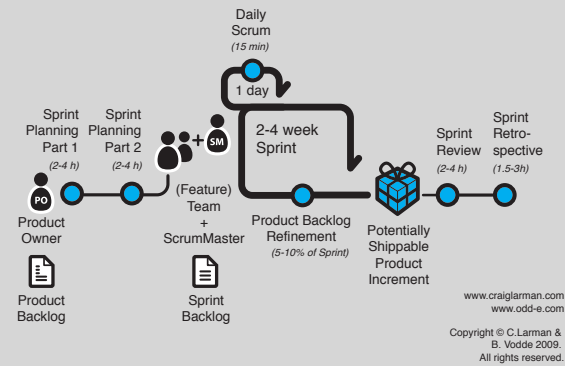
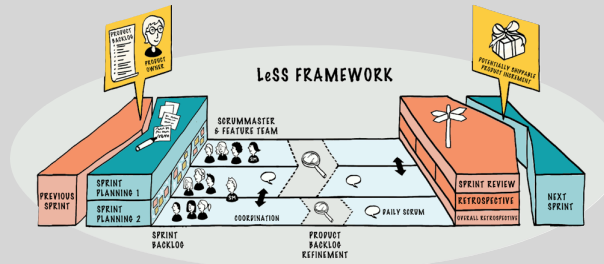


