Dependency Management Is Wasteful

Consider Dependency Reduction.
Synchronous Dependencies != Asynchronous Dependencies.

presented by: Gene Gendel

of

k s T s
Consulting

to

Abbott Curiosity Lectures

About Gene Gendel and KSTS Consulting

Gene Gendel is an organizational design consultant, adaptive & lean coach and trainer, and independent adviser to senior leadership. Gene is a widely recognized, world-class trainer/instructor, and is mainly focused on organizational design and product centricity, the experience for which he has gained, over the decades of deeply embedded coaching and consulting. Gene's clients represent a wide industry spectrum. Almost 15 of 20+ years of his professional experience Gene has dedicated to working with companies of various sizes and lines of business, trying to help them improve internal dynamics, organizational structure and becoming a better place for people to work in. Gene engages at *all* organizational levels: senior- and mid-level management, teams and individuals. In his work, Gene uses various methods, tools and techniques to amplify learning of other people and to ensure that his followers gain autonomy after Gene "coaches himself out of the job". Over the last decade, Gene's big focus has been on large financial institutions and consulting companies that struggle with moving away, from traditional budgets and portfolio/program/project work decomposition, towards more adaptive/flexible budgeting and better clearly products (product-centric, customer-focused development).

Gene is the co-founder of ***KSTS Consulting***, **the only company in the United States that has the credential of a *Certified LeSS Coaching Company***. Gene's, highly in-demand LeSS classes, both private and public, are based on advanced system modelling and rich exosystemic perspective that includes intra-organizational dynamics and market realities.

About Us

One Stop-Shop For:

❖ Organizational and team structure and dynamics
❖ Executive coaching and consulting
❖ Individual, single team and multi-team coaching
❖ Upskilling Scrum Masters and Product Owners
❖ HR advisory: building talent and defining careers
❖ Training: Agile, Kanban, Scrum, Large Scale Scrum
❖ Workshops: product discovery & business agility
❖ "SAFe Recovery" Program (gradual improvements)
❖ Complimentary lunch & learn sessions

www.keystepstosuccess.com

Follow us: https://www.linkedin.com/company/ksts-consulting/

Our Global Community

Join us: **https://www.meetup.com/large-scale-scrum-less-in-nyc/**

since 2015 | ~8000 members! - please, join us ☺

Alternatives to "Skillful" Dependency Management
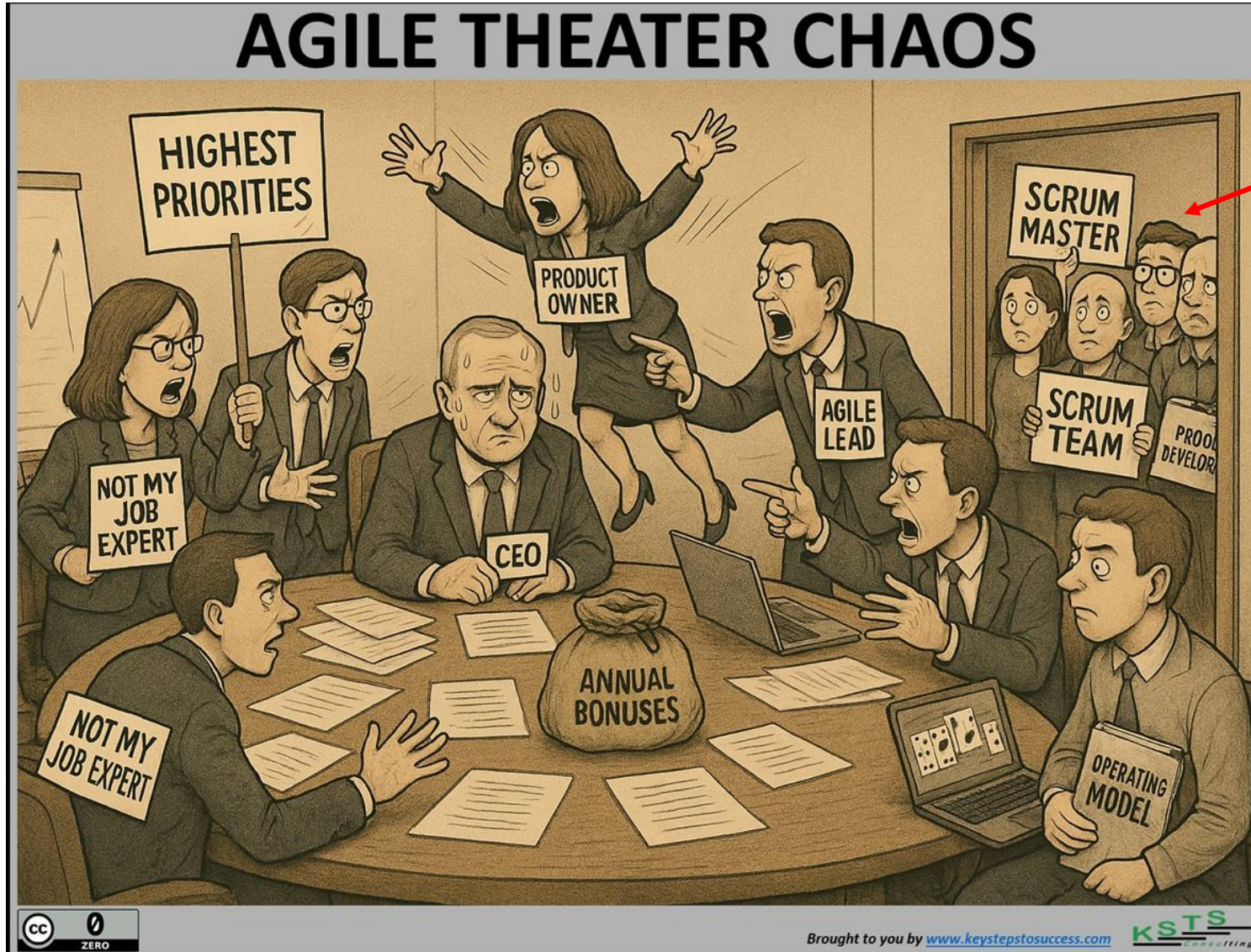
Alternatives to "Skillful" Dependency Management

My goals:
- **to increase**
- **to reduce**

Alternatives to "Skillful" Dependency Management

we care about

no so much

✦ AI Overview

In programming, synchronous dependencies mean that one operation must complete before the next can start, while asynchronous dependencies allow multiple operations to run concurrently. Synchronous execution is a blocking architecture, where each operation waits for a response before proceeding, while asynchronous execution is non-blocking, allowing tasks to run simultaneously. 🔗

