

What is "UNDONE" Department And How To Eradicate It?

Presented @ DeveloperWeek2021

About Gene

Gene Gendel is an organizational design specialist, agile/lean coach and trainer, consultant and adviser to senior leadership. Almost 15 of 20+ years of his professional experience – Gene dedicated to working with companies of various sizes and lines of business, trying to help them improve internal dynamics, organizational structure and and becoming a better place to work. 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 by others and to ensure that people gain autonomy after Gene “coaches himself out of the job”.

Gene is Certified Agile Coach (CAC), by Scrum Alliance – small elite community of [Scrum Alliance Certified Enterprise Coaches \(CEC\)](#). Today, he is the only CEC who resides in NY State. Gene is also one of the co-creators of [Team Level Coaching Certifications \(CTC\)](#) program for Scrum Alliance.

Gene is also one of very few [Certified LeSS Trainers \(CLT\)](#) that are based in the United States.






He is also well-recognized blogger and publisher. He is the author of [Adaptive Ecosystems: Collection of Independent Essays About Agility](#) and co-author of:

- [Agile Coaching: Wisdom from Practitioners](#) (free pdf).
- Best Agile Articles of [2017](#) and [2018](#)



Gene significantly contributes to global and local agile communities, where he influences people by running workshops, webinars, seminars, panel discussions, lunch & learn, coaching retreats, professional gatherings and other events. While teaching and coaching Large Scale Scrum (LeSS) to organizations, Gene has built the biggest in world (today) [LeSS meetup community](#) that is used as a free educational hub about LeSS available to many people.

Assets To Consider

- gg_nyc@yahoo.com (underscore) 
- <https://www.linkedin.com/in/ggnyc/> 
- <http://www.keystepstosuccess.com/contact-us/> 
- https://join.slack.com/t/kstsconsulting/shared_invite/zt-fykyy2wz-qrJkOgVYHAopwT2dq4ogCw (Free Slack channel – any Q&A 😊) 
- <https://www.meetup.com/Large-Scale-Scrum-LeSS-in-NYC/> 
- <http://www.keystepstosuccess.com/virtual-learning-training-coaching/>
- https://less.works/course_filters/courses-13



Today's Topic

📍 DeveloperWeek OPEN STAGE 1



* OPEN TALKS

ENGINEERING INNOVATION

DEVOPS SUMMIT



Gene Gendel

KSTS Consulting

Today, unfortunately, many agile teams are not yet able to create a true shippable increment every sprint.

This is because their Definition of Done (DoD) is weak (immature).

As such, teams have a lot of **UNDONE** work at the end of every sprint. This work may take a lot of different forms and is usually passed on to a 'special' **UNDONE** department or group to be handled. In Scrum and in Large Scale Scrum **UNDONE** department do not exist.

It may seem that reasons for having **UNDONE** department are purely technical [limitations]. But this is rarely so. For the most part, they are political and have to do with traditional organizational design and sphere of control.

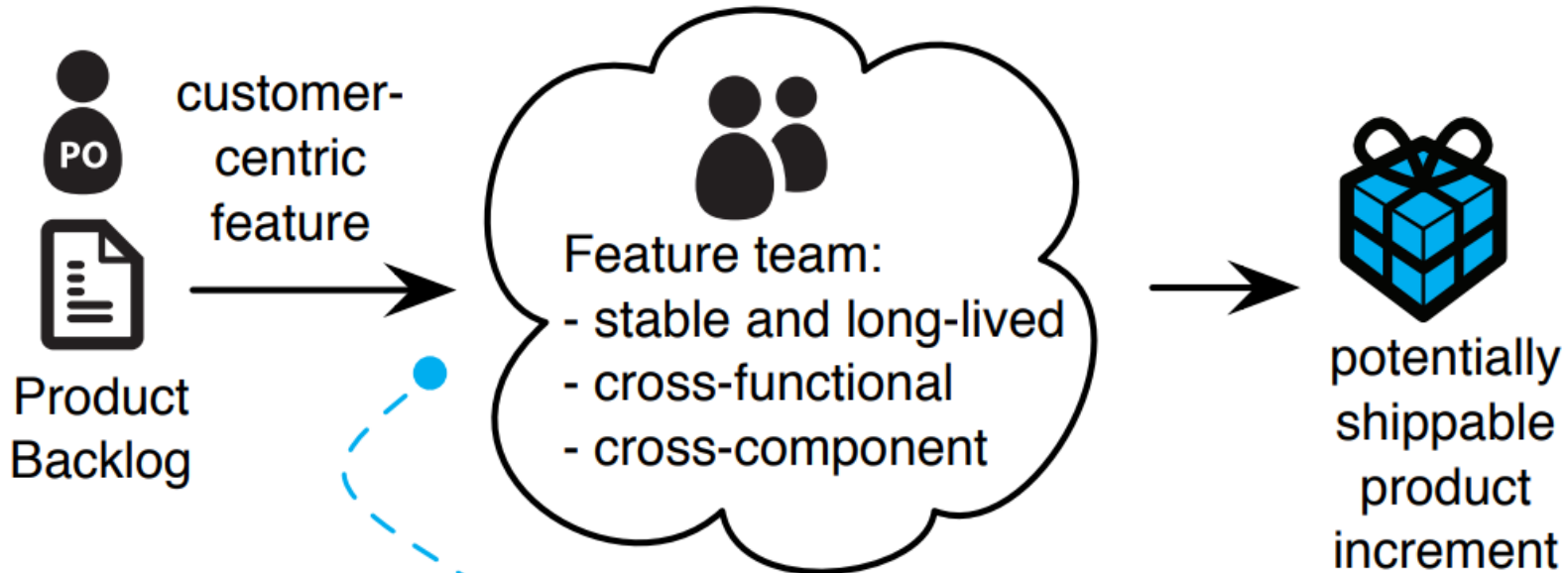
Team



<http://less.works> 

Source: <https://less.works/resources/graphics/index.html>

Team



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.

