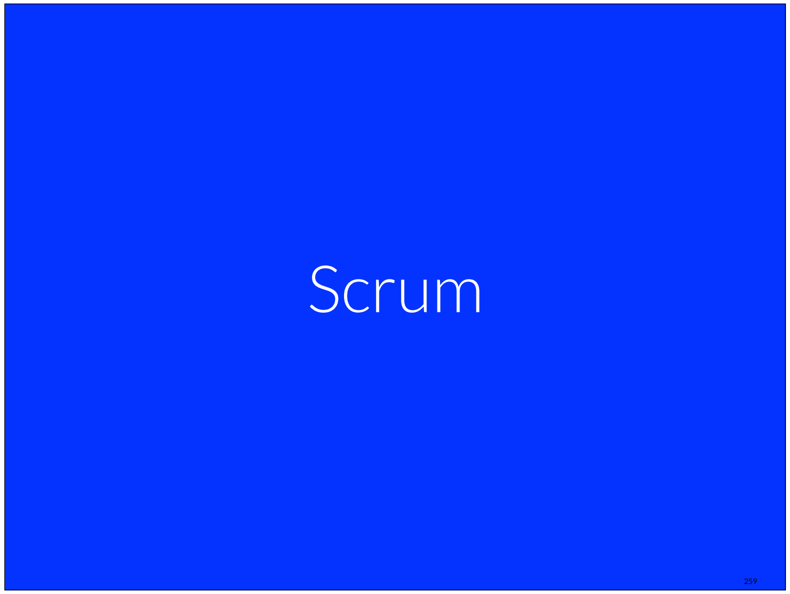
EXERCISE

- 1 person in team:
  - summarize empirical process control: what & why in development
- coach:
  - is there any connection between empirical process control and lean thinking?

257

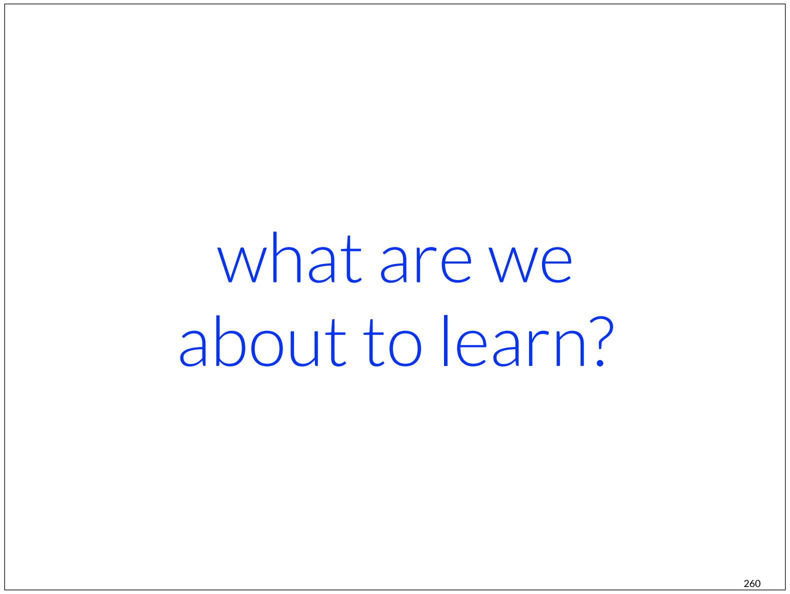


258



Scrum

259



what are we about to learn?

260

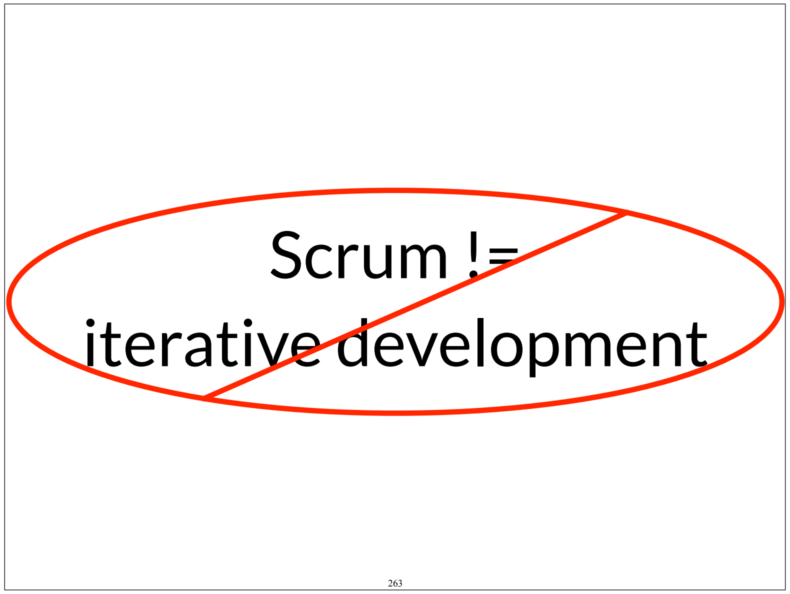
one of the first Certified Scrum Trainers, mentored by the creator of Scrum

evidence-based management

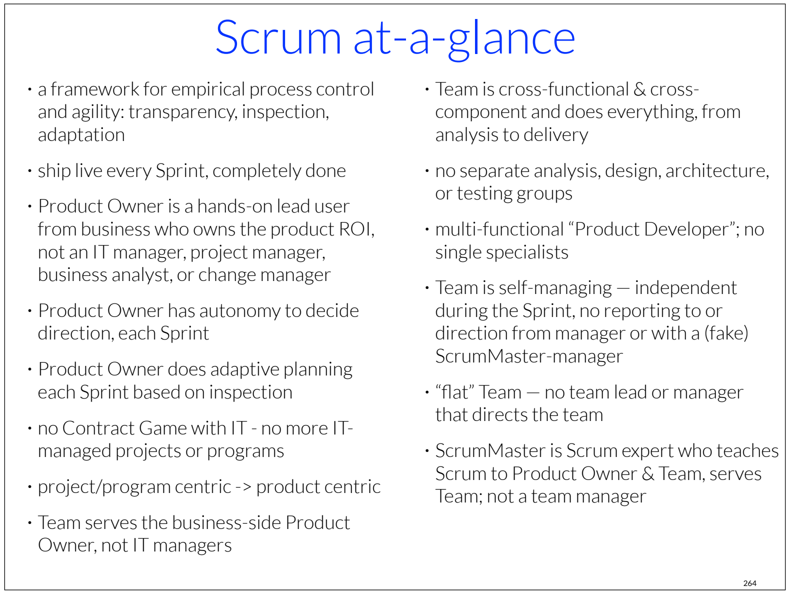
Harvard Business School

1986

HBR
JANUARY-FEBRUARY 1986
The New New Product Development Game
Hirotaka Takeuchi and Ikujiro Nonaka
Scrum



~~Scrum != iterative development~~

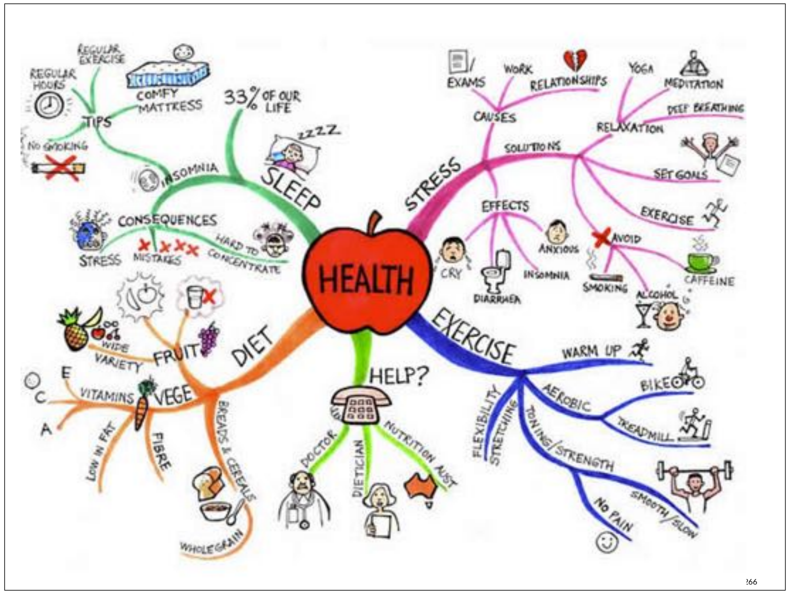


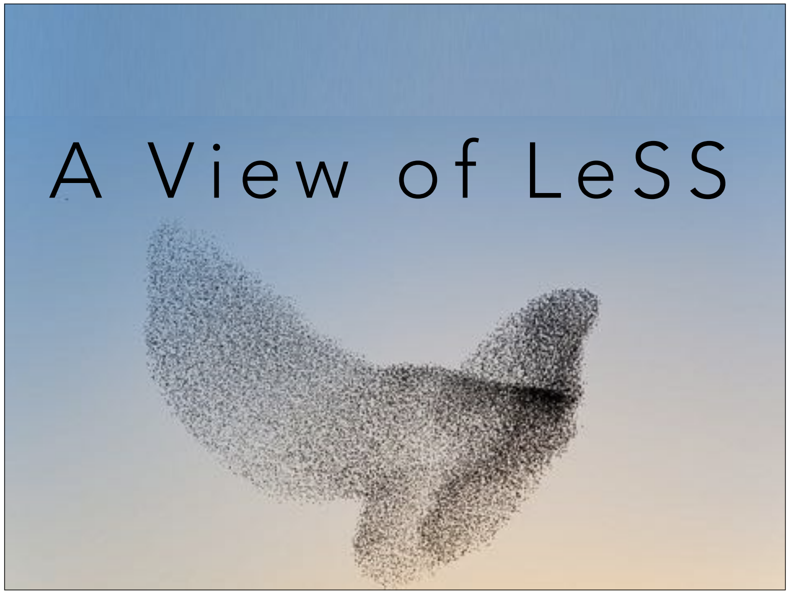
Scrum at-a-glance

- a framework for empirical process control and agility: transparency, inspection, adaptation
- ship live every Sprint, completely done
- Product Owner is a hands-on lead user from business who owns the product ROI, not an IT manager, project manager, business analyst, or change manager
- Product Owner has autonomy to decide direction, each Sprint
- Product Owner does adaptive planning each Sprint based on inspection
- no Contract Game with IT - no more IT-managed projects or programs
- project/program centric -> product centric
- Team serves the business-side Product Owner, not IT managers
- Team is cross-functional & cross-component and does everything, from analysis to delivery
- no separate analysis, design, architecture, or testing groups
- multi-functional "Product Developer"; no single specialists
- Team is self-managing — independent during the Sprint, no reporting to or direction from manager or with a (fake) ScrumMaster-manager
- "flat" Team — no team lead or manager that directs the team
- ScrumMaster is Scrum expert who teaches Scrum to Product Owner & Team, serves Team; not a team manager

EXERCISE

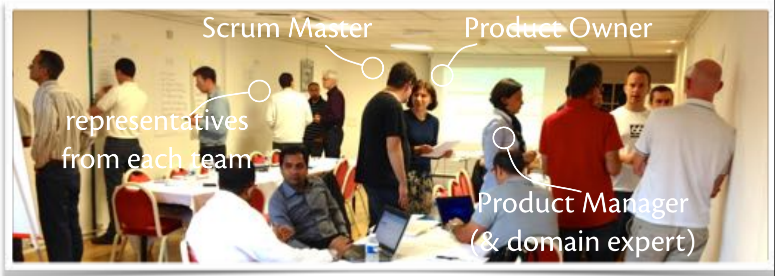
- coach
  - explore the degree of realization of Scrum in the organization





A View of LeSS

Sprint Planning One in LeSS


Scrum Master
Product Owner
representatives from each team
Product Manager (& domain expert)

Coordinating via Open Space in LeSS Huge

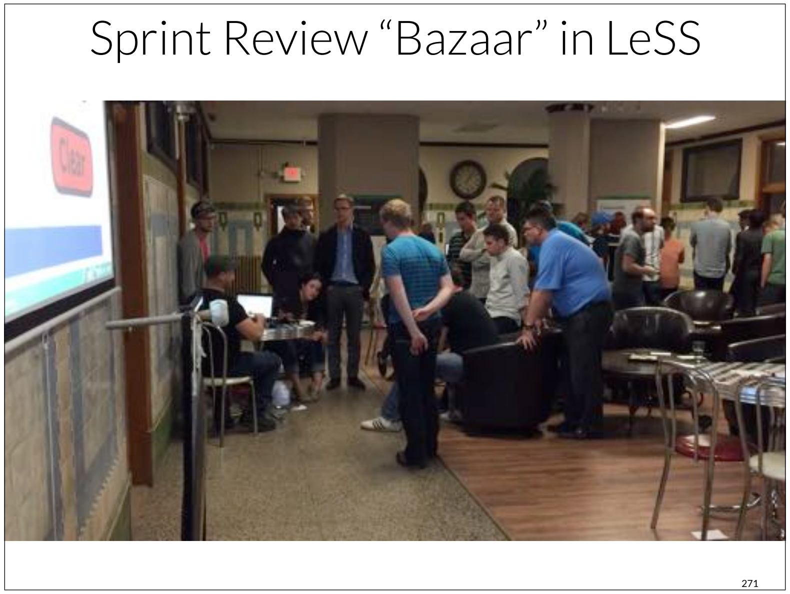
269

269

Multi-team PBR in LeSS

270

270



Sprint Review “Bazaar” in LeSS

271

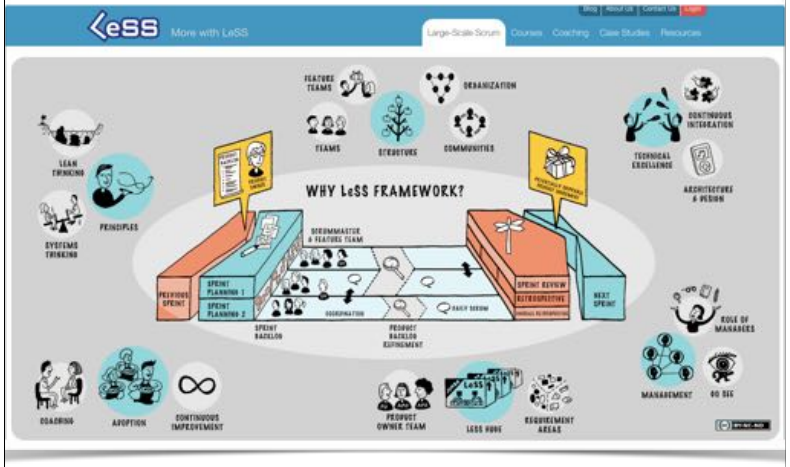
271

Overall Retrospective in LeSS (multisite)

272

272

Another View: less.works (explore it)



273

273

LeSS Overview

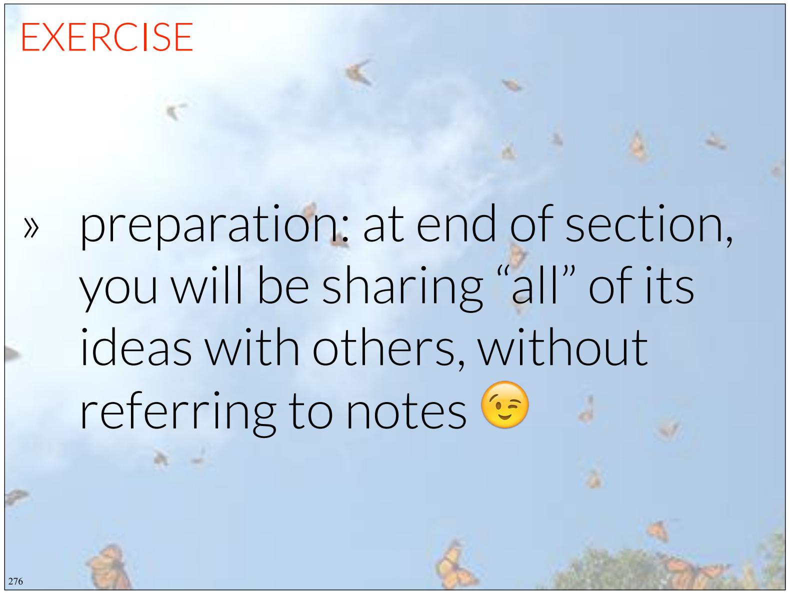
274

what are we
about to learn?

