

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

Presented @ Agility 2020

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 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-fykkyy2wz-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/trainers/listed-large-scale-im-usa-33/courses> 

Invitation For This Week



August 05-07

<https://less.works/course-details/less-basics-online-new-york-1928>

Link on Left hand side:

<https://www.keystepstosuccess.com/>

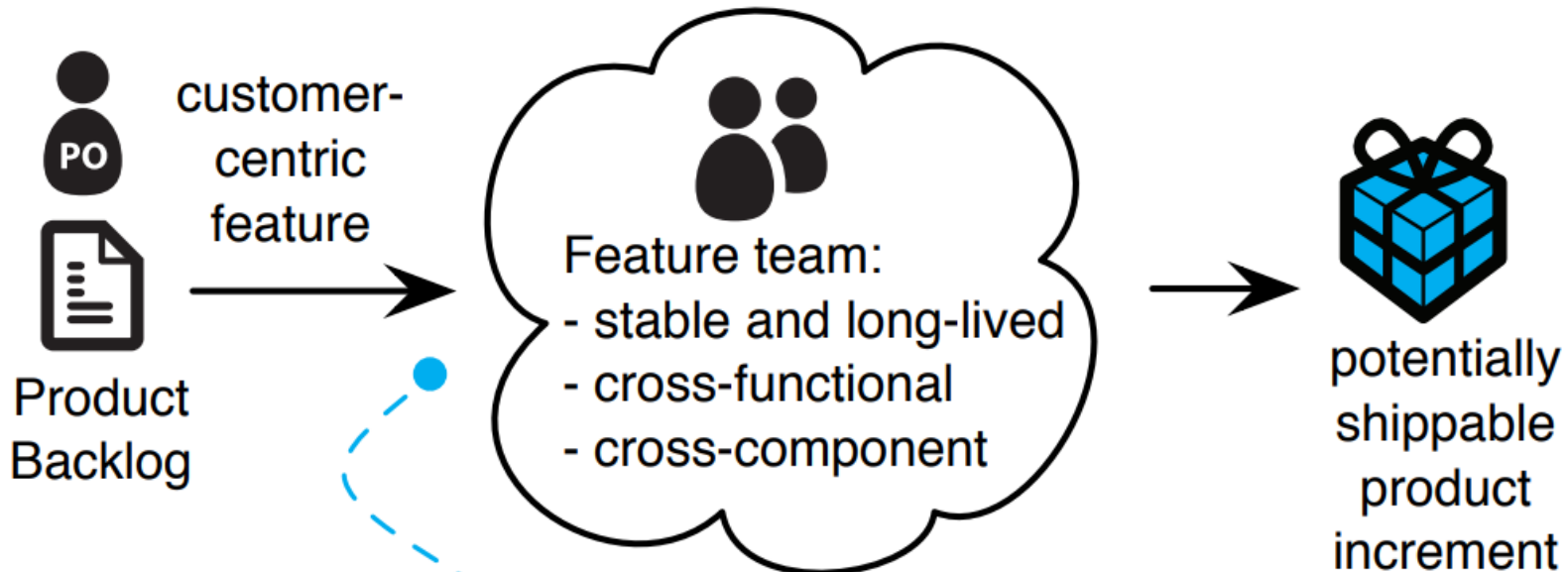
Team



<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.

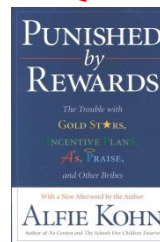
www.craiglarman.com
www.odd-e.com

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

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

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

Team Maturation

Types of teams


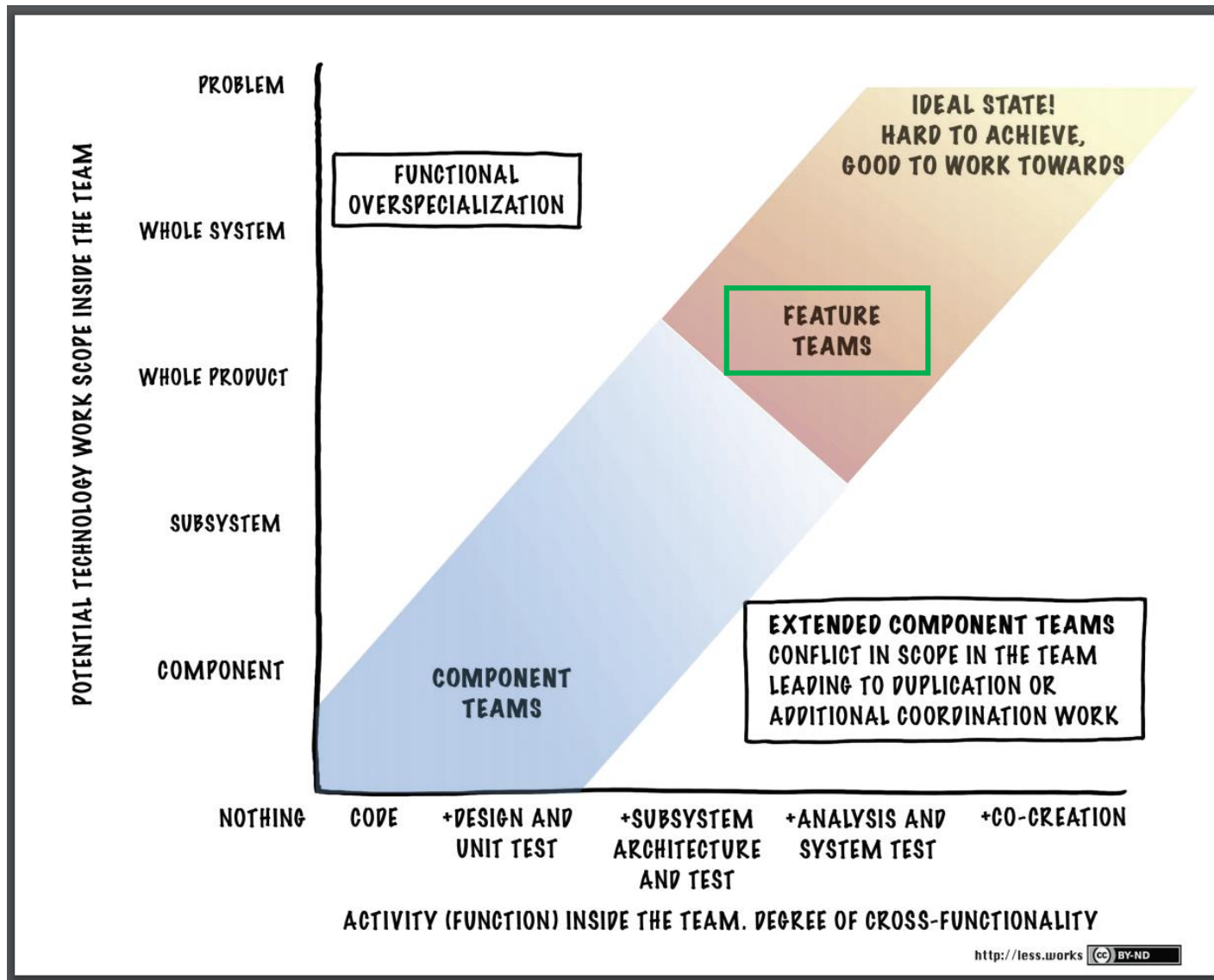
Setting overall direction				
Designing the team and its organizational context	Management Responsibility			
Monitoring and managing work process and progress		Team's Own Responsibility		
Executing the team task				
	Manager-led teams	Self-Managing teams	Self-Designing teams	Self-Governing teams

Figure 1. Types of Teams.