Source: <https://less.works/resources/graphics/index.html>

Technical Excellence

DoD



POTENTIALLY SHIPPABLE

- + TEST
- + STATIC ANALYSIS
- + INTEGRATION
- + PACKAGING
- + STAGING
- + CUSTOMER DOCUMENTATION
- + RISK EVALUATION
- + CODE
- + APPROVAL
- + UPDATE MARKETING MATERIAL
- + REGULATION
- + PREPARE FOR CUSTOMER FEEDBACK

- UNIT
- INTEGRATION
- USER ACCEPTANCE
- ACCEPTANCE
- SYSTEM
- PERFORMANCE
- STABILITY
- USABILITY
- STRESS
- MONKEY
- SMOKE

UNDERLINED ONES ARE THE DEFINITION OF DONE

Source: <https://less.works/resources/graphics/index.html>

Technical Excellence

Unfinished Work != Undone Work

Usually, Team's issue

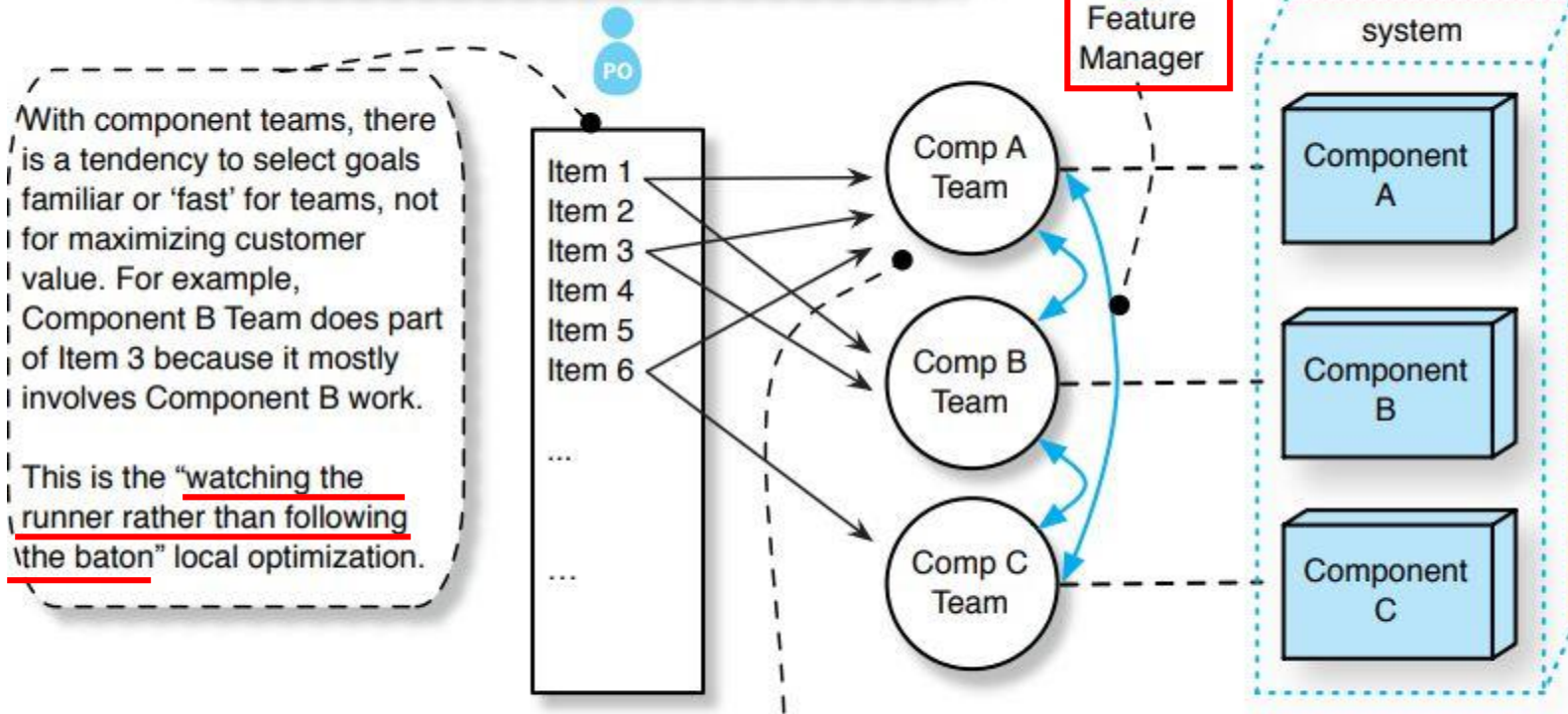
Usually, Organizational issue

Component Team

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.

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

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



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.

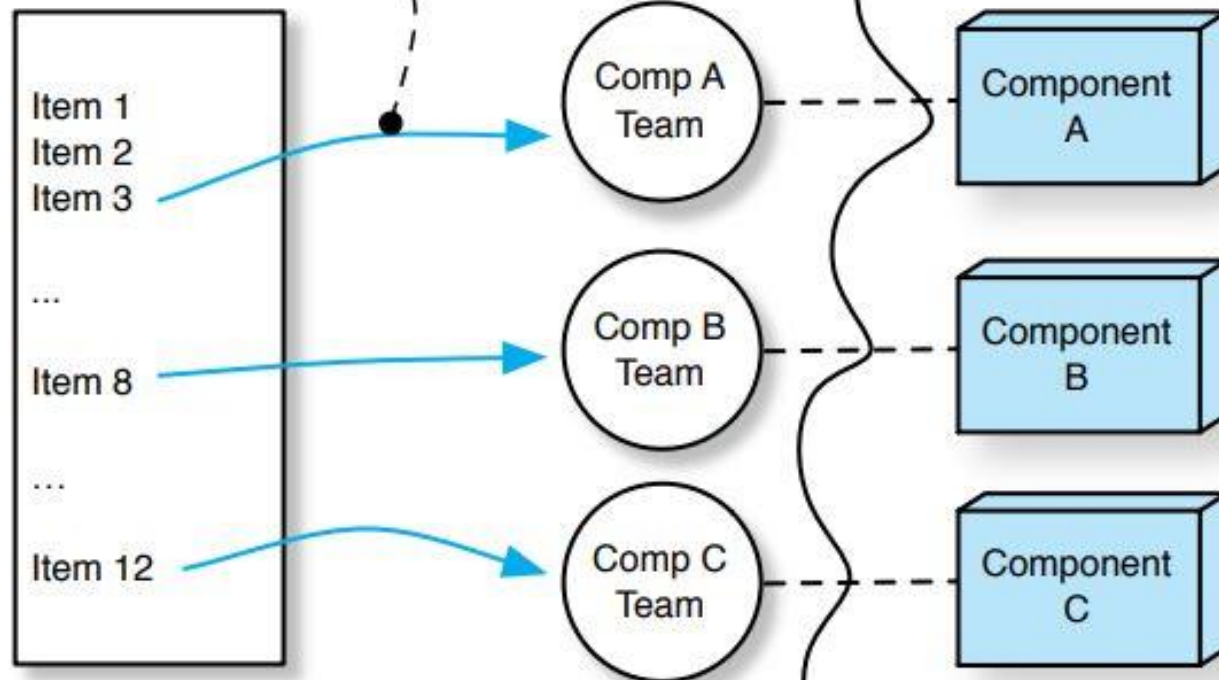
With component teams, there is increased delay, as one customer feature is split across multiple component teams for programming, and eventually transferred to a separate testing team for verification. There is handoff waste, and multitasking waste—as one component team may work on several features in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