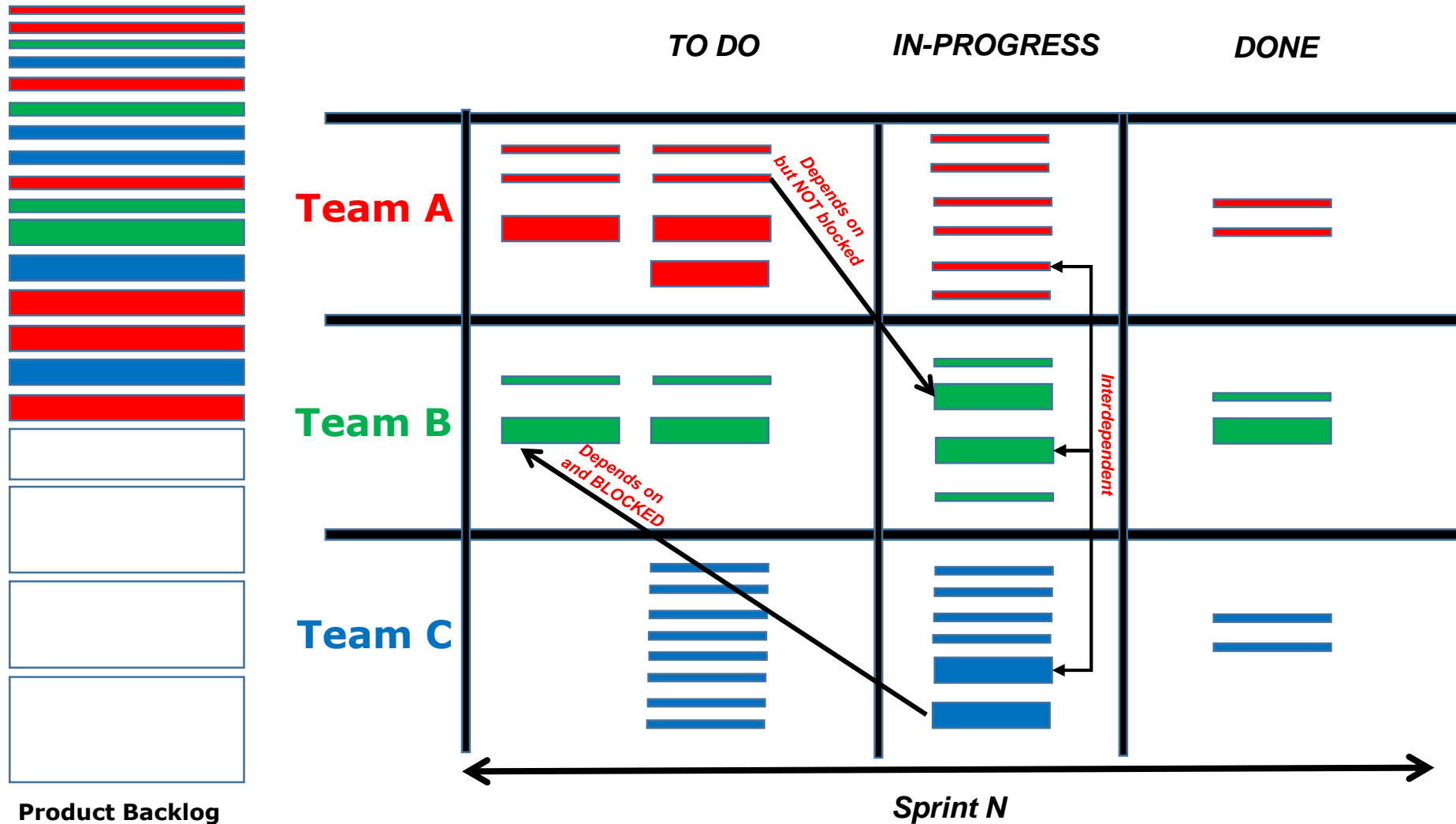# Alternatives to "Skillful" Dependency Management

Synchronized Sprint Cadence: Thoughtful Decision or Cargo Cult?
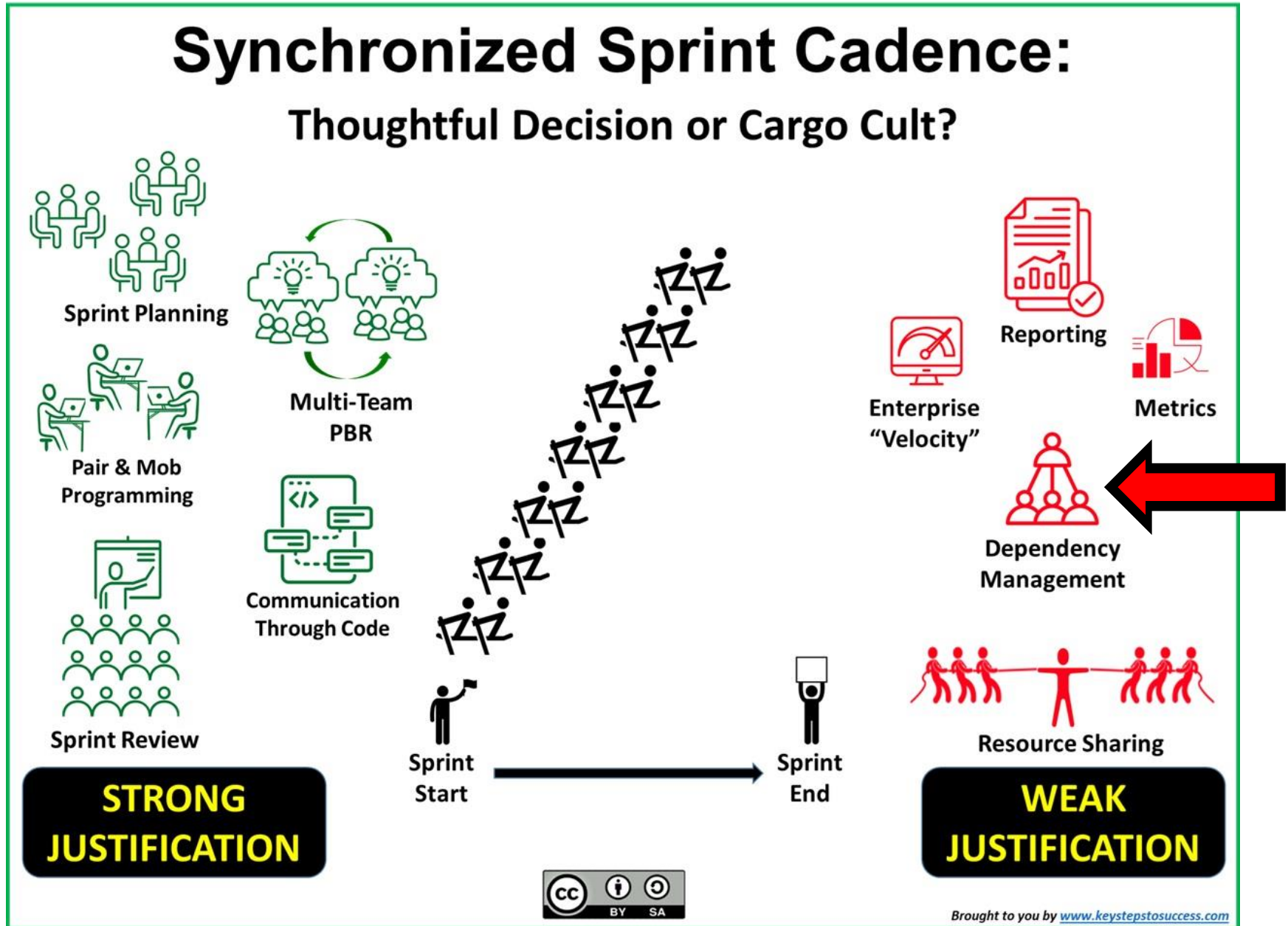
Alternatives to "Skillful" Dependency Management

Cross-Team Work Synchronous Dependencies

www.keystepstosuccess.com

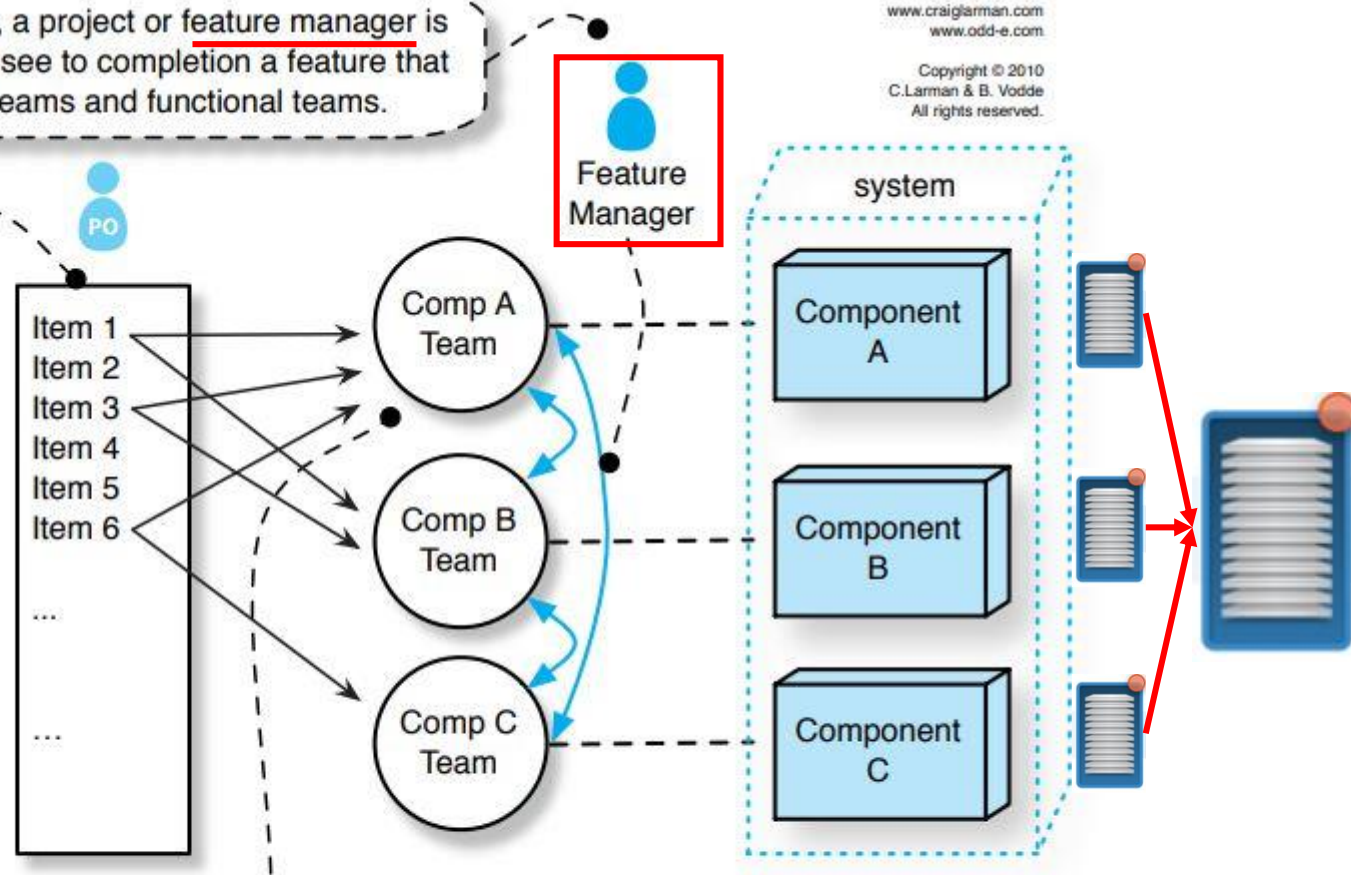# Alternatives to "Skillful" Dependency Management