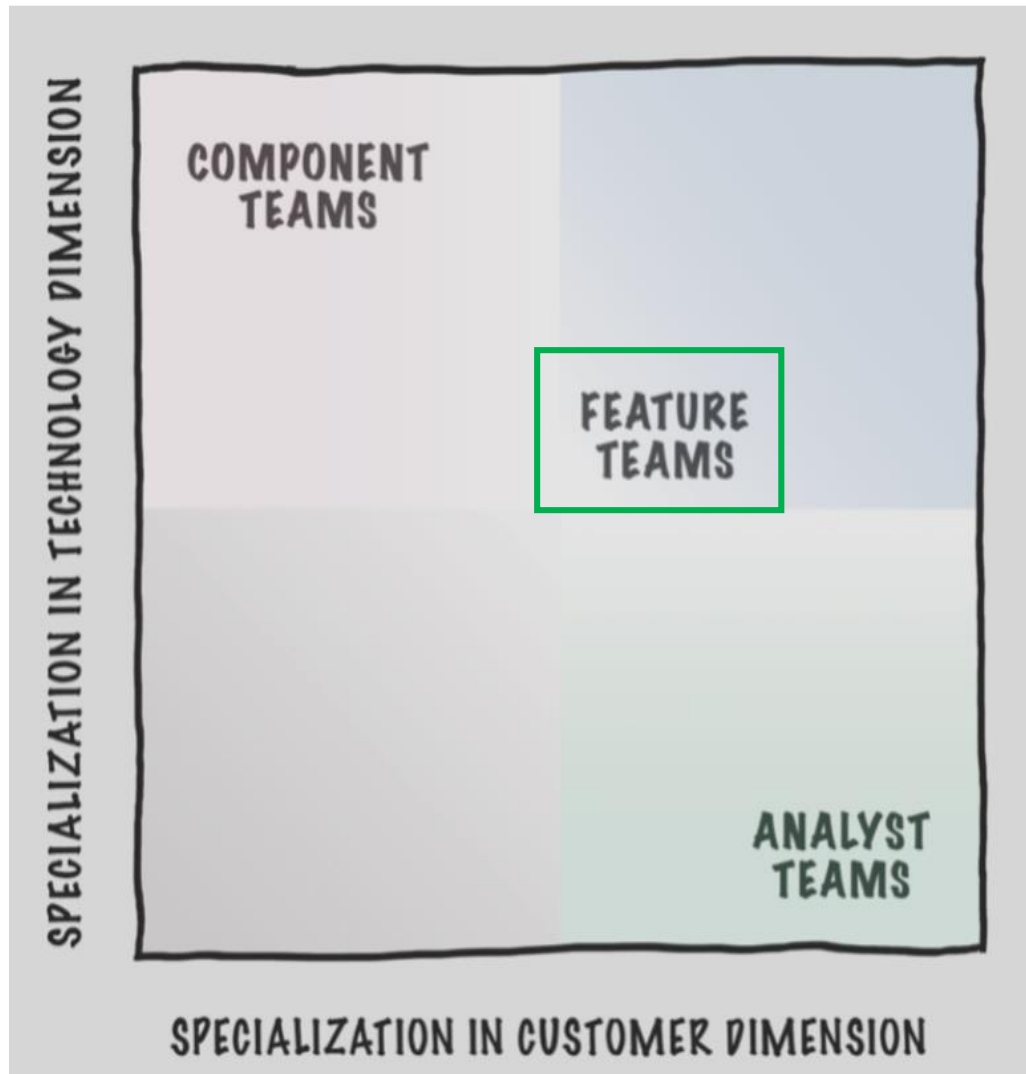
Source: <https://less.works/resources/graphics/index.html>

Feature Team Specialization



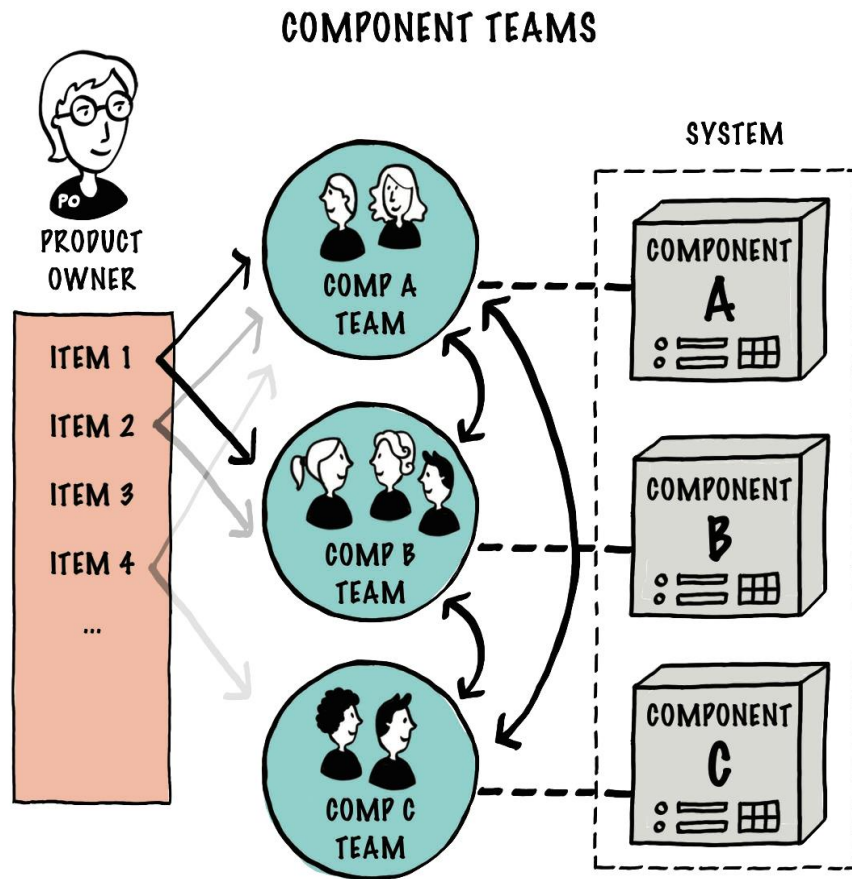
Source: <https://less.works/resources/graphics/index.html>

Feature Team Specialization

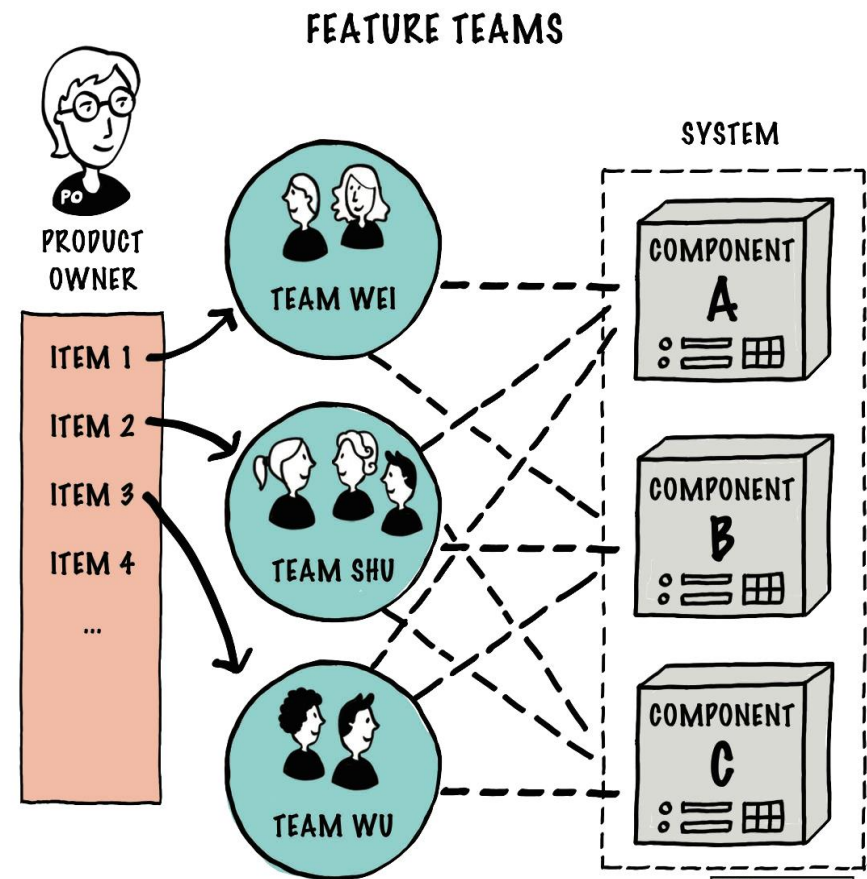


Source: <https://less.works/resources/graphics/index.html>

Component Teams vs. Feature Teams



Work from multiple teams is required to finish a customer-centric feature. These dependencies cause waste such as additional planning and coordination work, hand-offs between teams, and delivery of low-value items. Work scope is narrow.



Every team completes customer-centric items. The dependencies between teams are related to shared code. This simplifies planning but causes a need for frequent integration, modern engineering practices, and additional learning. Work scope is broad.