Source: <https://less.works/resources/graphics/index.html>

Component Team - Cont.

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



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

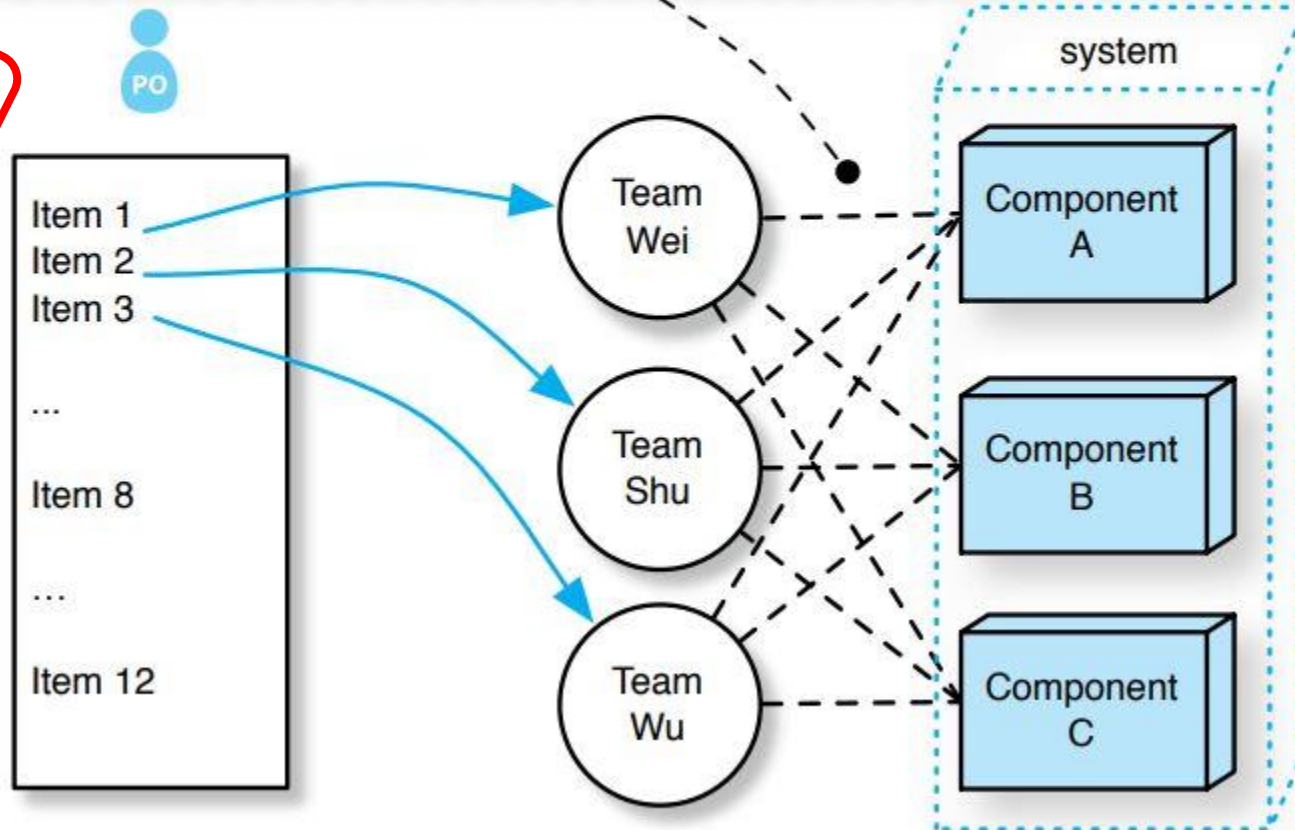
Copyright © 2009
C.Larman & B. Vodde
All rights reserved.