275

275

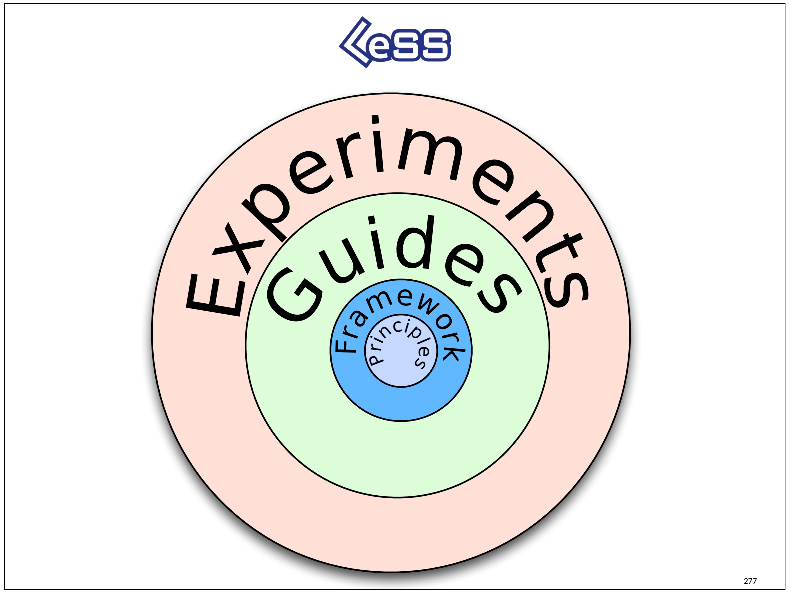
EXERCISE



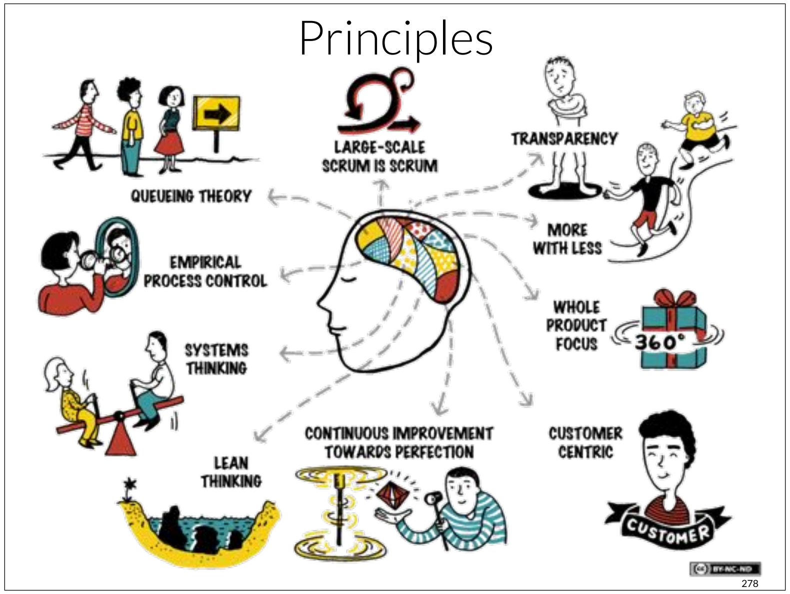
- » preparation: at end of section, you will be sharing "all" of its ideas with others, without referring to notes 😉

276

276

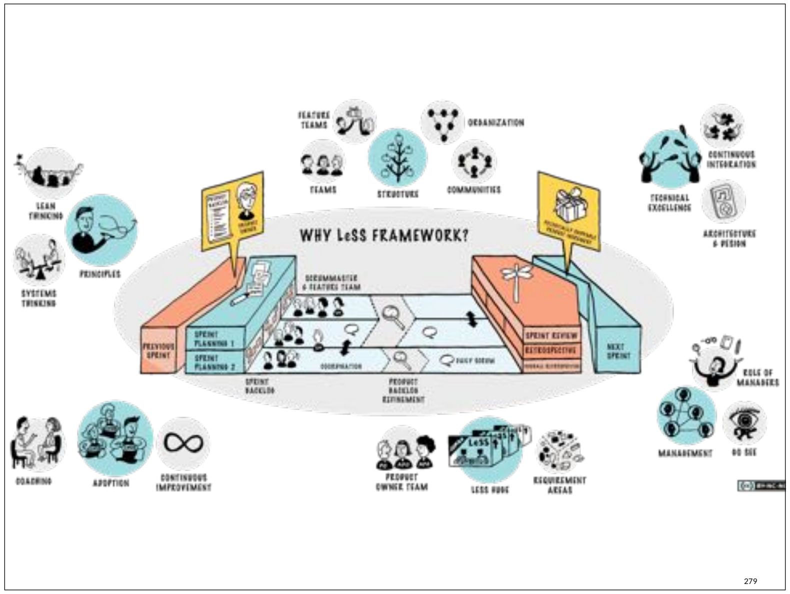


277

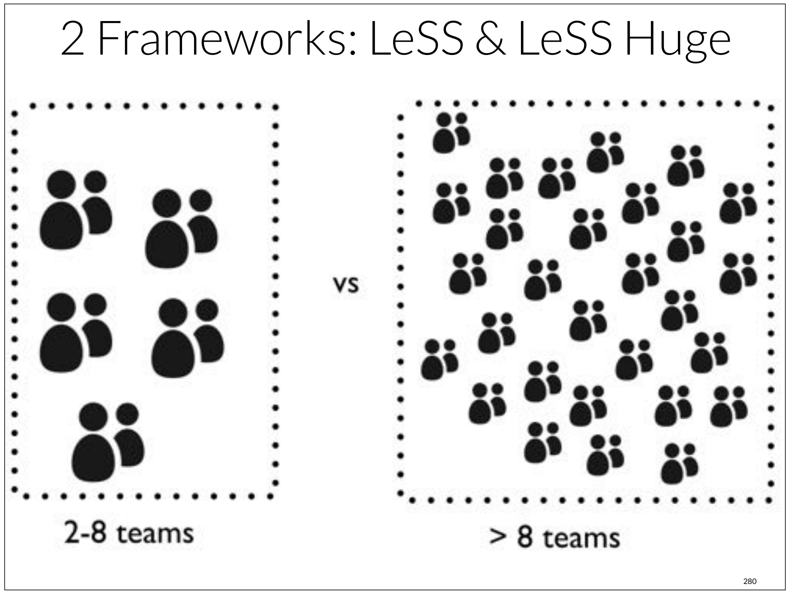


Principles

278



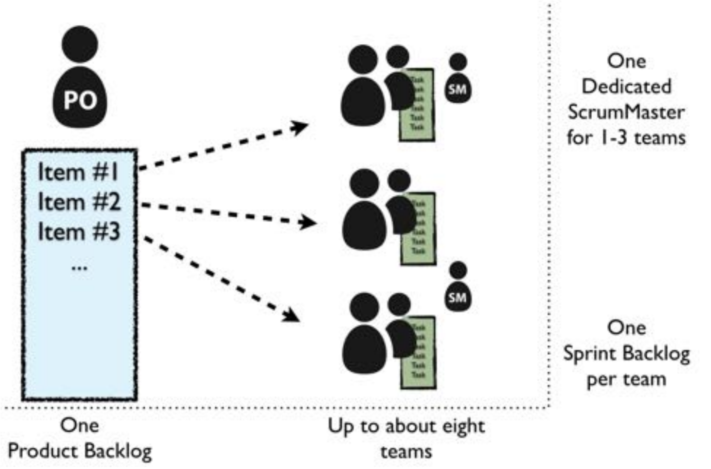
279



2 Frameworks: LeSS & LeSS Huge

280

(smaller) LeSS Framework



PO

Item #1
Item #2
Item #3
...

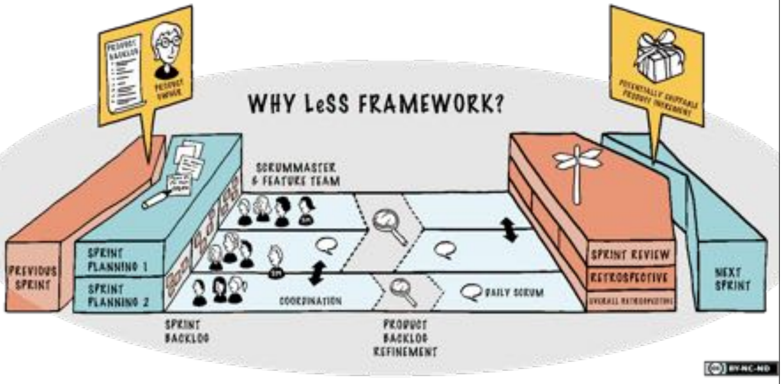
One Product Backlog

Up to about eight teams

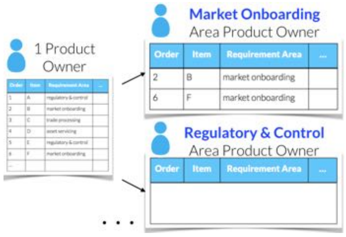
One Dedicated ScrumMaster for 1-3 teams

One Sprint Backlog per team

One Common Sprint, Shippable Product



LeSS Huge: Requirement Areas



1 Product Owner

Order	Item	Requirement Area	...
1	A	regulatory & control	
2	B	market onboarding	
3	C	trade processing	
4	D	asset servicing	
5	E	regulatory & control	
6	F	market onboarding	
...			

Market Onboarding Area Product Owner

Order	Item	Requirement Area	...
2	B	market onboarding	
6	F	market onboarding	

Regulatory & Control Area Product Owner

Order	Item	Requirement Area	...

. . .

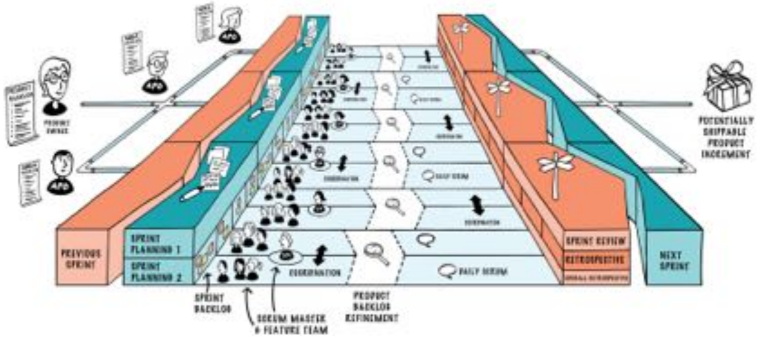
Market Onboarding
Area Feature Teams

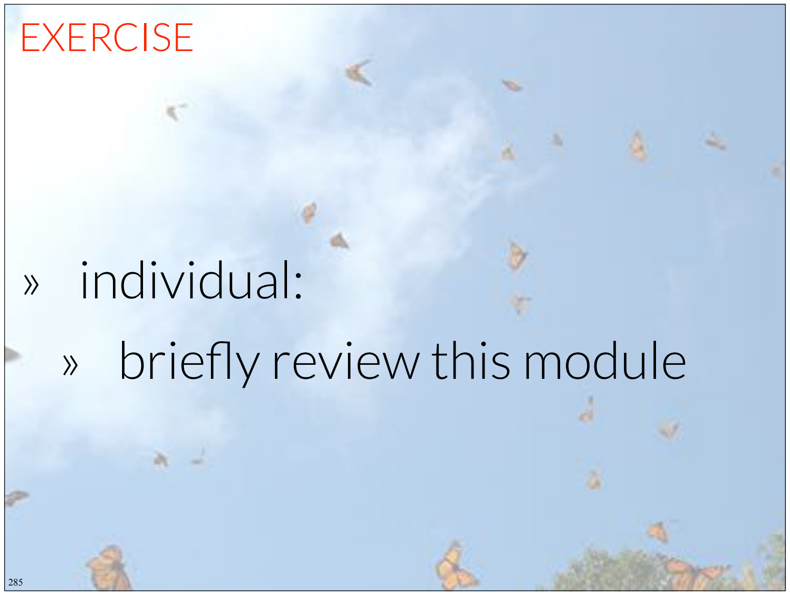
Scrum Feature Team | Scrum Feature Team | Scrum Feature Team
Scrum Feature Team | Scrum Feature Team | Scrum Feature Team

Regulatory & Control
Area Feature Teams

Scrum Feature Team | Scrum Feature Team | Scrum Feature Team

LeSS Huge: “Stack of LeSS”



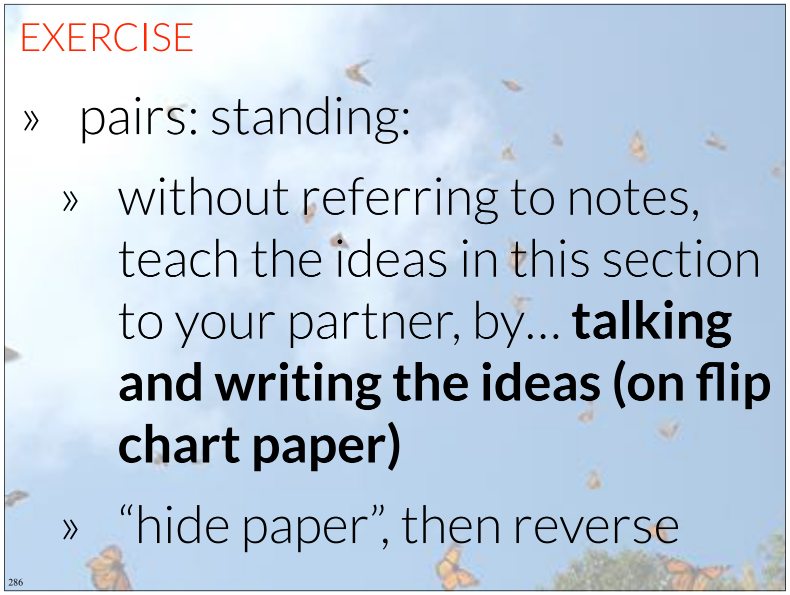


EXERCISE

- individual:
  - briefly review this module

285

285



EXERCISE

- pairs: standing:
  - without referring to notes, teach the ideas in this section to your partner, by… **talking and writing the ideas (on flip chart paper)**
  - “hide paper”, then reverse

286

286

EXERCISE

- team: standing:
  - **why** is there one **common** Sprint ending in a **shippable** product?
- coach: review

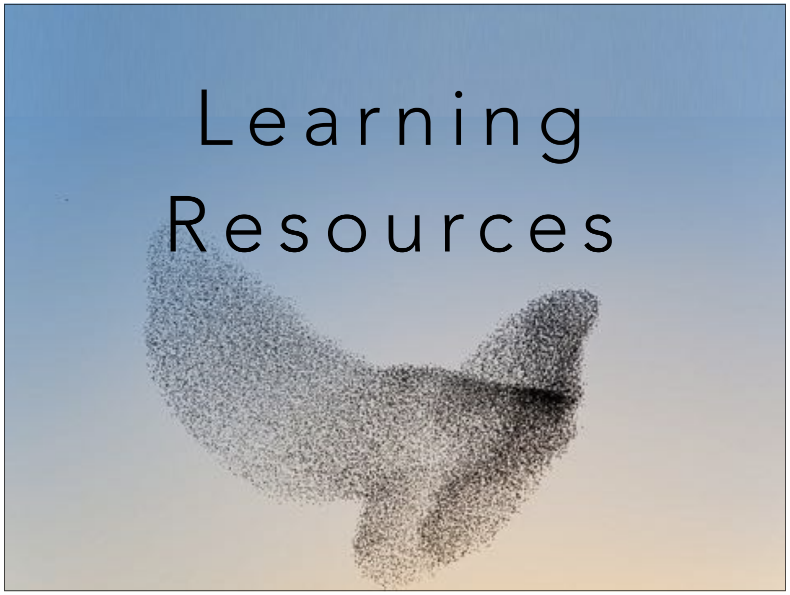
287

287

educate: **why**

288

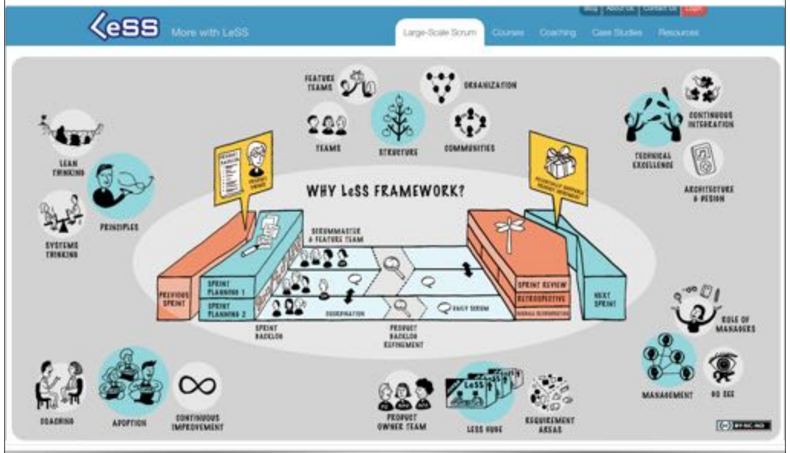
288



Learning Resources

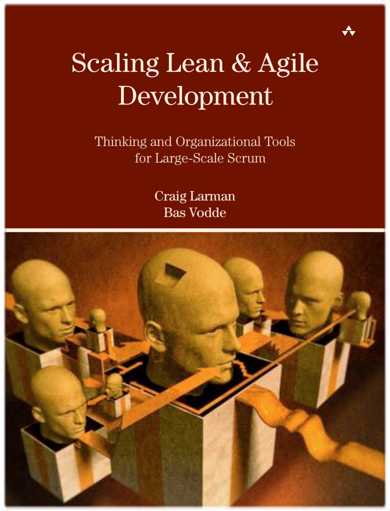
289

less.works



290

290



Scaling Lean & Agile Development

Thinking and Organizational Tools for Large-Scale Scrum

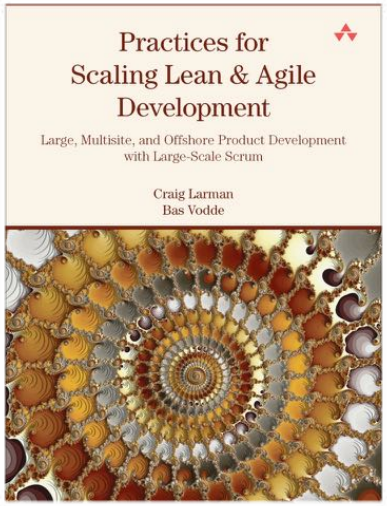
Craig Larman
Bas Vodde

Table of Contents

Thinking Tools

Organizational Tools

291



Practices for Scaling Lean & Agile Development

Large, Multisite, and Offshore Product Development with Large-Scale Scrum

Craig Larman
Bas Vodde

Table of Contents

Action Tools

292

CONTENTS

LeSS Structure

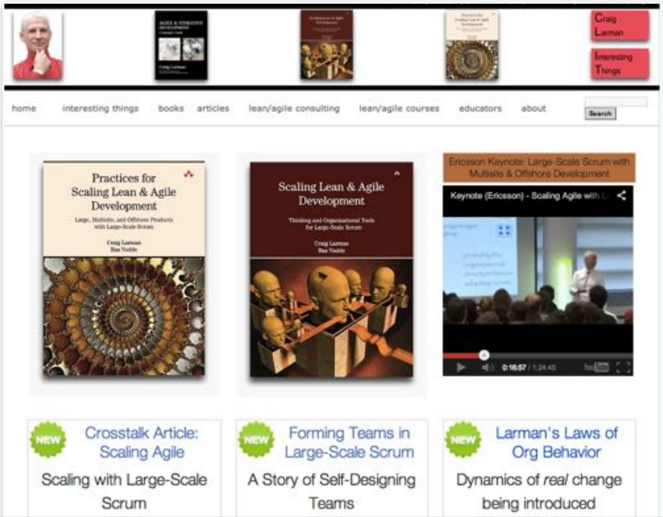
LeSS Product

LeSS Sprint

More or LeSS

Appendices

craiglarman.com



Craig Larman
Interesting Things
home
interesting things
books
articles
lean/agile consulting
lean/agile courses
educators
about
Search
Practices for Scaling Lean & Agile Development
Large, Multisite, and Offshore Products with Large-Scale Scrum
Craig Larman
Bas Vodde
Scaling Lean & Agile Development
Thinking and Organizational Tools for Large-Scale Scrum
Craig Larman
Bas Vodde
Ericsson Keynote: Large-Scale Scrum with Multisite & Offshore Development
Keynote (Ericsson) - Scaling Agile with L
0:16:57 / 1:24:45
NEW
NEW
NEW