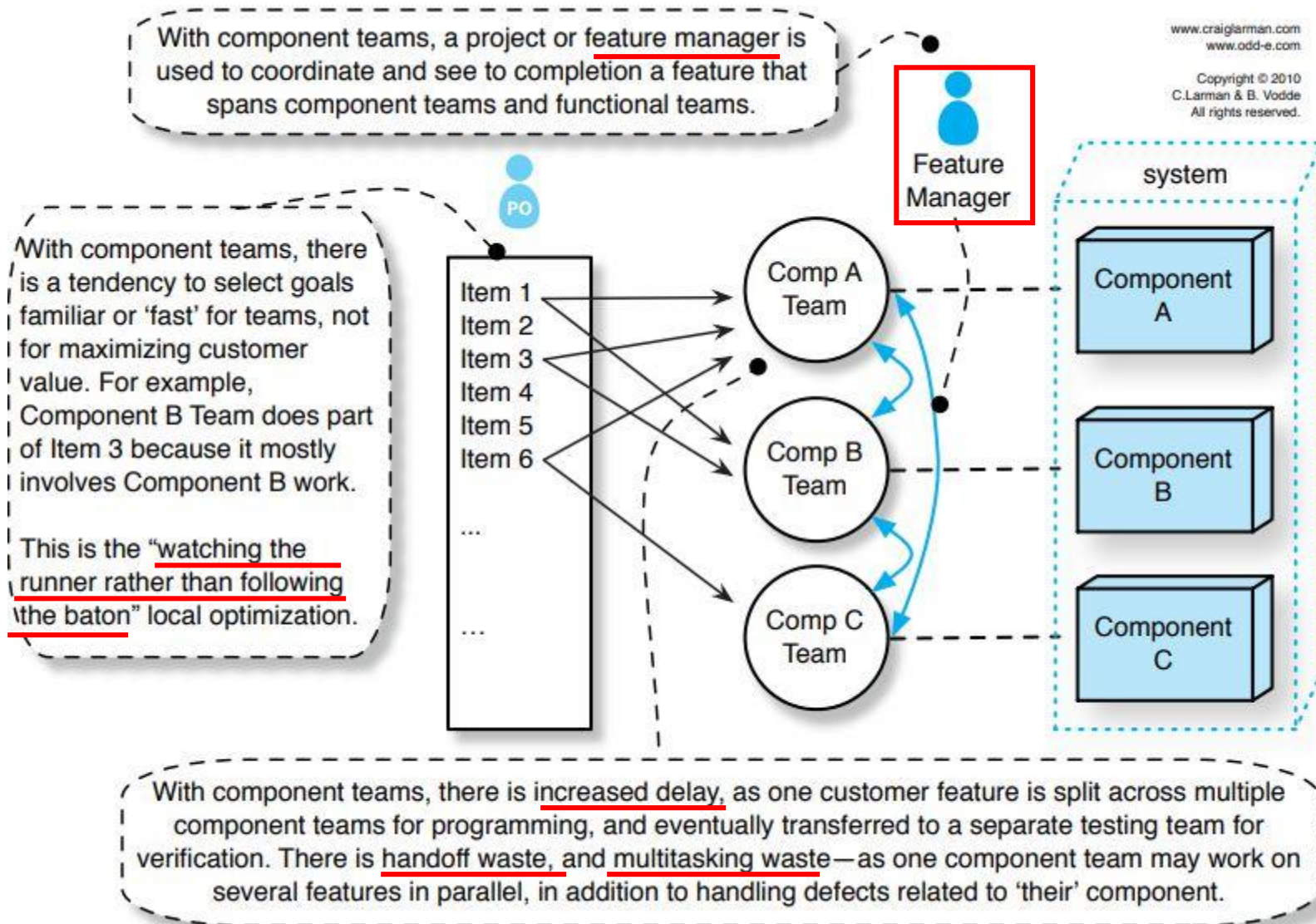
<http://less.works> CC BY-ND

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

Component Team

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

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

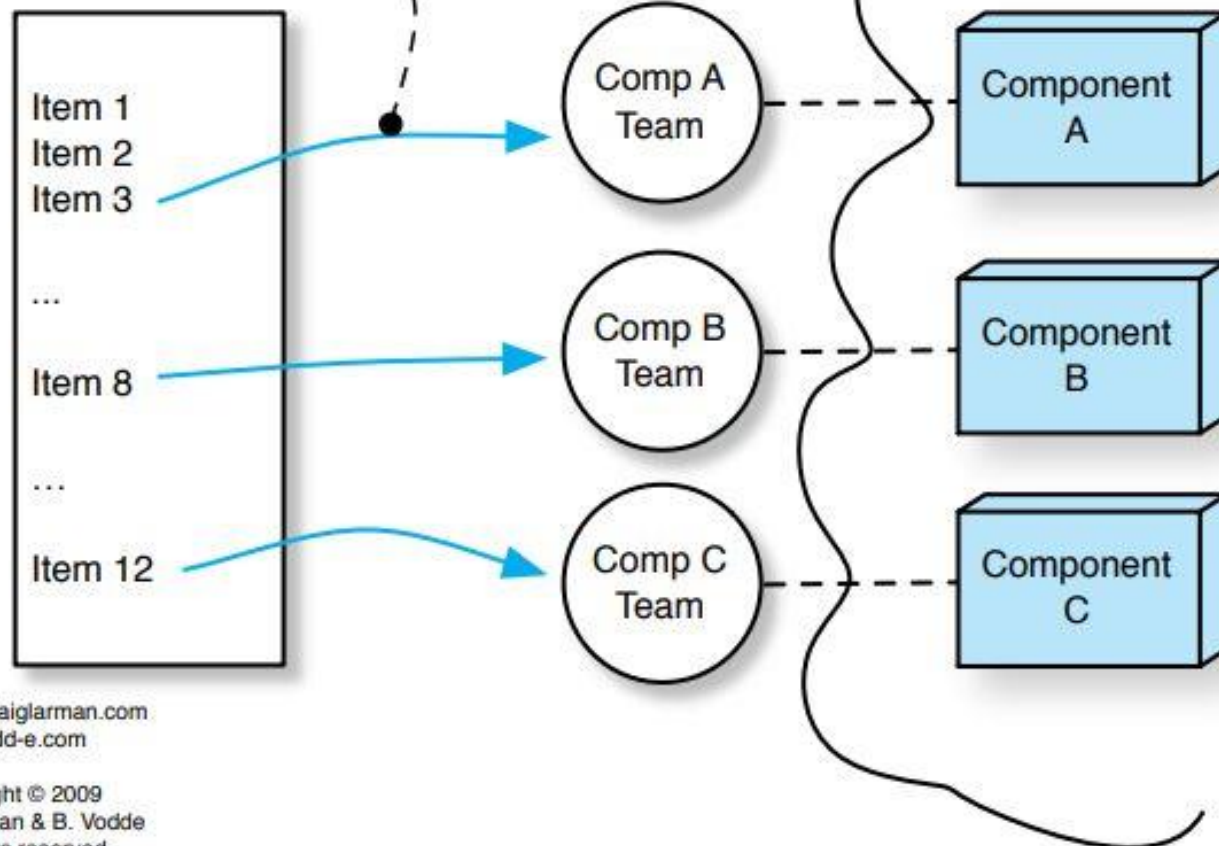


Source: <https://less.works/resources/graphics/index.html>

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

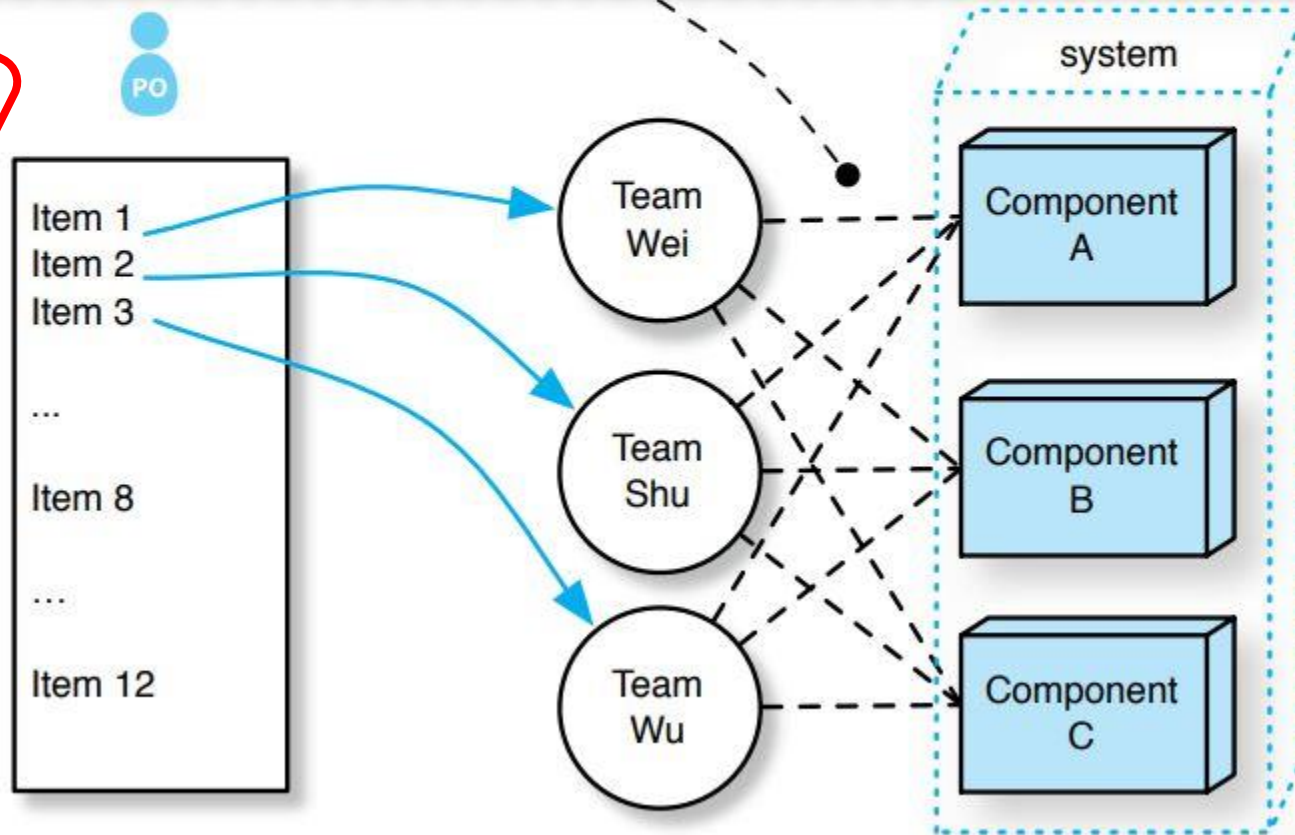
