



Certified LeSS Basics (CLB)

1- Day Review



Taking Care of Key Logistics



The Class

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.

About Training Content

- Created by: Gene Gendel
- Sourced from: <https://less.works/resources/graphics/index.html>
- Sourced from creative commons repository of Certified LeSS Trainers: <https://less.works/courses/become-less-trainer>
- Sourced from: <https://less.works/resources/learning-resources/books>
- Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>
- Sourced from: <https://www.keystepstosuccess.com/coach-tools/>

Conditions for Learning

Temporary PAUSE on:

PMI/PMO/PMP

Six Sigma/SAFe/Spotify

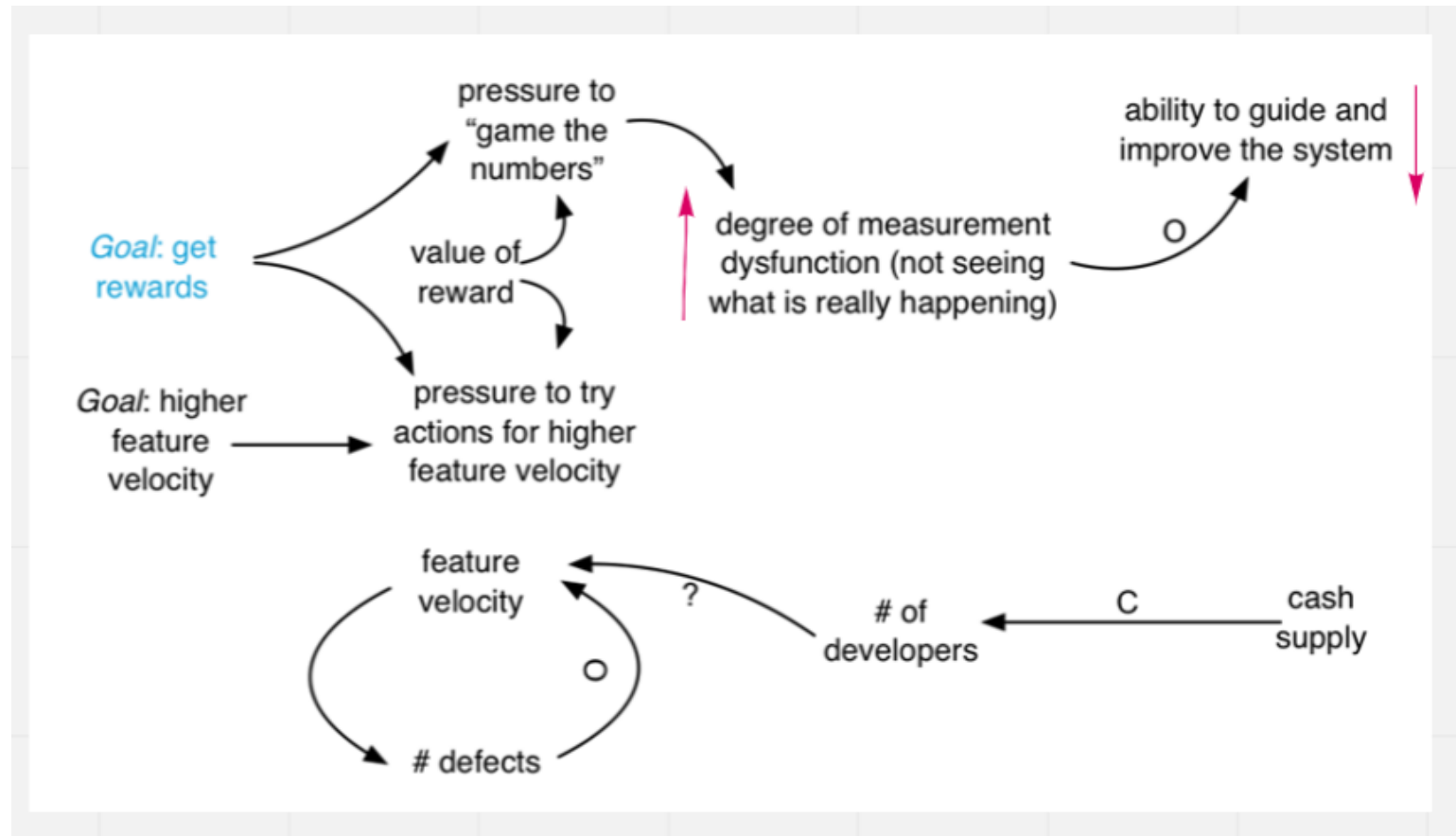
***Internal frameworks, methodologies,
“operating model”***

type of experience

Understanding System Thinking *and* System Modelling

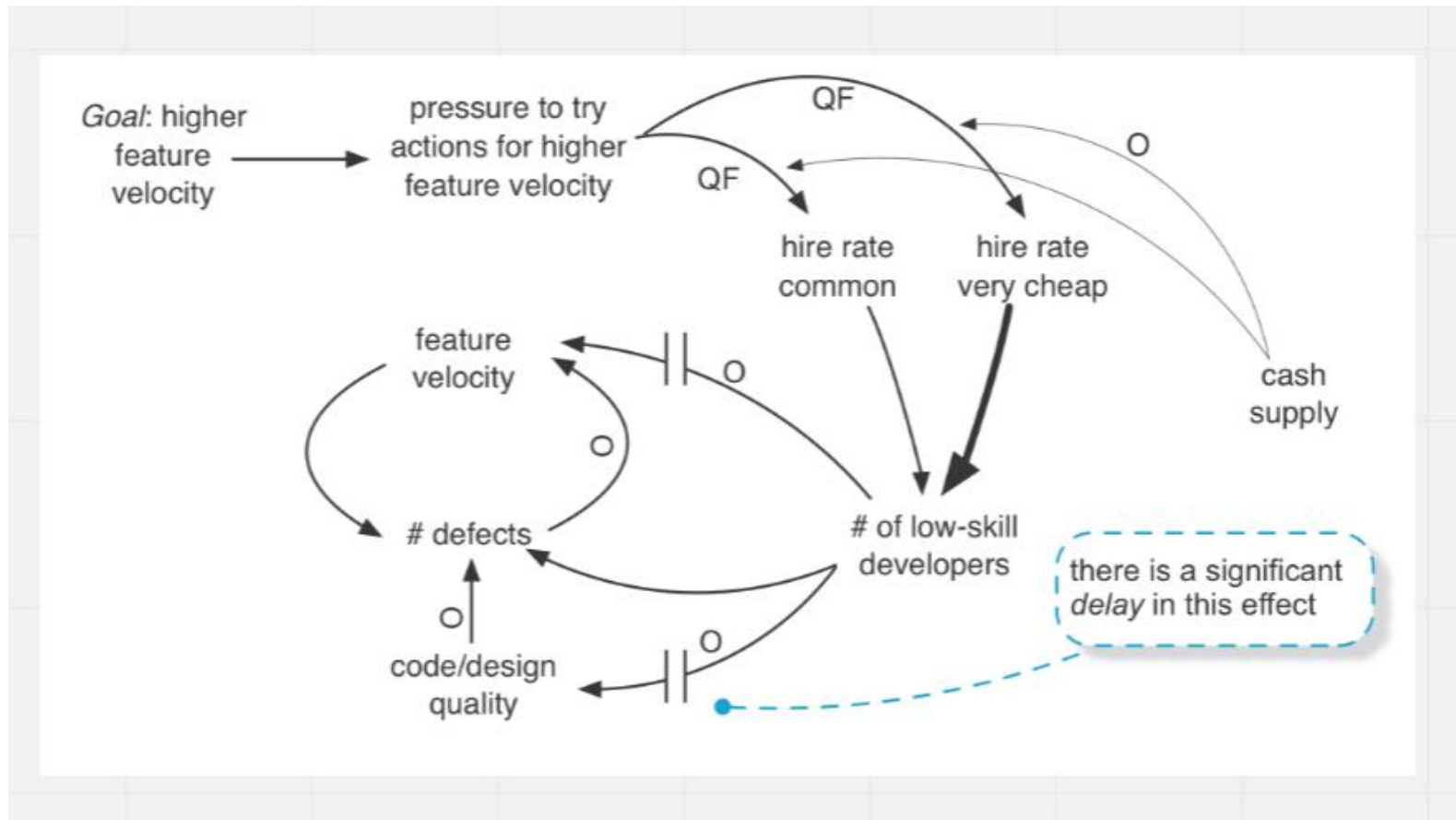
Causation **VS** ***Correlation***

System Modelling with CLD (example)



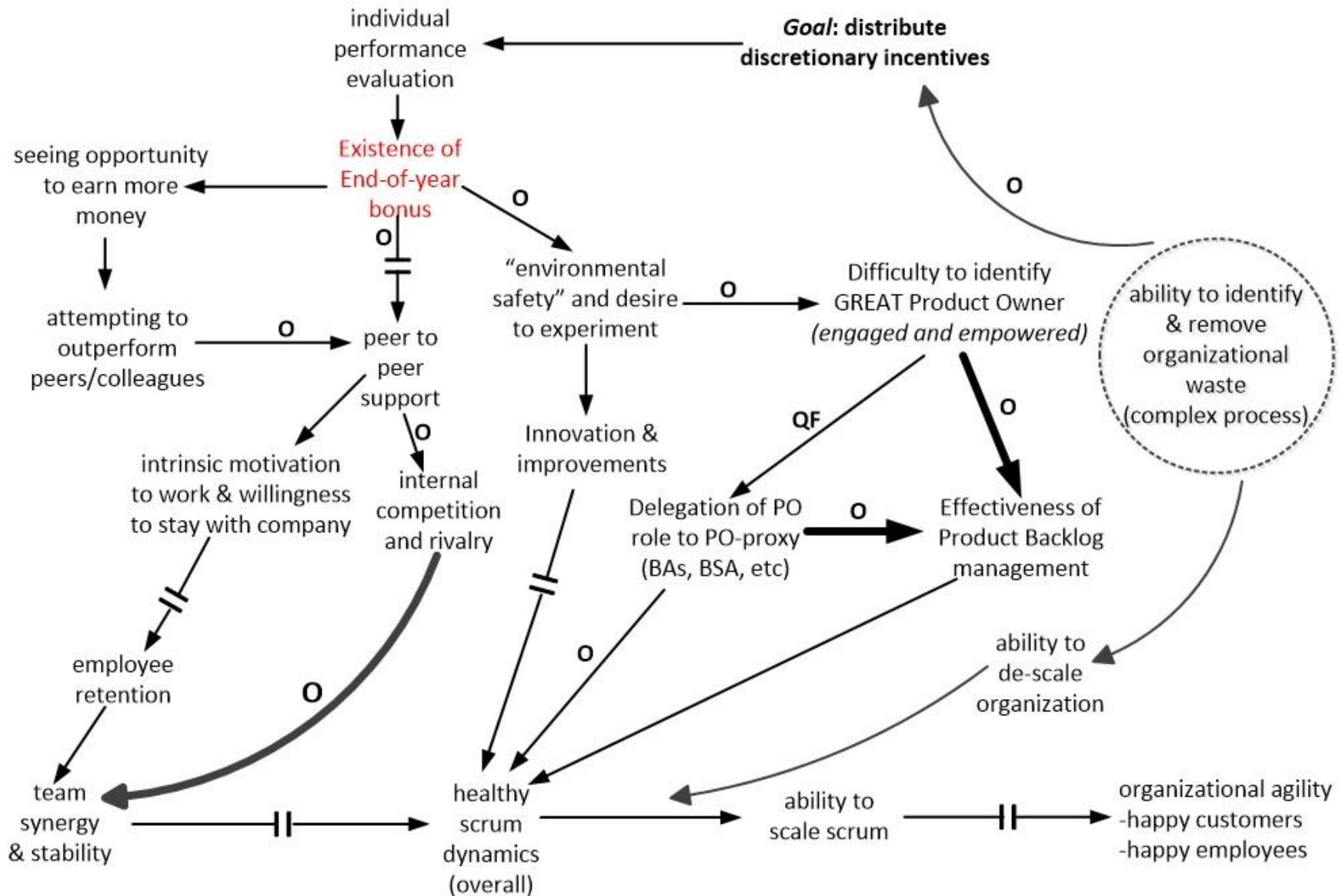
Sourced from: <https://less.works/less/principles/systems-thinking>

System Modelling with CLD (example)



Sourced from: <https://less.works/less/principles/systems-thinking>

System Modelling with CLD (example)

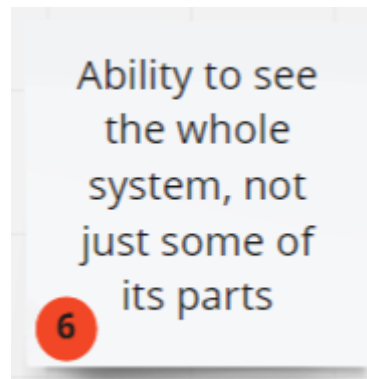


Sourced from: <https://www.keystepstosuccess.com/coach-tools/>

Causal Loop Diagrams (CLDs): Annotations

Here are some elements of CLDs that I use in my graphics:

- Goals — A high, overarching/strategic goal that needs to be achieved
- Variables — System elements that have an effect or influence on other system elements (other variables)
- Causal links — Arrows that connect two related variables
- Opposite effects — “O” annotation near an arrow; suggests that the effect of one variable on another is the *opposite* of what could be expected
- Delayed effect — “||” annotation that disrupts a causal link (arrow); it implies that there is a delayed effect of one variable by another variable
- Extreme effects — One variable has an extreme (beyond normal) effect on another variable; it is represented by a thick arrow
- Constraints — “C” annotation near arrow; implies that there is a constraint on a variable
- Quick-fix reactions — “QF” annotation near an arrow; action that brings about short-term, lower-cost effect



Errors & Omissions with Scaling

Fake Scrum & Fake Scaling

Scrum Anti-Patterns

- separate, phase-specific backlogs or single backlog with phase-specific items
- local optimization by single-skill specialists (e.g., PM, BA, QA, Architect, Developer)
- hand-overs, toll-gates, "internal contracts"
- long periods of down-time by specialists, when it is not "their phase" to work
- "water-scrum" / "scrum-fall"
- Very weak Definition of Ready & Done
- PSPI – takes many sprints to produce



Avoid This:

More obvious anti-pattern
Less obvious anti-pattern

- separate, component-specific backlogs or single backlog with component-specific items
- local optimization by component specialists (e.g. UI/UX, middle-tier, back-end, web service, architecture)
- hand-overs, toll-gates, "internal contracts"
- multiple non-development sprints needed to integrate all components and fix bugs
- Weak Definition of Ready & Done
- PSPI – takes many sprints to produce

In Scrum, in every Sprint, a team delivers Potentially Shippable Product Increment (PSPI). **This is fundamental for Scrum.** In order for this to happen, each team must possess all necessary attributes (skills, knowledge, domain expertise) required to get work fully DONE (potentially shippable). This is what makes Scrum - real Scrum. Many teams that lack the required Scrum attributes still attempt to sprint, however, effectiveness of such "sprint-like activities" is significantly reduced. **Not all anti-patterns of Scrum are equally obvious.**

1)

When Waterfall / Sequential Project Phase Team(s) "sprint", it looks like this:



2)

When Component Team(s) "sprint", it looks like this:



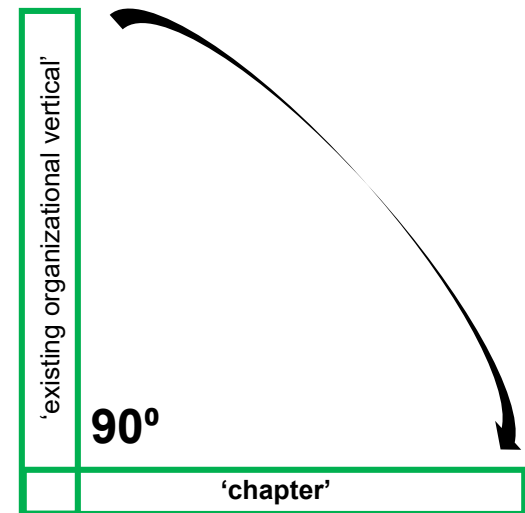
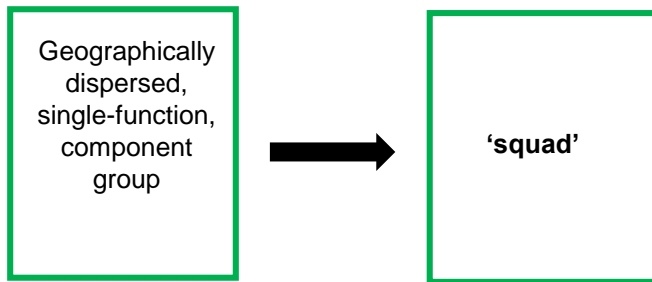
Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

- **Product definition is weak.** Applications and components that don't have strong customer alignment are treated as products
- **"Doing Scrum"** efforts are often a result of trying to meet goals of agile transformation (some annual % goals must be met), set at enterprise level
- **"Territorial" code ownership**
- Top-down, **"command & control"** governance, with little autonomy and self-management at team level
- Importance of Scrum dynamics and its roles are **viewed as secondary** to existing organizational structure blueprints
- **Too many single-specialty experts** and very few T-shaped workers
- **No meaningful HR changes** to support Scrum team design

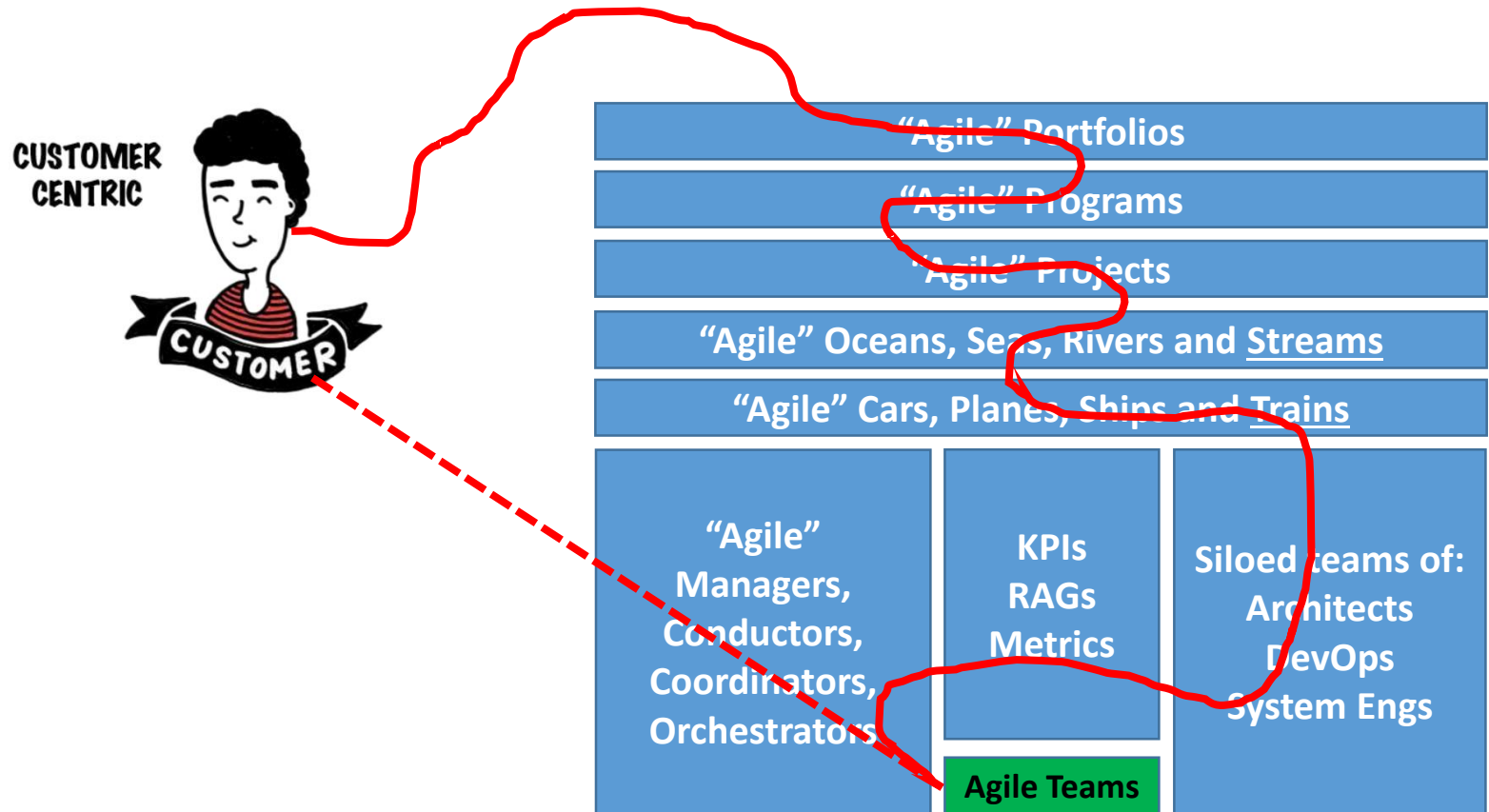
Instituting Fake Teams

Relabeling large, geographically dispersed, single-function specialist/component group, managed by engagement managers and technical leads (with groups, frequently coming from external vendors, bound by fixed-everything SOWs) into a 'squad'- is a **masquerade**

Taking an existing organizational vertical, flipping it on its side, and then calling it a 'chapter', while preserving all elements of the original structure (reporting lines, management relationships, individual performance by 'chapter leads', mandatory participation) - is a **masquerade**



Proximity of GEMBA to Customer



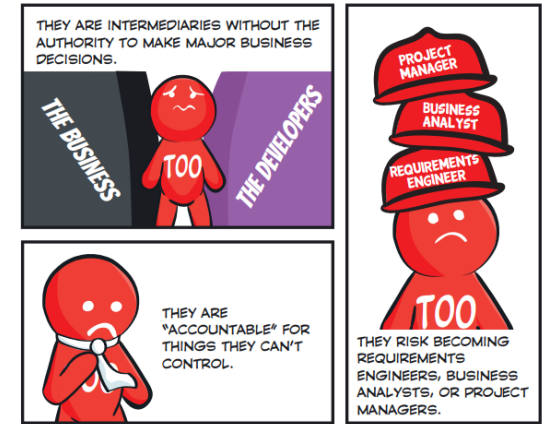
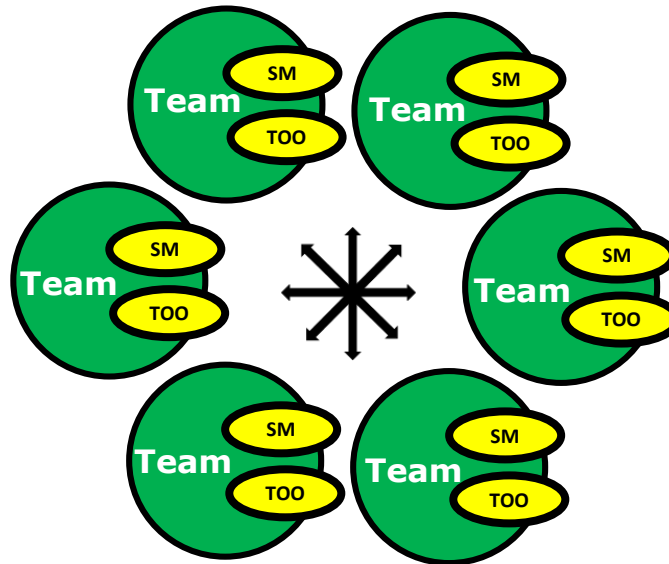
Created by: Gene Gendel

Version 348.54

- Where is customer – centricity?
- How **loooooong** is a feedback loop between customers and developers?
- How adaptive (agile, fast-reacting) can teams really be, to Customer's requests?
- What has changed from what you saw prior to "installation"?

<https://www.keystepstosuccess.com/2020/05/05-05-less-talks-dave-snowden-answering-tough-questions-qa/>

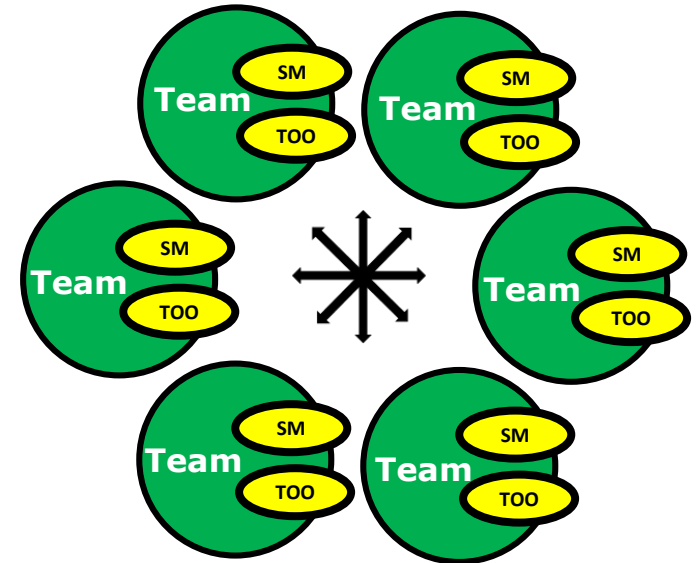
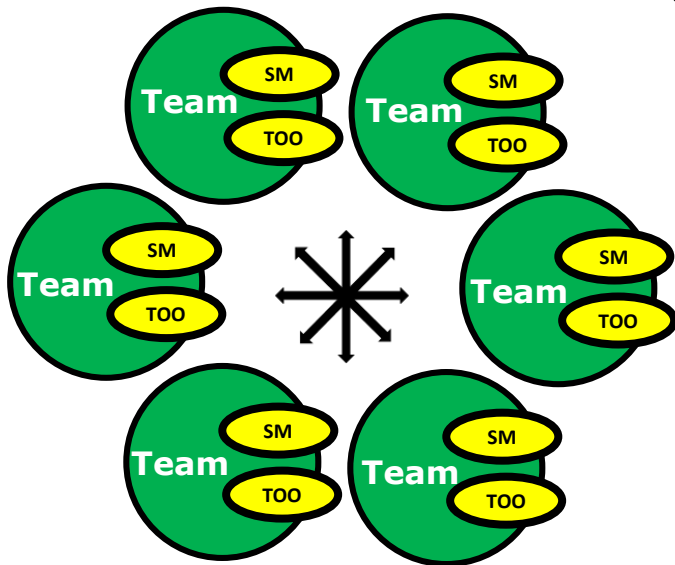
Fractal Scaling (TOO != PO ☺)



(C) 2019 Michael James (mj). All rights reserved.