<eSS

1

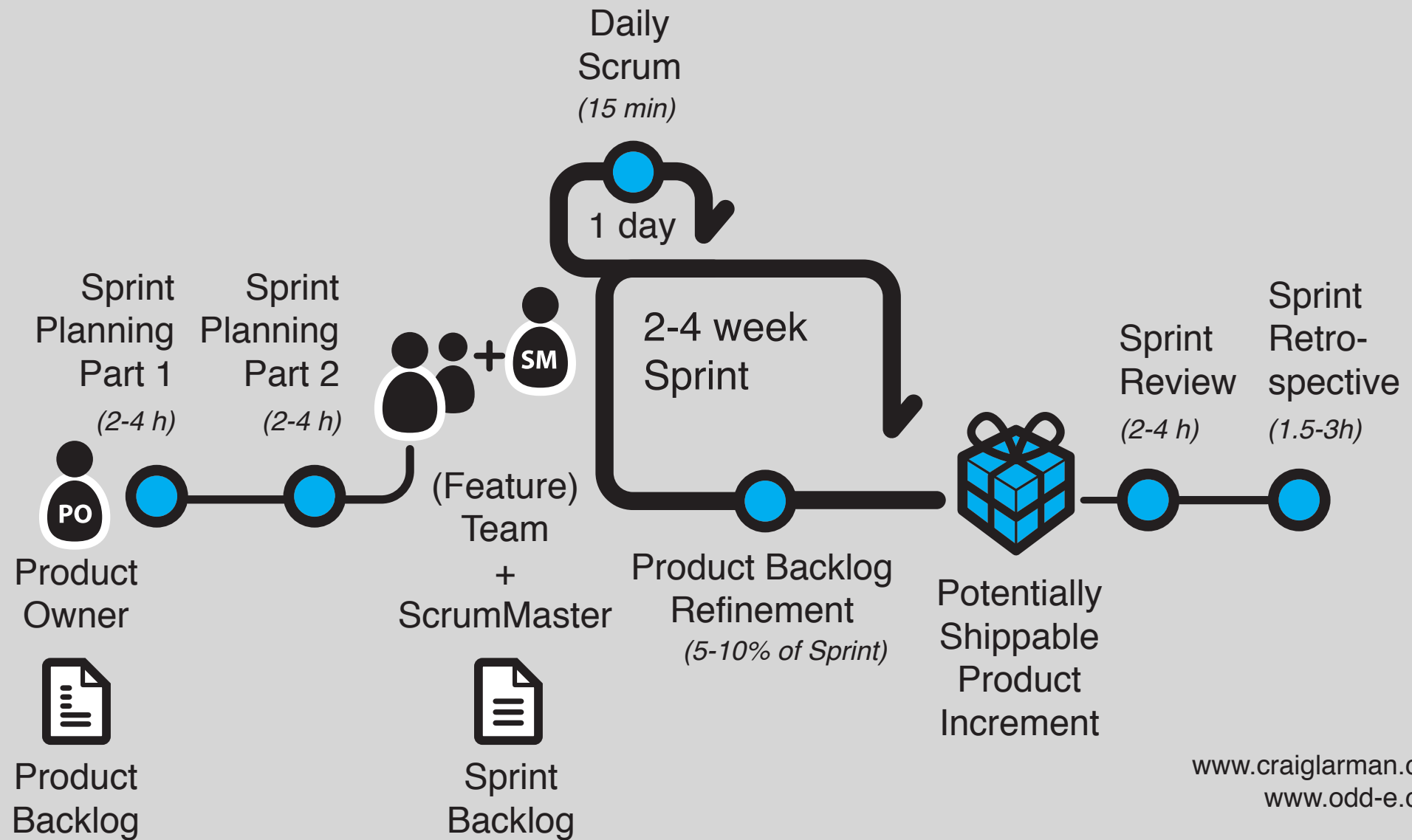


2



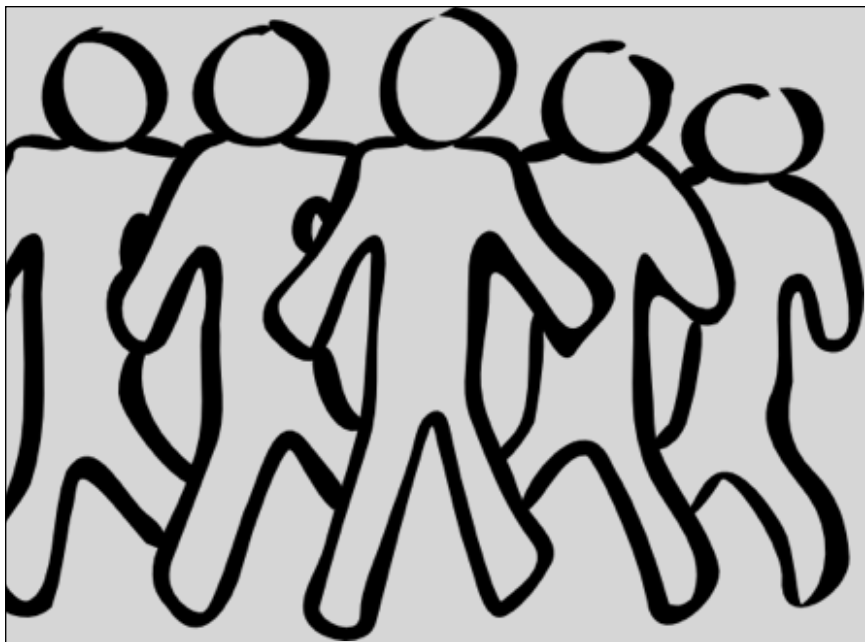
3

WHY?



www.craiglarman.com
www.odd-e.com

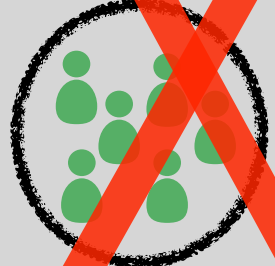
Copyright © C.Larman &
B. Vodde 2009.
All rights reserved.



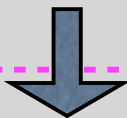