Source: <https://less.works/resources/graphics/index.html>

Feature Team

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

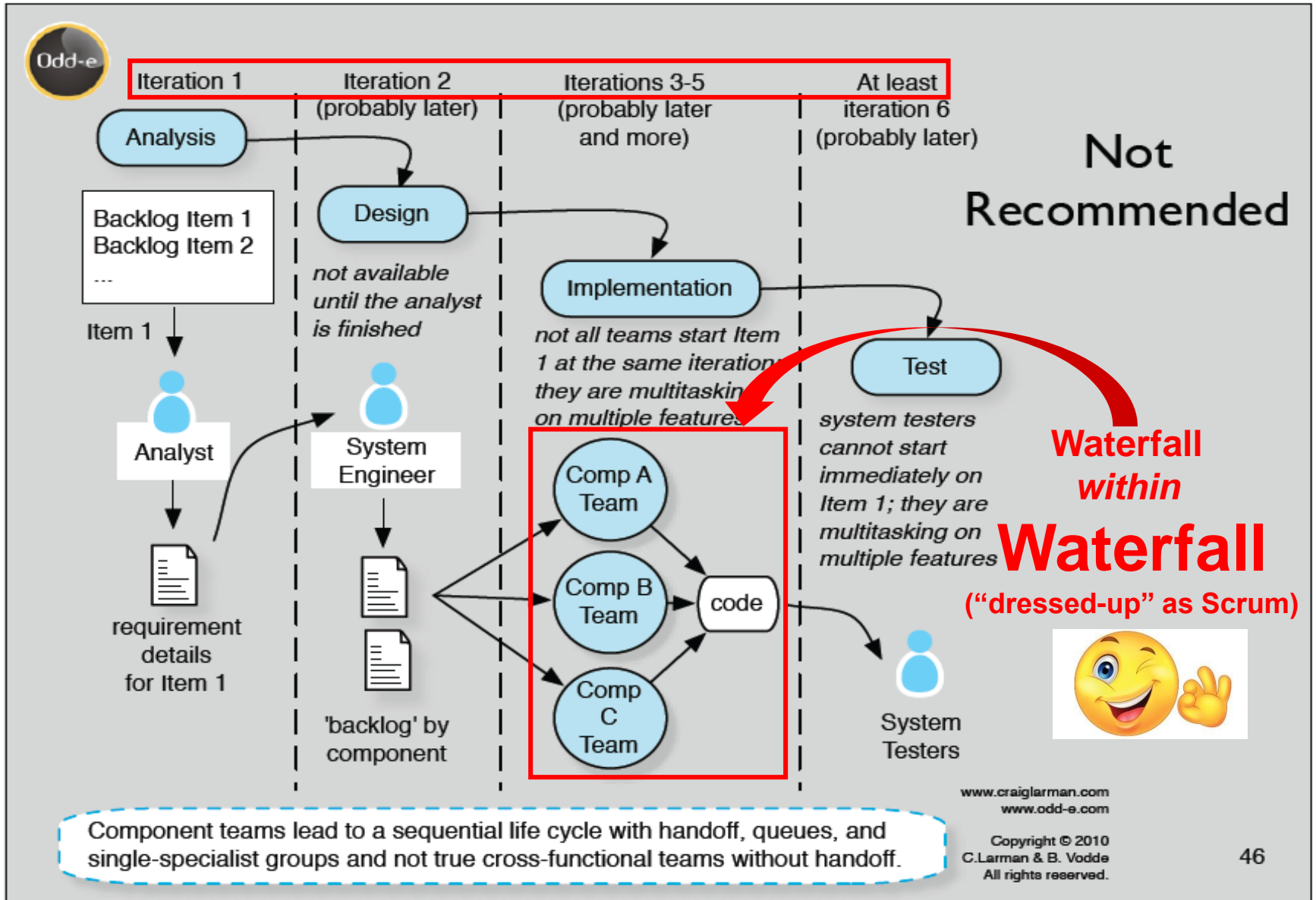


www.craiglarman.com
www.odd-e.com

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

Source: <https://less.works/resources/graphics/index.html>

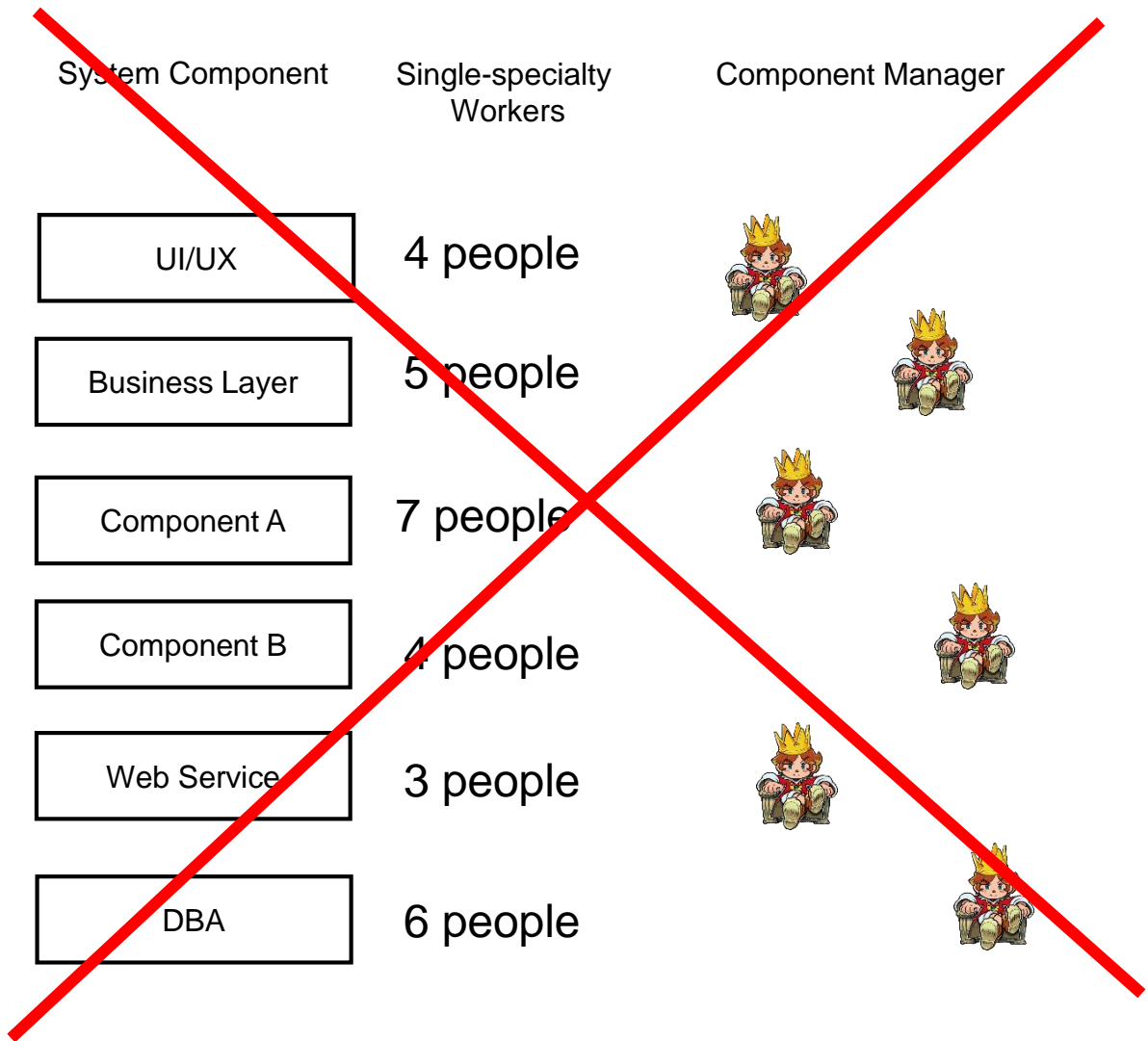
Mini-Waterfall in Scrum Clothes



Sourced from LeSS training materials of C. Larman and B. Vodde

No Team 'Tsars'