18

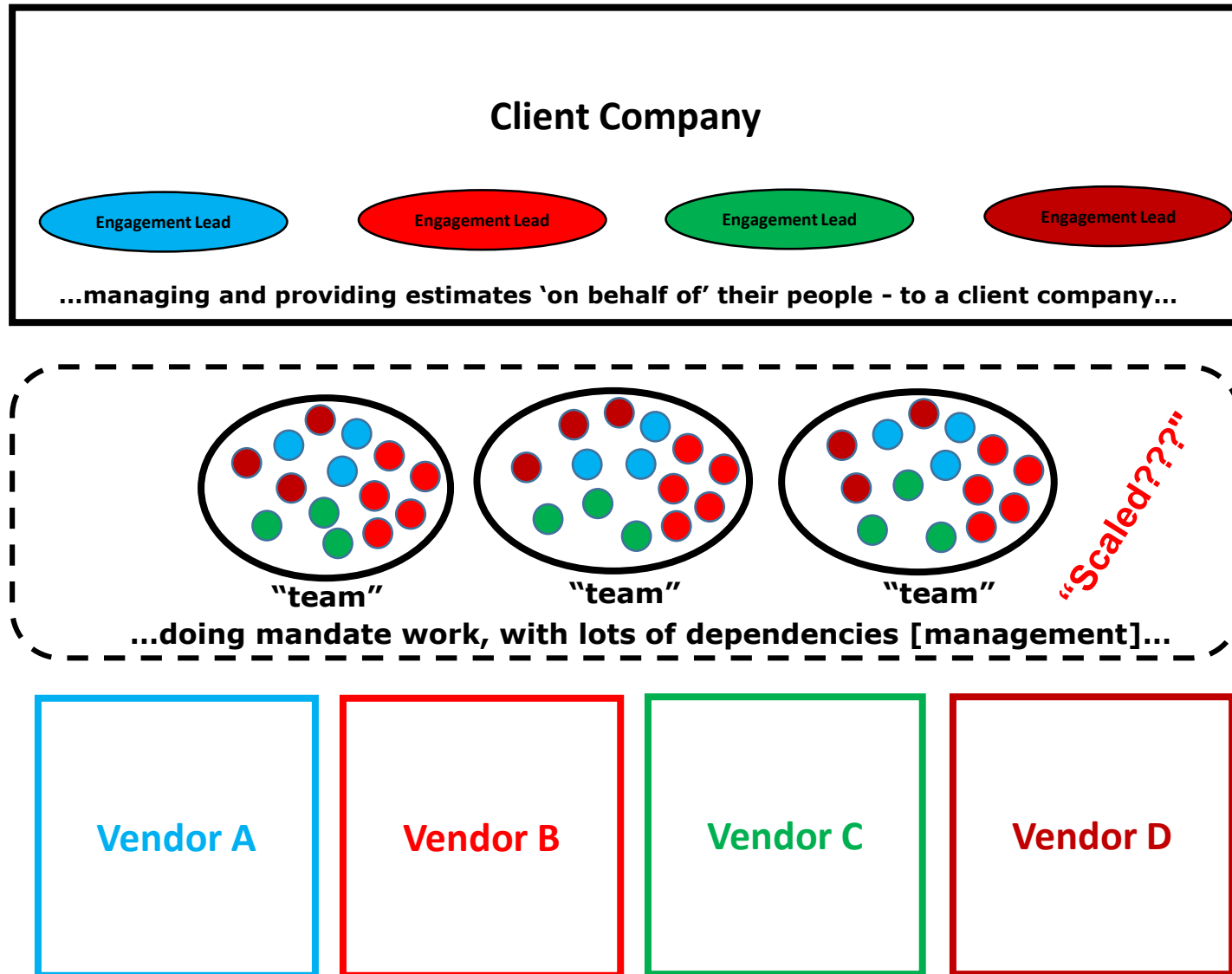
<https://seattlecscrum.com/Why-Scrum-Isnt-Making-Your-Company-Very-Agile/>



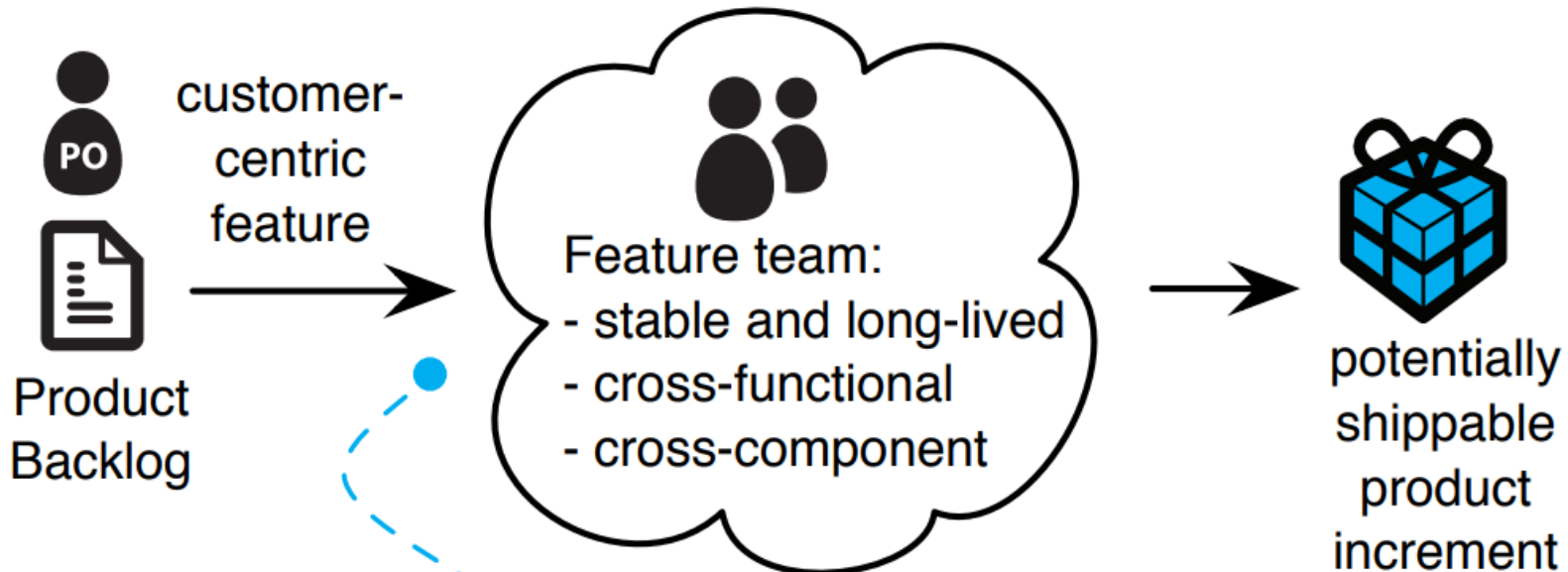
Team private backlogs. Coordination Overhead. Local Optimization.

Created by: Gene Gendel

“Scaling” Contractually-Bound Vendor Groups



LeSS is Scrum-based Approach



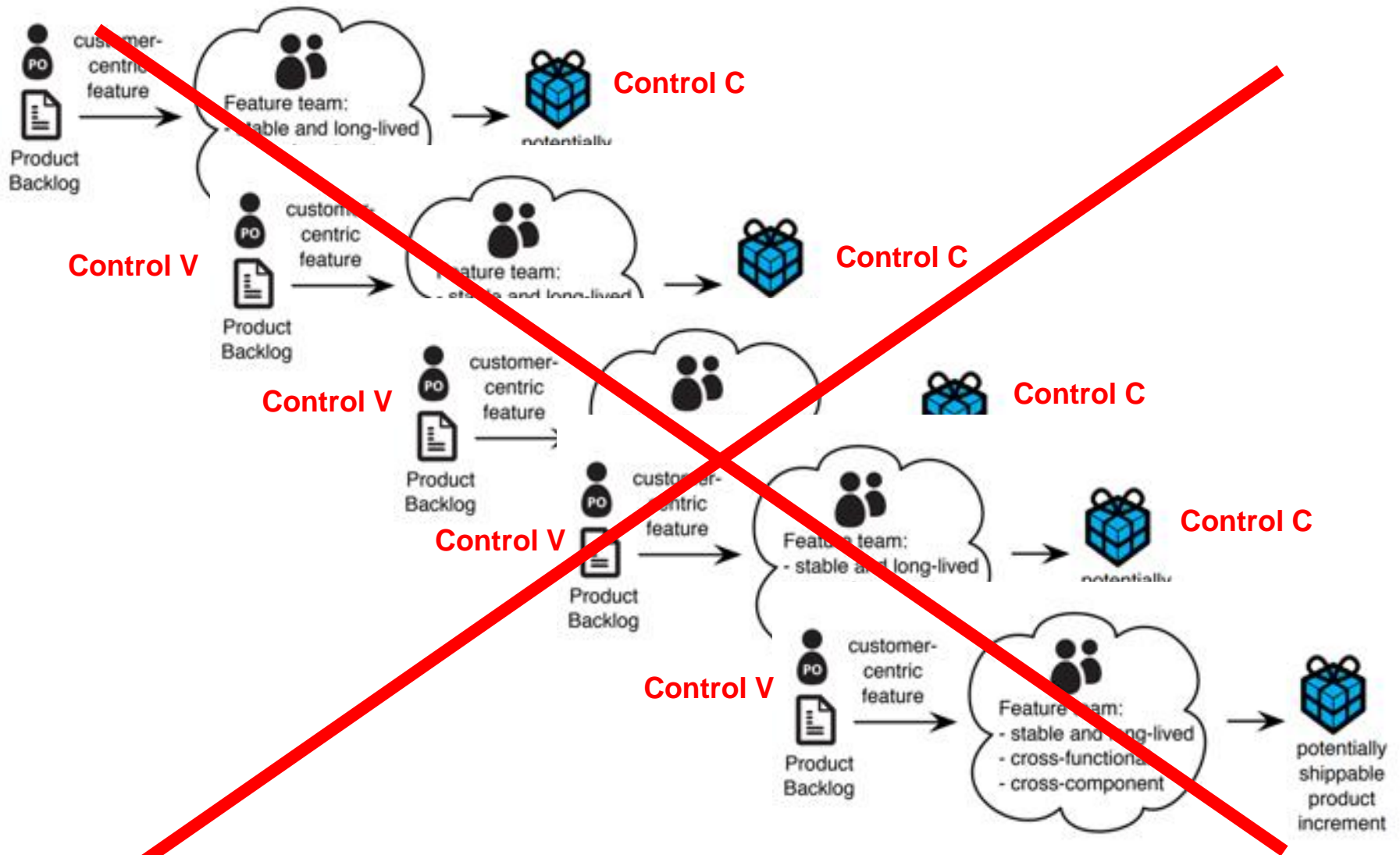
Team has the necessary knowledge and skills to complete an end-to-end customer-centric feature. If not, the team is expected to learn or acquire the needed knowledge and skill.