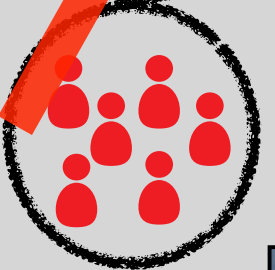
Architect



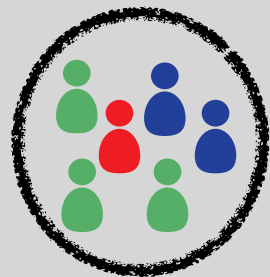
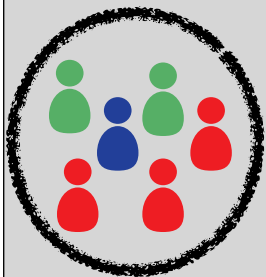
Dev.



Test

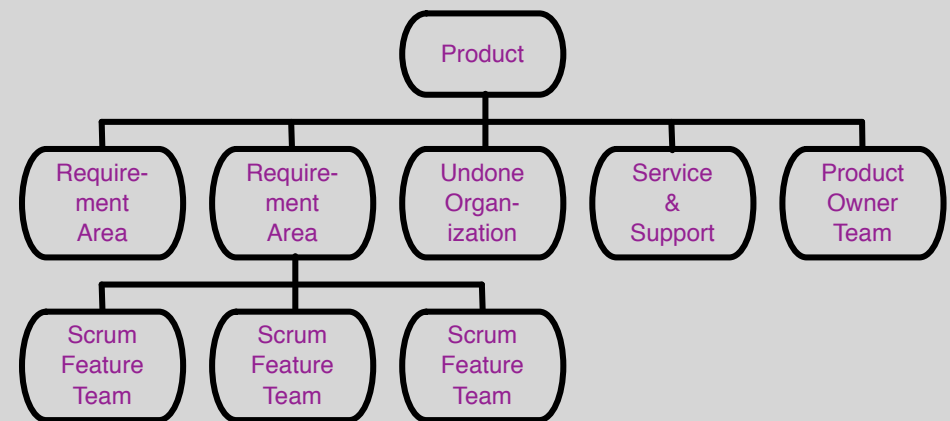
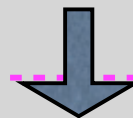
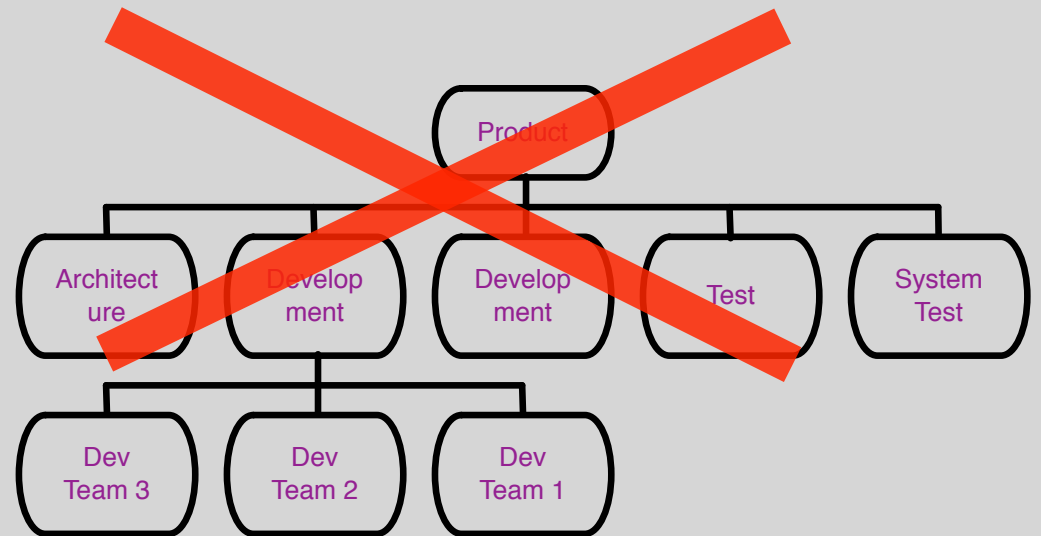