Crosstalk Article: Scaling Agile

Scaling with Large-Scale Scrum

Forming Teams in Large-Scale Scrum

A Story of Self-Designing Teams

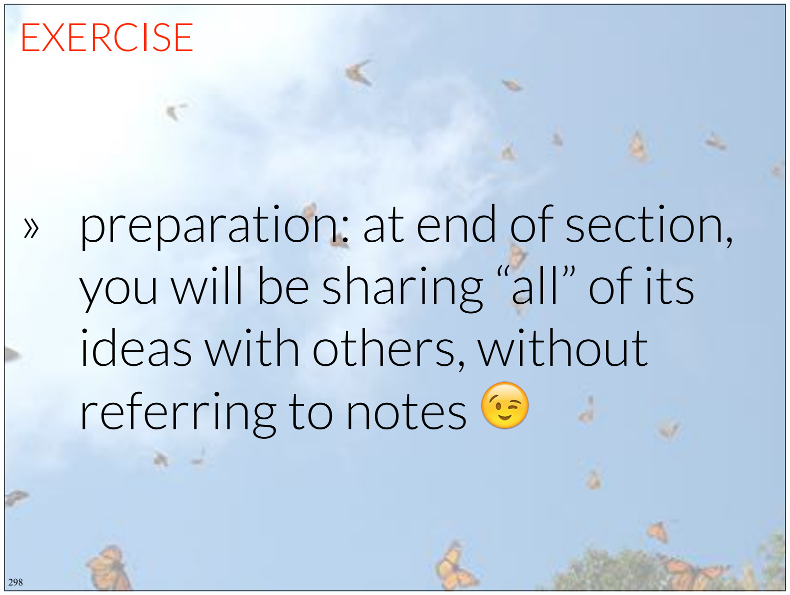
Larman's Laws of Org Behavior

Dynamics of *real* change being introduced

Adoption

First LeSS Adoption

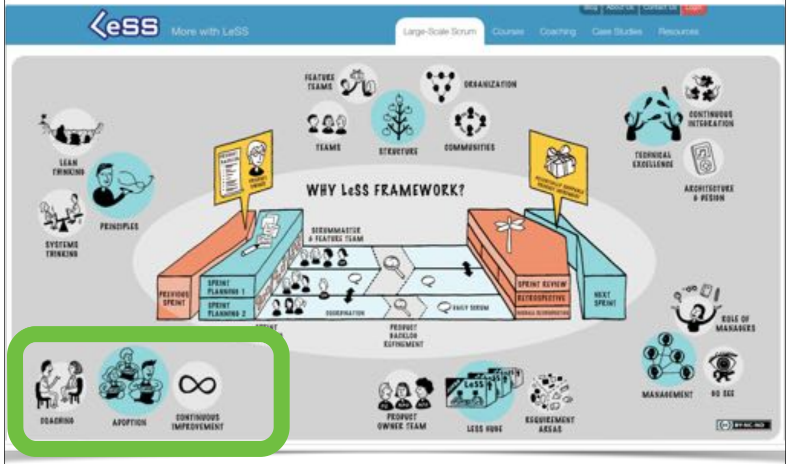
what are we
about to learn?

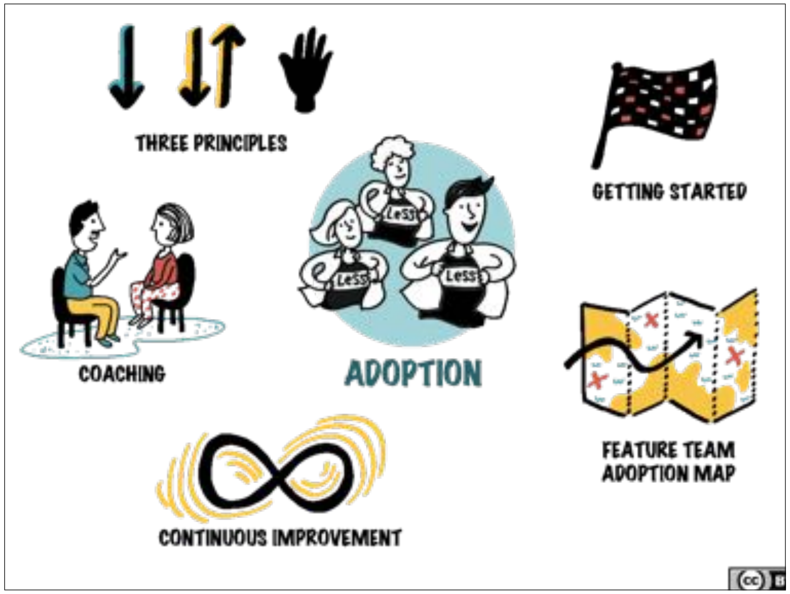


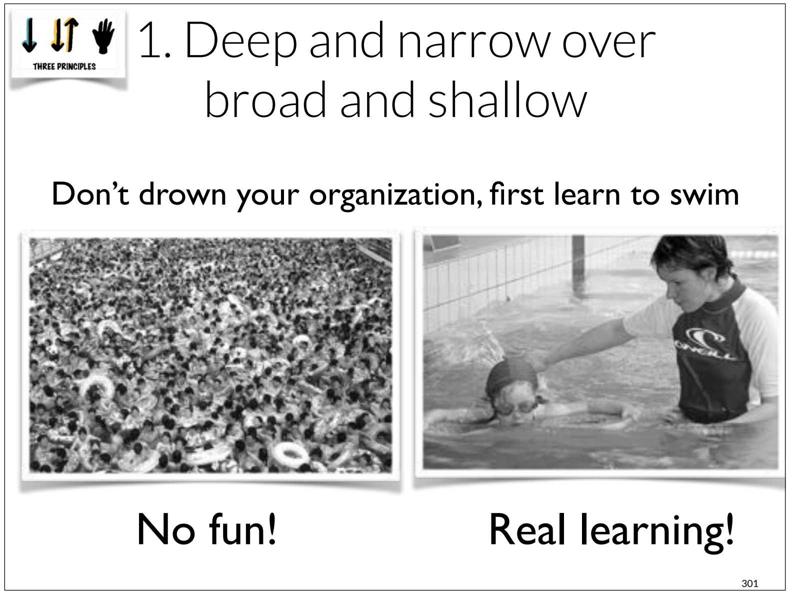
EXERCISE

» preparation: at end of section, you will be sharing “all” of its ideas with others, without referring to notes 😉

Adoption







1. Deep and narrow over broad and shallow

Don't drown your organization, first learn to swim

No fun!

Real learning!

301

301

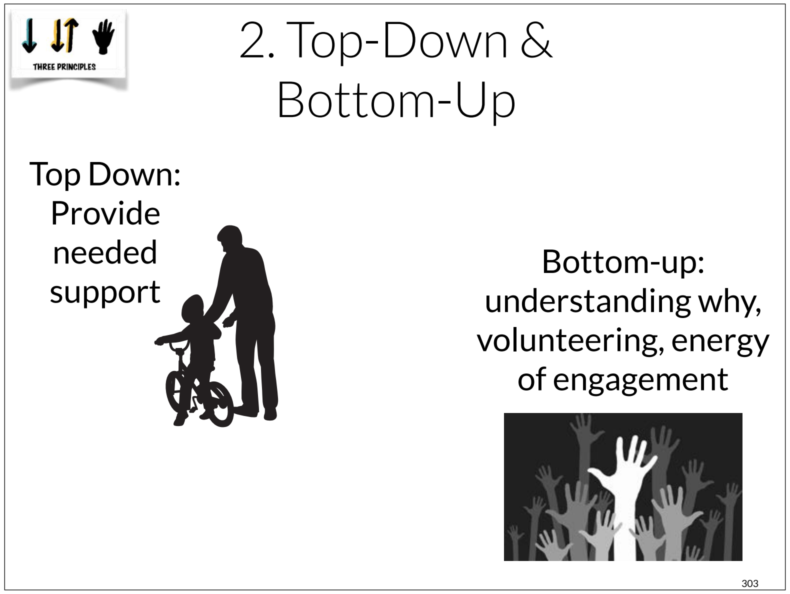
Scope of First Adoption

8- teams on **1 and only 1** product

preferably 1 site

302

302



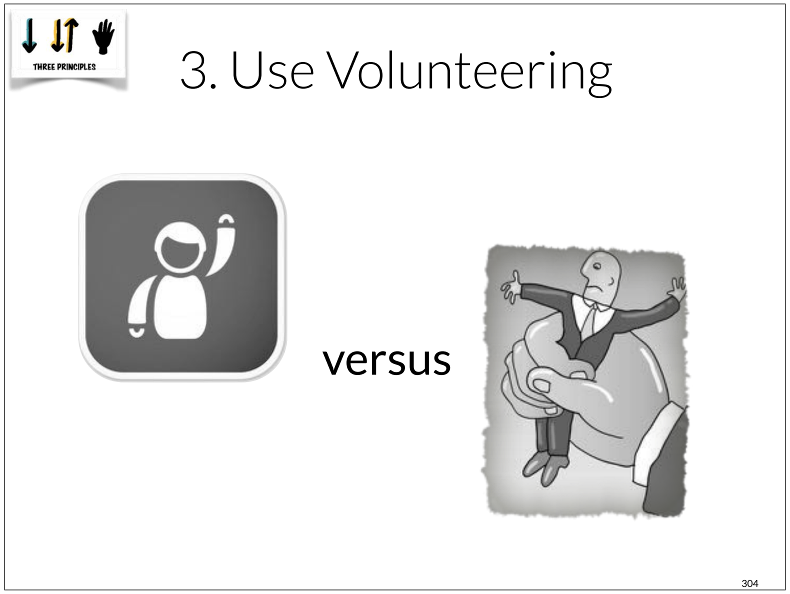
2. Top-Down & Bottom-Up

Top Down:
Provide
needed
support

Bottom-up:
understanding why,
volunteering, energy
of engagement

303

303



3. Use Volunteering

versus

304

304

2. Top-Down & Bottom-Up

Top Down:
Provide
needed
support

communication plan & strong “on message”

305

305

Getting Started

0. Educate Everyone: **why**

1. …

306

306

Educate Everyone

why

readings

educate **together** (not by role)

courses: (1) **Scrum**, (2) **LeSS**

307

307

Getting Started

0. …

1. Define/choose product for first LeSS adoption: **8- teams on 1 product**

2. …

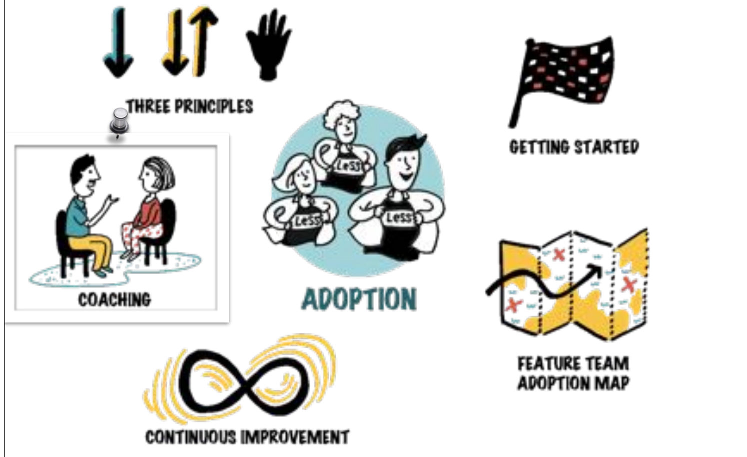
308

308

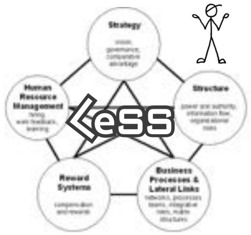
LeSS Rules

For the product group, establish the complete LeSS structure "at the start"; this is vital for a LeSS adoption.

First Group: INTENSIVE Coaching



Coaching & Coaches in LeSS

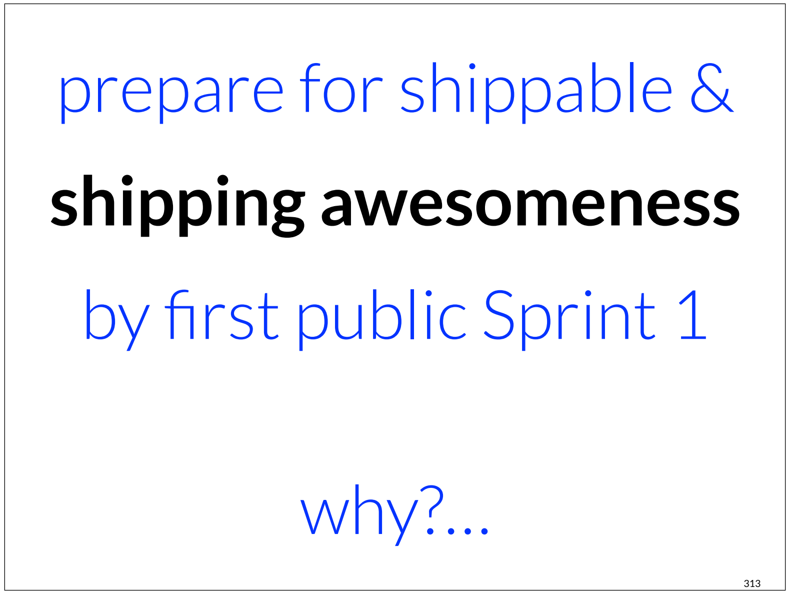


LeSS Coaching Activities for Adoption

1. Informed Consent for LeSS

2. LeSS Preparation

3. LeSS Sprint 1



prepare for shippable &

shipping awesomeness

by first public Sprint 1

why?...