Copyright © 2009
C. Larman & B. Vodde
All rights reserved.

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

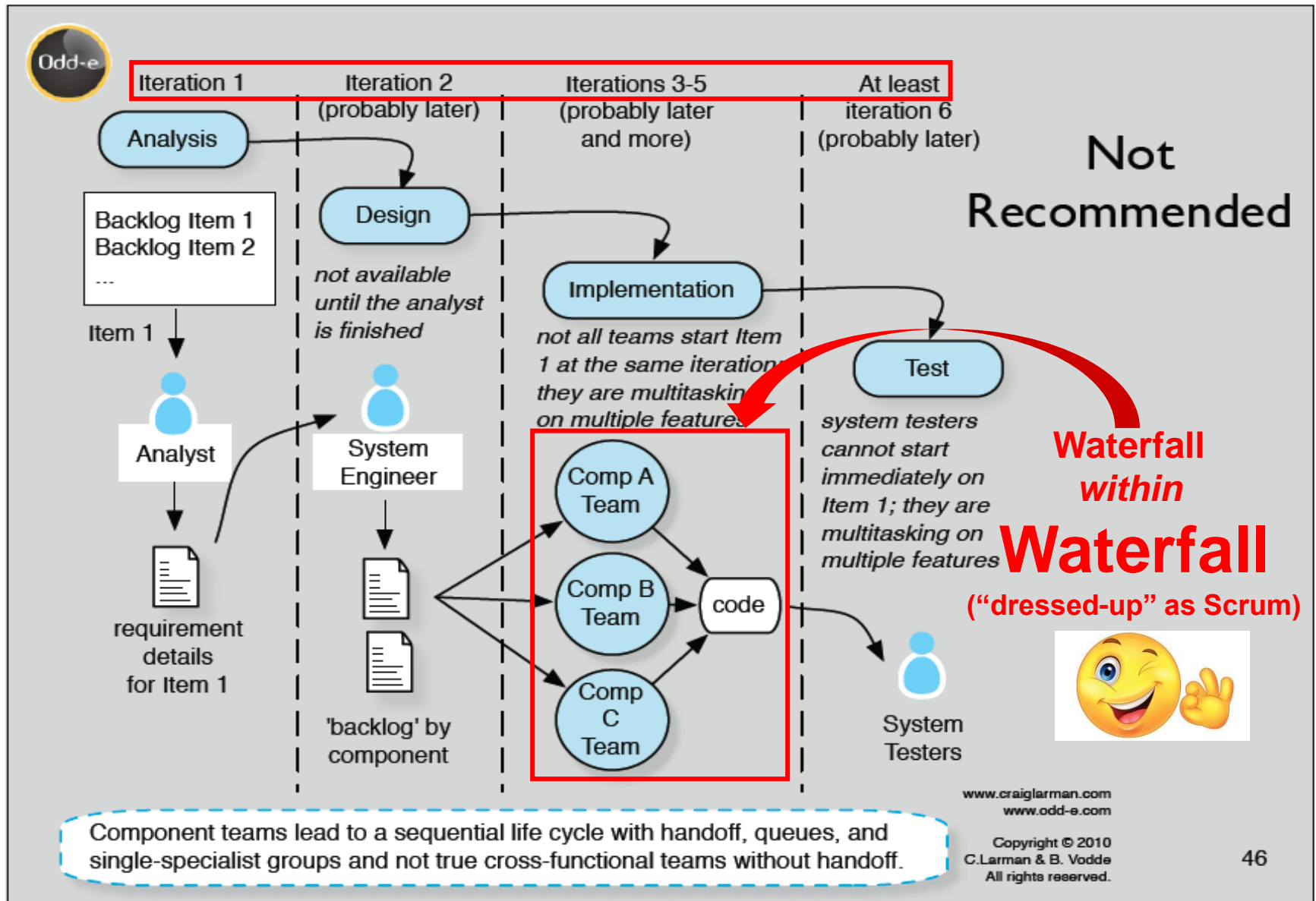


www.craiglarman.com
www.odd-e.com

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

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

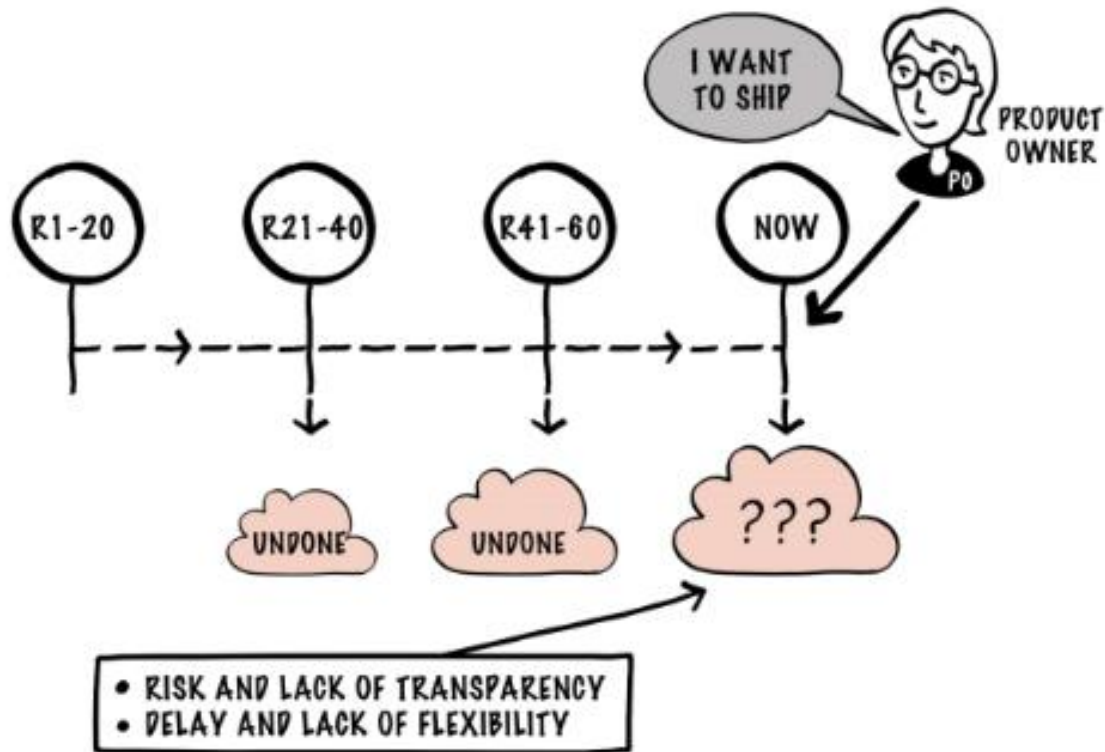
...This is not specific to
LeSS.....This is just good-
old simple Scrum...