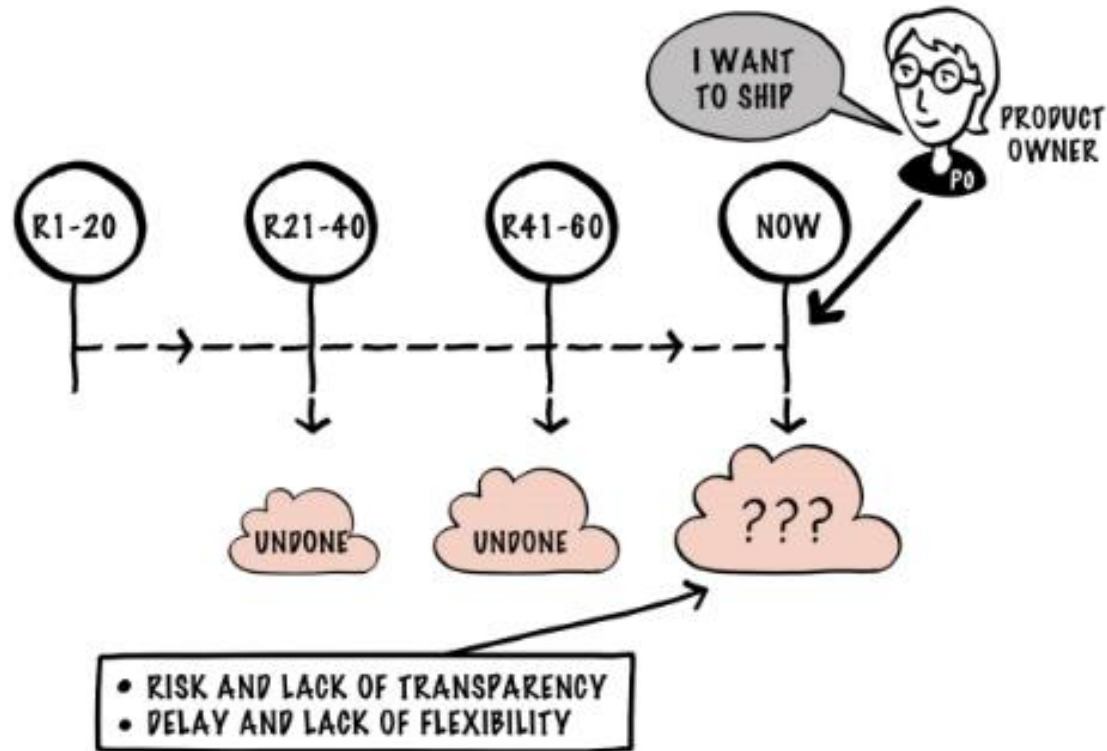
Component Teams



Classic Example of Local Optimization

Technical Excellence

Dealing with Un-Done Work

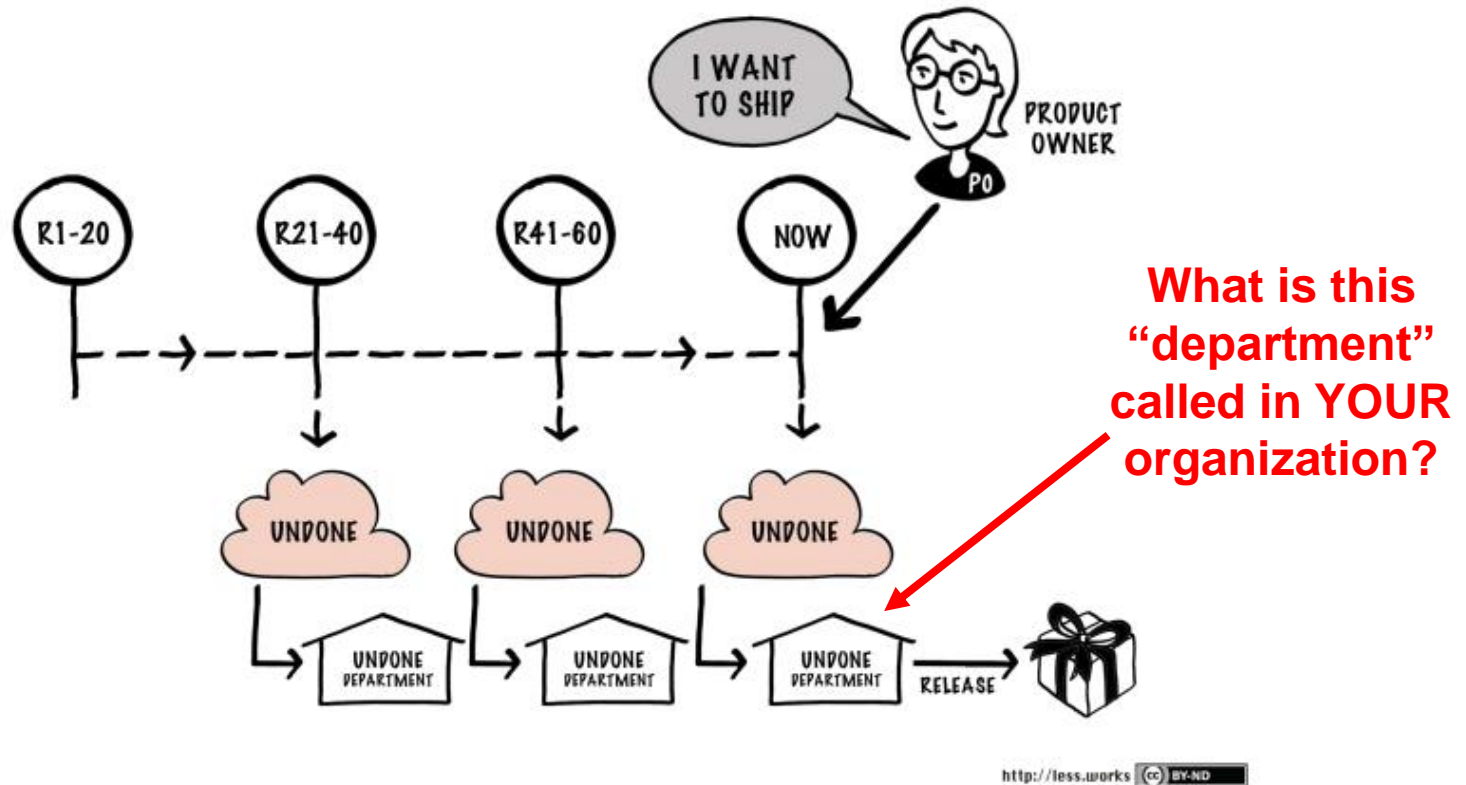