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

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

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

“Copy-Pasting” Is NOT Enterprise Scaling



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

Local Optimization in Agile “Big Bangs” - Instructions

Exercise

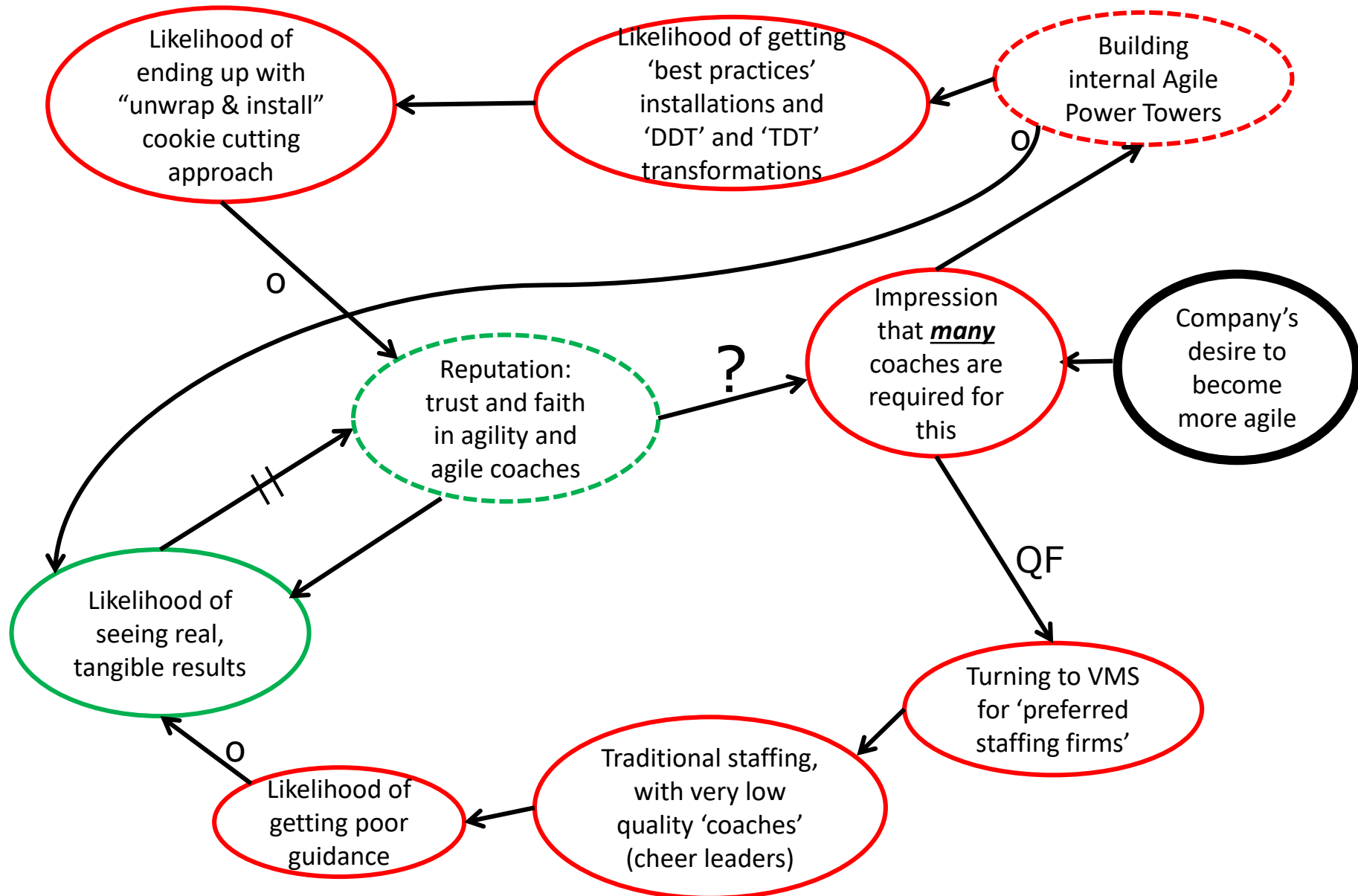
Duration: 10 min

Class: in-groups, brainstorm what could some reasons for **Big-Bang Superficial Agile Transformations** and what they may lead to.

Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in Agile “Big Bangs” - Exercise



Reading References

SAFe

- [05/05 – LESS TALKS: Dave Snowden: Answering Tough Questions](#)
- [Ken Schwaber: unSAFe at any speed](#)
- [Mike Cohn: L.A.F.A.B.L.E \(Large Agile Framework Appropriate for Big, Lumbering Enterprises\), by Mike Cohn](#)
- [S. Fe is not Agile. S. Fe is not even Scrum, by Mike Beedle](#)
- [Issues with SAFe, by Ron Jeffries](#)
- [“SAFe = shitty Agile for Enterprises”, by Martin Fowler](#)
- [SAFe: Market Share Increase, Rapid Growth, What Is The Recipe?, by Gene Gendel](#)
- [Dan Snowden: SAFe: the infantilism of management](#)
- [Does SAFe agree with the Agile Manifesto?, by Peter Merel](#)
- [SAFe ≠ AGILE, by Tom Mellor](#)
- [Dependencies, Scrum of Scrums, and SAFe, by Ron Jeffries](#)
- <https://ronjeffries.com/categories/safe/>, by Ron Jeffries
- [GOTO 2015: Agile is Dead, by Dave Thomas](#)
- [Why do so many companies seem to jump straight to SAFe when starting Agile?, by Mark Levison](#)
- [U.S. Air Force Questions about Agile /SAFe Memo? -highly discouraging from using rigid, prescriptive frameworks such as SAFe | \(Slide 12 screenshot\)](#)
- [CSO Memo on Agile – and SAFe, by Nicolas M. Chaillan \(US Air Force Chief Software Officer\)](#)
- [“It’s just a toolbox” – essentials and accidents in scaling agile, by Dr. Agilefant](#)
- [Scaling Agility or Bureaucracy, by Ari Tikka and Ran Nyman](#)
- [LeSS SAFe comparison, by Ari Tikka and Ran Nyman](#)
- [The Horror Of The Scaled Agile Framework, by Neil Killick](#)
- [You Don’t Need a Complicated Story Hierarchy, by Mike Cohn](#)
- [Let’s Acknowledge SAFe for What It Is....And Move On, by Mike Cottmeyer](#)
- [Revenge of the PMO, by Marty Cagan](#)
- [Why SAFe Is Not The Scaled Agile Approach You Need, by Renee Thoughton](#)
- [Remove References To Scrum From SAFe!, by Den Sunny](#)
- [Beware SAFe \(the Scaled Agile Framework for Enterprise\), an Unholy Incarnation of Darkness, by Sean Dexter](#)
- [10 Common Mistakes when Implementing SAFe, by Michael Küsters](#)
- [Steal “Agile”: Despicable Mission, Jacques Morali, Victor Willis, Henri Belolo](#)
- [How is SAFe different from Scrum/Agile project management?, by Peter Stevens](#)

Spotify “Model”:

- [Failed #SquadGoals, by Jeremiah Lee](#)
- [Case Study : When emulating Scaling Agile at Spotify went awry at Refinery29, by Andy Park](#)
- [The Spotify “Model”: Don’t Simply Copy-Paste, by Evan Campbell](#)
- [Spotify didn’t implement the Spotify model by copying Spotify...by Kent Beck](#)
- [“Spotify Model” – 10 lessons in transplantology, by Kate Hobler \(Terlecka\)](#)
- [Scaling Agile @ Spotify, with Tribes, Squads, Chapters & Guilds, by Henrik Kniberg & Anders Ivarsson](#)
- [Don’t Copy the Spotify Model, by Ben Linders](#)
- [Over-engineered “Agile Operating Model” by Gene Gendel](#)
- [Accelerated career path by “!@#%^ Coach”, by Gene Gendel](#)

<https://www.keystepstosuccess.com/2016/05/safe-market-share-increase-rapid-growth-what-is-the-recipe/>

Impact of LeSS On Org. Design/Structure

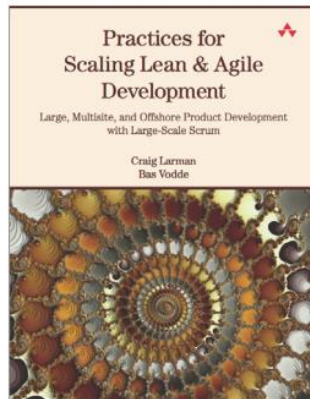
AGILE CONTRACTS PRIMER

Derived from the book...
*Practices for Scaling Lean & Agile Development:
Large, Multisite, & Offshore Product Development with Large-Scale Scrum*

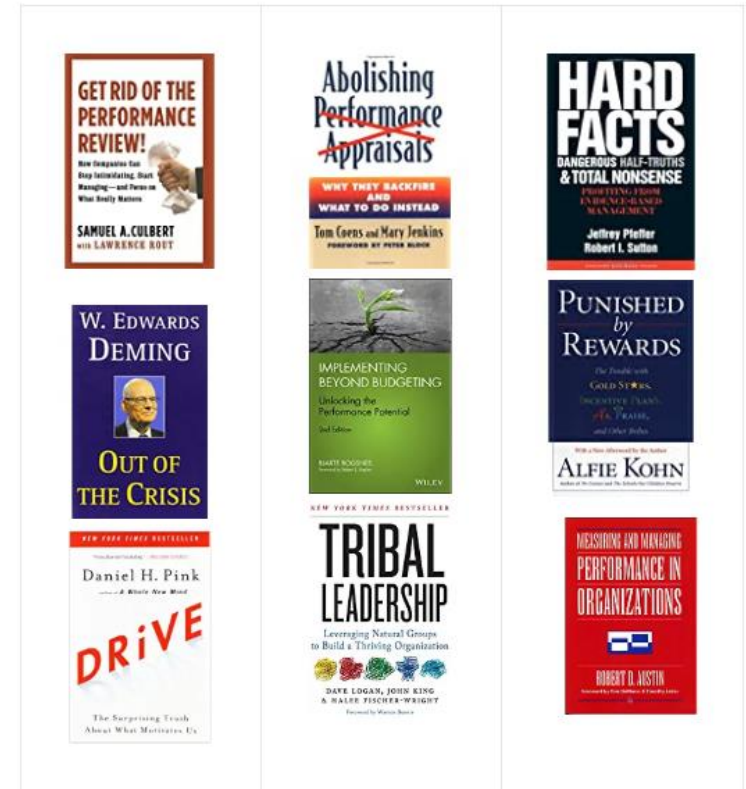
by Tom Arbogast, Craig Larman, and Bas Vodde

Version 5

Please send us comments for future versions, at www.agilecontracts.org.
Note: Check website for latest version; share the URL (rather than file) to keep up-to-date.



1	Introduction	1
2	Large-Scale Scrum	9
Action Tools		
3	Test	23
4	Product Management	99
5	Planning	155
6	Coordination	189
7	Requirements & PBIs	215
8	Design & Architecture	281
9	Legacy Code	333
10	Continuous Integration	351
11	Inspect & Adapt	373
12	Multisite	413
13	Offshore	445
14	Contracts	499



• Book Summaries

- <http://www.keystepstosuccess.com/2016/08/implementing-beyond-budgeting/>
- <http://www.keystepstosuccess.com/2016/02/quotes-from-get-rid-of-the-performance-review-how-companies-can-stop-intimidating-start-managing-and-focus-on-what-really-matters-by-culbert-samuel-a-laurence-rout/>
- <http://www.keystepstosuccess.com/2016/02/quotes-from-punished-by-rewards-the-trouble-with-gold-stars-incentive-plans-as-praise-and-other-bribes-by-alfie-cohn/>

• Proper Scaling of Scrum and Dynamic Financial Forecasting

- <http://www.keystepstosuccess.com/2018/02/proper-scaling-of-scrum-and-dynamic-financial-forecasting/>

• Gap Between Science and Business

- <http://www.keystepstosuccess.com/gap-between-science-and-business/>

De-Scaling With LeSS

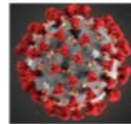
Market Reality: Lowered Tolerance To...

Tired of *Scaling*? 🤔

(...you are so not alone...)

**Maybe, it is time to
DE-Scale?** 🙌

<https://www.linkedin.com/feed/update/urn:li:activity:6691900420555608064/>



What is COVID-19 impact?