UNDERLINED ONES ARE THE DEFINITION OF DONE

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

Technical Excellence

Dealing with Un-Done Work

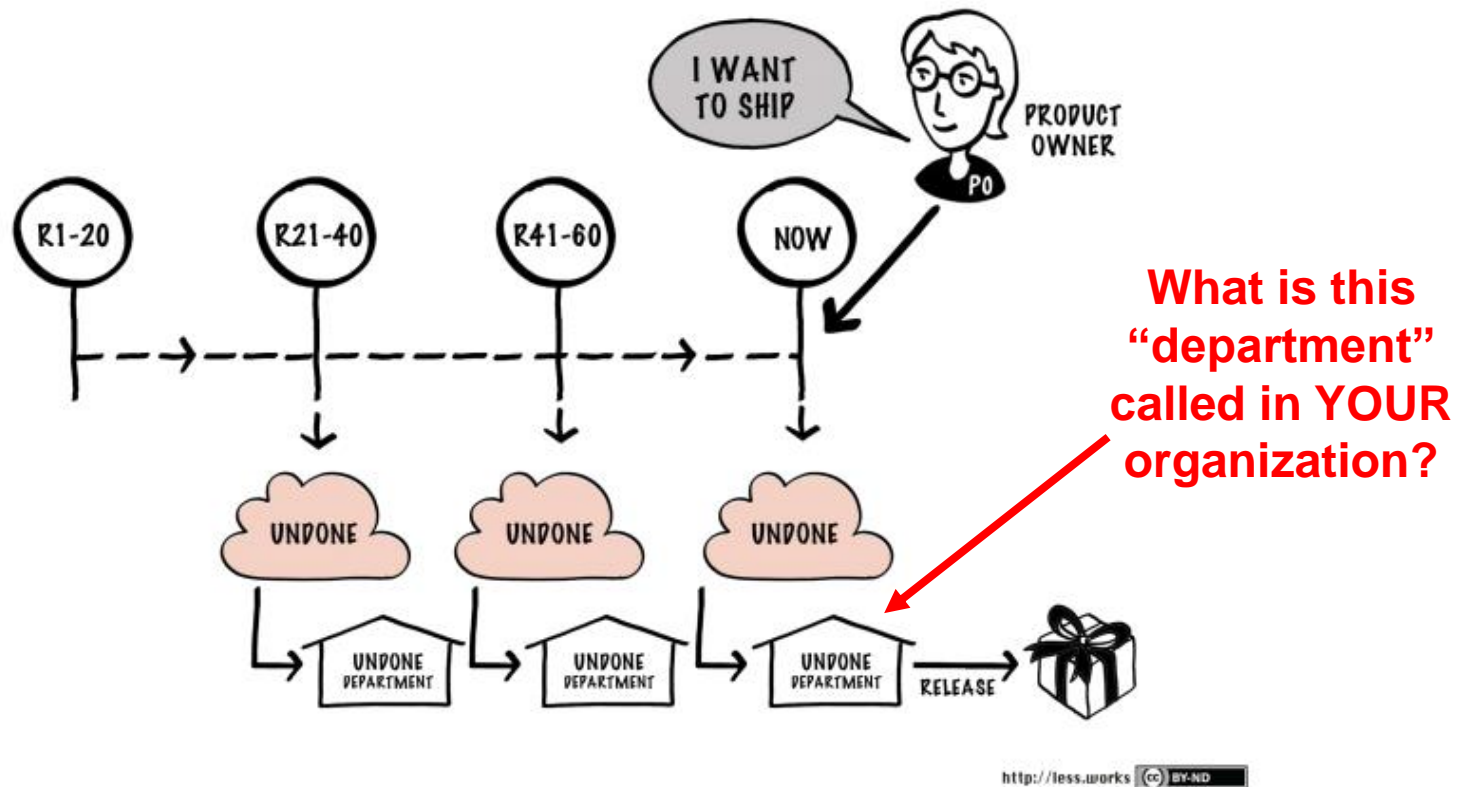


<https://less.works> 

Release Sprints

Technical Excellence