<https://less.works/resources/graphics/less-graphics.html>

Release Sprints

Technical Excellence

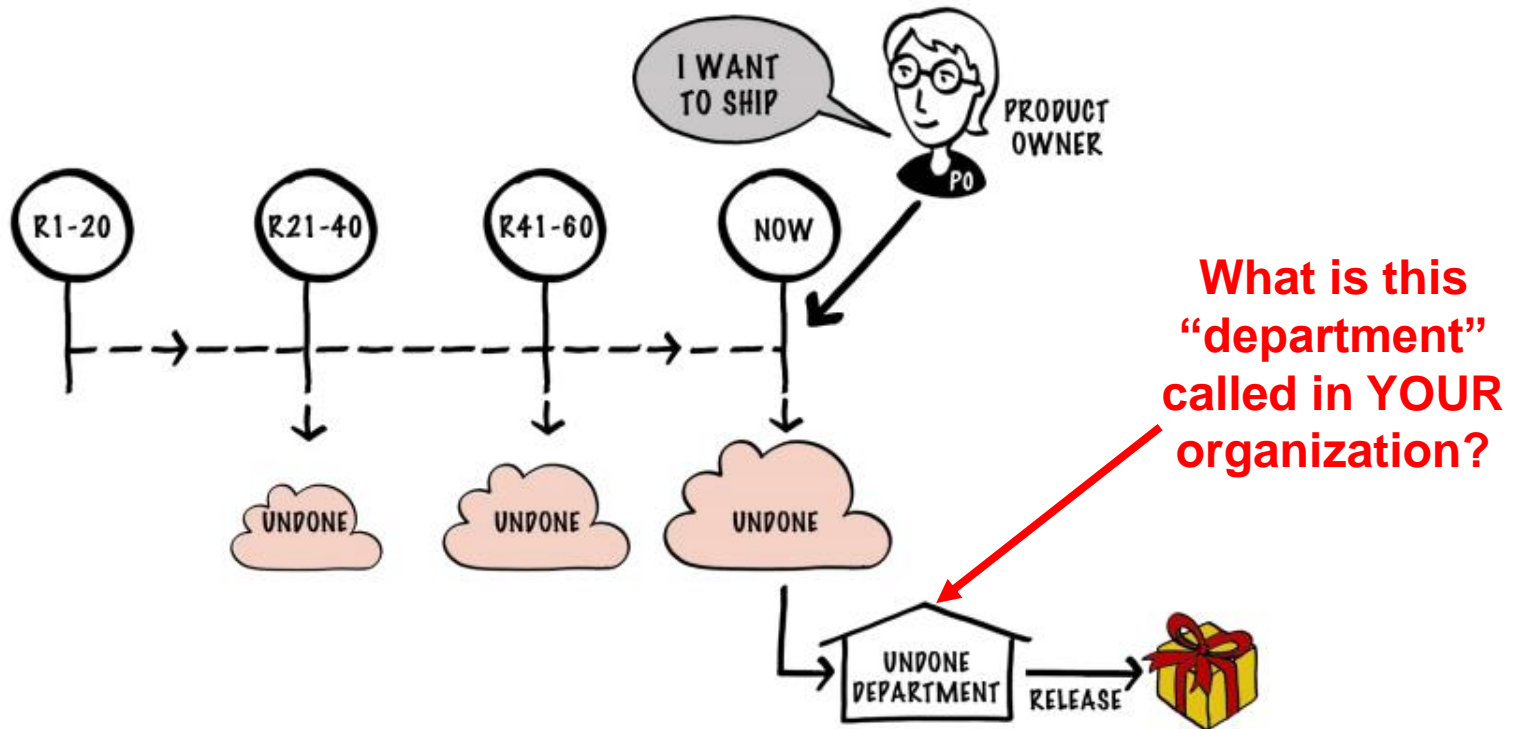
Dealing with Un-Done Work



Pipelining to Undone Department

Technical Excellence