With component teams, a project or feature manager is used to coordinate and see to completion a feature that spans component teams and functional teams.

With component teams, there is a tendency to select goals familiar or 'fast' for teams, not for maximizing customer value. For example, Component B Team does part of Item 3 because it mostly involves Component B work.

This is the "watching the runner rather than following the baton" local optimization.

Feature Manager

system

PO

Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
...
...

Comp A Team

Comp B Team

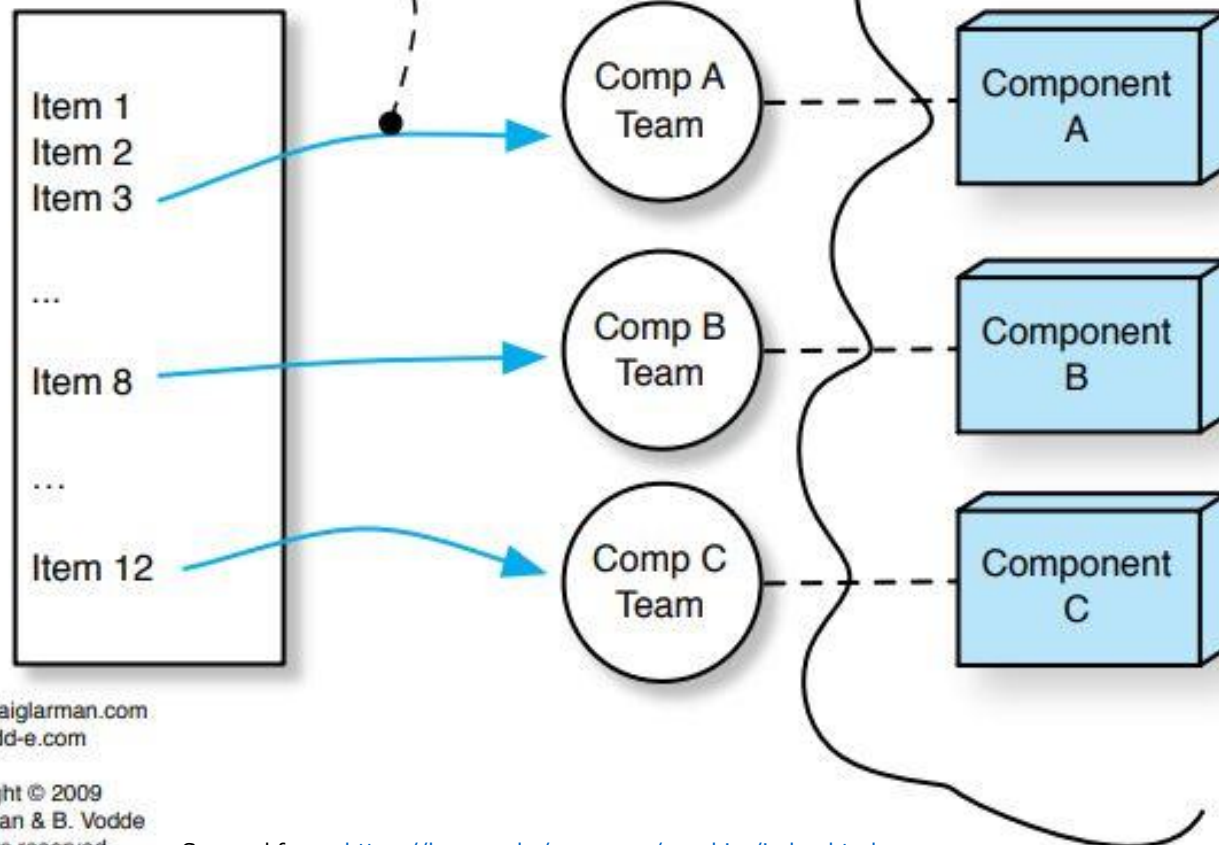Comp C Team

Component A

Component B

Component C

With component teams, there is increased delay, as one customer feature is split across multiple component teams for programming, and eventually transferred to a separate testing team for verification. There is handoff waste, and multitasking waste—as one component team may work on several features in parallel, in addition to handling defects related to 'their' component.

Sourced from: https://less.works/resources/graphics/index.html

Alternatives to "Skillful" Dependency Management

With component teams, there is a tendency to select goals familiar for people, not for maximizing customer value. For example, Component A Team does Backlog Item 3 because it mostly involves Component A work.
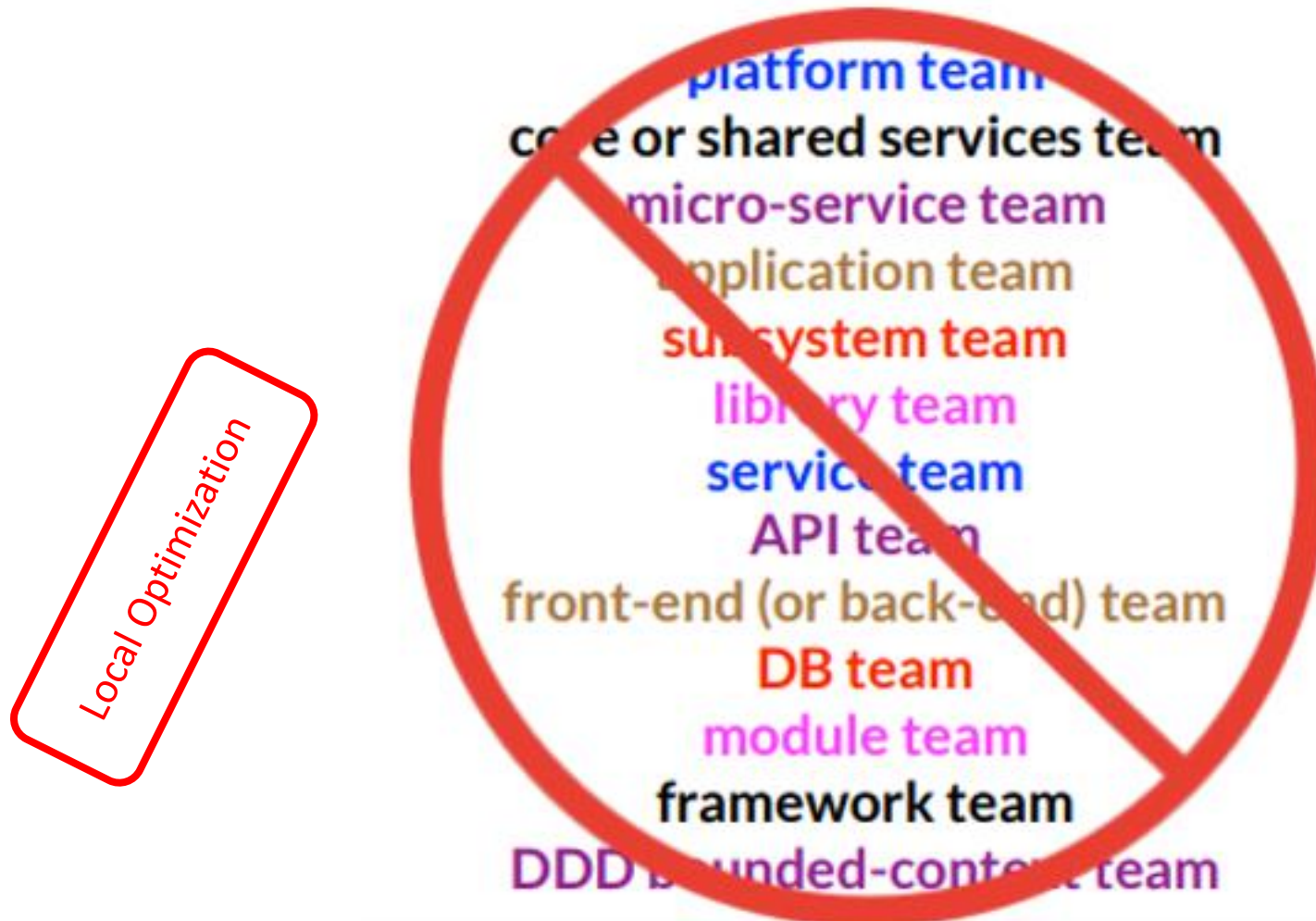
Local Optimization