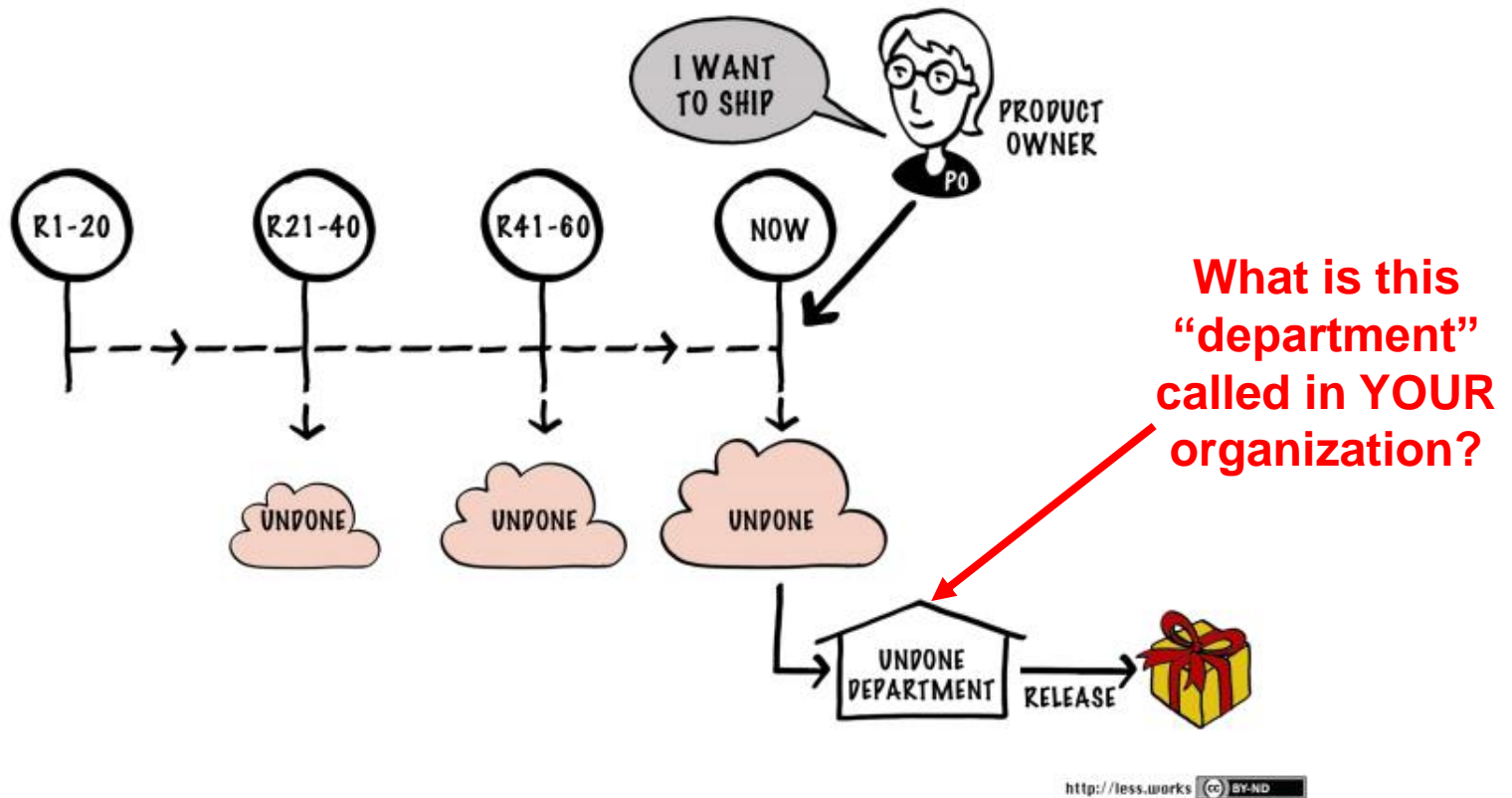
Dealing with Un-Done Work



Pipelining to Undone Department

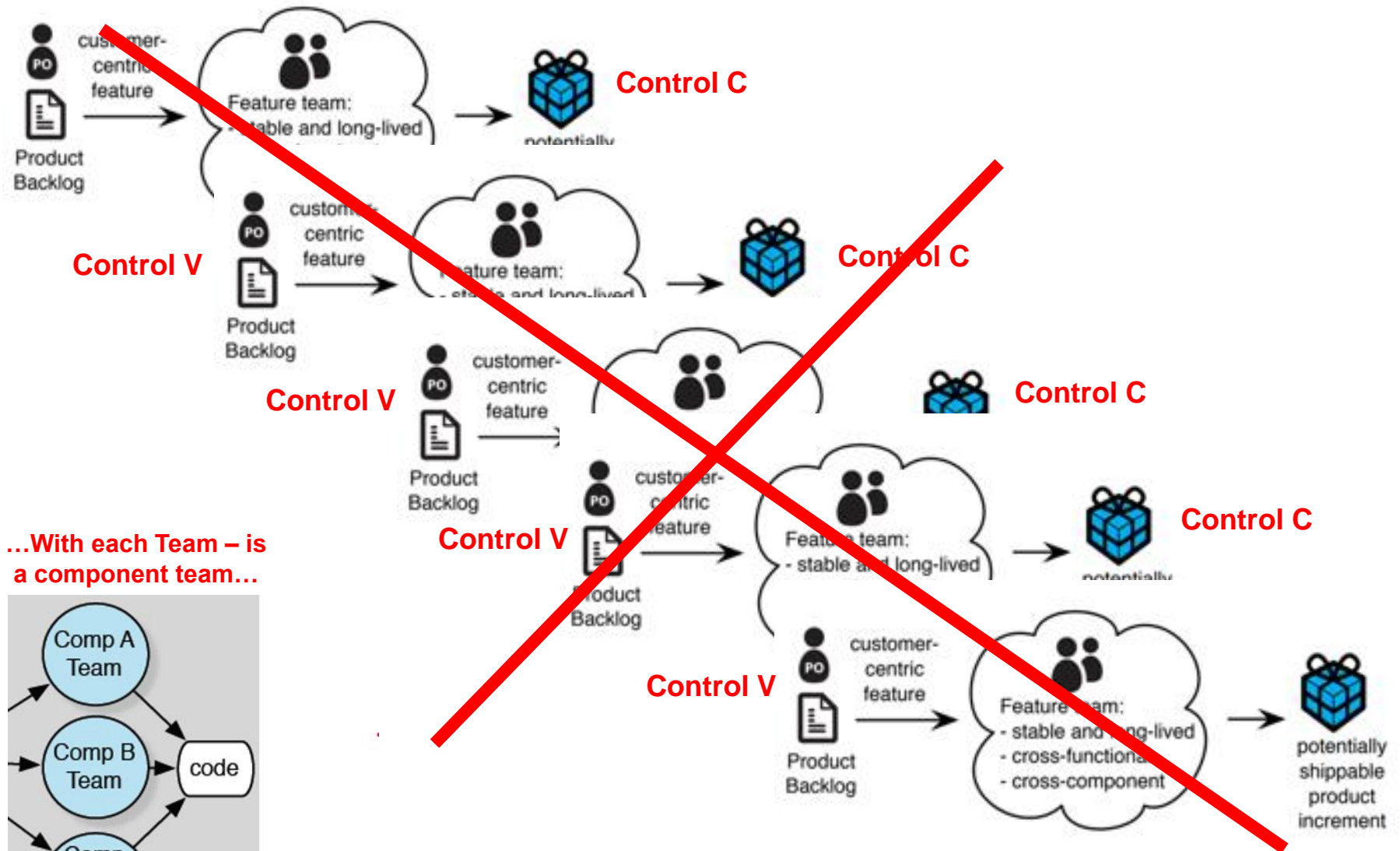
Technical Excellence

Dealing with Un-Done Work

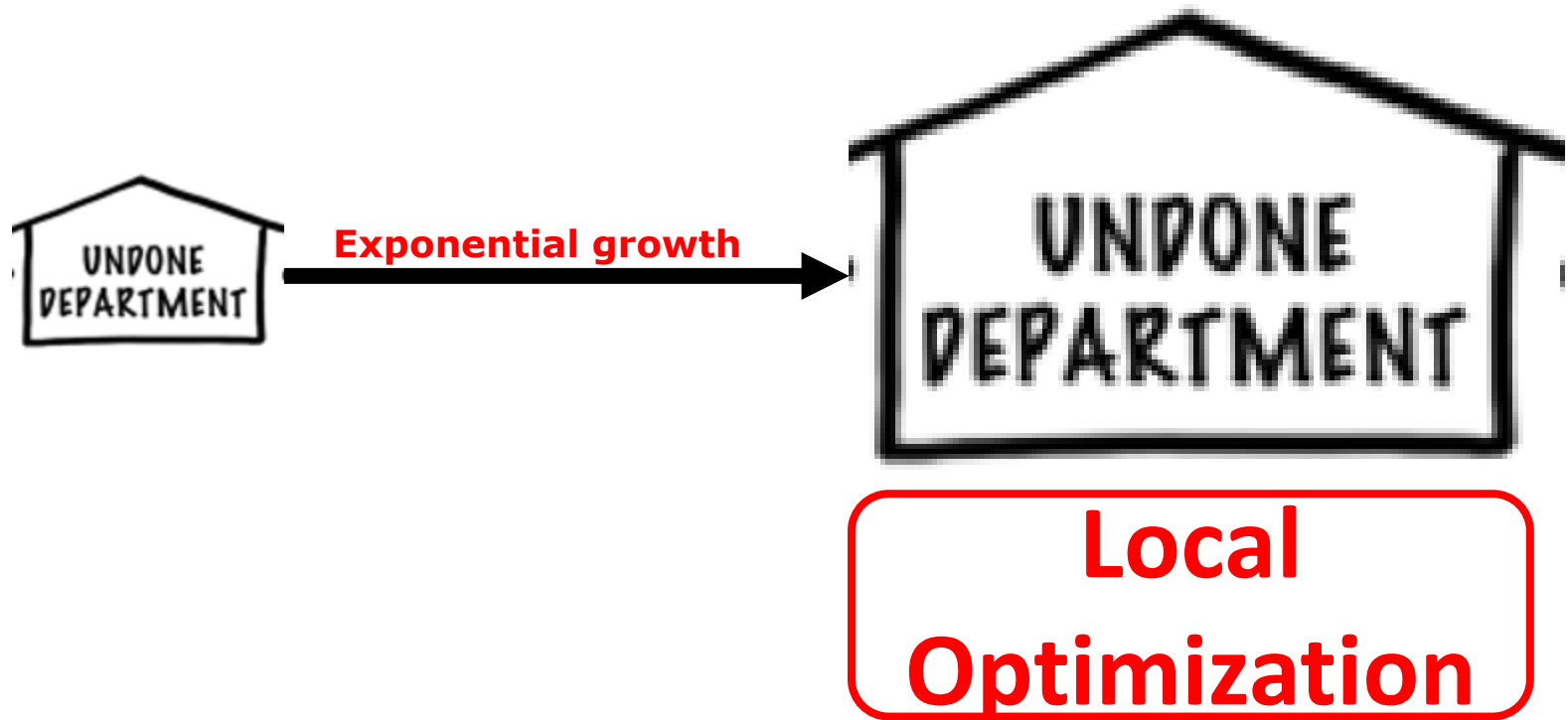


Undone Department Finalizes

What Is Fake Scaling?