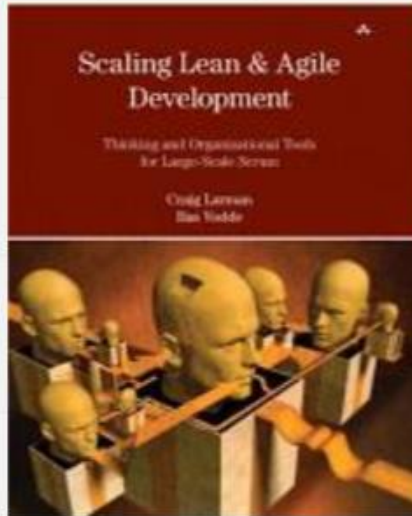
Degree of
freedom from
cognitive bias and
anchorage to
what is "in trend
& style")

2

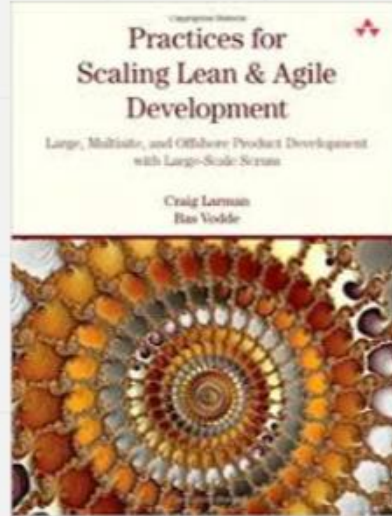
LeSS IS Scrum

(Size, Sprint Cadence, Dynamics)

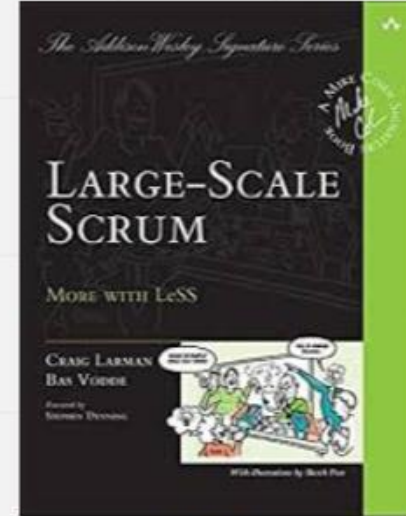
History of LeSS



2008



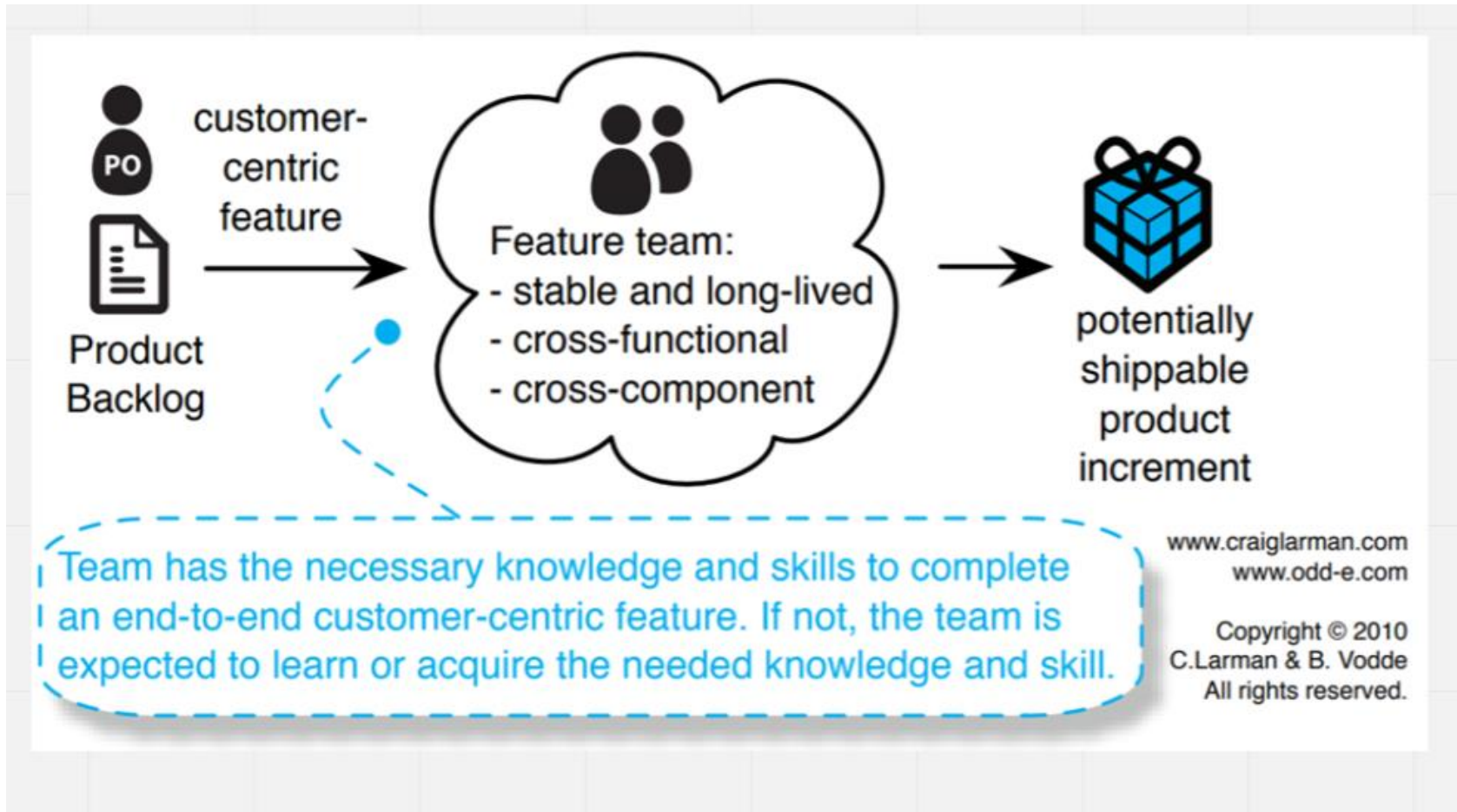
2010



2016

Depth, duration,
authenticity and
quality of
1 collected
historical
knowledge

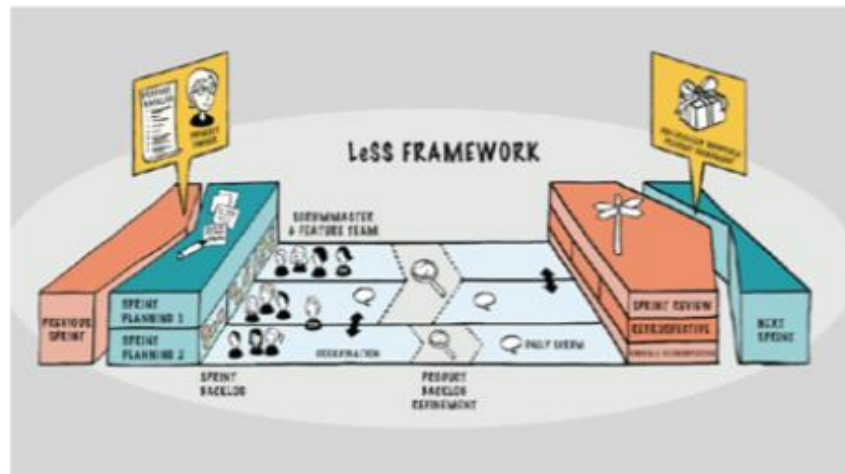
Good-Old Scrum



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

Boundaries of LeSS

- 2-8 Teams
- LeSS is no more than 70+/- a few, people: don't try to bite more than you can swallow
- **One Product. One Product Owner. One Backlog**
- LeSS adoptions require months of preparation and then a 'FLIP'.



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

The Horses Analogy

LeSS is not:

- Many teams doing their own so-called “Scrum”, and
- Working on different so-called “products”
- Delivering components that need further integration
- Working for many so-called “Product Owners” (“TOO”)
- Maintaining private, silo-ed and dependent backlogs

Many people with different strategy, vision, mission. Competing with each other.



LeSS is:

- 2-8 teams, working together in real Scrum, and:
- Working on the same, widely defined product
- Delivering fully integrated, end-to-end features
- Working for only one REAL Product Owner
- Maintaining only one, shared backlog

Only one person with shared strategy, vision, mission. No internal competition.

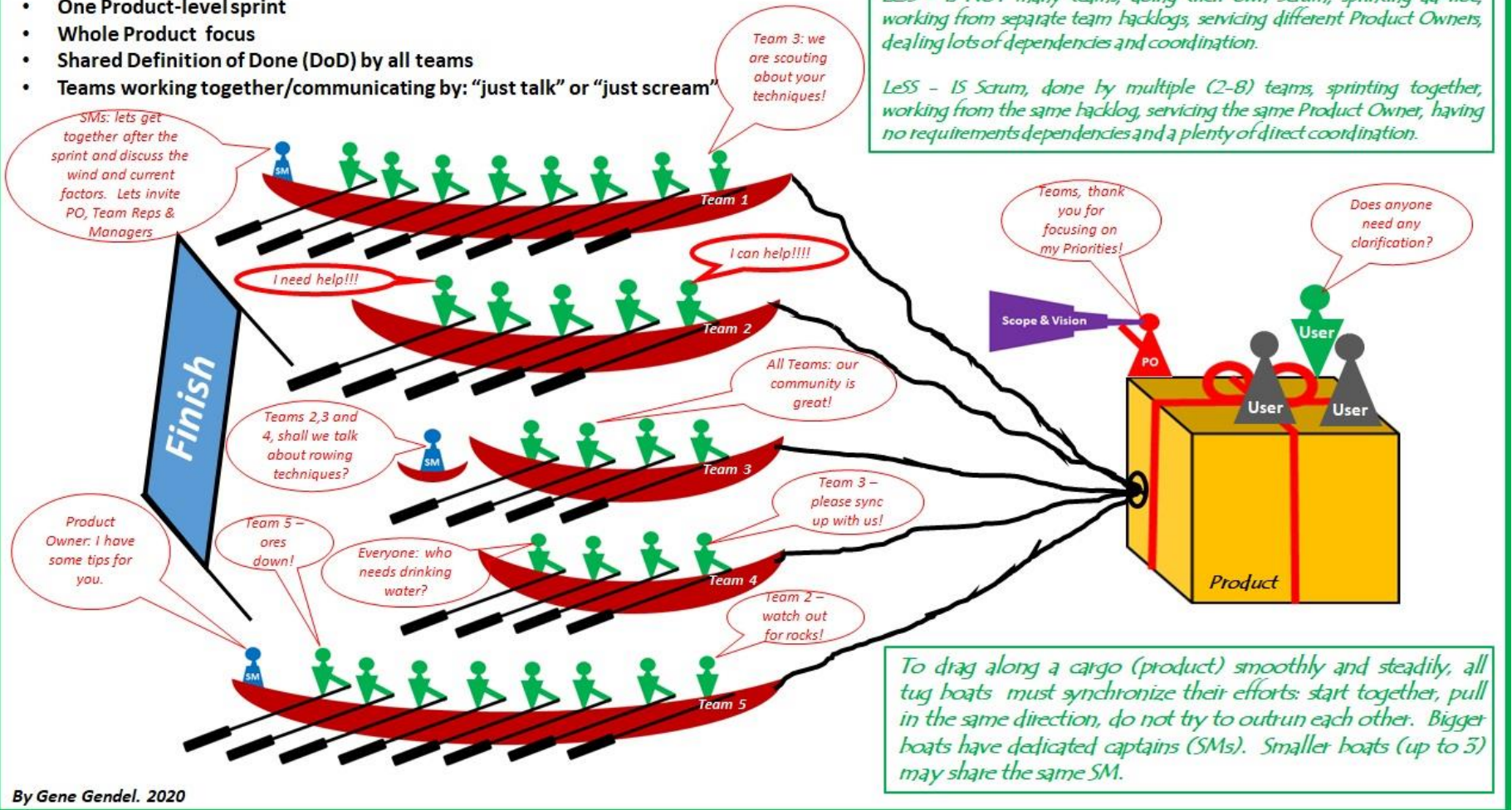


One LeSS Sprint: 2 – 8 Teams

- One Product-level sprint
- Whole Product focus
- Shared Definition of Done (DoD) by all teams
- Teams working together/communicating by: “just talk” or “just scream”

LeSS - is NOT many teams, doing their own Scrum, sprinting ad-hoc, working from separate team backlogs, servicing different Product Owners, dealing lots of dependencies and coordination.

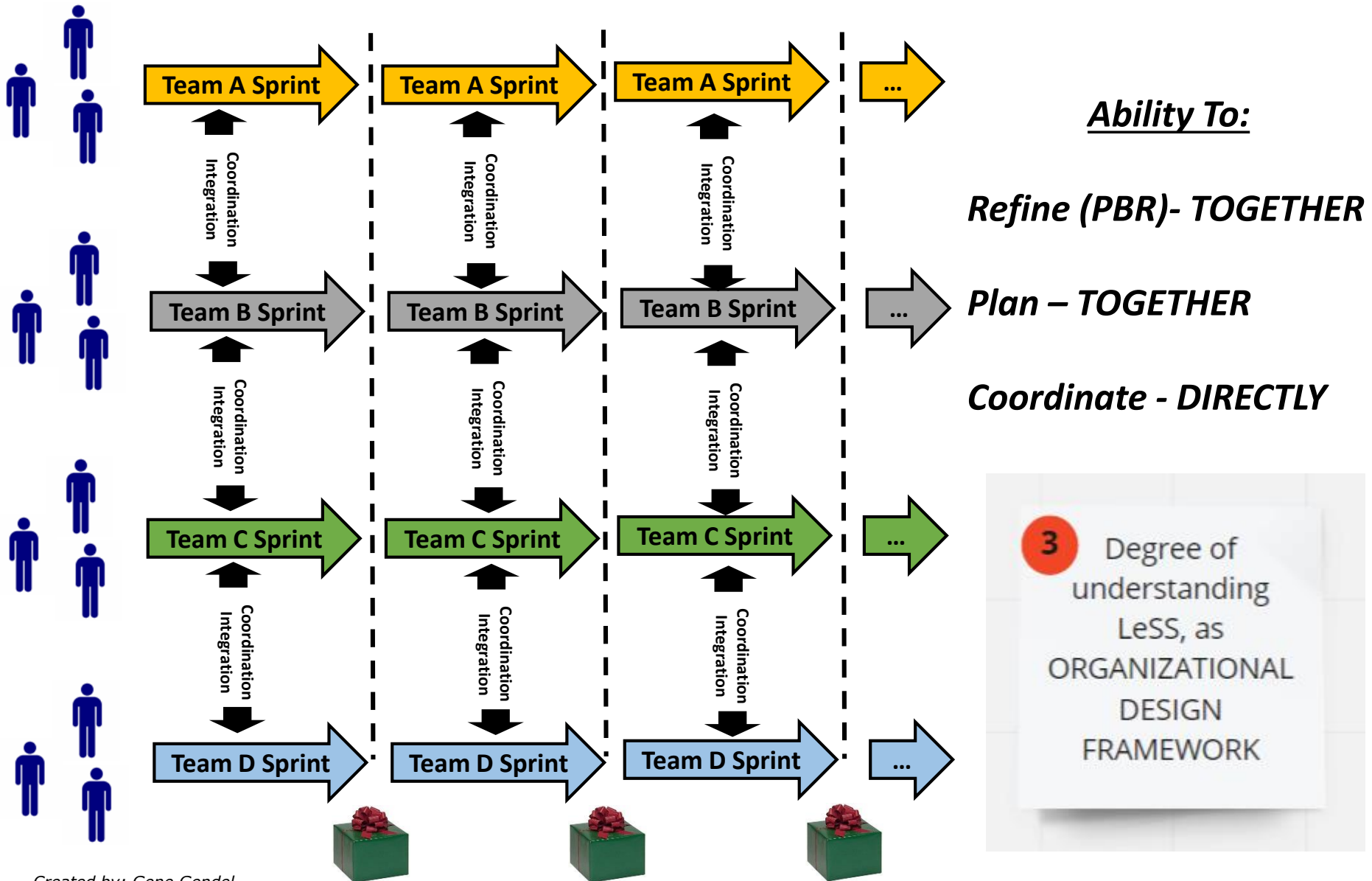
LeSS - IS Scrum, done by multiple (2-8) teams, sprinting together, working from the same backlog, servicing the same Product Owner, having no requirements dependencies and a plenty of direct coordination.