System

Item 1
Item 2
Item 3

...

Item 8

...

Item 12

Comp A Team

Comp B Team

Comp C Team

Component A

Component B

Component C

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

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

Sourced from: https://less.works/resources/graphics/index.html

Gene Gendel, Certified Enterprise & Team Coach (CEC-CTC), Certified LeSS Trainer (CLT)

www.keystepstosuccess.com

Alternatives to "Skillful" Dependency Management

Synchronous Dependencies Become a HUGE Problem When….

Local Optimization

platform team
core or shared services team
micro-service team
application team
subsystem team
library team
service team
API team
front-end (or back-end) team
DB team
module team
framework team
DDD bounded-context team

Sourced from creative commons repository of Certified LeSS Trainers: https://less.works/courses/become-less-trainer

Alternatives to "Skillful" Dependency Management
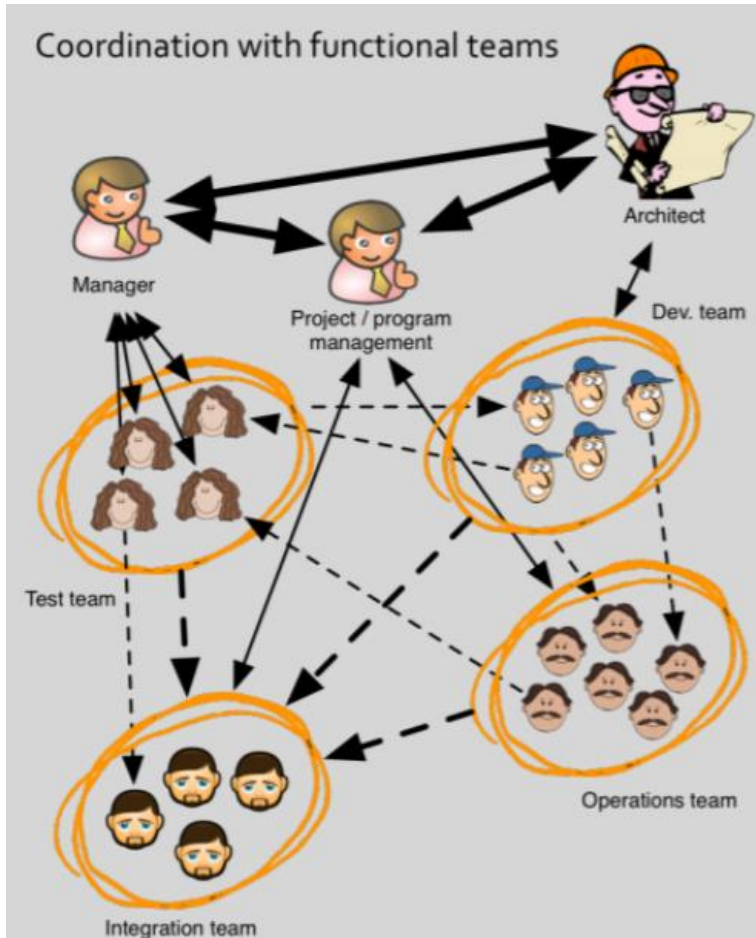
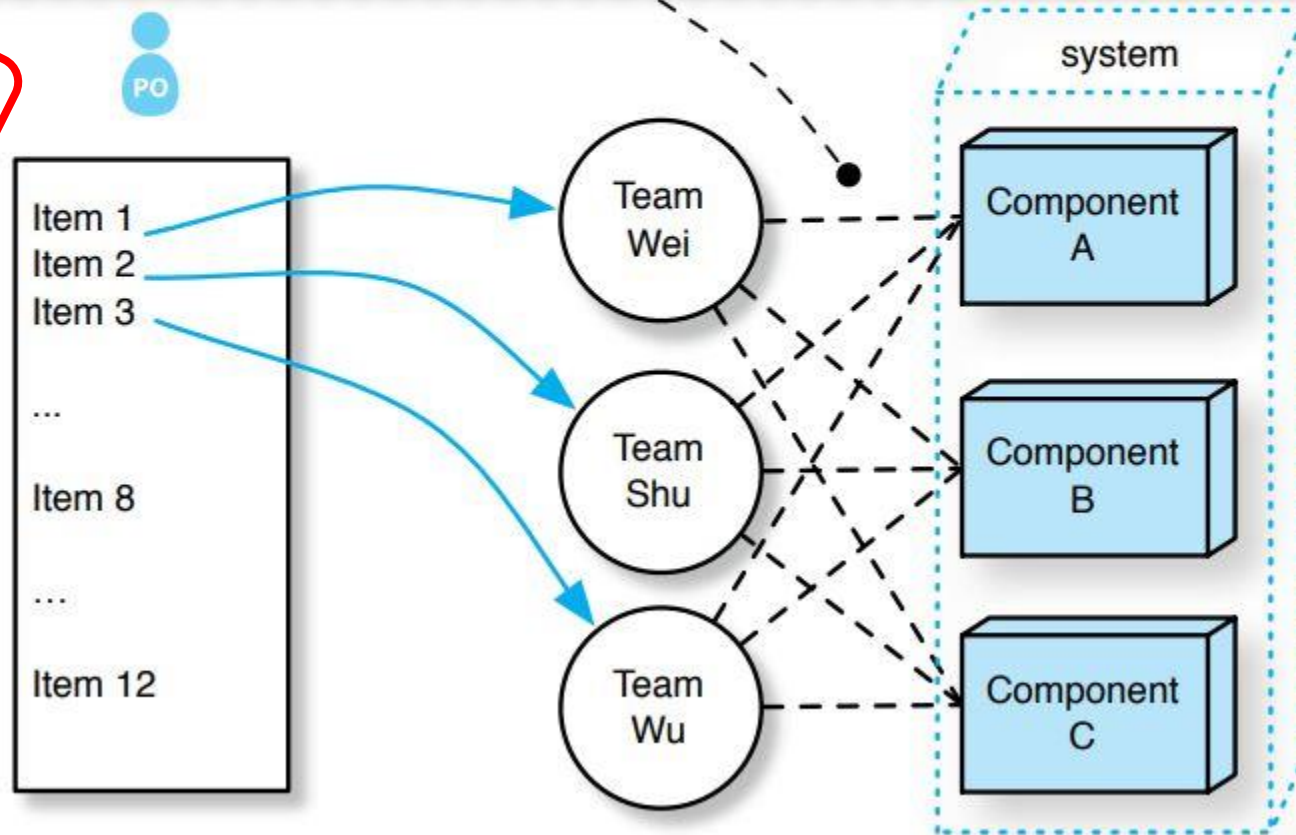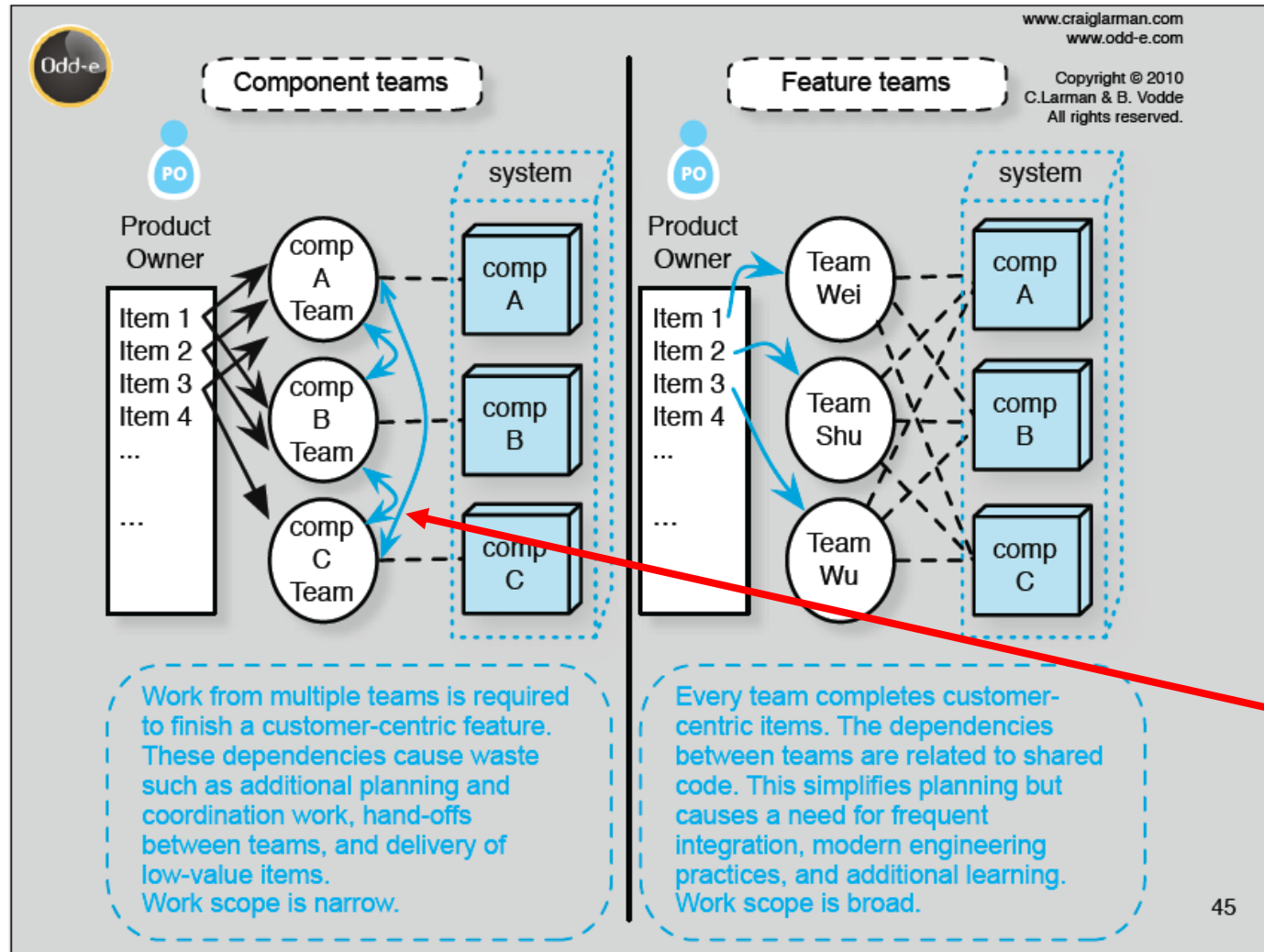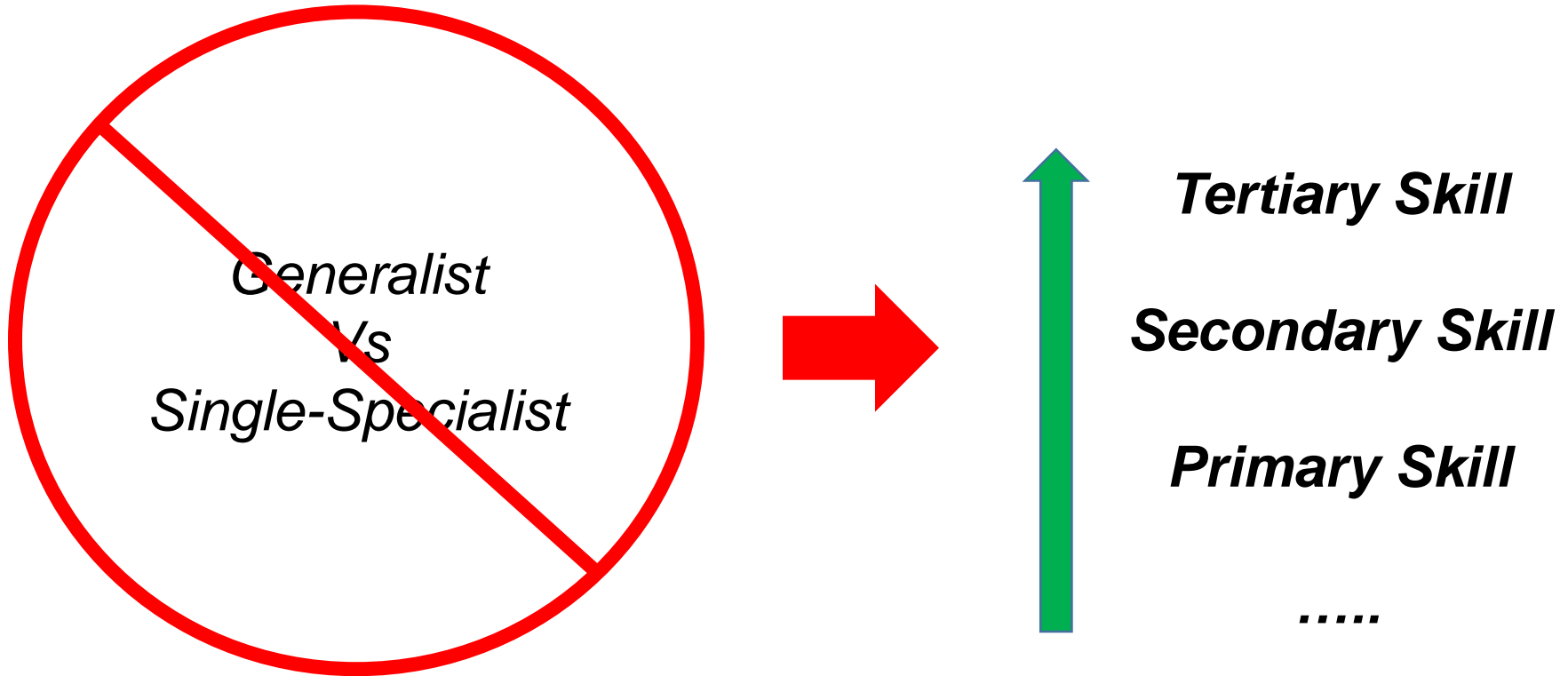*Hard Synchronous Dependencies* | *Synchronous Dependencies With Learning Opportunities*

Sourced from creative commons repository of Certified LeSS Trainers: https://less.works/courses/become-less-trainer