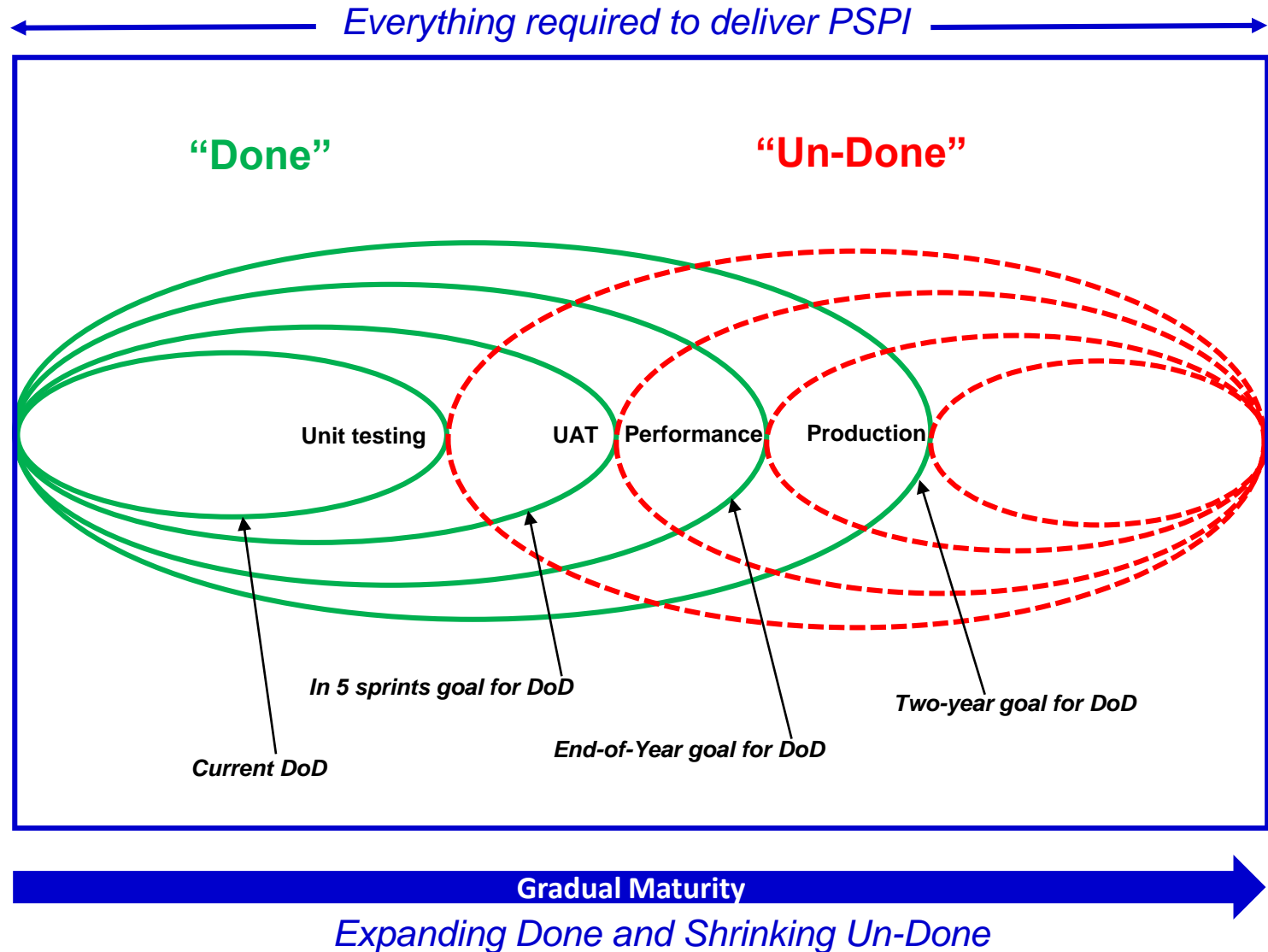
“Undone” Work, With Fake Scaling?



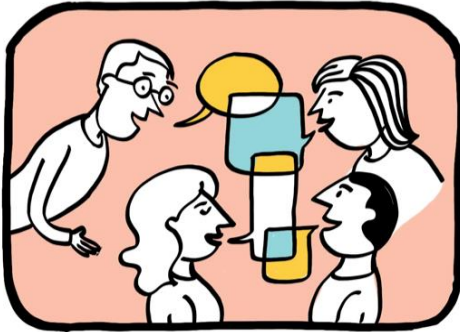
How to Eradicate “Undone” Department

**GRADUALLY,
REMOVE
UNDONE
DEPARTMENT**

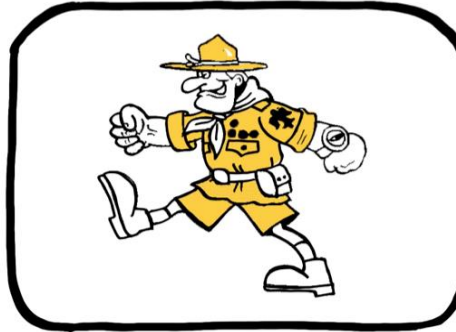
How to Eradicate “Undone” Department



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

Less is More

Scaling Organizational Adaptiveness (a.k.a. “Agility”) with Large Scale Scrum (LeSS)

Organizational De-Scaling / Flattening

Organizational de-scaling (flattening) takes months and years to complete and from a high perspective, looks like a gradual process.

However, throughout this long process, there are many phases (bursts) of: comprehensive preparation, followed by a organizational “flipping”.



Diagram created by Gene Gendel

LeSS

Simplified (flatter) and **inverted** (senior leadership is in supportive function) **organizational design**. **System Optimization**. Reduction of: silos, hand-overs. Scrum is implemented by coordinated, feature-centric teams (2-8), building the same, widely defined Product/serving the same Product Owner. Scrum is the main building block of IT org. structure. Teams are collocated. Teams strive to ‘own’ their decisions, not ‘rent’ from others. Adoption of LeSS is “**deep & narrow**”, not “**broad and shallow**”; no Big Bang transformation attempts.

Multi-site development - used for multiple locations. Reliance of technical **Mentoring** and **Communities**. No subsystem code ownership. Reduction of ‘undone’ work. Focus on **Customer value**. Support of Senior Leadership. **Involvement of HR**, supporting the concept of **job security, not role security**.

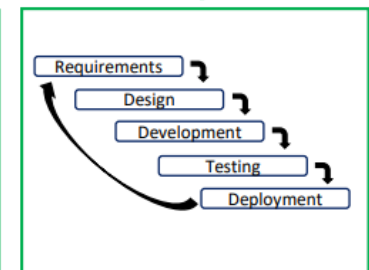
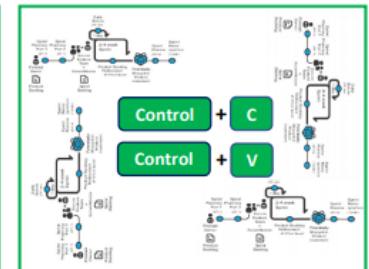
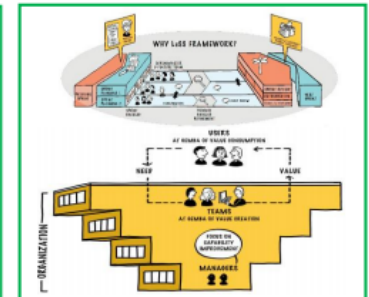
Transition from independent basic Scrum, performed by multiple teams to LeSS, is a desirable approach, when a product is widely defined and real Product Owner is identified

Scrum

Copy-paste scaling (no conscious scaling strategy) of Scrum throughout an organization: many teams doing their ‘own’ Scrum. Frequently manifested, when there is an internal competition between departments or company areas, on e.g. “*who is going to have more scrum teams by e.g. end-of-year*”. True product definition is weak. Cases of using **Scrum for component-centric development** are frequent (often, the result of trying to meet goals of agile transformation (% annually), set at enterprise level. This results in massive integration problems that need to be resolved by adding “*integration/stabilization*” sprints. Importance of **Scrum dynamics and roles** is viewed as **secondary**, to existing organizational structures and blueprints. Too many single-specialty experts and very few T-shaped workers. **No meaningful HR changes**.

Waterfall

Complex organizational design. Domains of single-function expertise, ownership and control. Numerous silos, hand-offs and translational layers between component teams, single-function specialty groups, and technology and business, overall. **Internal contracts and blame-shifting** strongly prevail. Long cycle “from concept to cash”. **Local optimization** by single-specialty workers/departments. **Theory X management** – is the primary way to manage/treat individuals. **HR** – putting a lot of emphasis on performance appraisals and individual ownership; manipulating workers with **subjective bonuses**; fostering an environment of **internal competition** and “*I am the best and you are not*” attitude. Weak definition of product value, from a stand-point of a paying customer. Manifestation of **Larman's Laws of Organizational Behavior**.