Team 2



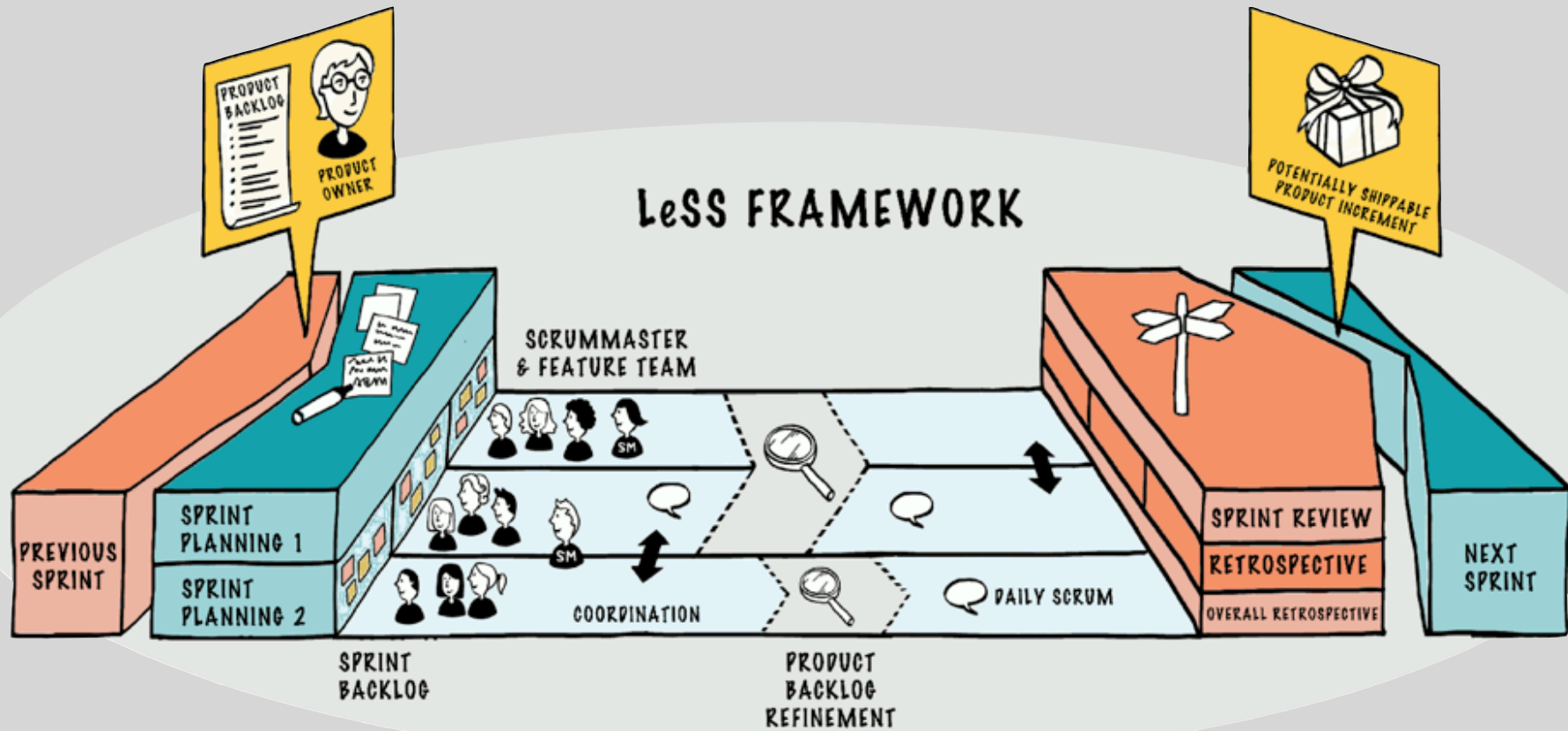
Team 1

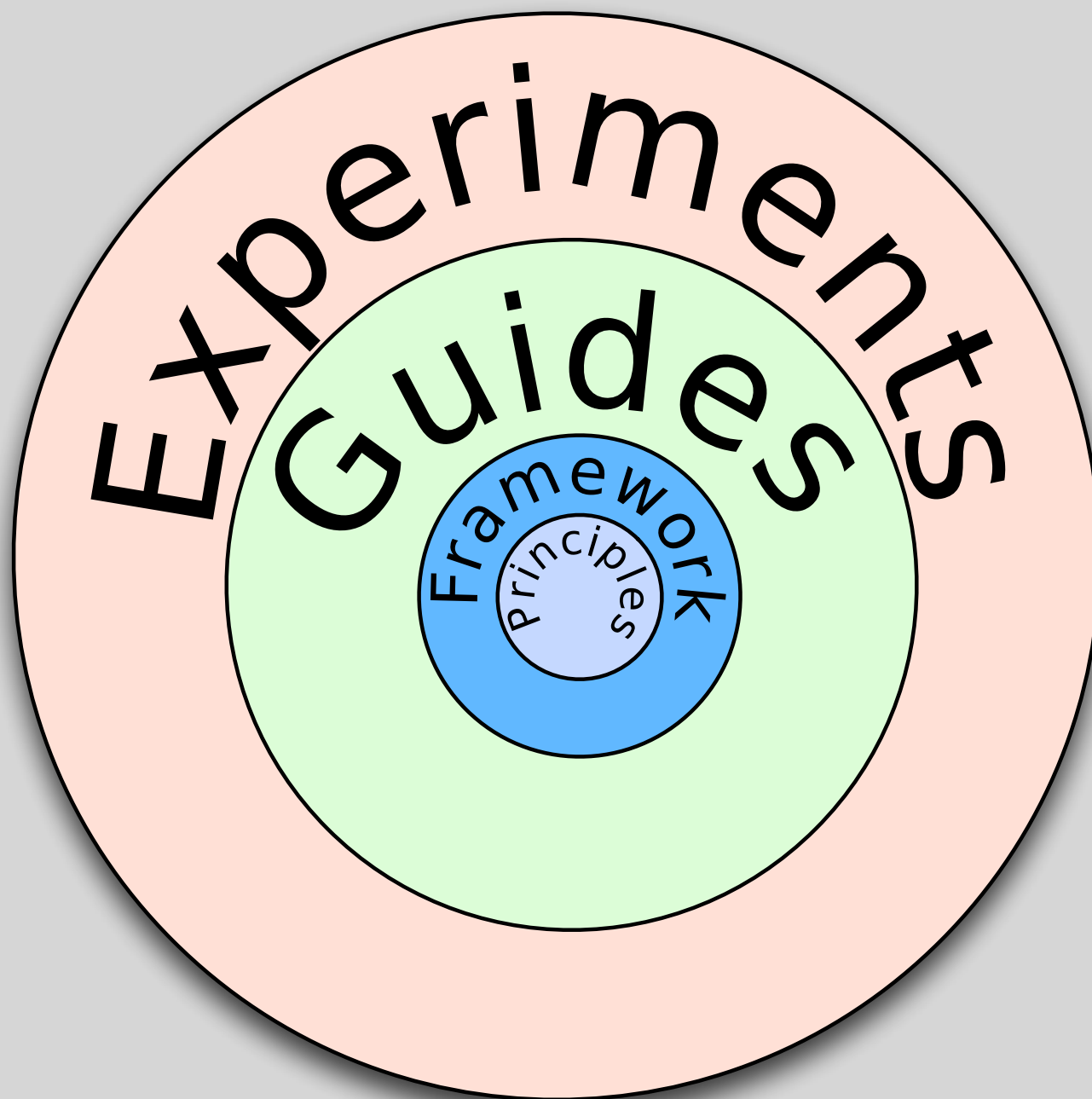
Team 3





LeSS FRAMEWORK















AGILE & ITERATIVE DEVELOPMENT

A Manager's Guide



Craig Larman

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



DEBUGGING THE DEVELOPMENT PROCESS

Practical
Strategies
for Staying
Focused,
Hitting Ship
Dates, and
Building
Solid Teams



STEVE MAGUIRE
Author of *Writing Solid Code*



Dynamics of Software Development



"Don't Flip the
Bezo Bit" and
53 More Rules
for Delivering
Great Software
on Time