Gene Gendel, Certified Enterprise & Team Coach (CEC-CTC), Certified LeSS Trainer (CLT)
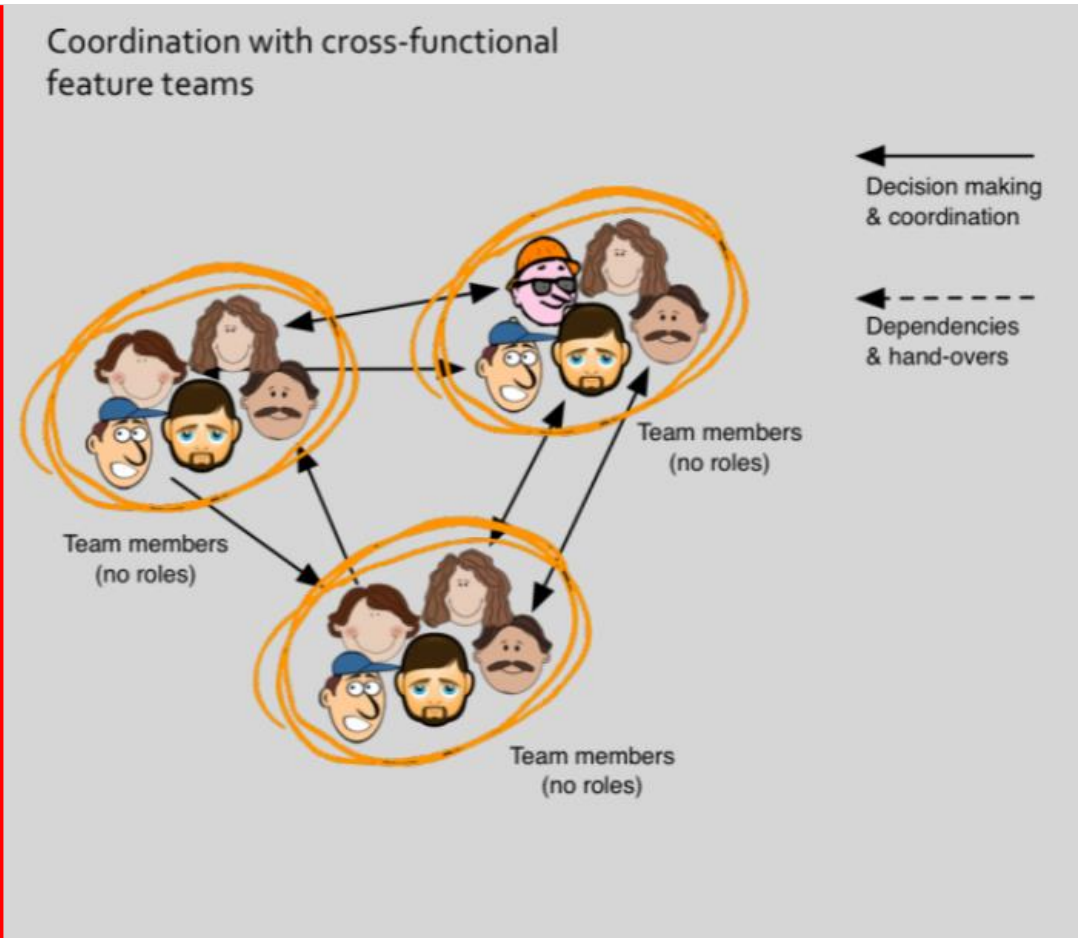
Alternatives to "Skillful" Dependency Management

With feature teams, teams can always work on the highest-value features, there is less delay for delivering value, and coordination issues shift toward the shared code rather than coordination through upfront planning, delayed work, and handoff. In the 1960s and 70s this code coordination was awkward due to weak tools and practices. Modern open-source tools and practices such as TDD and continuous integration make this coordination relatively simple.

System Optimization

PO

Item 1
Item 2
Item 3
...
Item 8
...
Item 12

Team Wei
Team Shu
Team Wu

system
Component A
Component B
Component C

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

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

Sourced from: https://less.works/resources/graphics/index.html

Alternatives to "Skillful" Dependency Management

Hard Synchronous Dependencies

Synchronous Dependencies With Learning Opportunities

Coordination Costs of Hard Synchronous Dependencies is Astronomic $$$$$$$$$$

Sourced from creative commons repository of Certified LeSS Trainers: https://less.works/courses/become-less-trainer

Gene Gendel, Certified Enterprise & Team Coach (CEC-CTC), Certified LeSS Trainer (CLT)

www.keystepstosuccess.com

Alternatives to "Skillful" Dependency Management

Synchronous Dependencies: from HARD to SOFT

~~Generalist
Vs
Single-Specialist~~

Tertiary Skill

Secondary Skill

Primary Skill

.....

Alternatives to "Skillful" Dependency Management

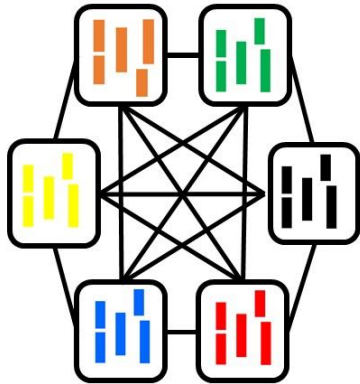*Synchronous Dependencies: from HARD to SOFT*

I-shaped → T-shaped → M-shaped

Alternatives to "Skillful" Dependency Management

Synchronous Dependencies: from VERY HARD to HARD to SOFT
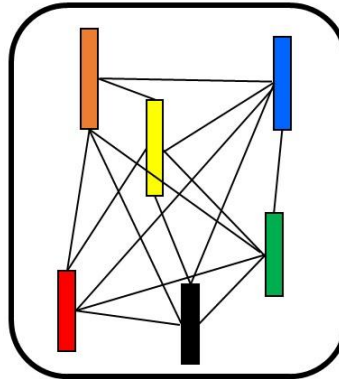
Explicit (Visible) vs. Implicit (Hidden) Hand-Offs

Functional Teams
made of
I-shaped developers

Cross-Functional Team
made of
I-shaped developers

Cross-Functional Team
made of
T-shaped developers

- Cross-team dependencies
- Explicit hand-offs
- Very long Cycle Time
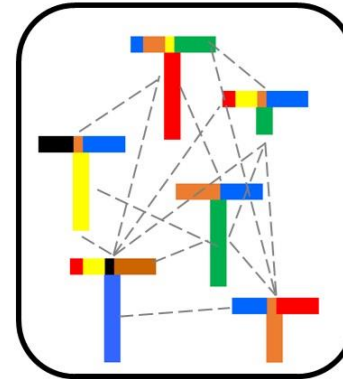- High integration risks
- Low learning opportunities

- Intra-team dependencies
- Implicit hand-offs
- Shorter Cycle Time
- Lower integration risks
- Higher learning opportunities

- No hard dependencies
- No hand-offs
- Shortest Cycle Time
- Lowest integration risks
- Highest learning opportunities

very HARD   *HARD*   *SOFT*

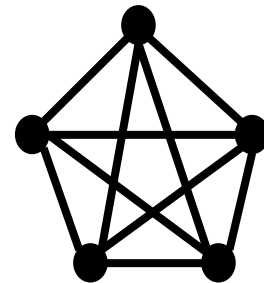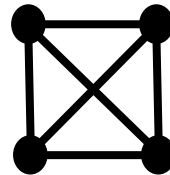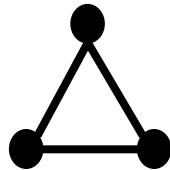Brought to you by www.keystepstosuccess.com

Alternatives to "Skillful" Dependency Management

$$N = \frac{n \times (n-1)}{2}$$
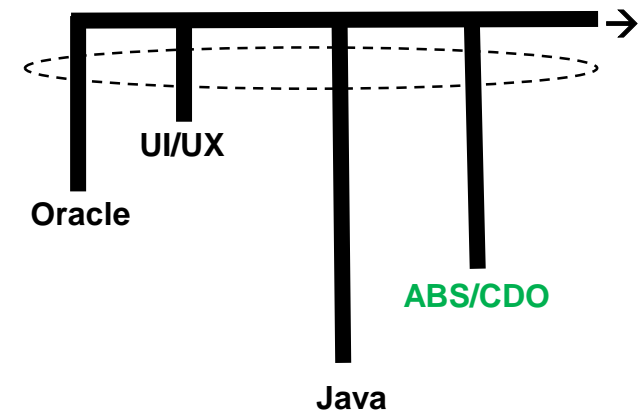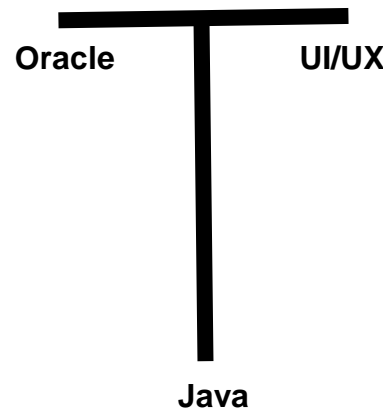
n - number of communication nodes
N – number of communication channels

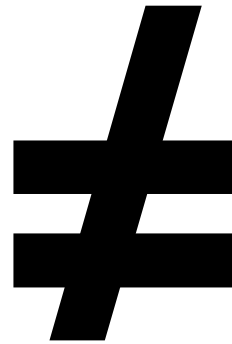# Alternatives to "Skillful" Dependency Management

- A **cross-functional team** is a group of people with different skills and expertise who work together to achieve a *common goal*.