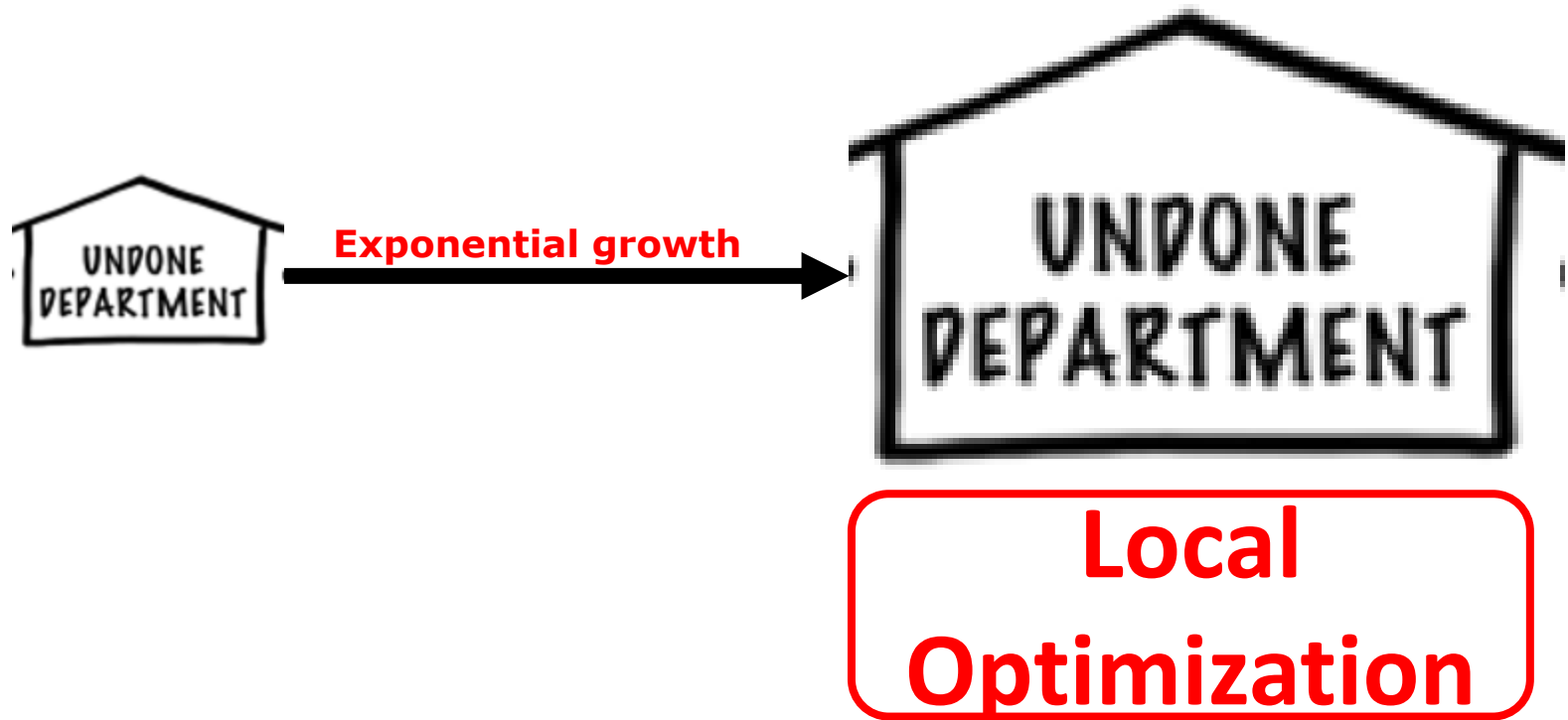
Dealing with Un-Done Work



<http://less.works> BY-ND

Undone Department Finalizes

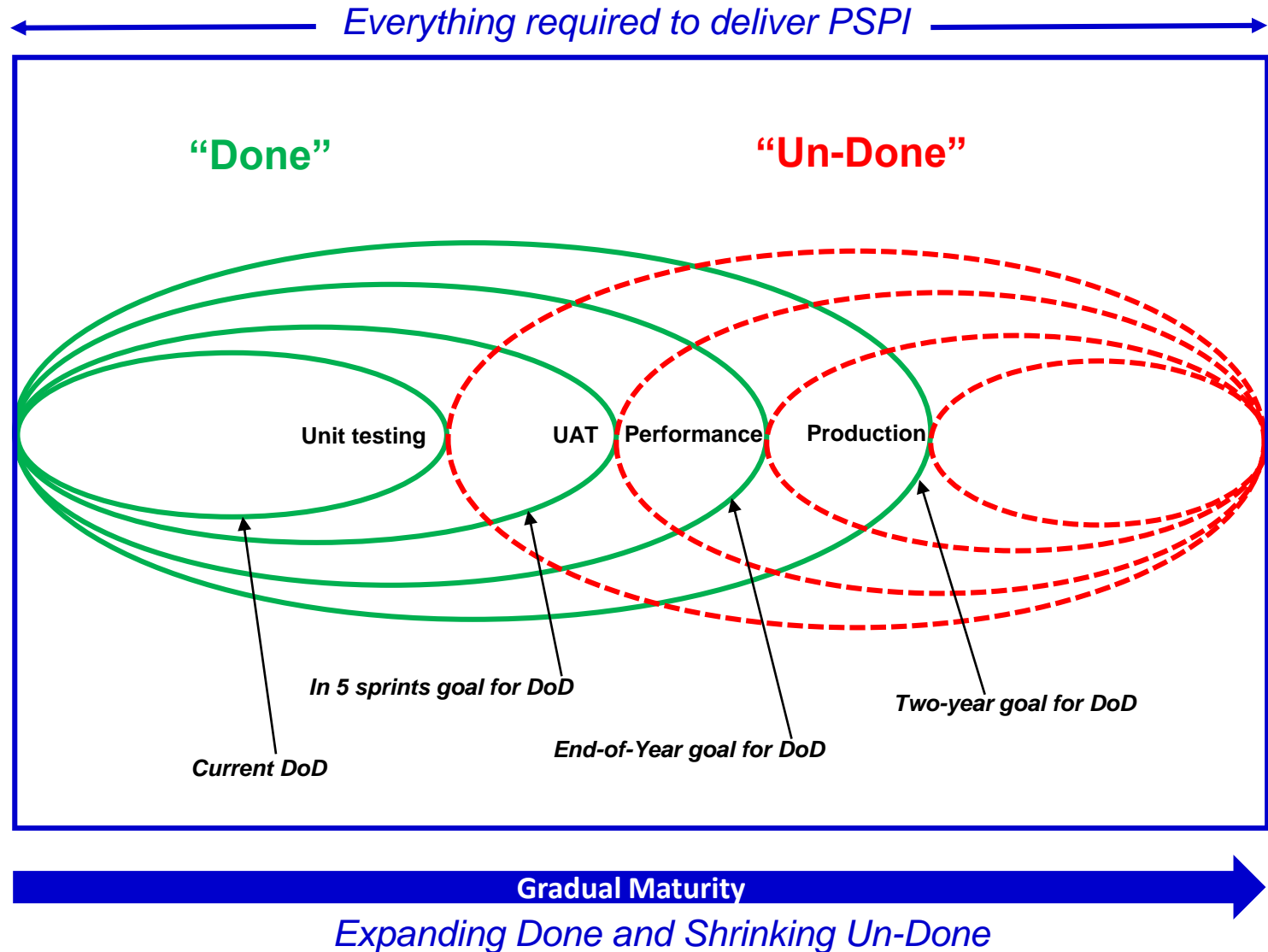
“Undone” Work, With Fake Scaling?