Jim McCarthy

Foreword by Denis Gilbert,
Head Coach, Microsoft Visual C++

Microsoft Press

XP and Large Distributed Software Projects

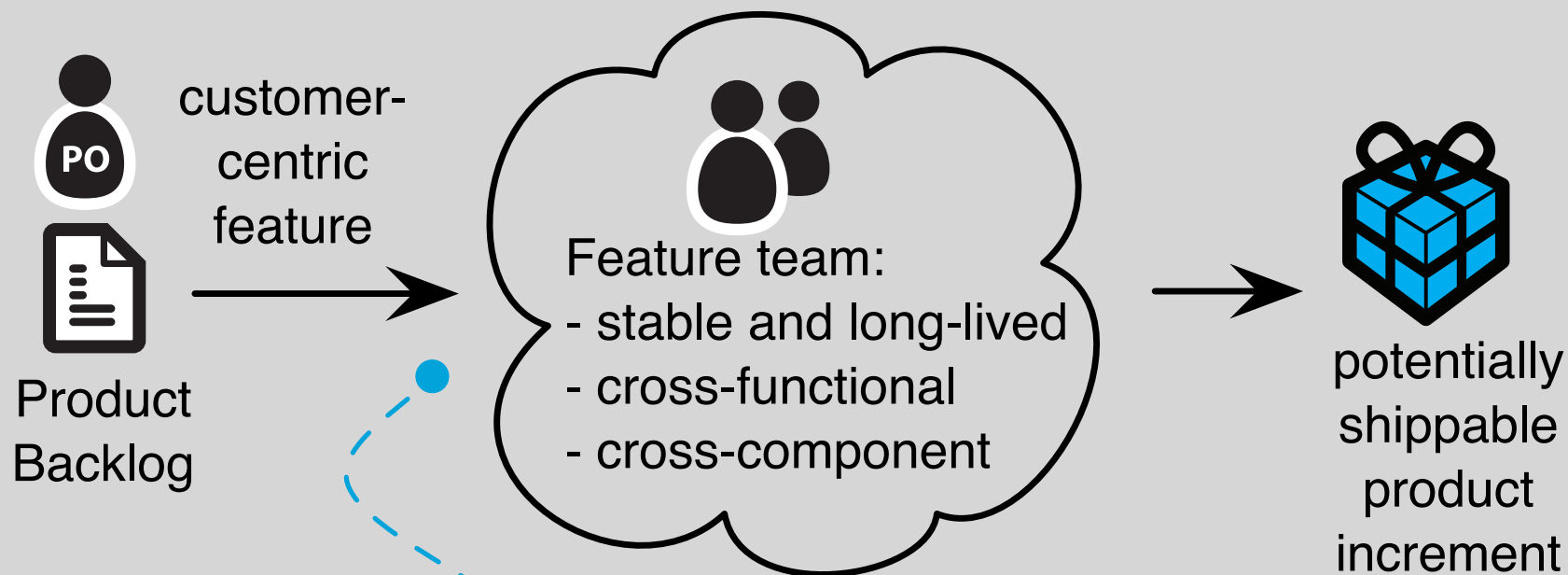
—Even-André Karlsson and Lars-Göran Andersson

XP ideas have mainly been used in small colocated projects. Within Ericsson's GSM division, however, projects are both large and distributed. In this chapter we discuss some experience that we have had on applying ideas related to XP—such as daily builds, frequent iterations, planning increments based on customer value, and automatic testing—to this type of project. We discuss how our approach relates to the original XP ideas. Furthermore we suggest some additional practices that we have found useful in this context. We also discuss how the aspects of XP that we have not tried could be used, and what effect they would have.

Component Teams



Feature teams



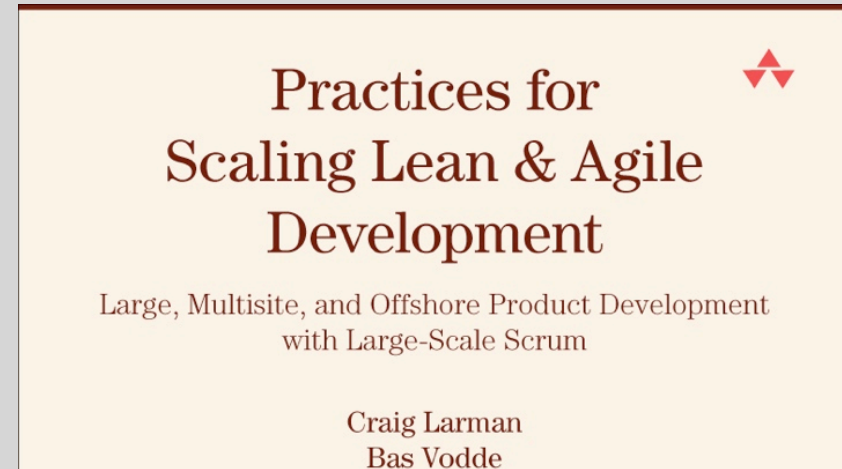
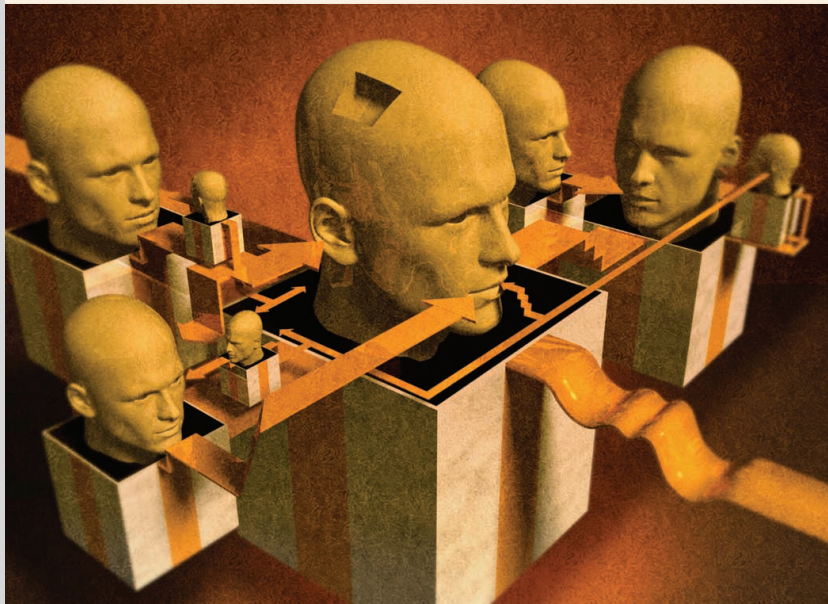
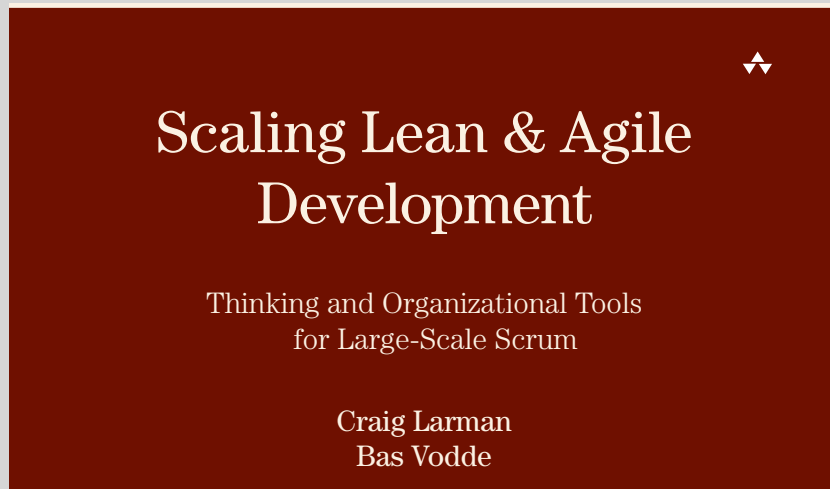
Team has the necessary knowledge and skills to complete an end-to-end customer-centric feature. If not, the team is expected to learn or acquire the needed knowledge and skill.