- A **cross-functional developer** is a member of a cross-functional team, which is a group of people with diverse skills working together to achieve a *common goal*.

- A **full-stack team** is a group of developers who work together to create a *software product* from start to finish, including both the front-end and back-end.

- A **full-stack developer** is a programmer who works on all aspects of software *product* development, from front-end to back-end.

Alternatives to "Skillful" Dependency Management

False Equivalency Kills Innovation

$450,000 NOT spent on education and cross-training of developers ≠ **$50M LOST in market opportunities, to competitors**
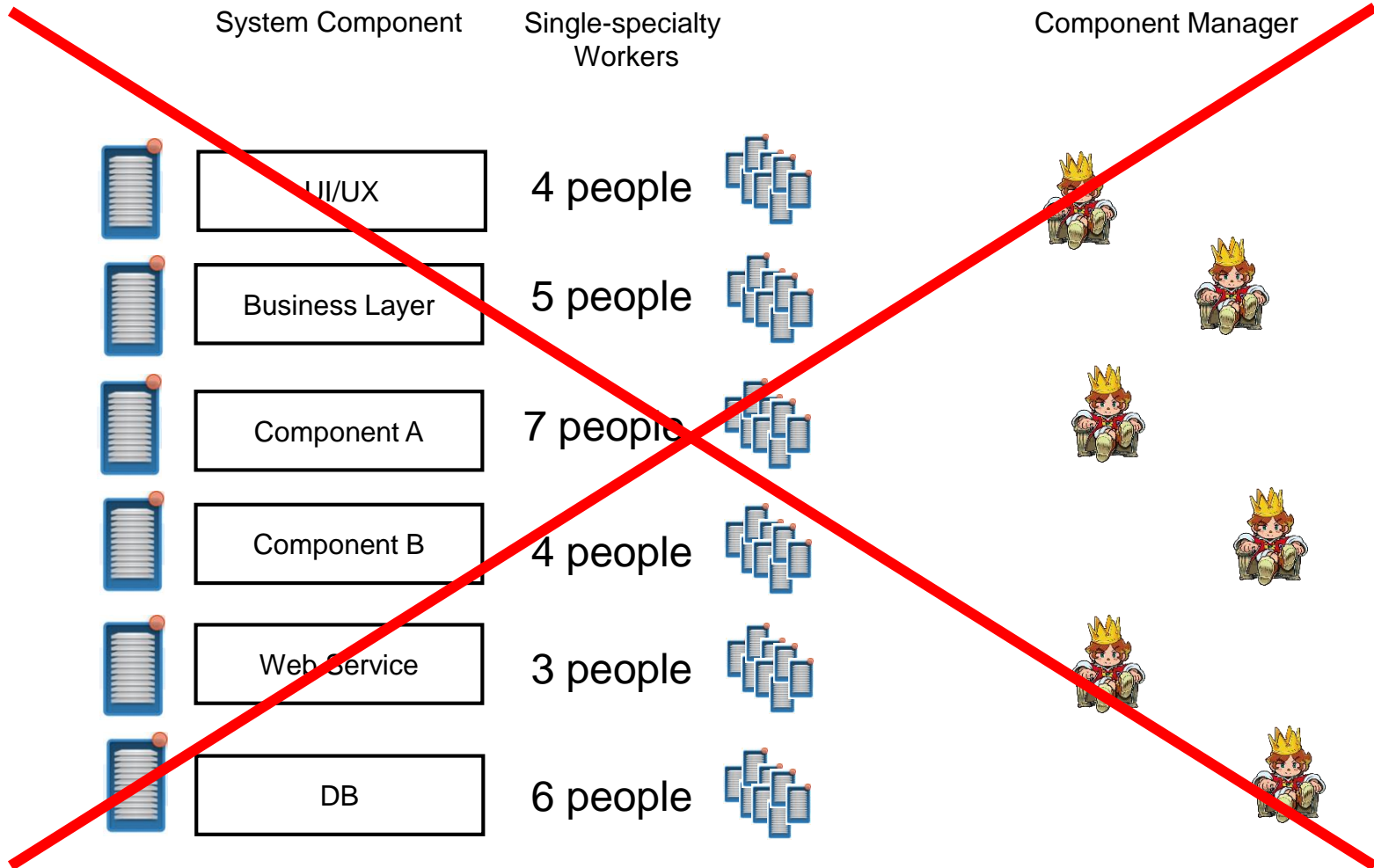
…such a weak argument and poor judgement by a manager…

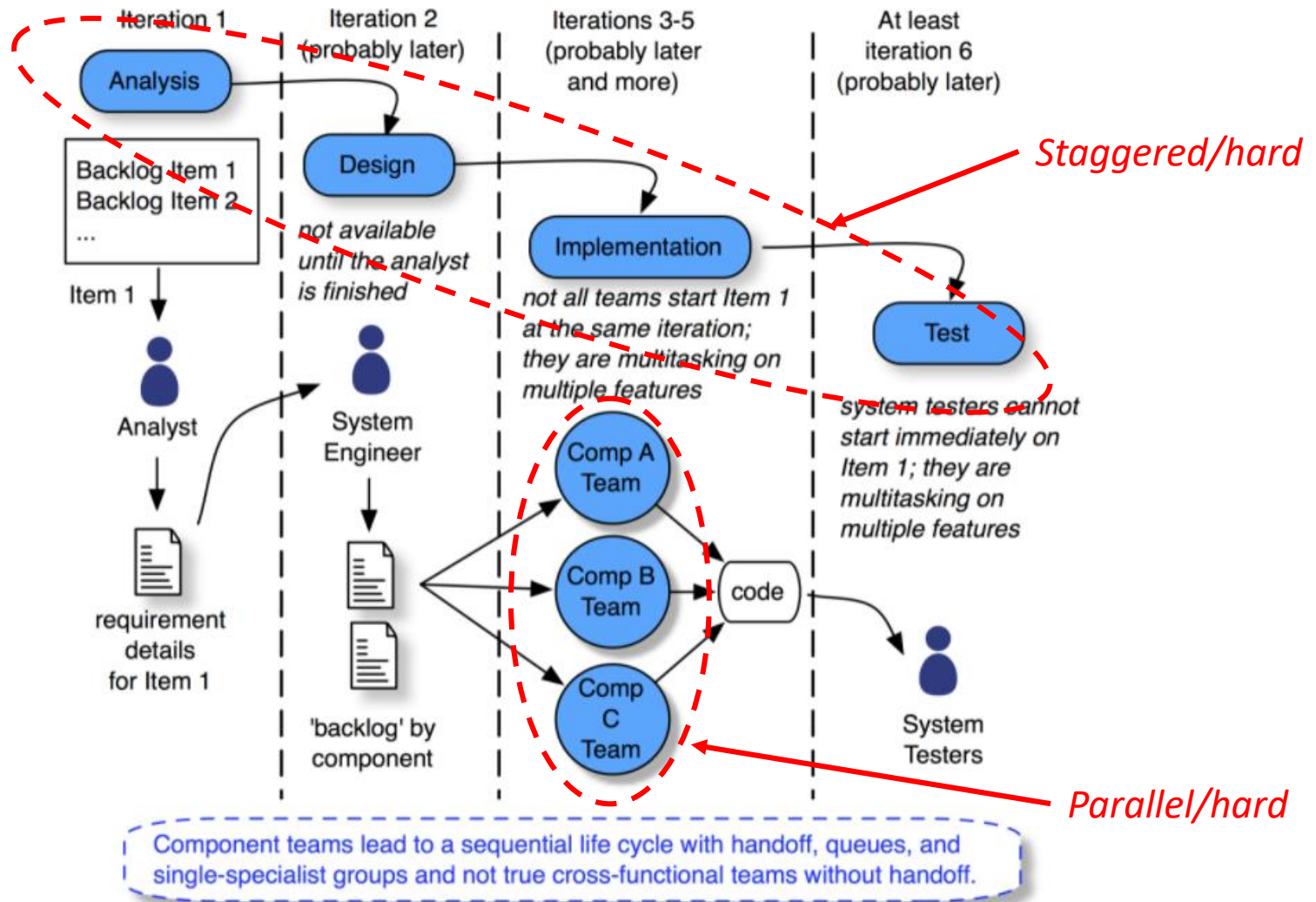**BIG IDEA:** stop being penny-wise and near-sighted, when making long-term organizational design decisions