By Gene Gendel. 2020

Sourced from: <https://less.works/trainer-graphical-representations/gene-gendel/less-adoptions.jpg>

One LeSS Sprint: 2 – 8 Teams

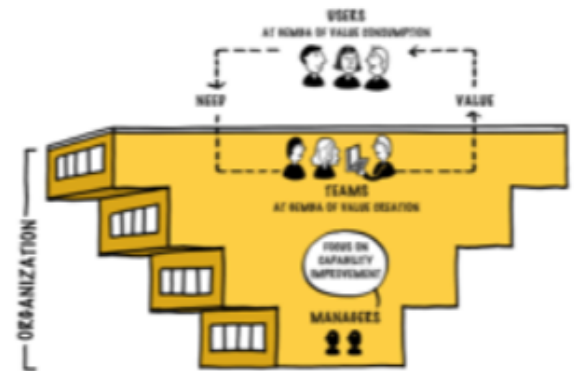


Created by: Gene Gendel

Three LeSS Principles

➤ Deep and narrow over broad and shallow

➤ Top-down and bottom-up

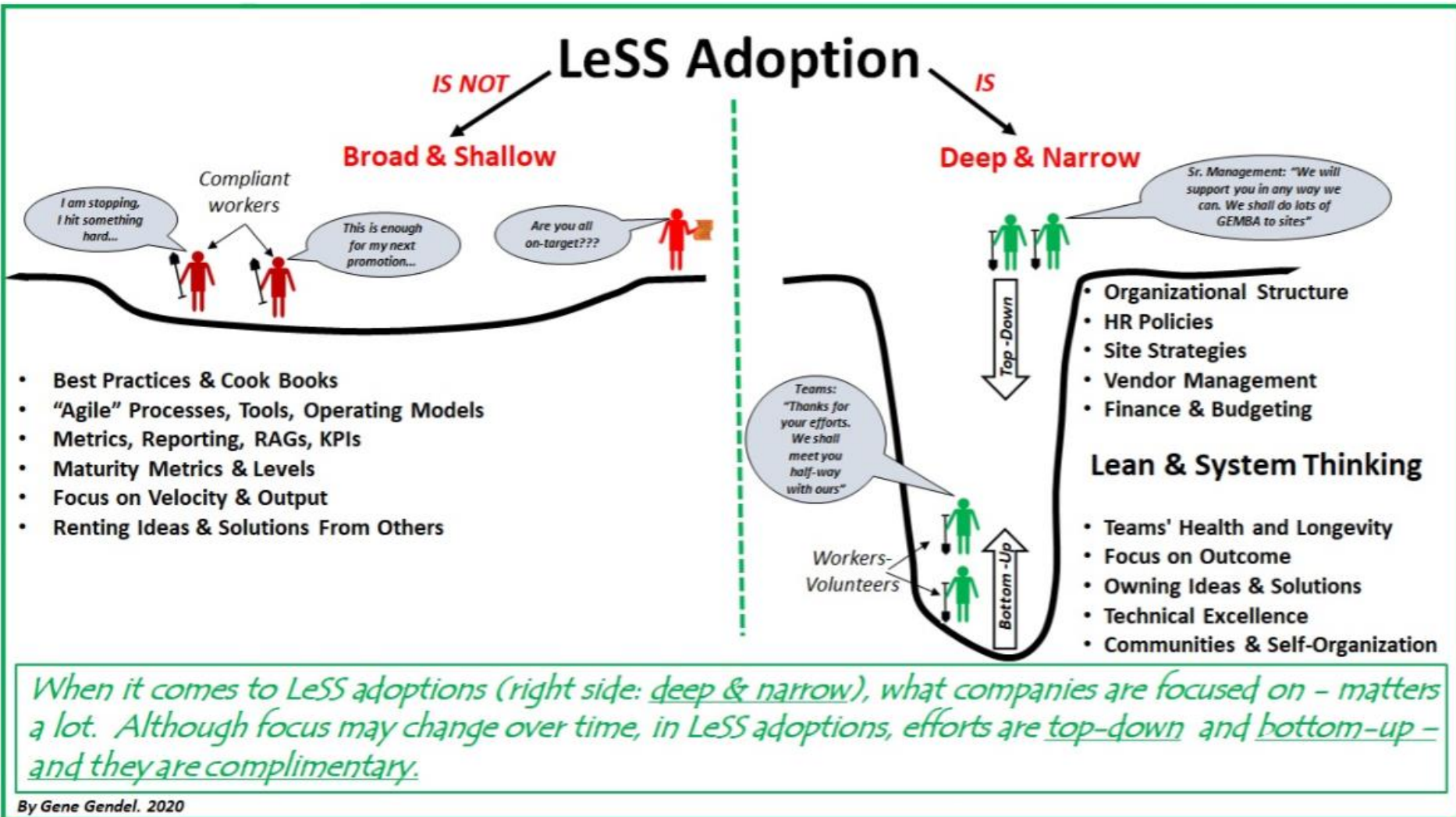


➤ Use volunteering



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

Three LeSS Principles



Sourced from: <https://less.works/trainer-graphical-representations/gene-gendel/one-sprint.jpg>

Organizational Design & Local Optimization

Local Optimization

Organizational STRUCTURE –

is the 1st Order Factor (Variable) that has impact on everything else inside ECOSYSTEM.

Everything else just follows: culture, mindset, behaviors, norms, values, processes, policies

This is what some less experienced agile “enthusiasts” do not stress, when they coach senior management 😞

Local Optimization

Larman's Laws of Organizational Behavior

([Spanish translation](#))([Russian translation](#)) ([Japanese translation](#))([Persian/Farsi translation](#))([Polish translation](#))([Portuguese translation](#))

After decades of observation and organizational consulting, here are *Larman's Laws of Organizational Behavior*. These are observations rather than laws to follow ;)

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 redefining or overloading 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”, “revolutionary”, “religion”, and “needing pragmatic customization for local concerns” — which deflects from addressing weaknesses and manager/specialist status quo.
4. As a corollary to (1), if after *changing the change* some managers and single-specialists are still displaced, they become “coaches/trainers” for the change, frequently reinforcing (2) and (3).
5. (in large established orgs) *Culture follows structure*. And in tiny young orgs, structure follows culture.

Elaboration:

Sourced from: https://www.craiglarman.com/wiki/index.php?title=Larman%27s_Laws_of_Organizational_Behavior

Local Optimization

(“This Is Not My Job”)



Sourced from: <http://www.tsmplug.com/games/pes-2014-download/>

Local Optimization

...whereas, **Global optimization** refers to finding the optimal value of a given function among all possible solution...

... **Local optimization** finds the optimal value within the neighboring set of candidate solution...

<https://www.igi-global.com/dictionary/from-optimization-to-clustering/45858>

[Pentagon Wars – Bradley Fighting Vehicle Evolution](#)



Local Optimization

"Everyone is busy and working so hard. Yet, the system is delivering slow and Users are not happy"

How could that be?



Sourced from: <https://www.youtube.com/watch?v=5unMIXg6WL4>

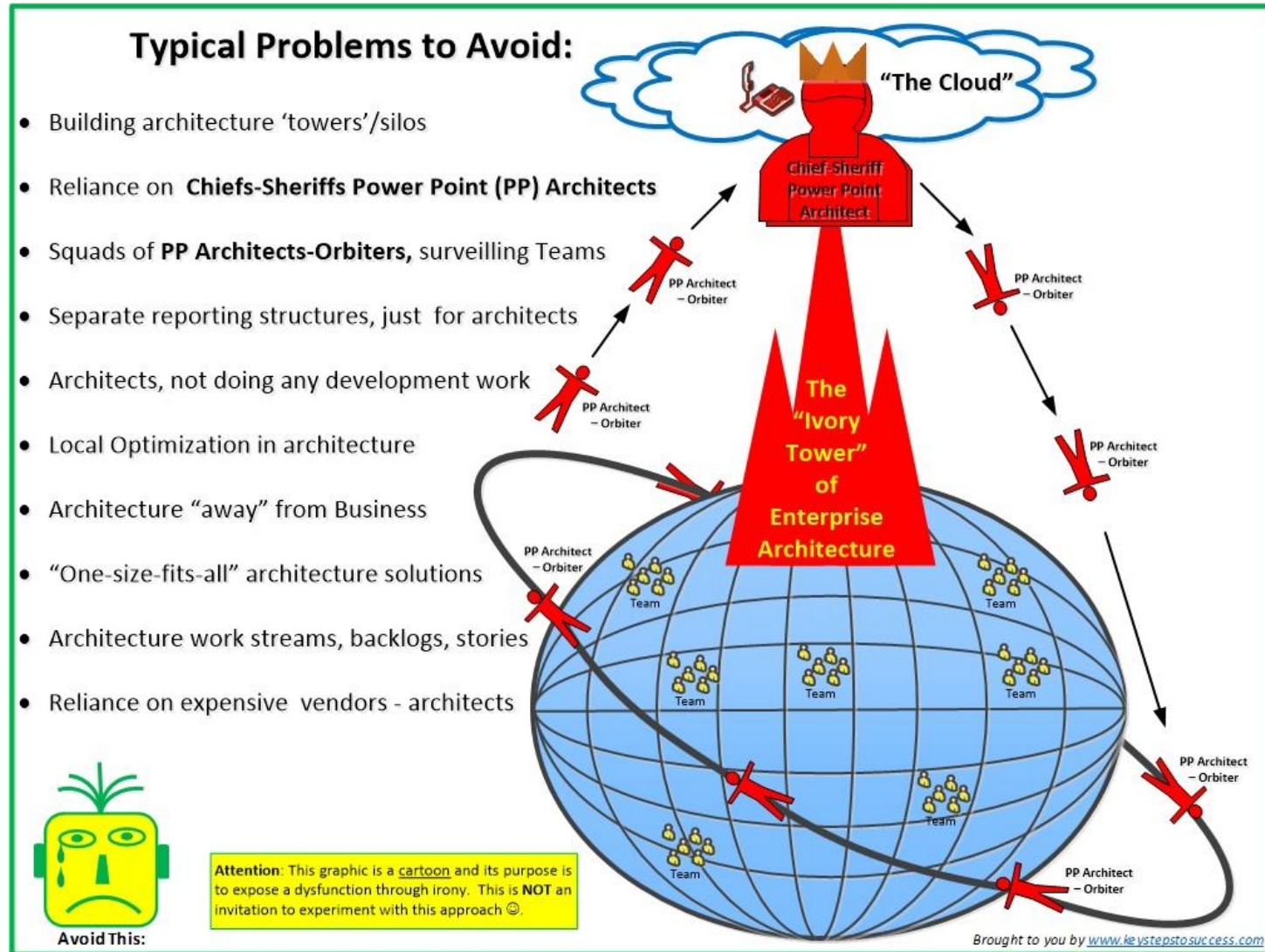
Local Optimization

Frequently Heard Justifications:

- Efficient
- Productive
- Best
- Good
- Optimized
- Ideal
- Cost-Effective



Local Optimization In Roles: “Agile Architect”