How to Eradicate “Undone” Department

**GRADUALLY,
REMOVE
UNDONE
DEPARTMENT**

How to Eradicate “Undone” Department



Team Maturation

Types of teams


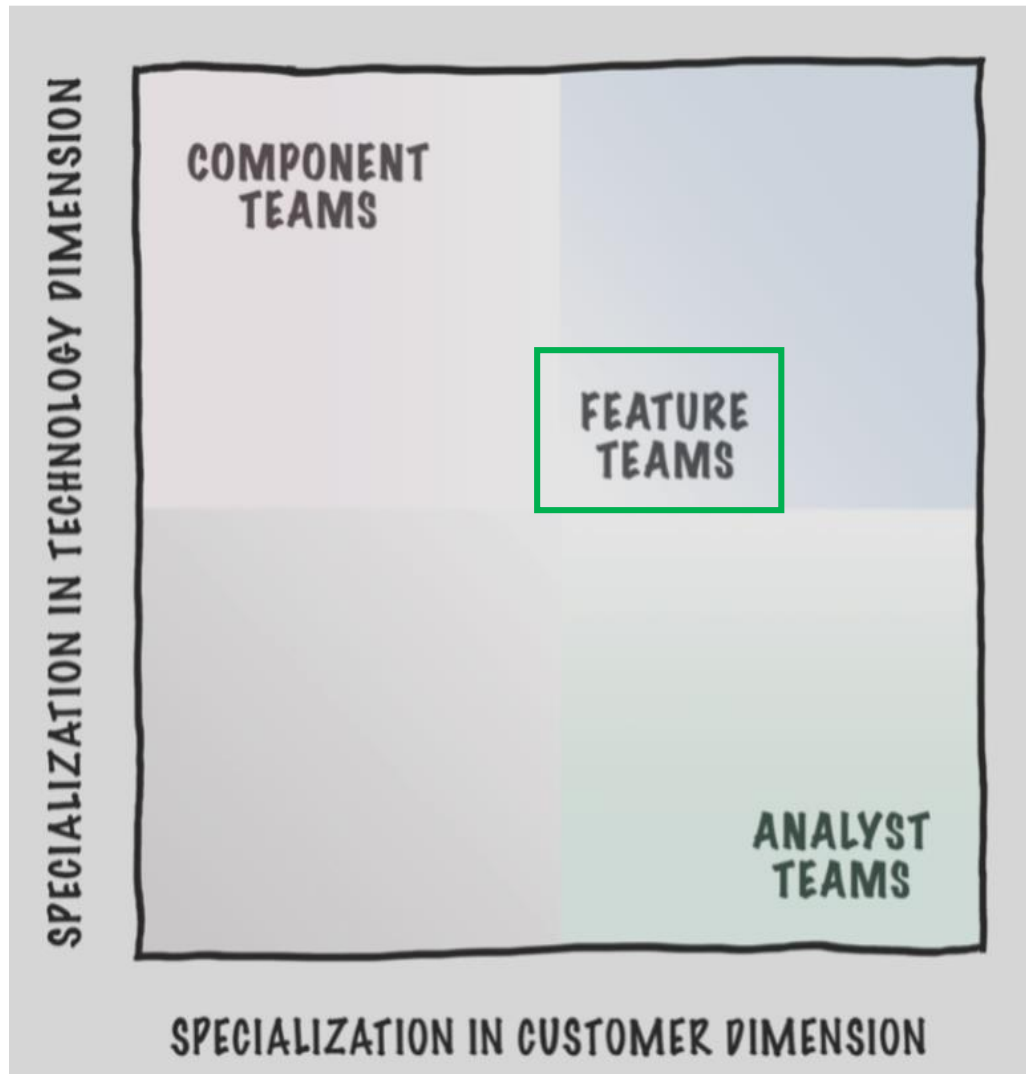
Setting overall direction				
Designing the team and its organizational context	Management Responsibility			
Monitoring and managing work process and progress		Team's Own Responsibility		
Executing the team task				
	Manager-led teams	Self-Managing teams	Self-Designing teams	Self-Governing teams

Figure 1. Types of Teams.

Source: <https://less.works/resources/graphics/index.html>

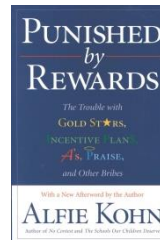
Feature Team Specialization



Source: <https://less.works/resources/graphics/index.html>

Real Team vs. Fake Team

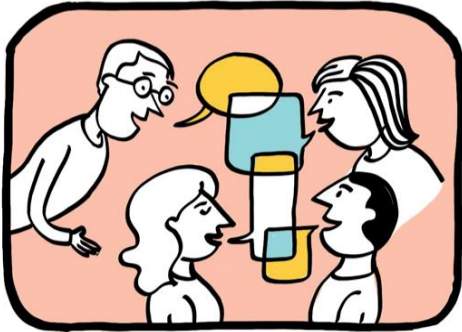
Want MORE of	Want LESS of
Stable teams that stay together and work on features for a long time	Temporary groups, created on a feature or project
Share all responsibility, as a team	Individual responsibility only for 'their' part based on specialty
Self-management and self-direction	PM-controlled work
Simple, flattened organizations. No matrix	Matrix organizational with resource pools
Full dedication a team	Resource sharing and part time "membership"



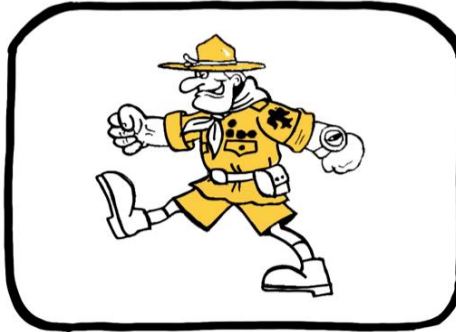
Real Team vs. Fake Team

Want MORE of	Want LESS of
Responsible Teams	Single-Function Roles
Focus on Customer	Documentation and “Contracts”
Teams that understand Requirements	Dedicated Analysts
Team ownership, inspection and adaptation	“Best practices” and prescriptive manuals
“Owning”	“Renting”
Well understood existing/simple roles and processes	Additional, vaguely defined roles and processes

How to Eradicate “Undone” Department



JUST TALK



SCOUT



OPEN SPACE



TRAVELER



COMMUNITIES



COMPONENT MENTOR

<https://less.works/resources/graphics/index.html>

<http://less.works> CC BY-ND

Q & A