313

shipping speaks louder than words

314

ship every Sprint

315

street cred

316

Early Street Credibility

significant product

Early Street Credibility

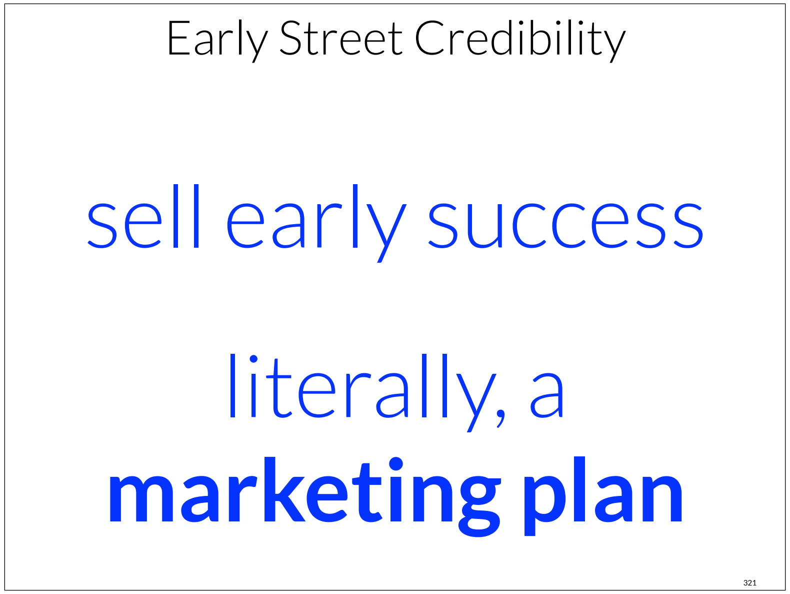
prepare, even for 1-2 months

Early Street Credibility

1+ **private** Sprints with Temporary Fake Product Owner

Early Street Credibility

real Product Owner who sells success

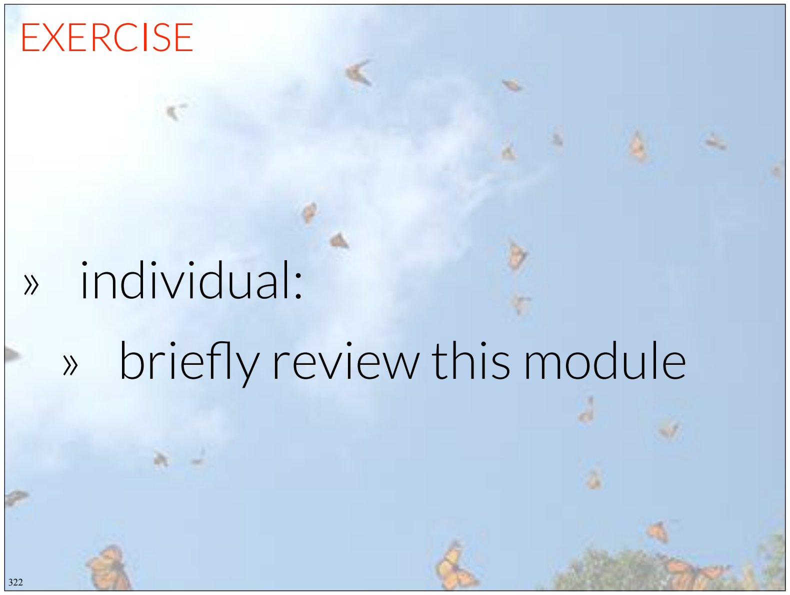


Early Street Credibility

sell early success

literally, a **marketing plan**

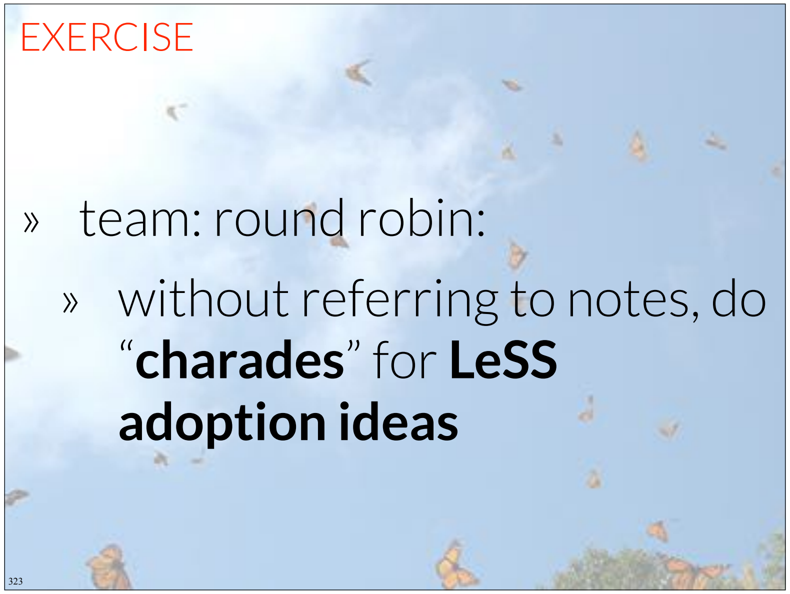
321



EXERCISE

- individual:
  - briefly review this module

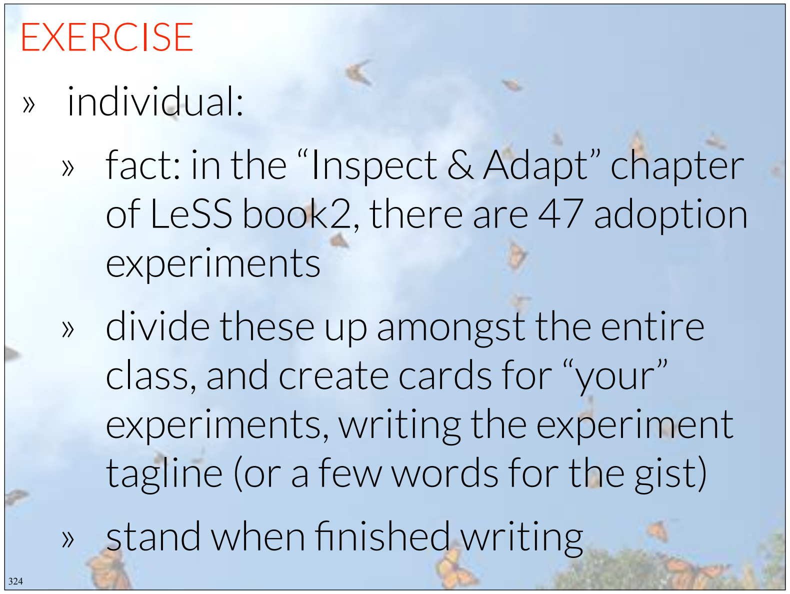
322



EXERCISE

- team: round robin:
  - without referring to notes, do "**charades**" for **LeSS adoption ideas**

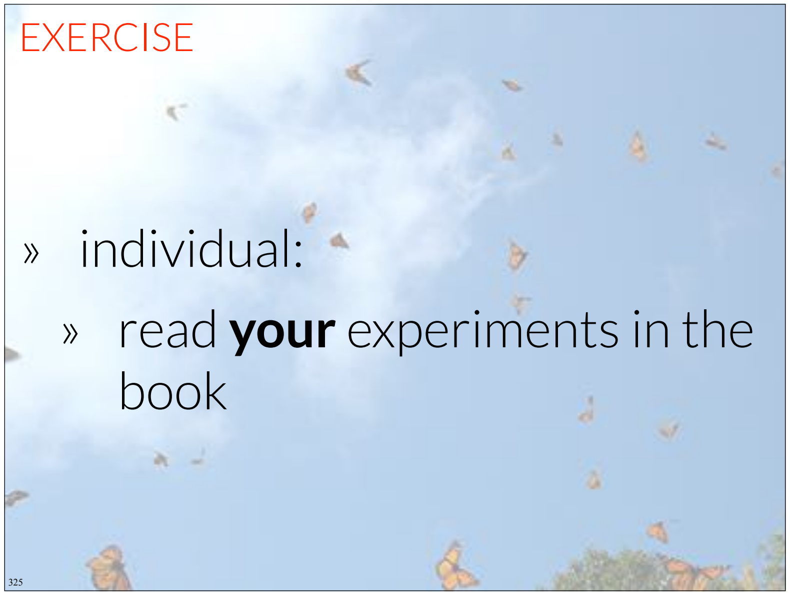
323



EXERCISE

- individual:
  - fact: in the "Inspect & Adapt" chapter of LeSS book2, there are 47 adoption experiments
  - divide these up amongst the entire class, and create cards for "your" experiments, writing the experiment tagline (or a few words for the gist)
  - stand when finished writing

324



EXERCISE

- individual:
  - read **your** experiments in the book

325

EXERCISE

- class: round robin:
  - **briefly**, share all your experiments with the class, and for each, summarize **why**

326

before? …

327

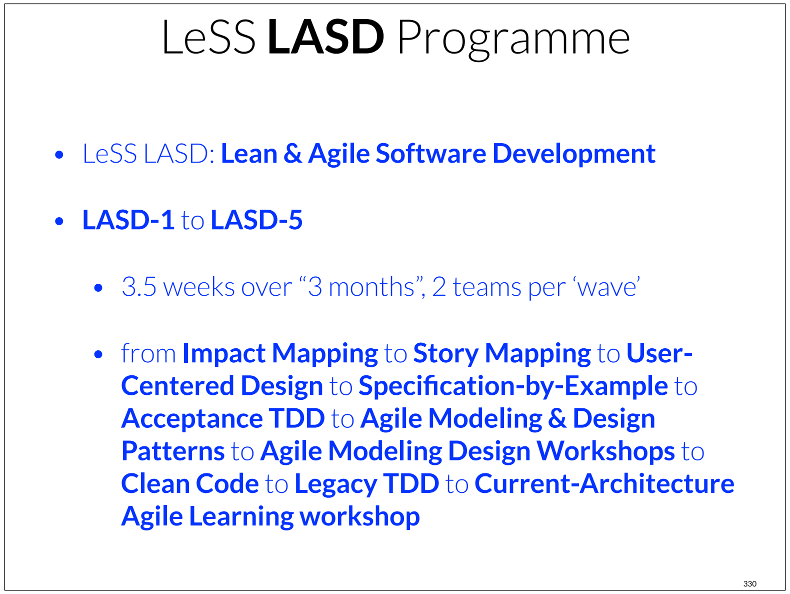
Pre-Adoption: **Build Interest**

- “LeSS 1” **book** to key people
- send LeSS **video** links to people
- **internal champions**? agile coach & senior manager
- outside **expert** — “you’re never a prophet in own land”
- **events** to build interest:
  - –*LeSS Practitioner* (preferred), *LeSS for Executives*, and/or *Less LeSS*
  - –Deep-Dive LeSS Q&A session with key people

328

during? …

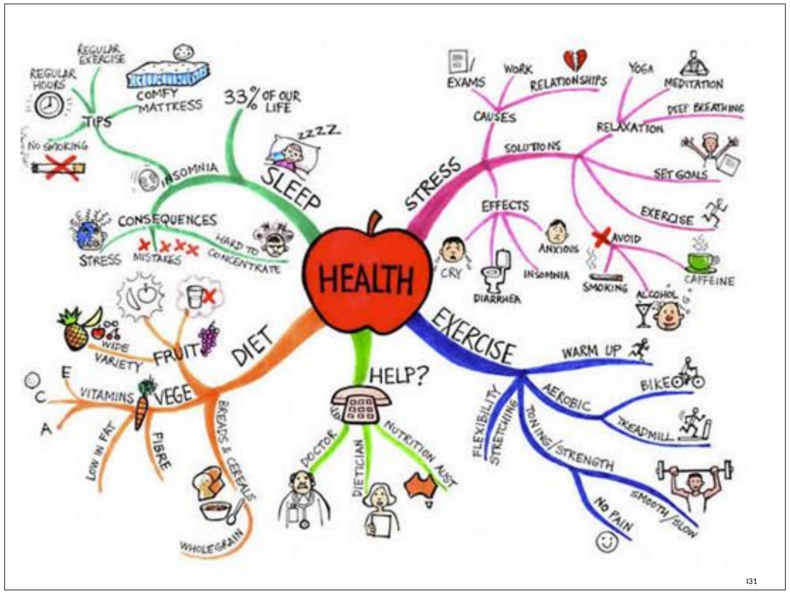
329



LeSS **LASD** Programme

- LeSS LASD: **Lean & Agile Software Development**
- **LASD-1** to **LASD-5**
  - 3.5 weeks over “3 months”, 2 teams per ‘wave’
  - from **Impact Mapping** to **Story Mapping** to **User-Centered Design** to **Specification-by-Example** to **Acceptance TDD** to **Agile Modeling & Design Patterns** to **Agile Modeling Design Workshops** to **Clean Code** to **Legacy TDD** to **Current-Architecture Agile Learning workshop**

330

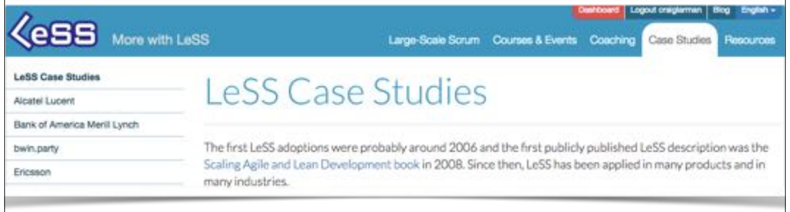


331

LeSS Stories

332

3 Types & the Case Studies



LeSS Case Studies

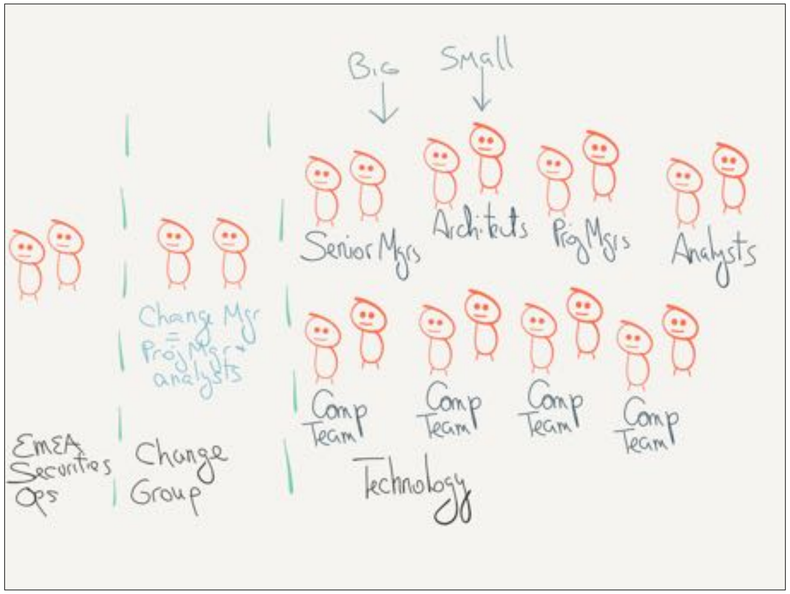
The first LeSS adoptions were probably around 2006 and the first publicly published LeSS description was the Scaling Agile and Lean Development book in 2008. Since then, LeSS has been applied in many products and in many industries.

333

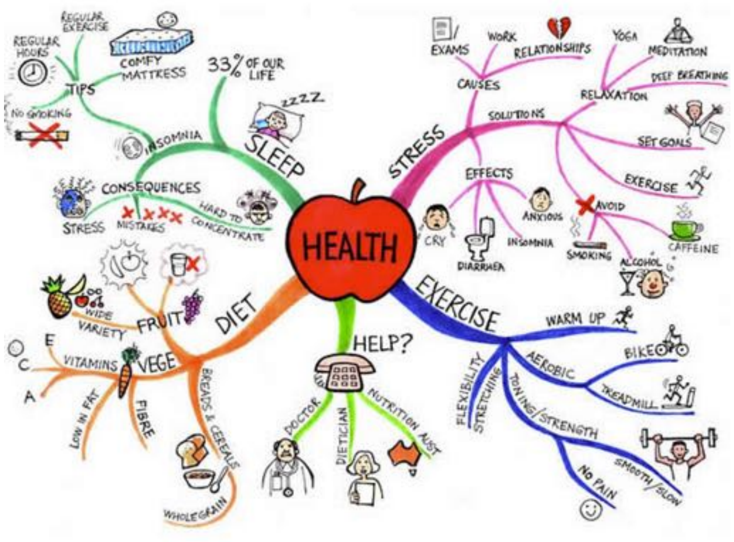
type & fireside story…

334

335



336



337

Systems Thinking

338

what are we about to learn?

339

Systems Thinking

a SYSTEM

see the whole, especially its dynamics across time

optimize the whole (not local optimization or local efficiency)

see delays

see mental models

understand system dynamics with causal loop models

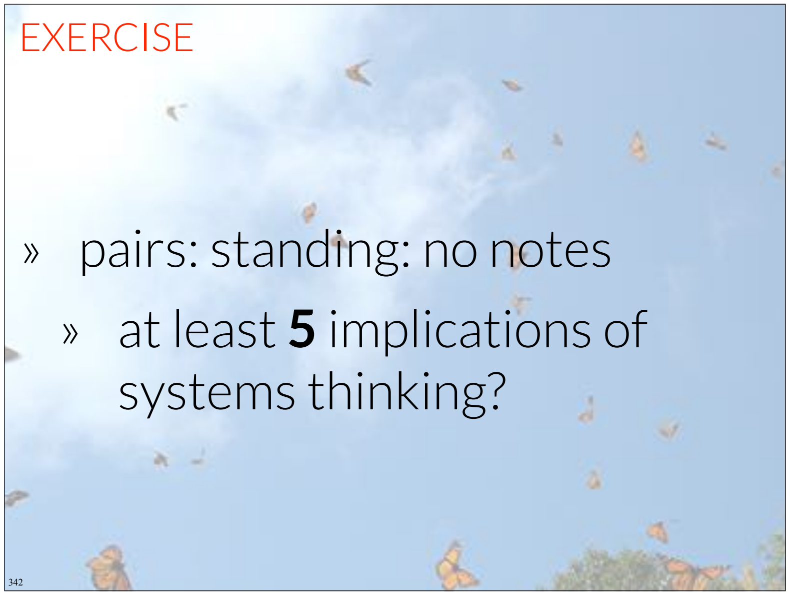
340

watch the ball

optimize the whole

not the players

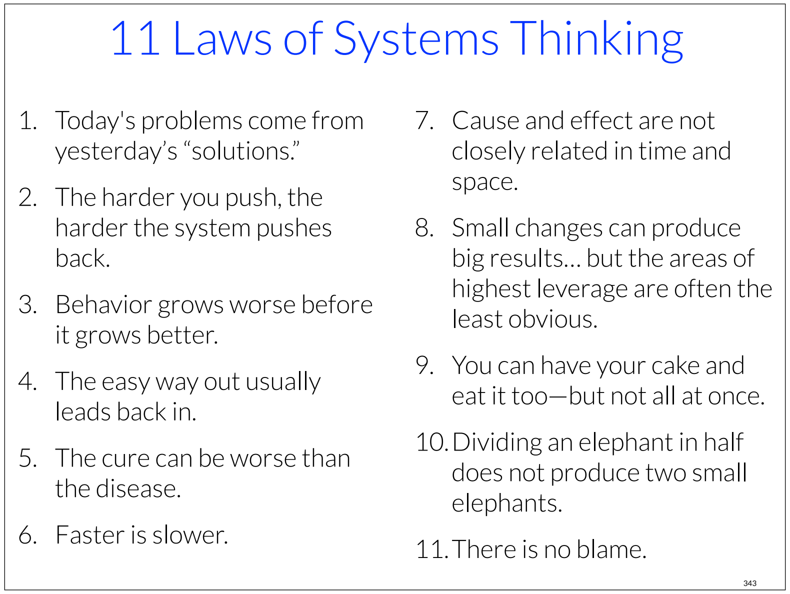
341



EXERCISE

- pairs: standing: no notes
  - at least **5** implications of systems thinking?