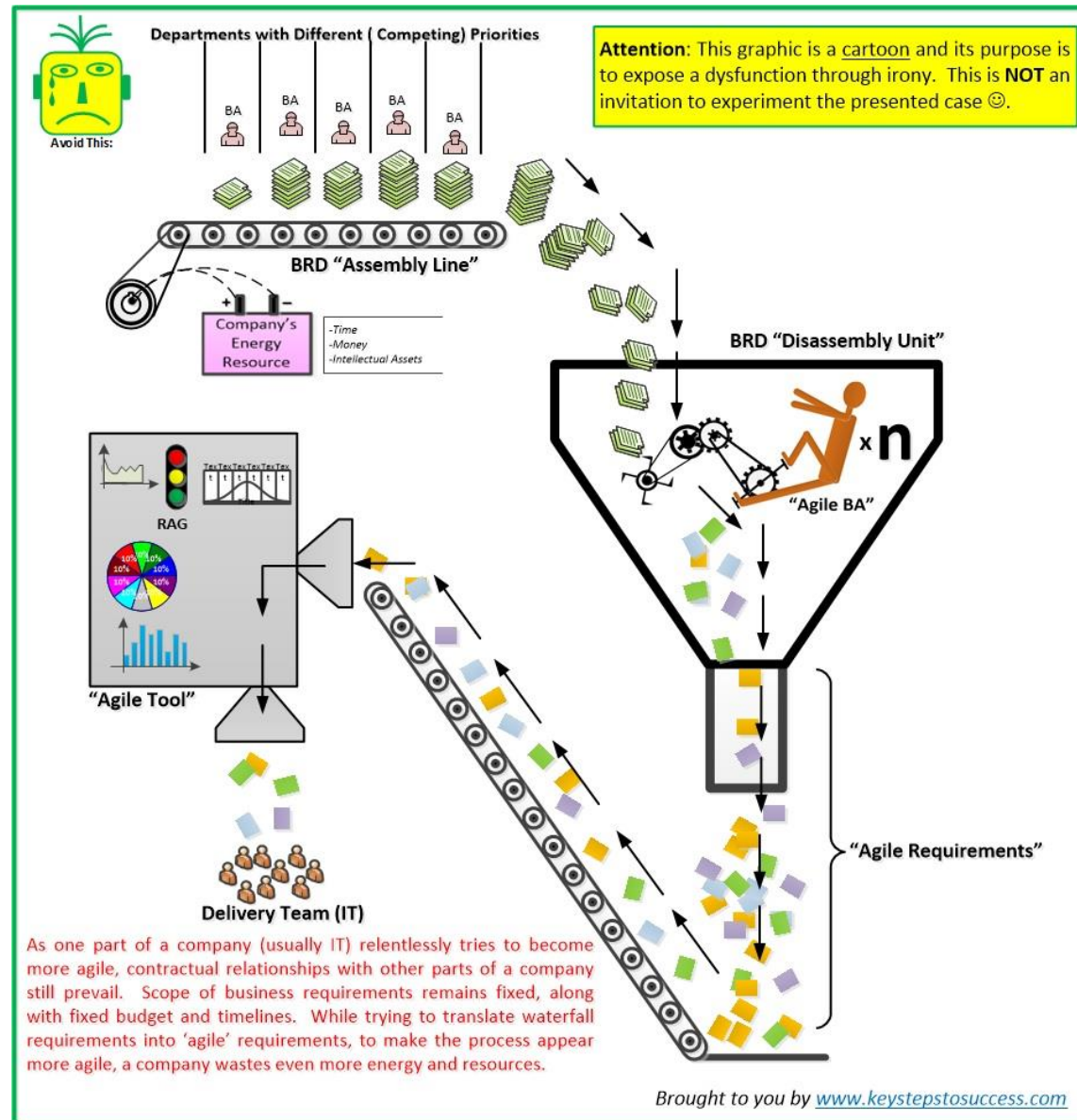
Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

Local Optimization In UI/UX Design



Sourced from: <https://www.meetup.com/Design-Sprint-NYC/>

Local Optimization In Roles: “Agile BA”



Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

Seeing (Hearing) Local Optimization in...

Team Structures

Org. Structures

Documentation

Definition of Done

Backlogs

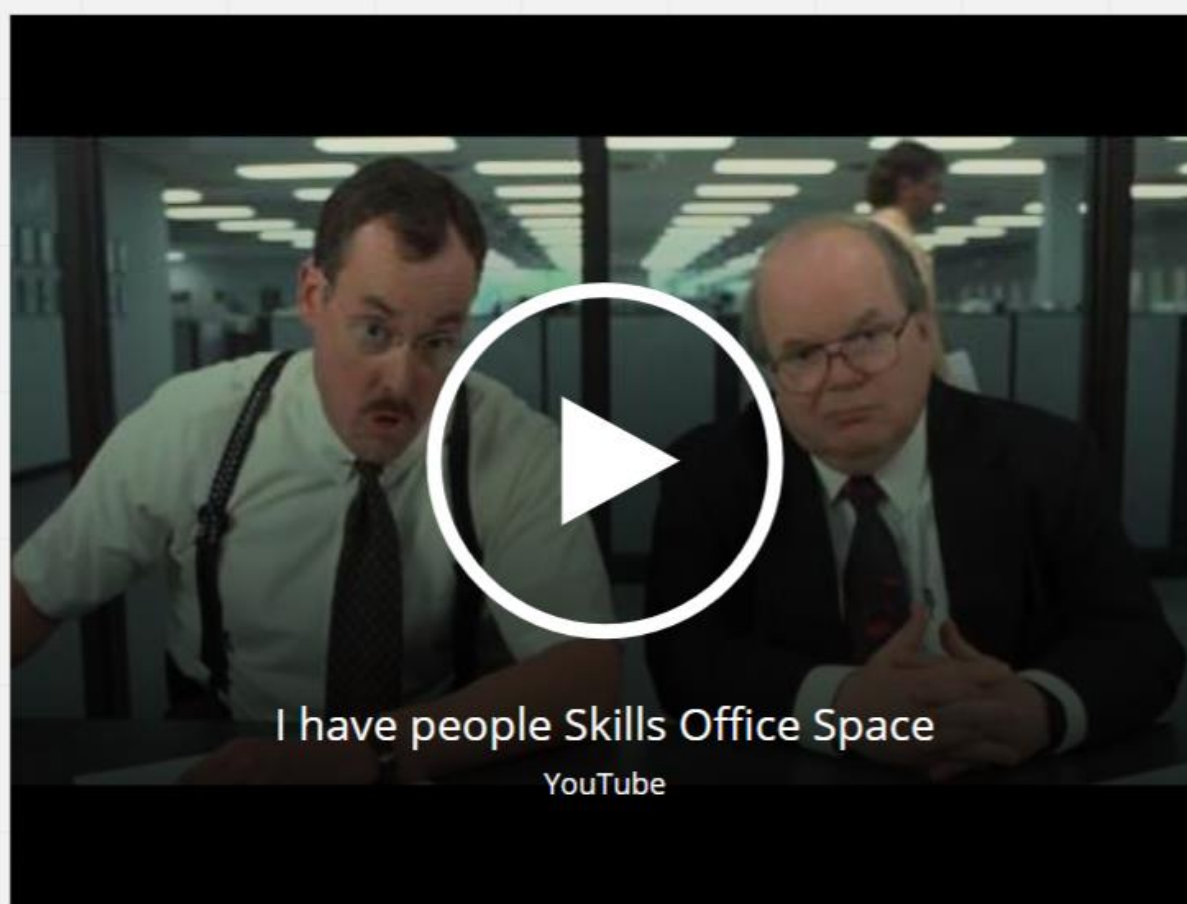
Role Definitions

Product Design

Goals & Metrics

Also at: [https://less.works/less/principles/systems-thinking.html#Seeing\(andHearing\)LocalOptimization](https://less.works/less/principles/systems-thinking.html#Seeing(andHearing)LocalOptimization)

Local Optimization



Sourced from: <https://www.youtube.com/watch?v=hNuu9CpdjIo>

Exercise

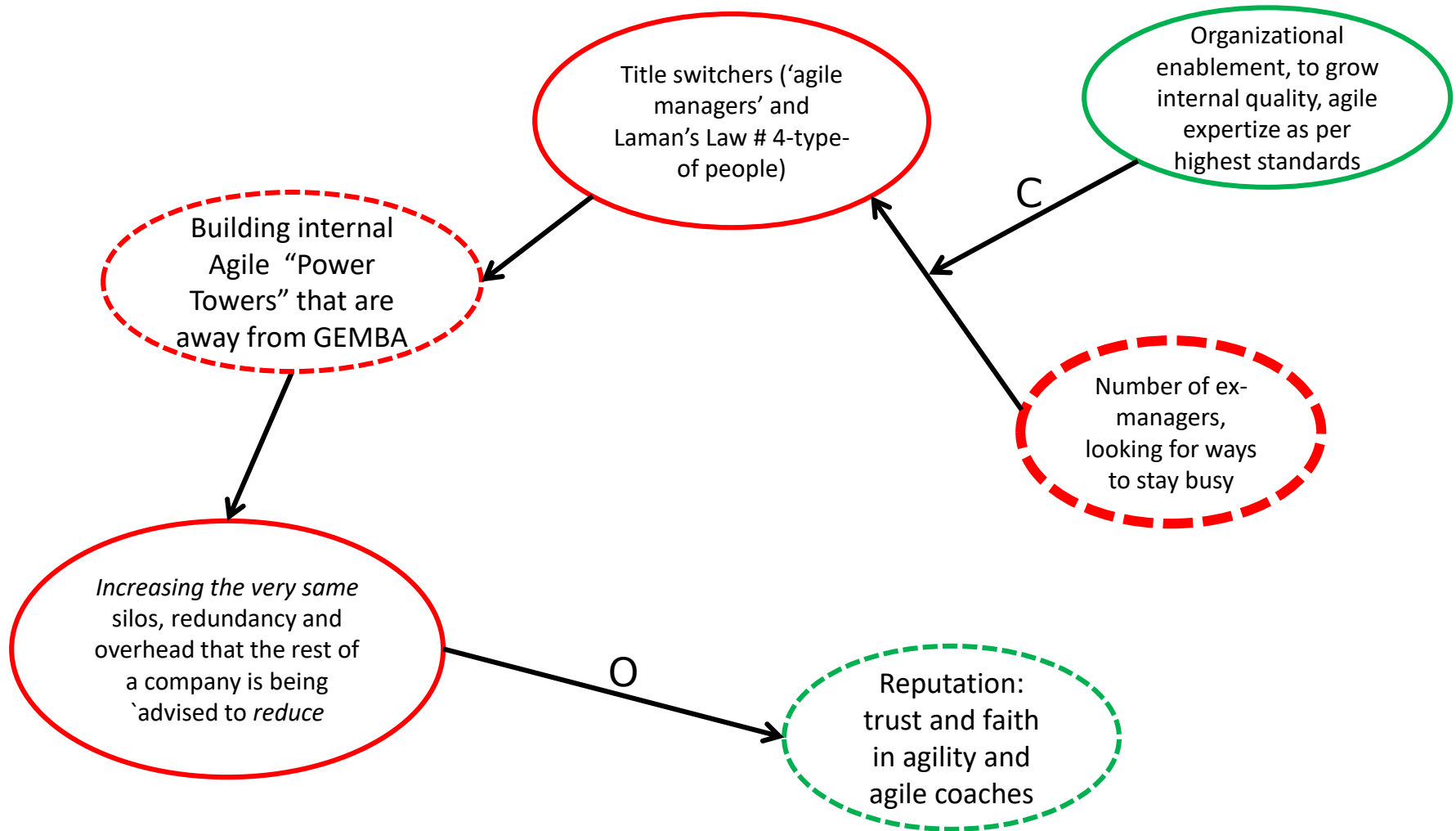
Duration: 10 min

Class: in-groups, brainstorm some of the most common examples of ***Local Optimization in Agile Leadership*** in your respective organizations. Work with provided system variables to create a model.

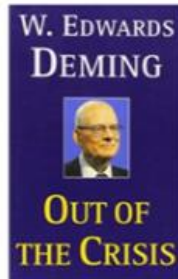
Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in Agile Leadership - Exercise

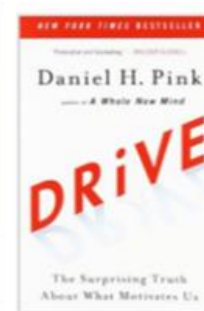
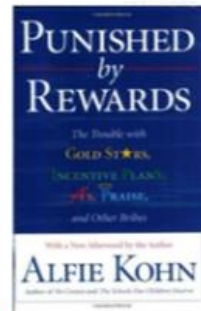
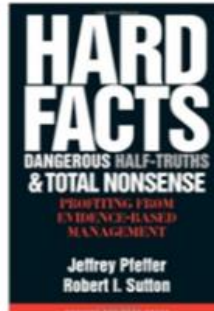
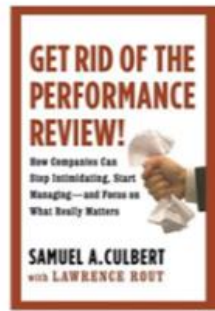
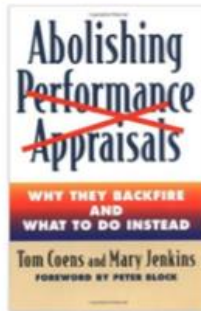


Local Optimization - HR



Gap Between Science and Business

"The idea of a merit rating is alluring. The sound of the words captivates the imagination: pay for what you get; get what you pay for; motivate people to do their best, for their own good. The effect is exactly the opposite of what the words promise."