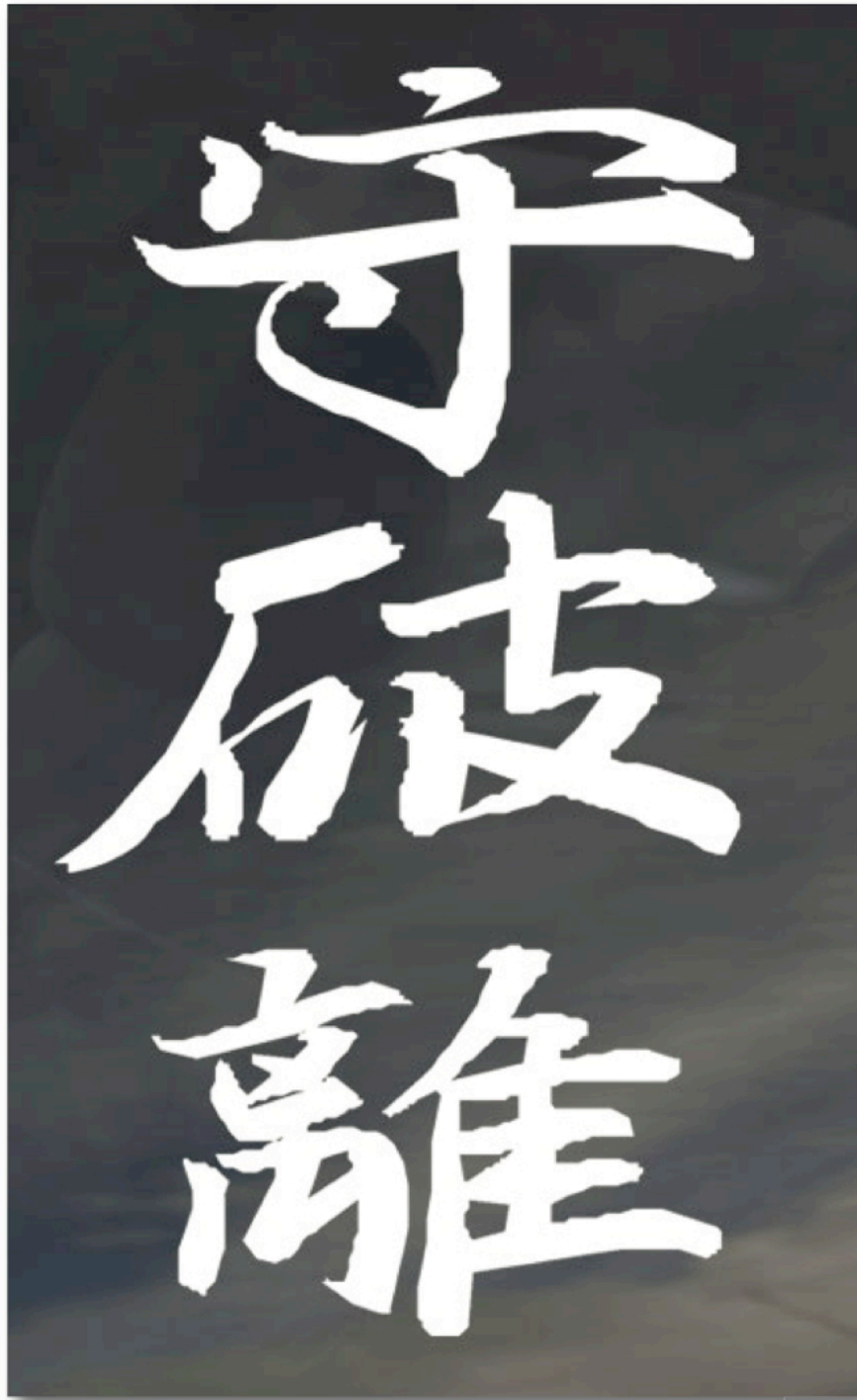
www.craiglarman.com
www.odd-e.com

Copyright © 2010
C.Larman & B. Vodde
All rights reserved.