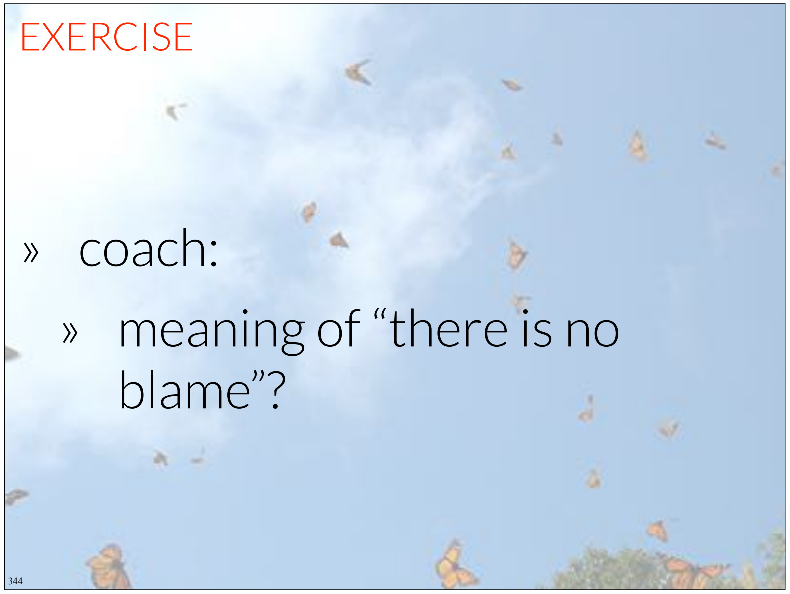
342



11 Laws of Systems Thinking

1. Today's problems come from yesterday's “solutions.”
2. The harder you push, the harder the system pushes back.
3. Behavior grows worse before it grows better.
4. The easy way out usually leads back in.
5. The cure can be worse than the disease.
6. Faster is slower.
7. Cause and effect are not closely related in time and space.
8. Small changes can produce big results… but the areas of highest leverage are often the least obvious.
9. You can have your cake and eat it too—but not all at once.
10. Dividing an elephant in half does not produce two small elephants.
11. There is no blame.

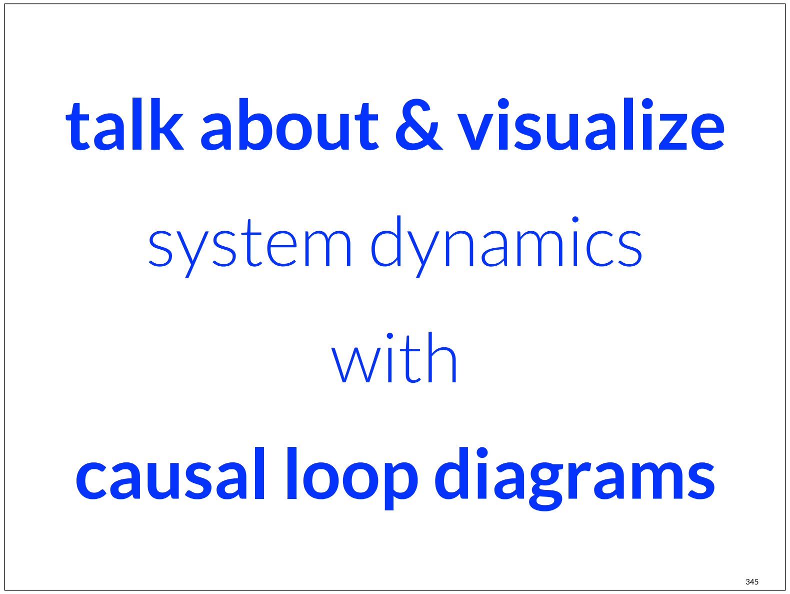
343



EXERCISE

- coach:
  - meaning of “there is no blame”?

344



talk about & visualize

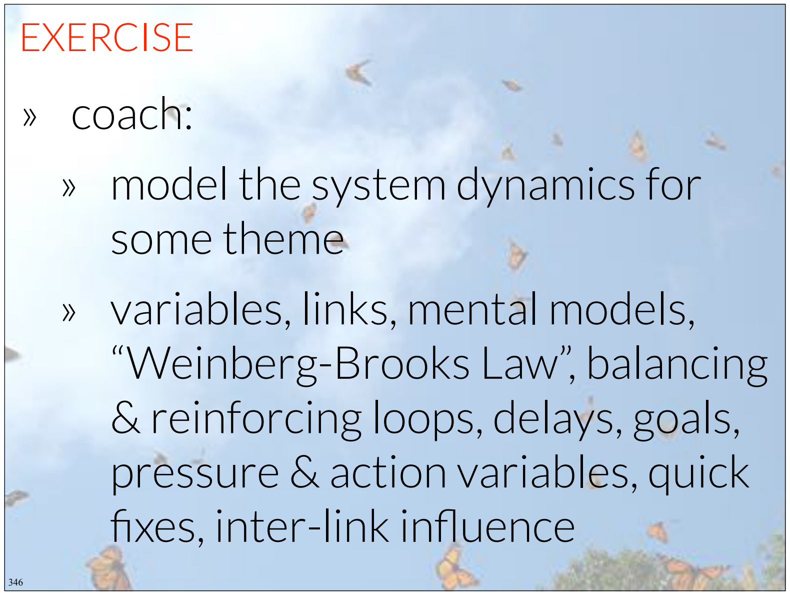
system dynamics

with

causal loop diagrams

345

345

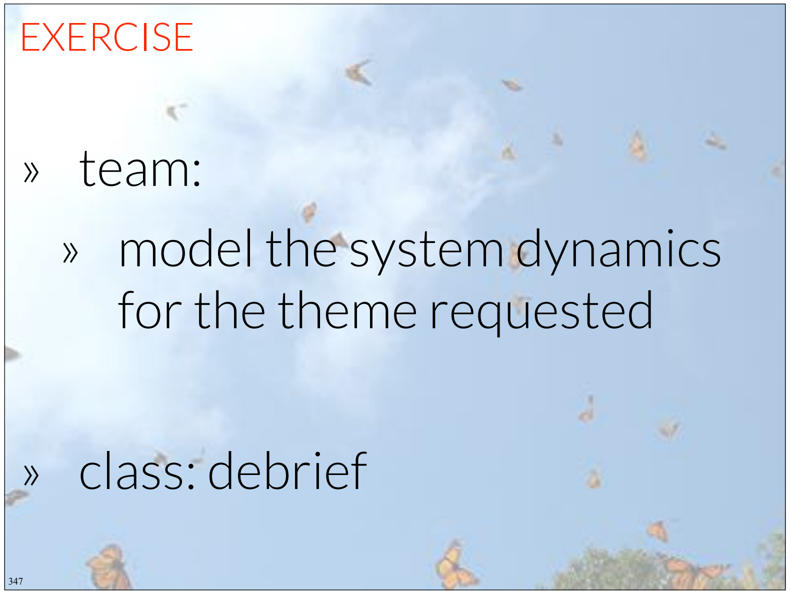


EXERCISE

- coach:
  - model the system dynamics for some theme
  - variables, links, mental models, “Weinberg-Brooks Law”, balancing & reinforcing loops, delays, goals, pressure & action variables, quick fixes, inter-link influence

346

346

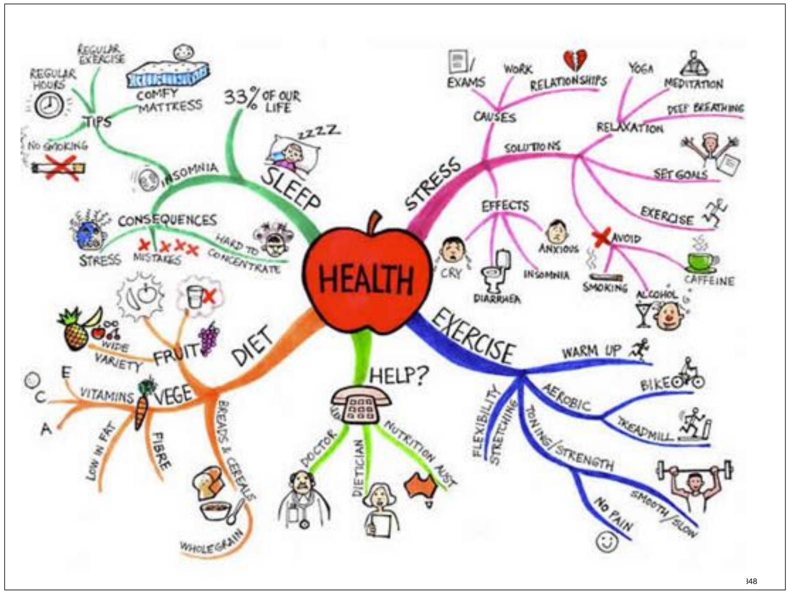


EXERCISE

- team:
  - model the system dynamics for the theme requested
- class: debrief

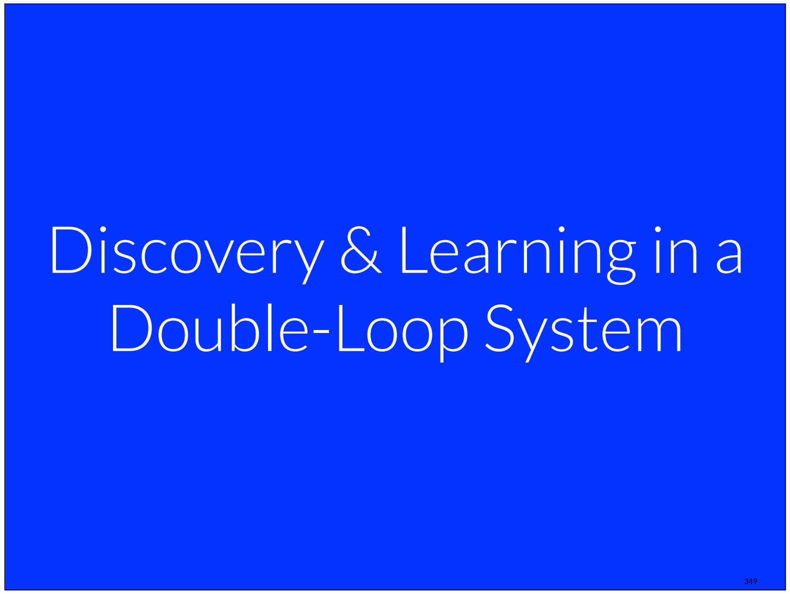
347

347



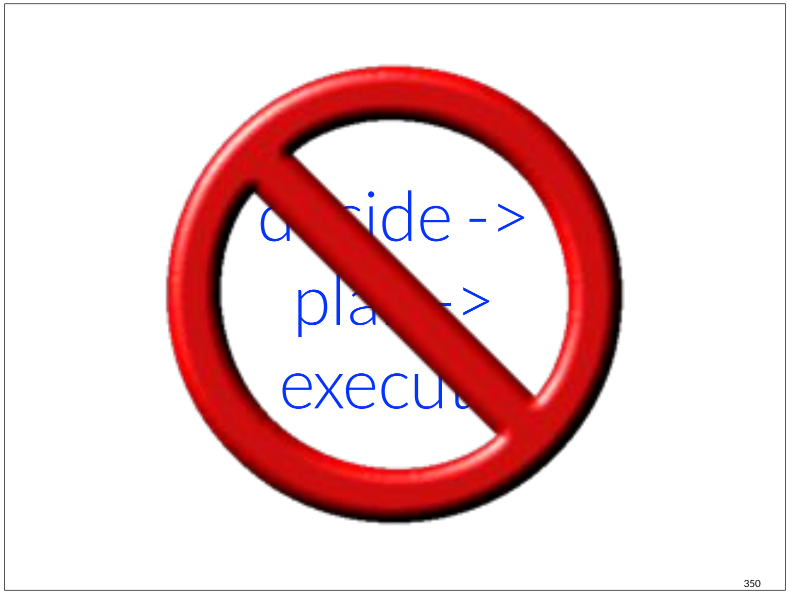
348

348



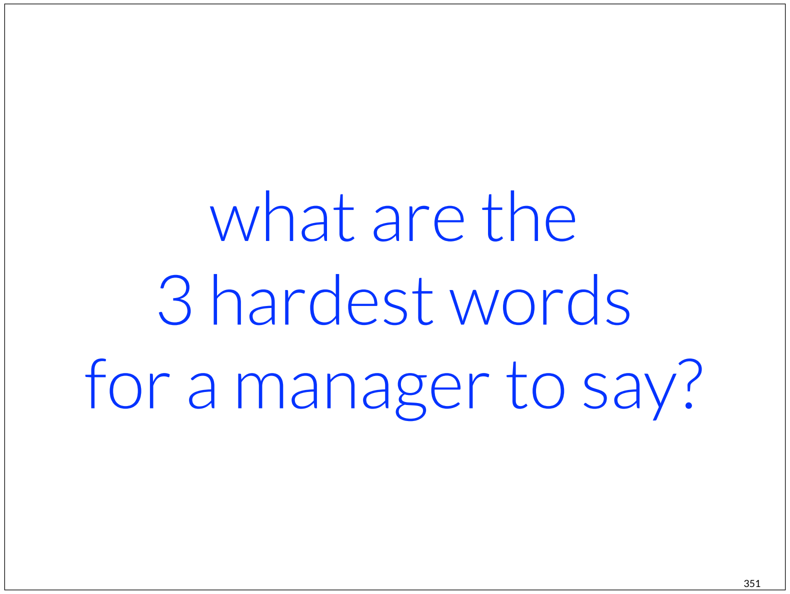
Discovery & Learning in a Double-Loop System

349



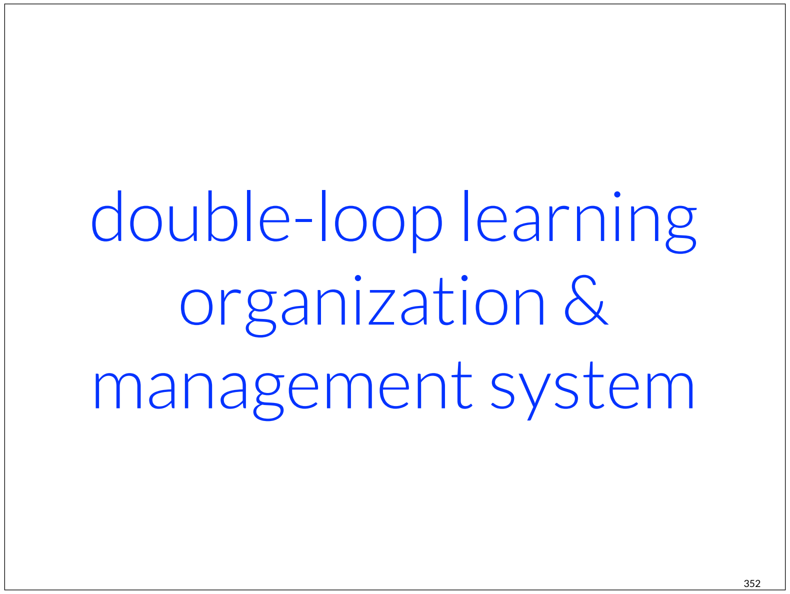
d ide ->
pla ->
execu

350



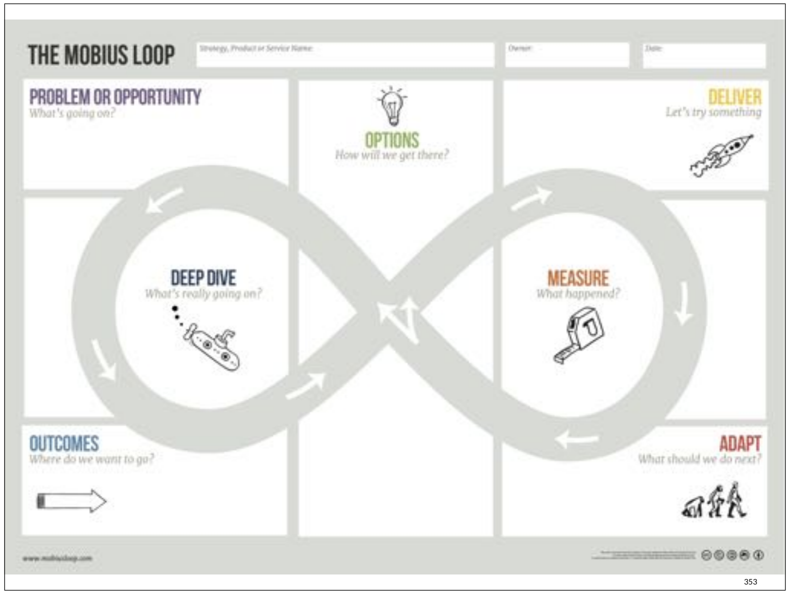
what are the
3 hardest words
for a manager to say?