Graphics are courtesy of <https://less.works>

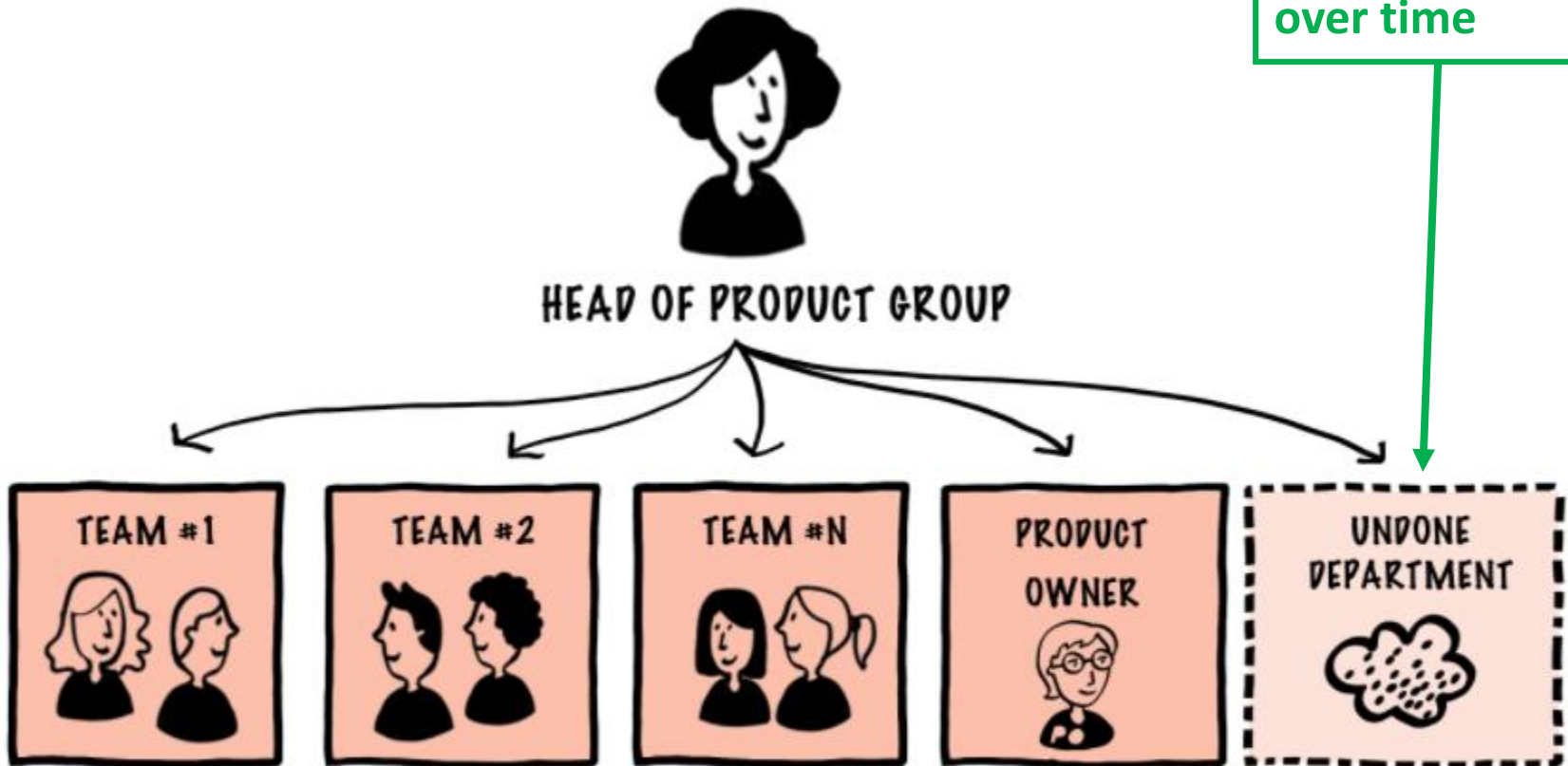
Scaling Scrum

Waterfall

Source: http://www.keystepstosuccess.com/wp-content/uploads/2018/02/scrum_scaling_org_descaling-1.pdf

Relationships in LeSS

To be dissolved
over time



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

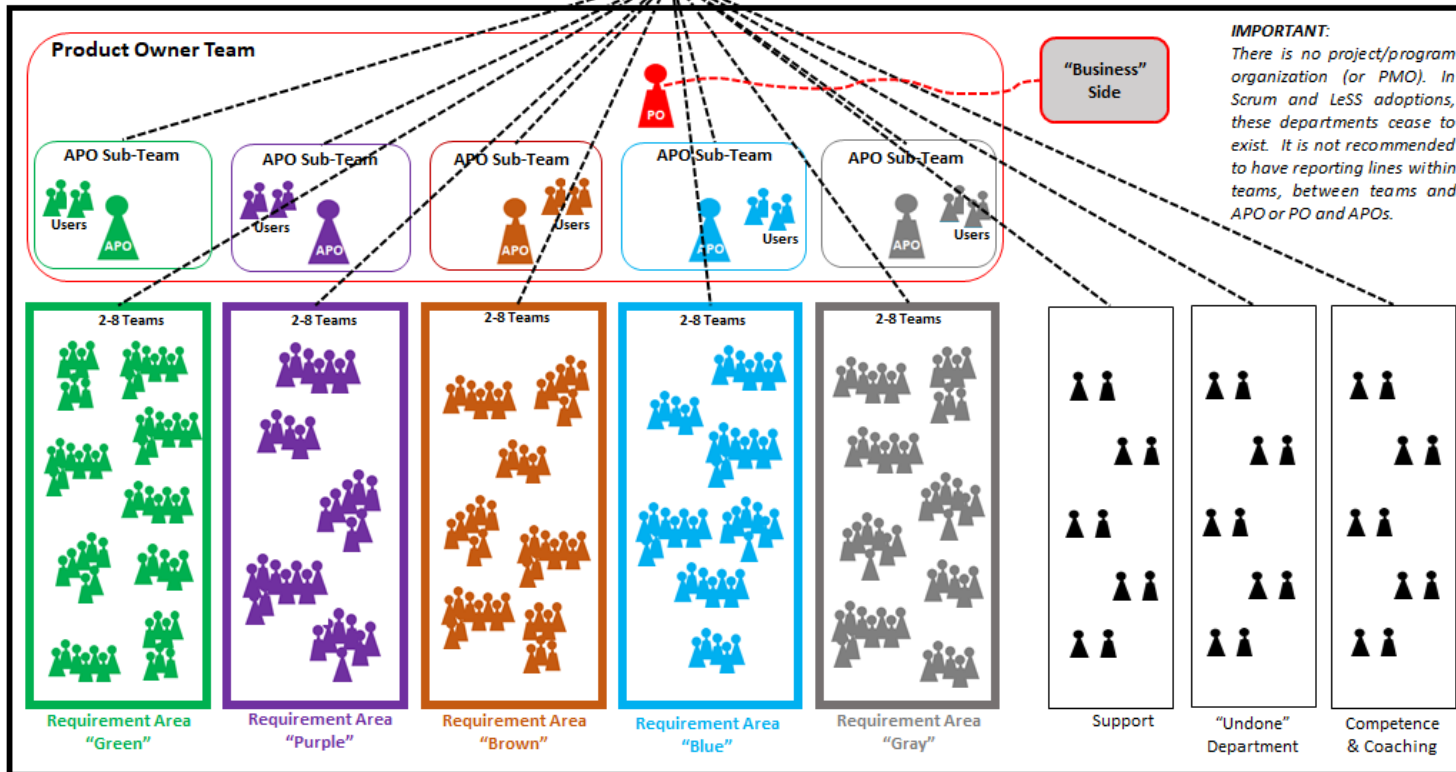
<http://less.works>

Relationships in LeSS

Guide: TRY.... Keeping LeSS Huge Organizational Reporting Structure As Flat and Simple As Possible

Product Owner Team— Product Owner plus all Area Product Owners. Conceptually, the same as in a LeSS structure. In huge LeSS Huge, there are also APO Sub-Teams that are, essentially, what is Product Owner team is in LeSS. Each APO Sub-Team consists of Area Product Owner and customers/users that can help APO with clarification and other daily work (**NOT** departments of analysts, specification writers, UI/UX designers, or architects).

Head of Product Group - most likely, a traditional “head of product group”. Whereas the whole group supports teams by Go See and help them remove obstacles and improve, Head of Product Group plays the *role of the line manager* of all the teams, whatever that is called in your organization. Product Owner Team members (including APO Sub-Teams) and all developers from teams in requirement areas, as well as people in other supportive organizational structures (e.g. support, ‘undone’ department, competence & coaching) -all report to Head of Product Group. **Note:** Since LeSS organizations don’t have matrix structures, there are no dotted-line managers.



Undone department - e.g. test, QA, architecture, or business analysts - they should not exist in the smaller LeSS framework groups; rather, they should be integrated into the teams. Undone department is still frequently seen, in some LeSS adoptions, as they often cross organizational boundaries. A goal in every LeSS adoption should be to remove an undone department.

Support - In a LeSS Huge organization, it typically centralizes some support because of the massive volume of work. Still, this department should be as small as possible with the attitude “How can we help?” rather than “Take it this way!” This department is less likely to exist in smaller LeSS organizations, where teams support each other without the need for a separate, centralized group.

Competence and Coaching (technical, organizational) - group consists of skillful practitioner-experts who actively Go See and observe how people work. They pair up and work with people to discover their training and coaching needs.

By Gene Gendel. 2020