LeSS Books





Shu
(following)

Ha
(breaking
away)

Ri
(mastery)

Prescription or Ownership?



LeSS Rules (October 2015)

LeSS Framework Rules

The LeSS framework applies to products with 2-“8” teams.

LeSS Structure

- Structure the organization using real teams as the basic organizational building block.
- Each team is (1) self-managing, (2) cross-functional, (3) co-located, and (4) long-lived.
- The majority of the teams are customer-focused feature teams.
- ScrumMasters are responsible for a well-working LeSS adoption. Their focus is towards the Teams, Product Owner, organization, and development practices. A ScrumMaster does not focus on just one team but on the overall organizational system.
- A ScrumMaster is a dedicated full-time role.
- One ScrumMaster can serve 1-3 teams.
- In LeSS, managers are optional, but if managers do exist their role is likely to change. Their focus is the value-delivering capability of the product development system rather than the specific scope of a product.

LeSS Guides

LeSS BOOK CHAPTER 2:
INTRODUCTION



LEAN
THINKING



SYSTEMS
THINKING



PRINCIPLES



COACHING



ADOPTION



CONTINUOUS
IMPROVEMENT



FEATURE
TEAMS



TEAMS



STRUCTURE



ORGANIZATION



COMMUNITIES



CONTINUOUS
INTEGRATION



TECHNICAL
EXCELLENCE



ARCHITECTURE
& DESIGN



ROLE OF
MANAGERS



MANAGEMENT



GO SEE



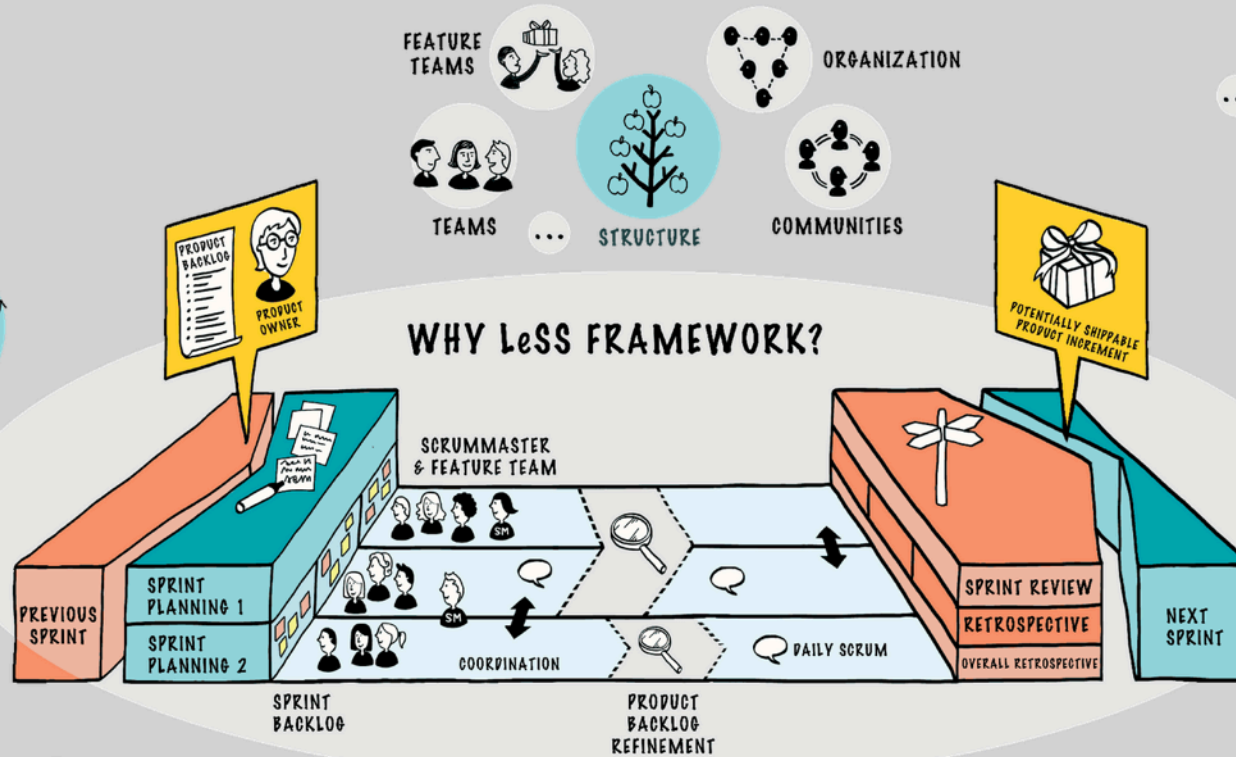
PRODUCT
OWNER TEAM



LESS HUGE



REQUIREMENT
AREAS

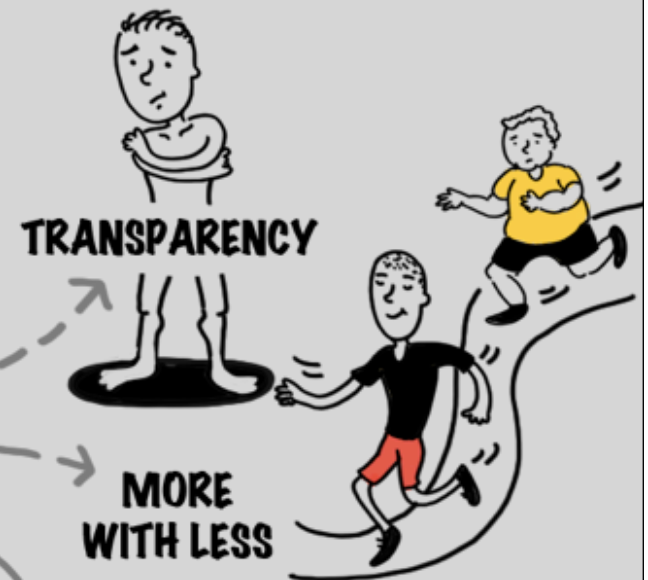




QUEUEING THEORY



LARGE-SCALE
SCRUM IS SCRUM



TRANSPARENCY

MORE
WITH LESS



EMPIRICAL
PROCESS CONTROL



SYSTEMS
THINKING



LEAN
THINKING



CONTINUOUS IMPROVEMENT
TOWARDS PERFECTION

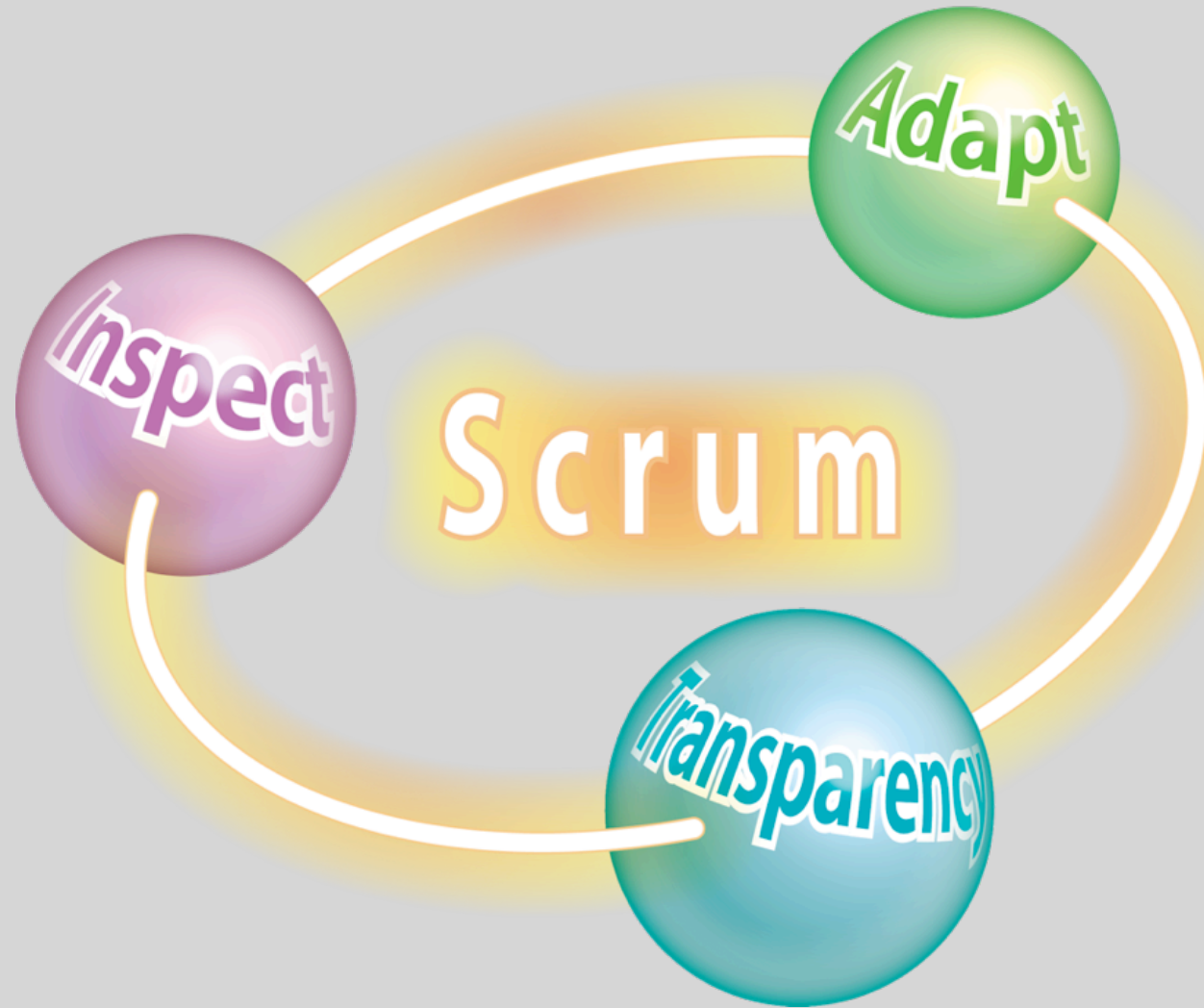


WHOLE
PRODUCT
FOCUS

CUSTOMER
CENTRIC



Large Scale Scrum is Scrum



MORE
with
LESS.



More with Less

Build Your Method Up -
Don't Tailor It Down