351



double-loop learning
organization &
management system

352



THE MOBIUS LOOP

Strategy, Product or Service Name:

Owner:

Date:

PROBLEM OR OPPORTUNITY
What's going on?

OPTIONS
How will we get there?

DELIVER
Let's try something

DEEP DIVE
What's really going on?

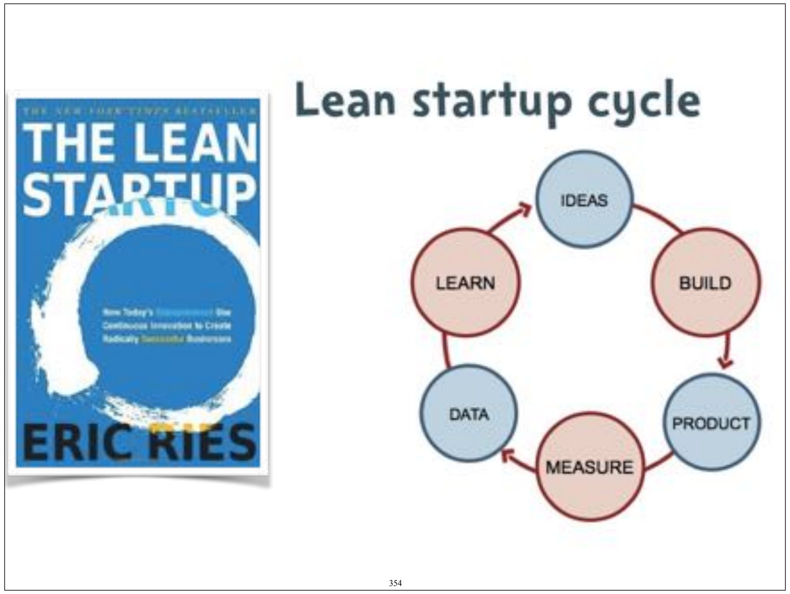
MEASURE
What happened?

OUTCOMES
Where do we want to go?

ADAPT
What should we do next?

www.mobiusloop.com

353



Lean startup cycle

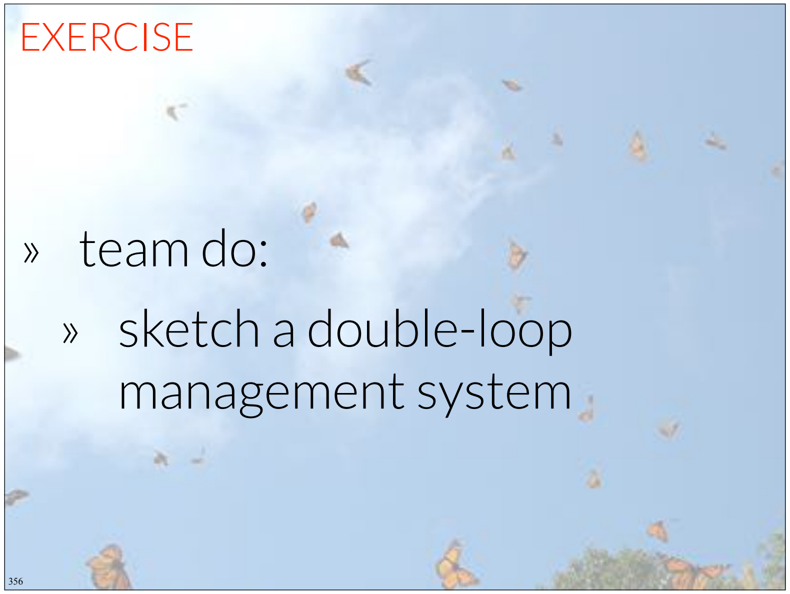
THE NEW YORK TIMES BESTSELLER
THE LEAN STARTUP
How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses
ERIC RIES

IDEAS
BUILD
PRODUCT
MEASURE
DATA
LEARN

354

the scientific method applied to business

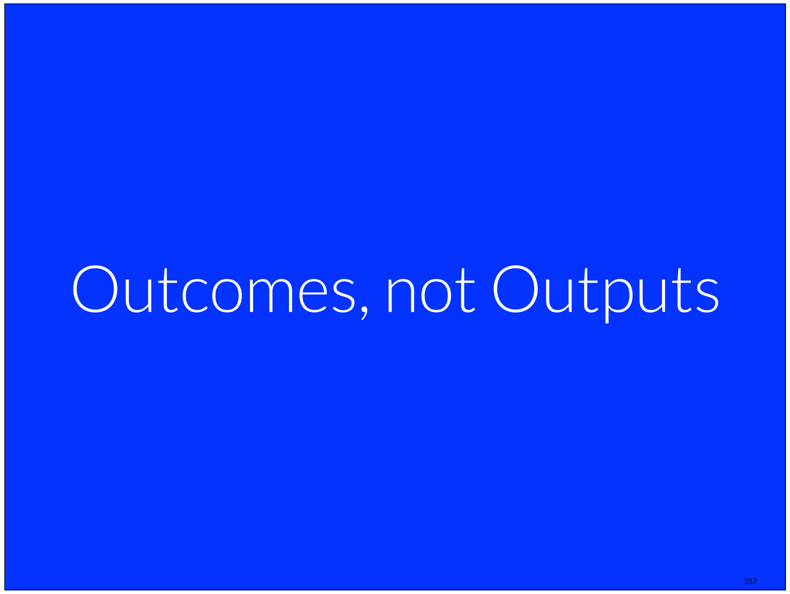
355



EXERCISE

- team do:
  - sketch a double-loop management system

356



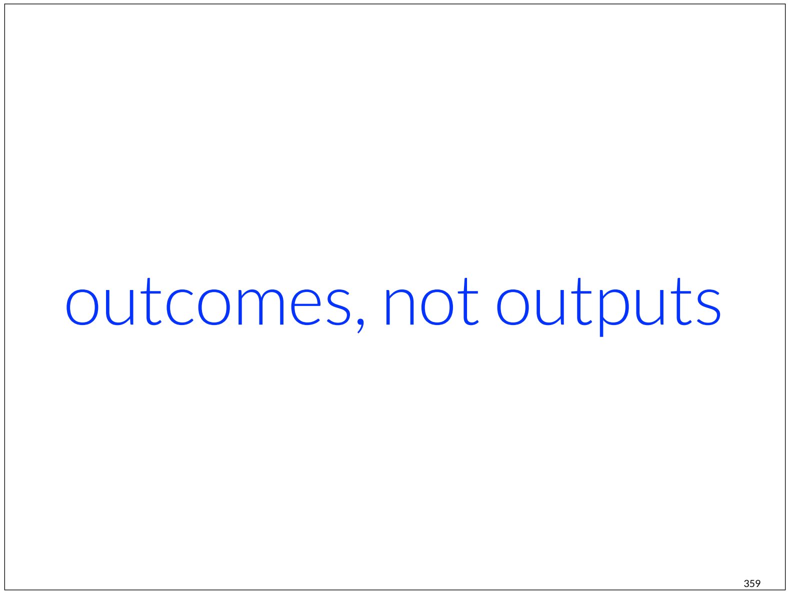
Outcomes, not Outputs

357

features
stories
epics

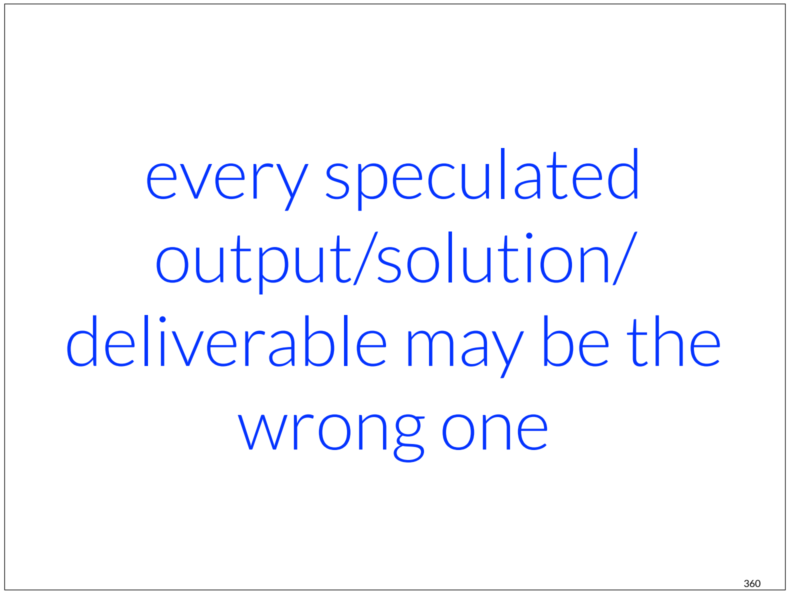
shopping list of "the solutions"

358



outcomes, not outputs

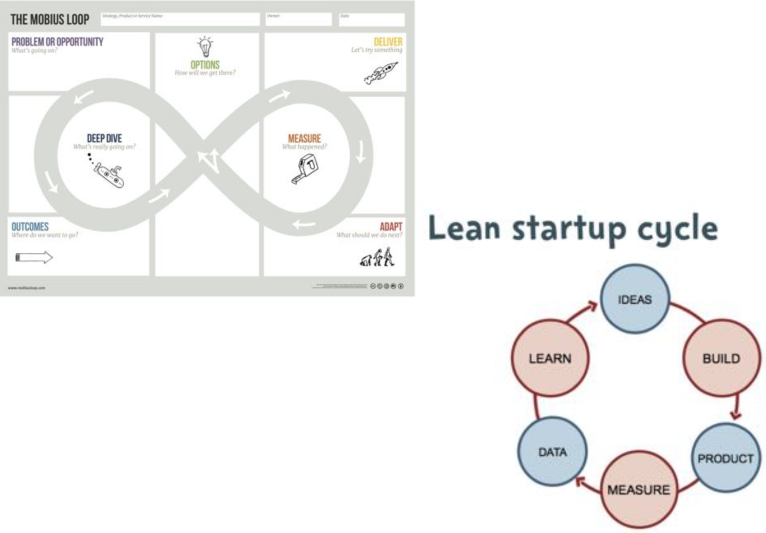
359



every speculated output/solution/ deliverable may be the wrong one

360

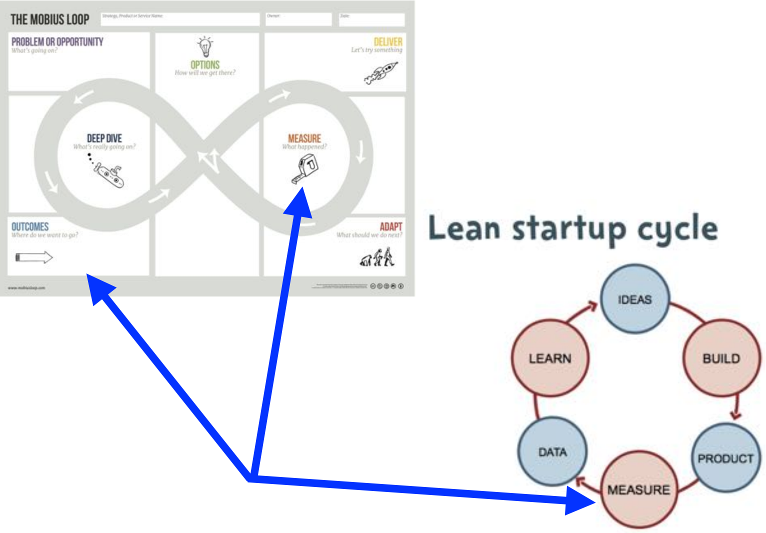
Outcomes, not Outputs in a Discovery & Learning Double-Loop System



outcomes, not outputs

Impact Mapping: **Always** with Others

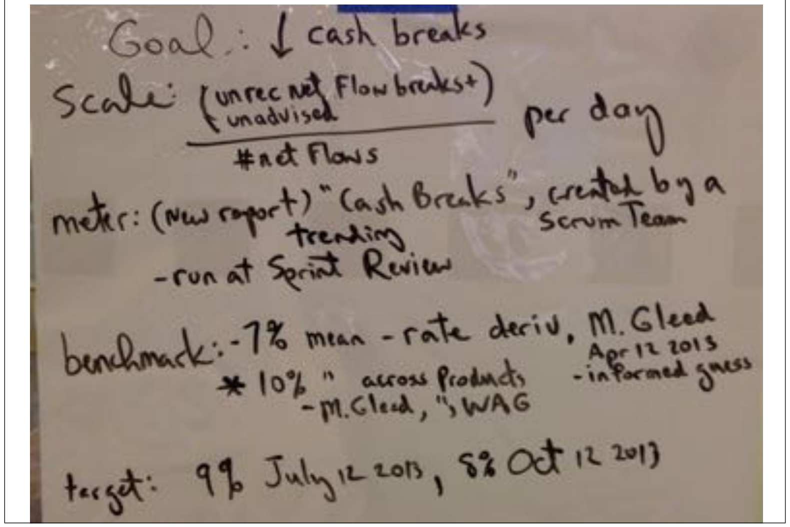
365



Lean startup cycle

366

outcome: quantification



Goal: ↓ cash breaks

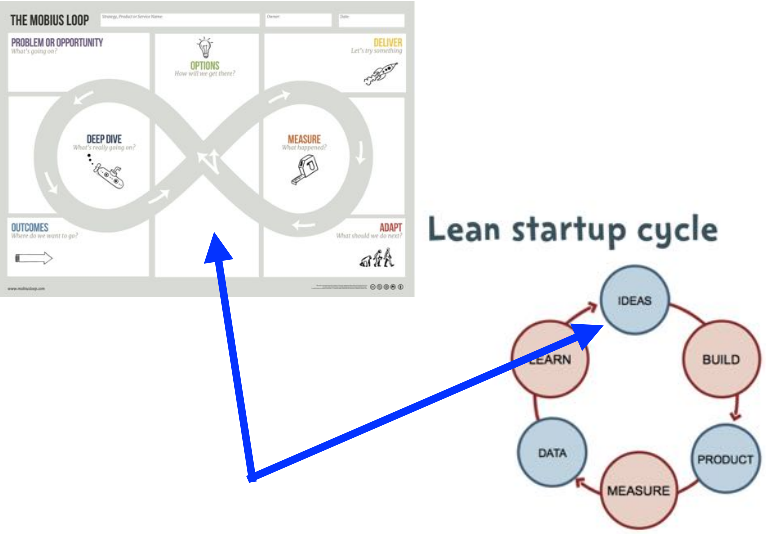
Scale: (unrec net Flow breaks + unadvised) / #net Flows per day

meter: (New report) "Cash Breaks", created by a Scrum Team
trending
- run at Sprint Review

benchmark: -7% mean - rate deriv, M. Gleed Apr 12 2013
* 10% " across Products - M. Gleed, ", WAG - informed guess