Alternatives to "Skillful" Dependency Management
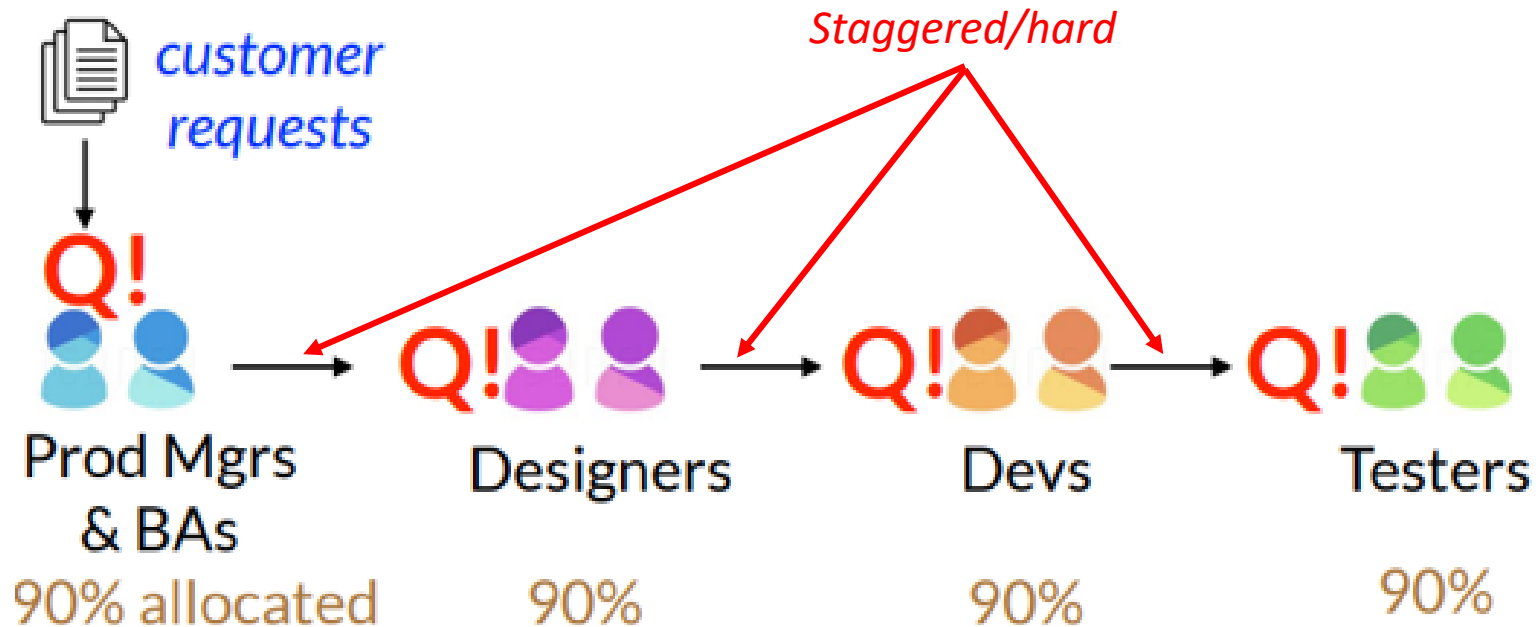

"Beneficiaries" of Hard Synchronous Dependencies

System Component	Single-specialty Workers	Component Manager
UI/UX	4 people	
Business Layer	5 people	
Component A	7 people	
Component B	4 people	
Web Service	3 people	
DB	6 people	

www.keystepstosuccess.com

Alternatives to "Skillful" Dependency Management

Synchronous Dependencies

Gene Gendel, Certified Enterprise & Team Coach (CEC-CTC), Certified LeSS Trainer (CLT)

Alternatives to "Skillful" Dependency Management

Staggered/hard

customer requests

Q! Prod Mgrs & BAs — 90% allocated

Q! Designers — 90%

Q! Devs — 90%

Q! Testers — 90%

in a system with "servers" and Qs, plus variability, as *server resource utilization* increases ("people are 90% allocated"), total duration INCREASES

Alternatives to "Skillful" Dependency Management
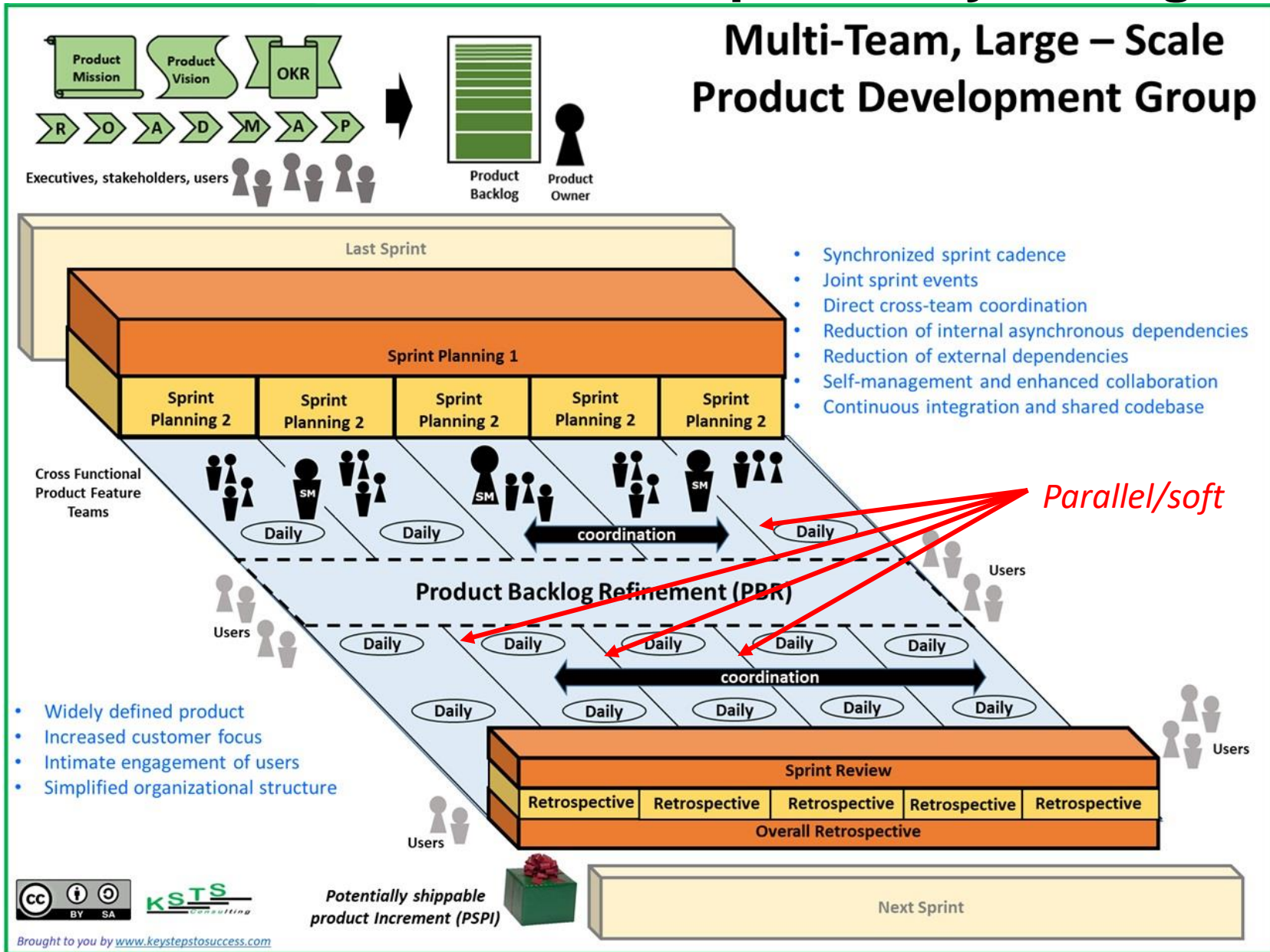
Upcoming
June 17-19 (3 sessions x 3 hours)

Registration from: https://www.keystepstosuccess.com/large-scale-scrum-training/

Q &A

Synopsis

On my site: https://www.keystepstosuccess.com/

On Facebook: https://www.facebook.com/people/KSTS-Consulting/61561671969644/

On LinkedIn:

- **Gene Gendel: https://www.linkedin.com/in/ggnyc/**

- **KSTS Consulting: https://www.linkedin.com/company/ksts-consulting/**

On Meetup:
https://www.meetup.com/large-scale-scrum-less-in-nyc/ (close to 6500)