target: 9% July 12 2013, 8% Oct 12 2013

367



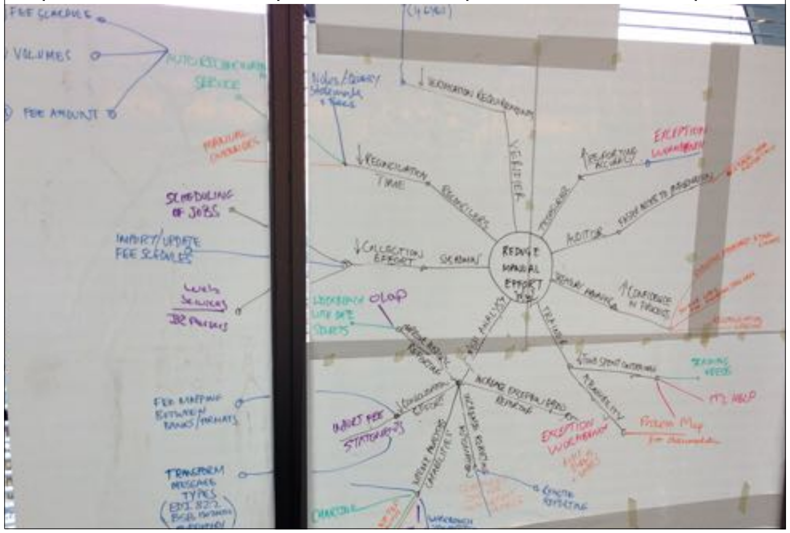
Lean startup cycle

368

Speculated Options: Impacts & Outputs

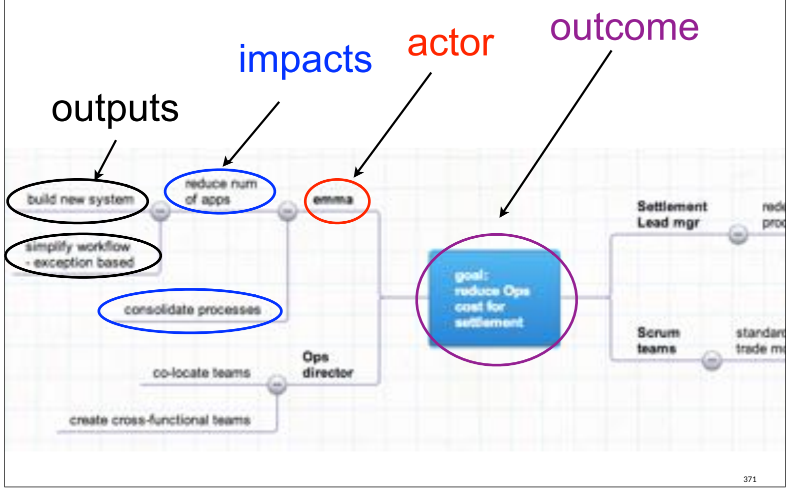
369

Speculated Options: Impacts & Outputs



370

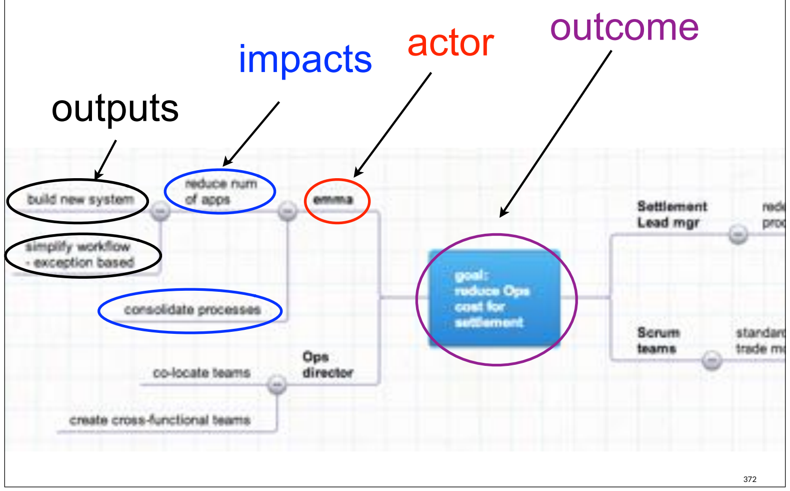
Impact Map Elements



371

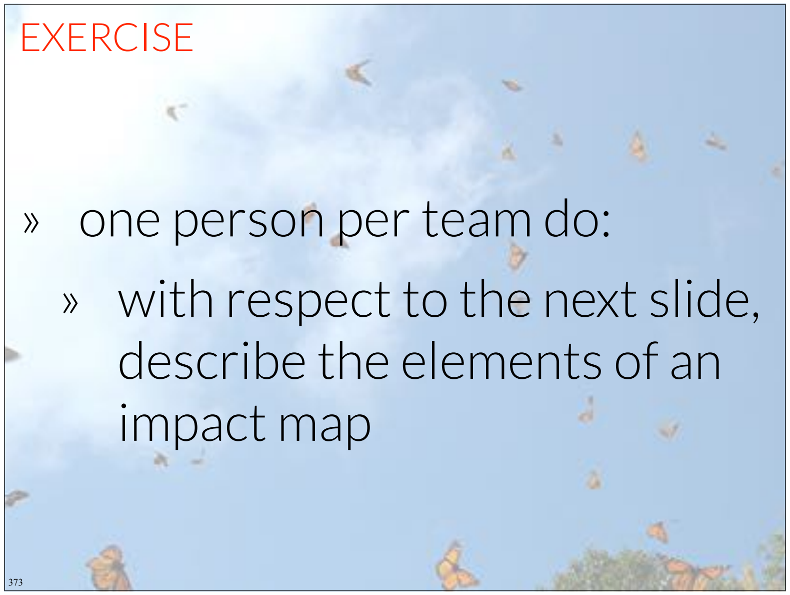
371

Many Speculated Impacts & Outputs



372

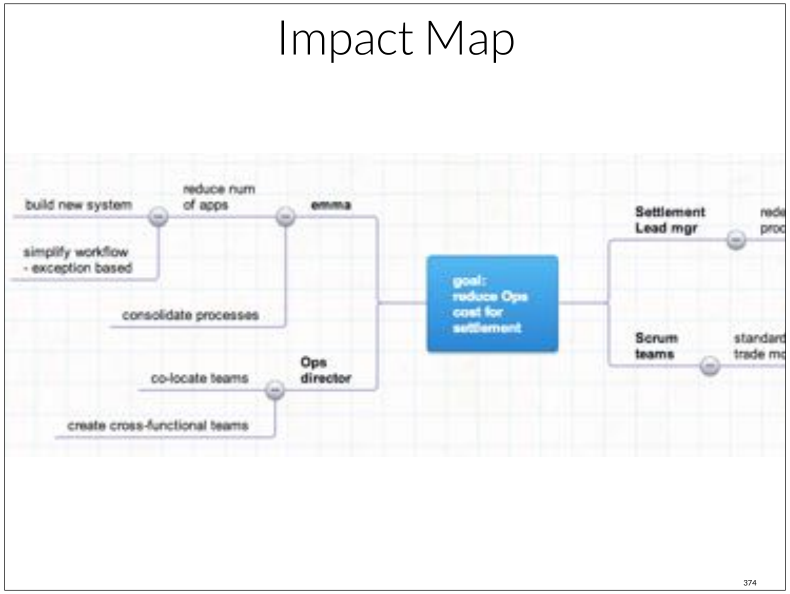
372



EXERCISE

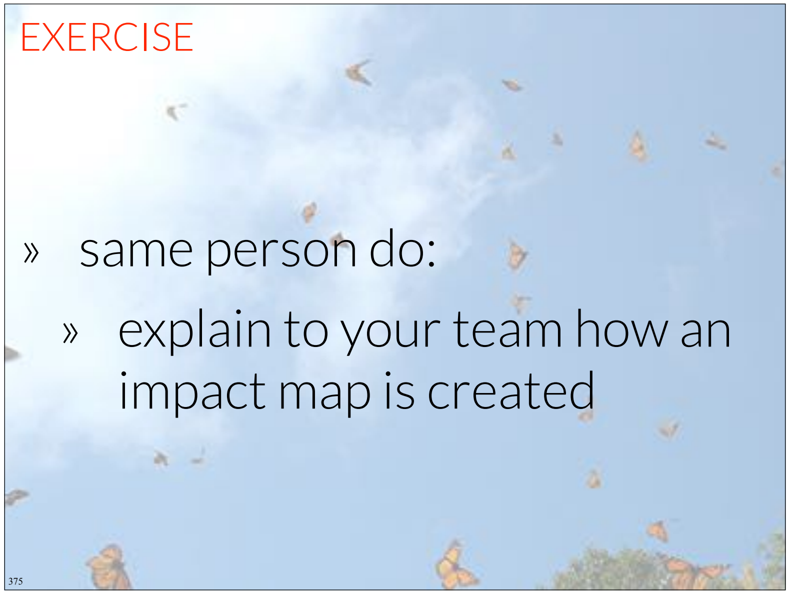
- one person per team do:
  - with respect to the next slide, describe the elements of an impact map

373



Impact Map

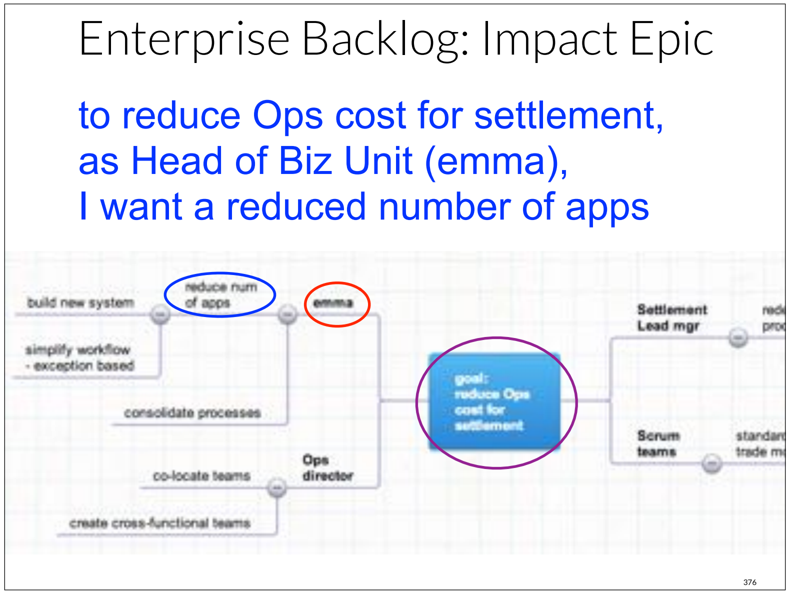
374



EXERCISE

- same person do:
  - explain to your team how an impact map is created

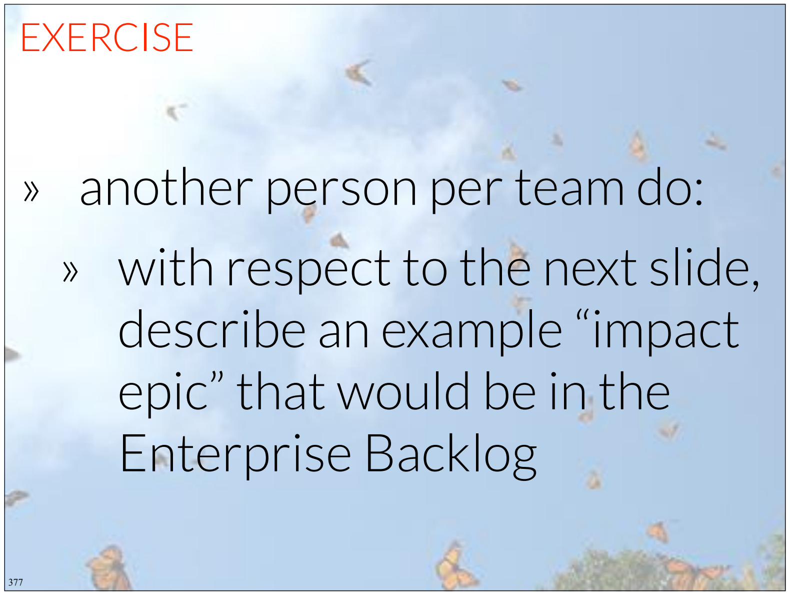
375



Enterprise Backlog: Impact Epic

to reduce Ops cost for settlement,
as Head of Biz Unit (emma),
I want a reduced number of apps

376



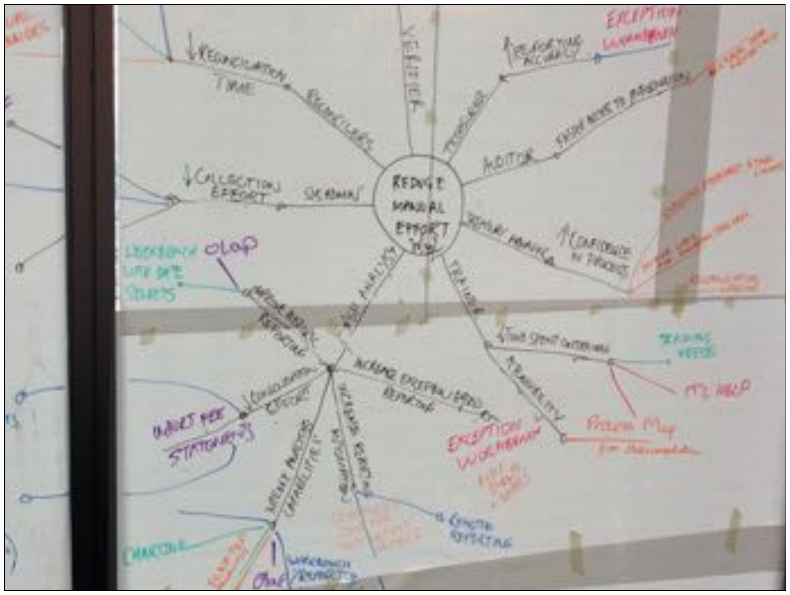
EXCEPTION

EXERCISE

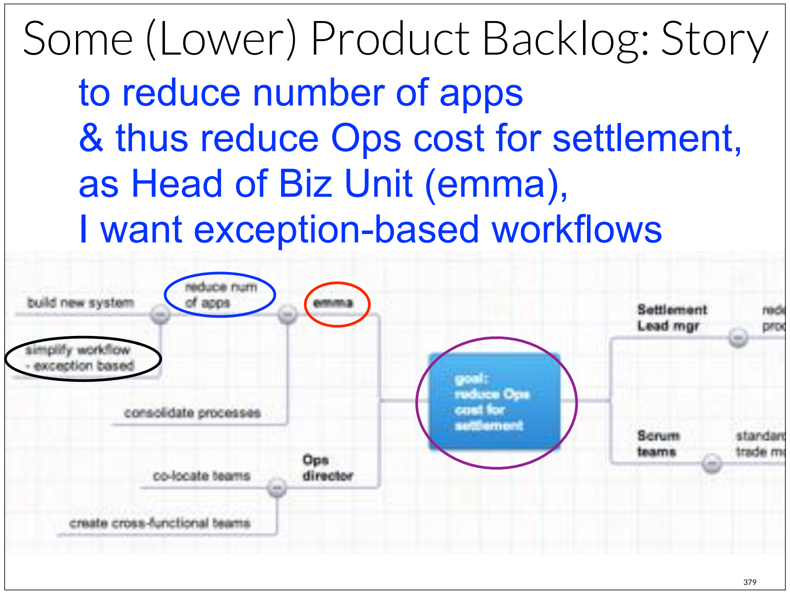
- another person per team do:
  - with respect to the next slide, describe an example "impact epic" that would be in the Enterprise Backlog

377

377



378



Some (Lower) Product Backlog: Story

to reduce number of apps
& thus reduce Ops cost for settlement,
as Head of Biz Unit (emma),
I want exception-based workflows

build new system

reduce num of apps

emma

simplify workflow - exception based

consolidate processes

goal: reduce Ops cost for settlement

Settlement Lead mgr

Scrum teams

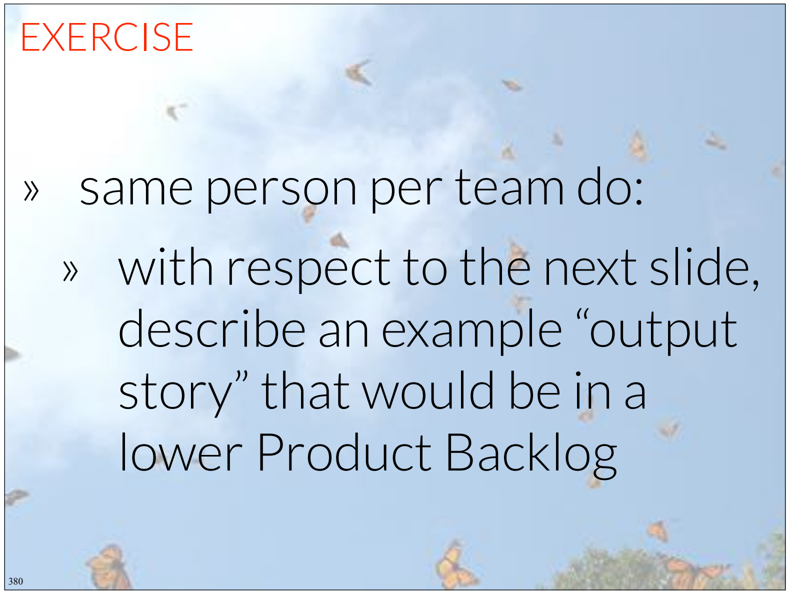
Ops director

co-locate teams

create cross-functional teams

379

379

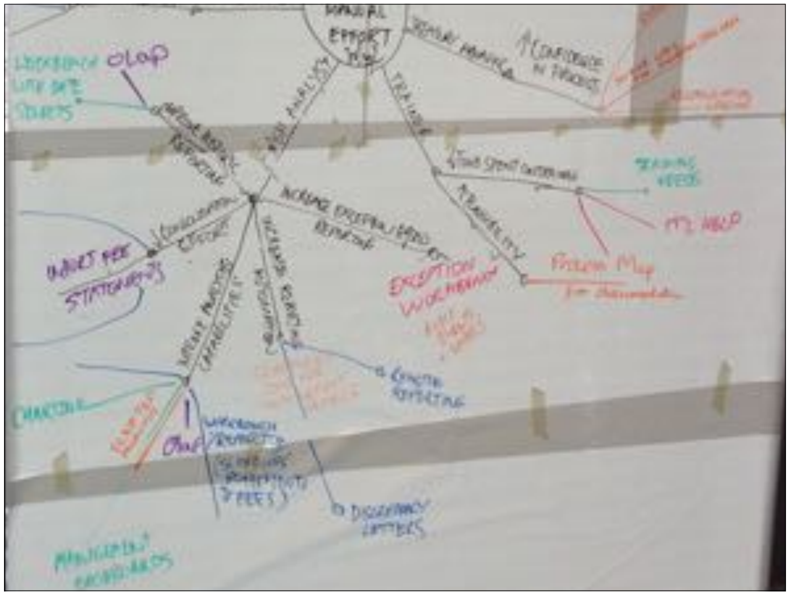


EXERCISE

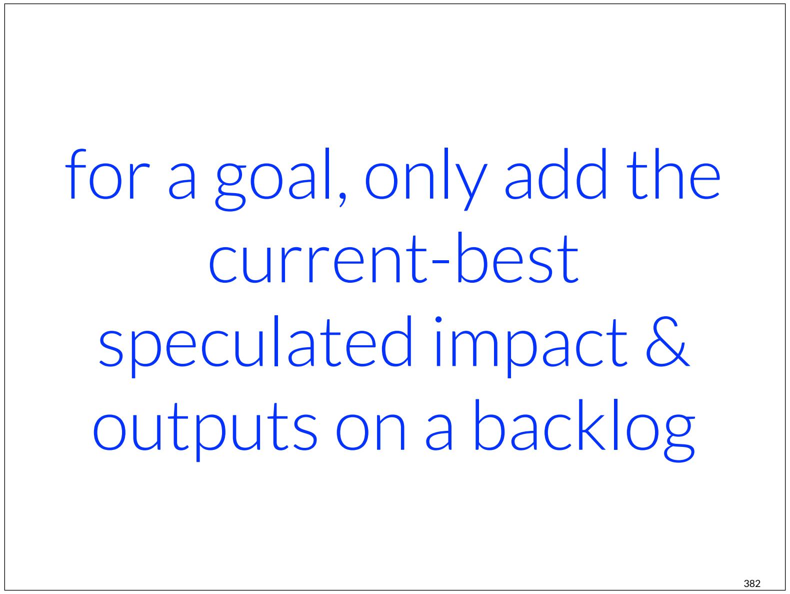
- same person per team do:
  - with respect to the next slide, describe an example "output story" that would be in a lower Product Backlog

380

380

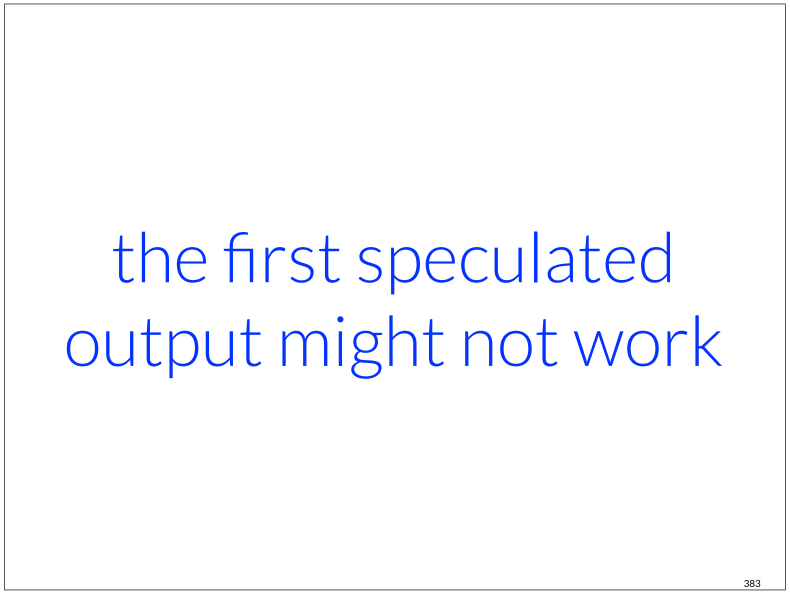


381



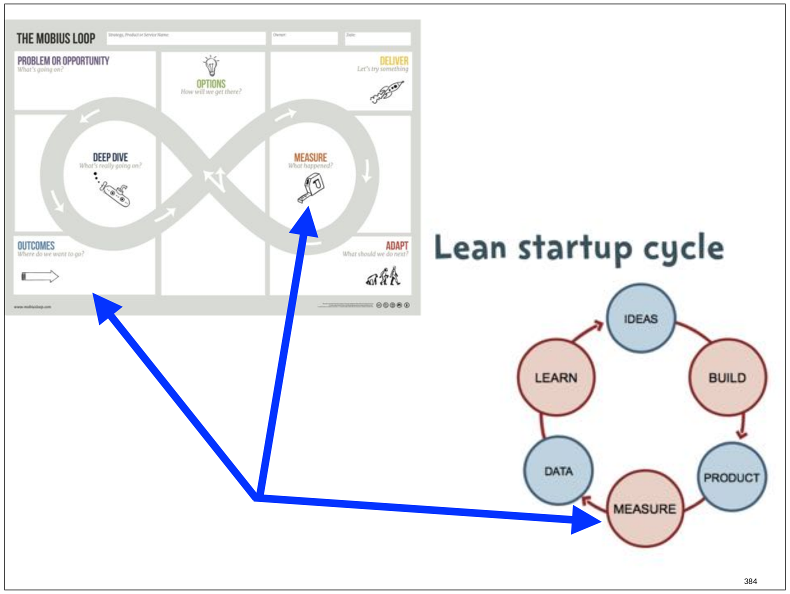
for a goal, only add the current-best speculated impact & outputs on a backlog

382



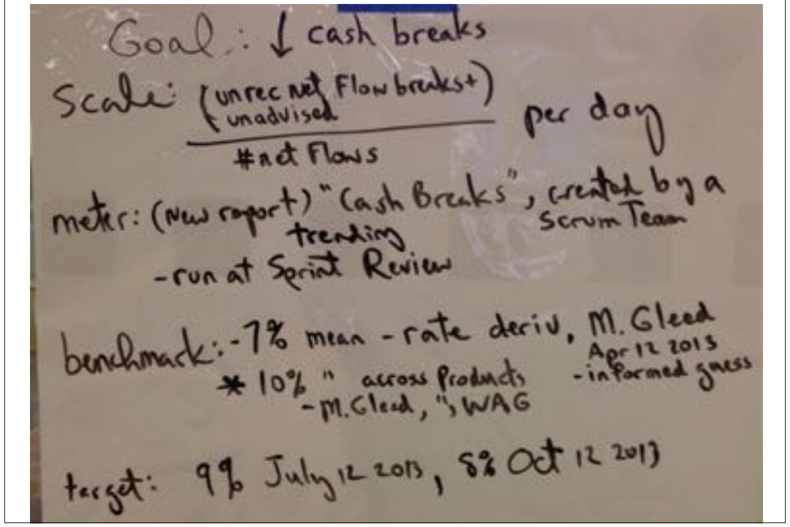
the first speculated output might not work

383



384

impact/outcome: measurement



Goal: ↓ cash breaks

Scale: (unrec net Flow breaks+ / unadvised) / #net Flows per day

meter: (New report) "Cash Breaks", created by a Scrum Team
trending
-run at Sprint Review

benchmark: -7% mean -rate deriv, M. Gleed Apr 12 2013 -informed guess
* 10% " across Products -M.Gleed, ", WAG

target: 9% July 12 2013, 8% Oct 12 2013

385

every Sprint, you measure the impact of delivering an output

this requires delivering early and often, and measuring impacts

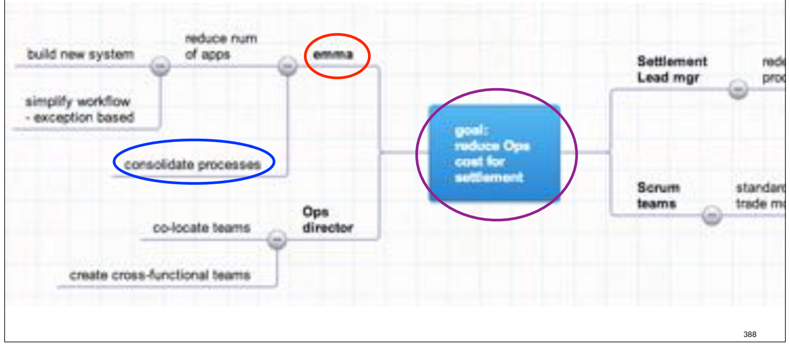
386

the speculated output didn't work,
so…

387

Enterprise Backlog: Next Alternative

to reduce Ops cost for settlement,
as Head of Biz Unit (emma),
I want consolidated processes



388

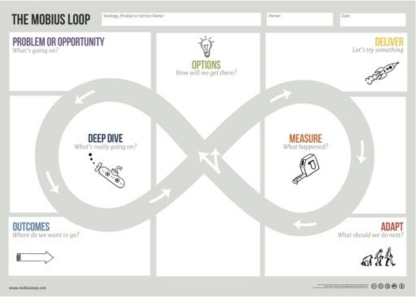
1. de case

2. prove

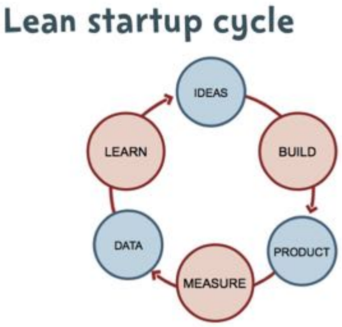
3. pla

4. e

389



learning & discovery & change



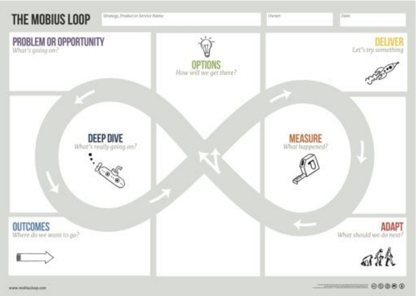
390

Beyond Budgeting

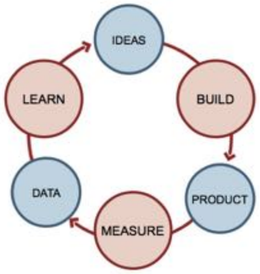
391

traditional “annual operating budget” financial control systems inhibit…

392



Lean startup cycle



learning & discovery & change

there's a growing body of CFOs and other financial leaders who recognize there is another way…

Beyond Budgeting Institute
From Command & Control to Empower & Adapt

Tel +44 20 3755 3692

About Membership News Events Resources Education Partnerships Contact
Log In

JOIN THE BEYOND BUDGETING ROUND TABLE
BBRT Membership is very valuable. It offers support and a shared learning network of active evaluaters, implementers and users of the Beyond Budgeting management model.
Read More

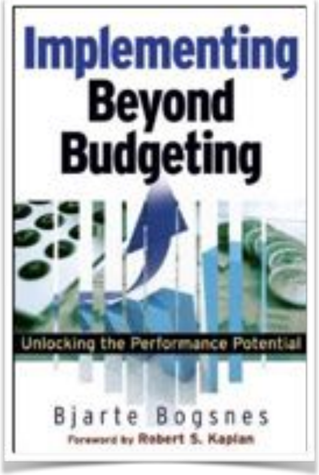
Sign Up Today
Subscribe to our newsletter and stay updated on developments in and around our network
Sign Up Now
Should you have any questions, feel free to call us on +44 20 3755 3692

Our Global Beyond Budgeting Network
MAERSK SKF Statnett HILTI VOLVO

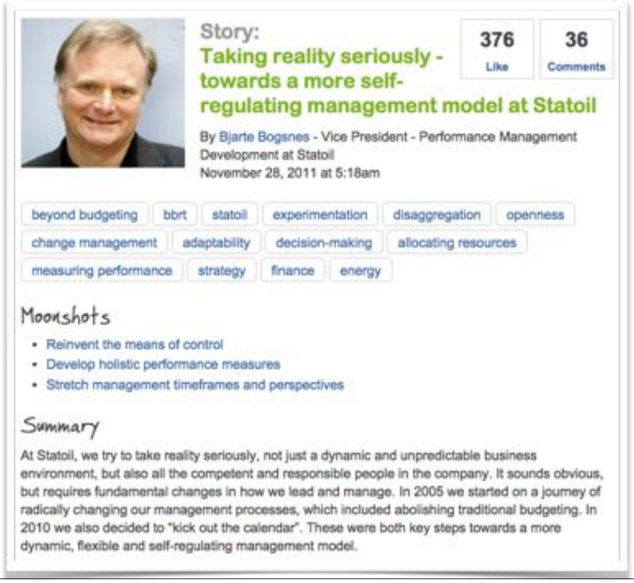
Latest News
New Beyond Budgeting Partner in Iceland
10 March, 2015
The case for moving beyond traditional budgeting
10 March, 2015
Next European Members' Meeting
10 February, 2015
Beyond Budgeting in Financial Times

Upcoming Events
6th Controlling, Performance & Reporting Forum
19 March, 2015
Agile Management Congress
24 March, 2015
Beyond Budgeting
8 April, 2015
Beyond Budgeting Workshop
10 April, 2015

Join the BBRT
Since its foundation in 1998, the BBRT has built a strong global membership programme comprising a broad network of leading organisations, universities, business professionals, academics and thought leaders. BBRT Membership ...



Implementing Beyond Budgeting
Unlocking the Performance Potential
Bjarte Bogsnes
Foreword by Robert S. Kaplan



Story:

Taking reality seriously - towards a more self-regulating management model at Statoil

By Bjarte Bogsnes - Vice President - Performance Management Development at Statoil
November 28, 2011 at 5:18am

Moonshots

- Reinvent the means of control
- Develop holistic performance measures
- Stretch management timeframes and perspectives

Summary

At Statoil, we try to take reality seriously, not just a dynamic and unpredictable business environment, but also all the competent and responsible people in the company. It sounds obvious, but requires fundamental changes in how we lead and manage. In 2005 we started on a journey of radically changing our management processes, which included abolishing traditional budgeting. In 2010 we also decided to "kick out the calendar". These were both key steps towards a more dynamic, flexible and self-regulating management model.

Beyond Budgeting: Sample Practices

Separate budget purposes: (1) target setting, (2) forecasting & (3) resource allocation

Redesign each on its base purpose

Remove calendar rhythms wherever possible (e.g., remove annual operating budget, ...)

Remove individual bonus

relative measures

rather than budget cost limits, monitor costs and adapt/intervene if necessary; educate and trust “financially responsible people”

Beyond Budgeting: Sample Principles

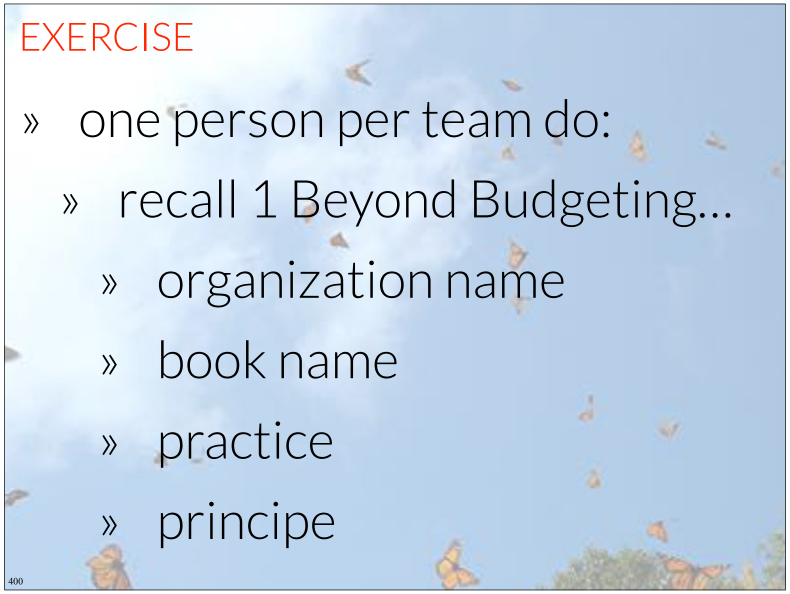
Trust teams to regulate their performance; don't micro-manage them

Make planning a continuous and inclusive process; not a top-down annual event

Coordinate interactions dynamically; not through annual budgets

Make resources available just-in-time; not just-in-case

Base controls on fast, frequent feedback; not budget variances



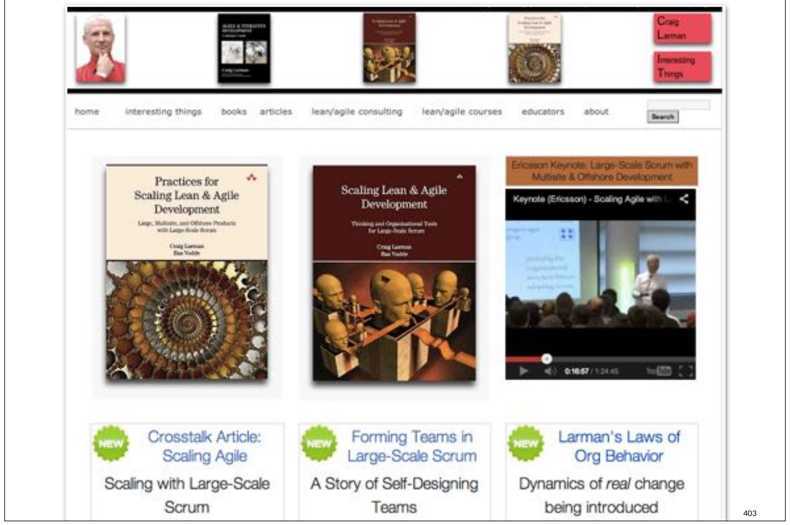
EXERCISE

- one person per team do:
  - recall 1 Beyond Budgeting...
    - organization name
    - book name
    - practice
    - principe

closing

thank you!

craiglarman.com



Practices for Scaling Lean & Agile Development

Large, Multisite, and Offshore Products with Large-Scale Scrum

Craig Larman
Bas Vodde

Scaling Lean & Agile Development

Thinking and Organizational Tools for Large-Scale Scrum

Craig Larman
Bas Vodde

Ericsson Keynote: Large-Scale Scrum with Multisite & Offshore Development

Keynote (Ericsson) - Scaling Agile with L

0:16:57 / 1:24:45

NEW Crosstalk Article: Scaling Agile

Scaling with Large-Scale Scrum

NEW Forming Teams in Large-Scale Scrum

A Story of Self-Designing Teams

NEW Larman's Laws of Org Behavior

Dynamics of *real* change being introduced