Overall tolerance
[degree of] to
Local
Optimization
(entire system)

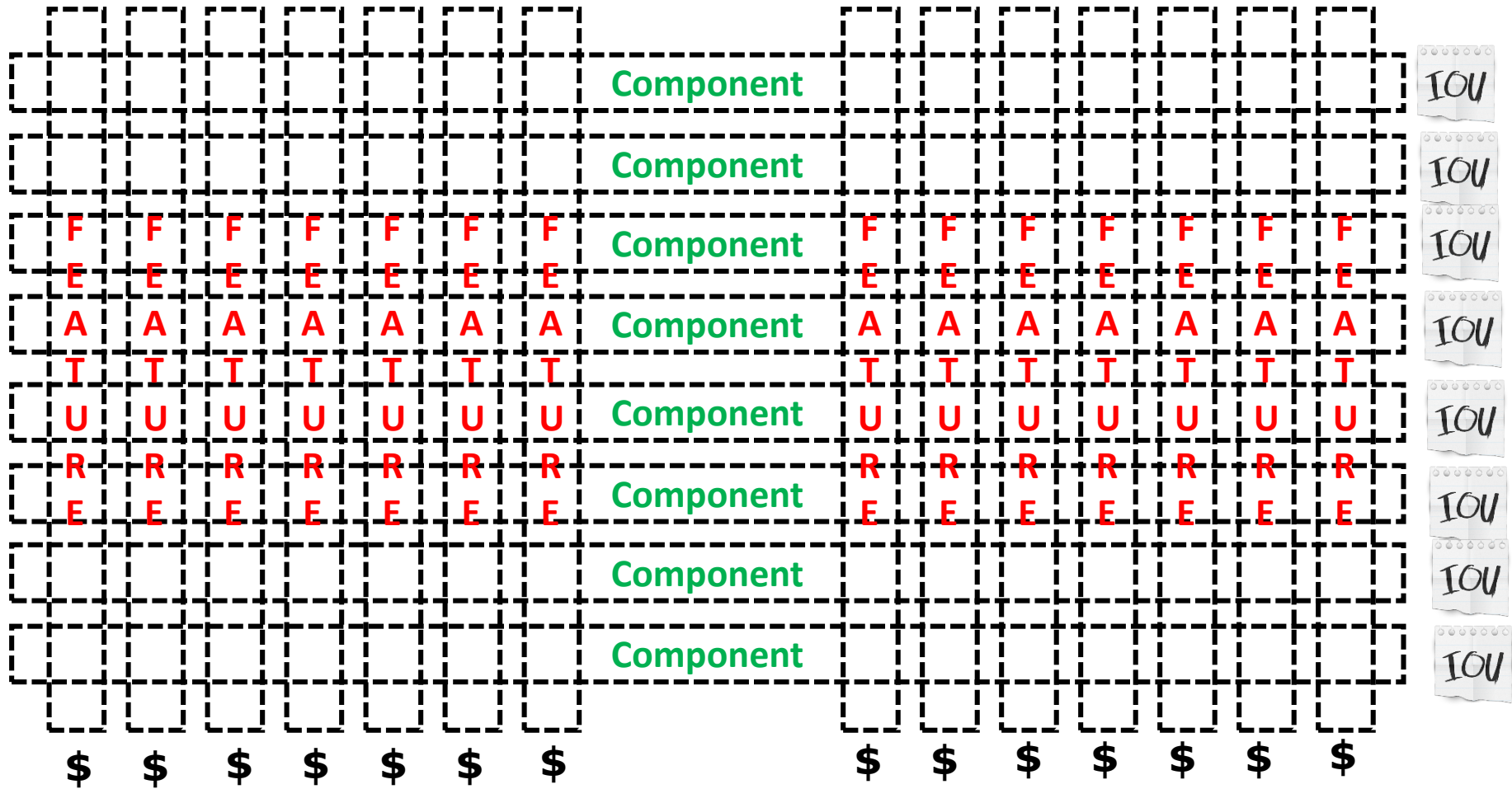
7

HR
involvement
& support

8

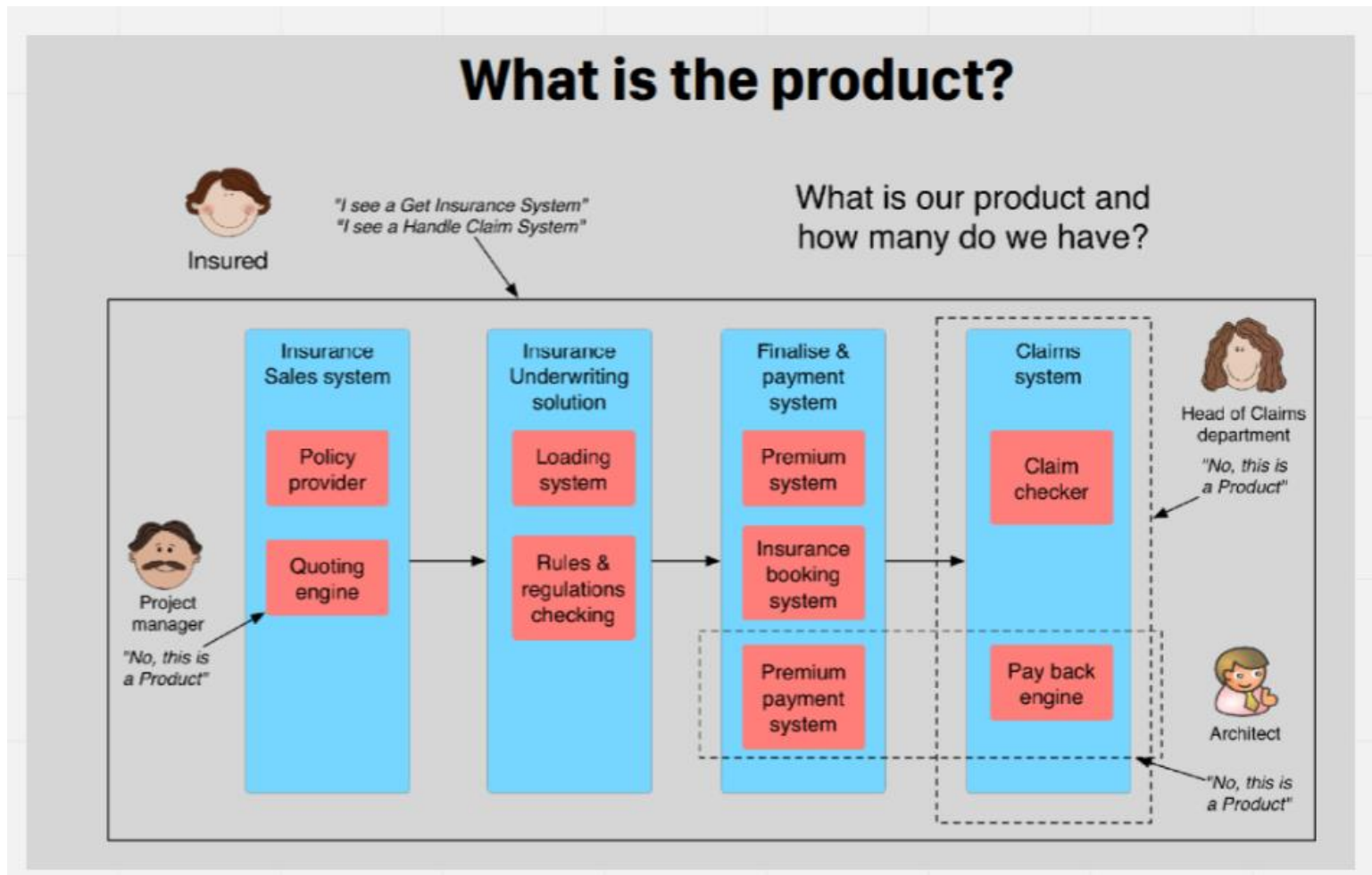
LeSS Product vs Project/Program/Portfolio

Which Product “Dimension” Holds Real Value?



Created by: Gene Gendel

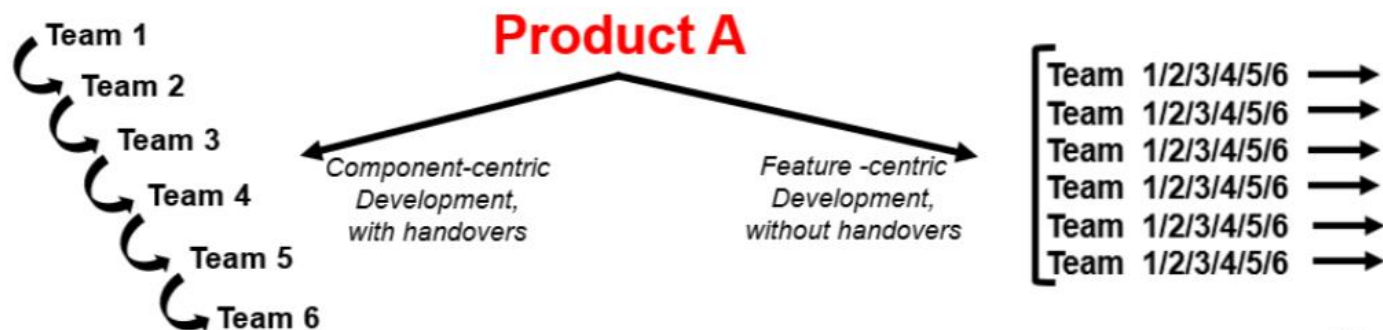
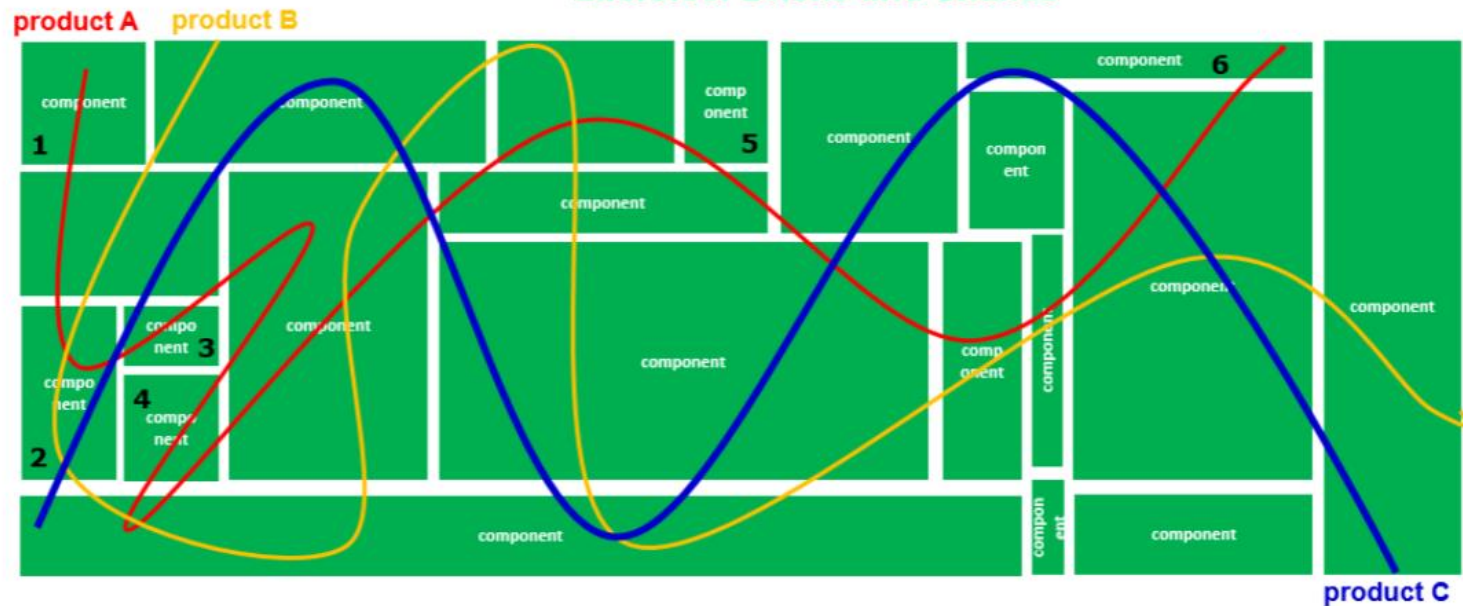
= Which Product “Dimension” Holds Real Value?



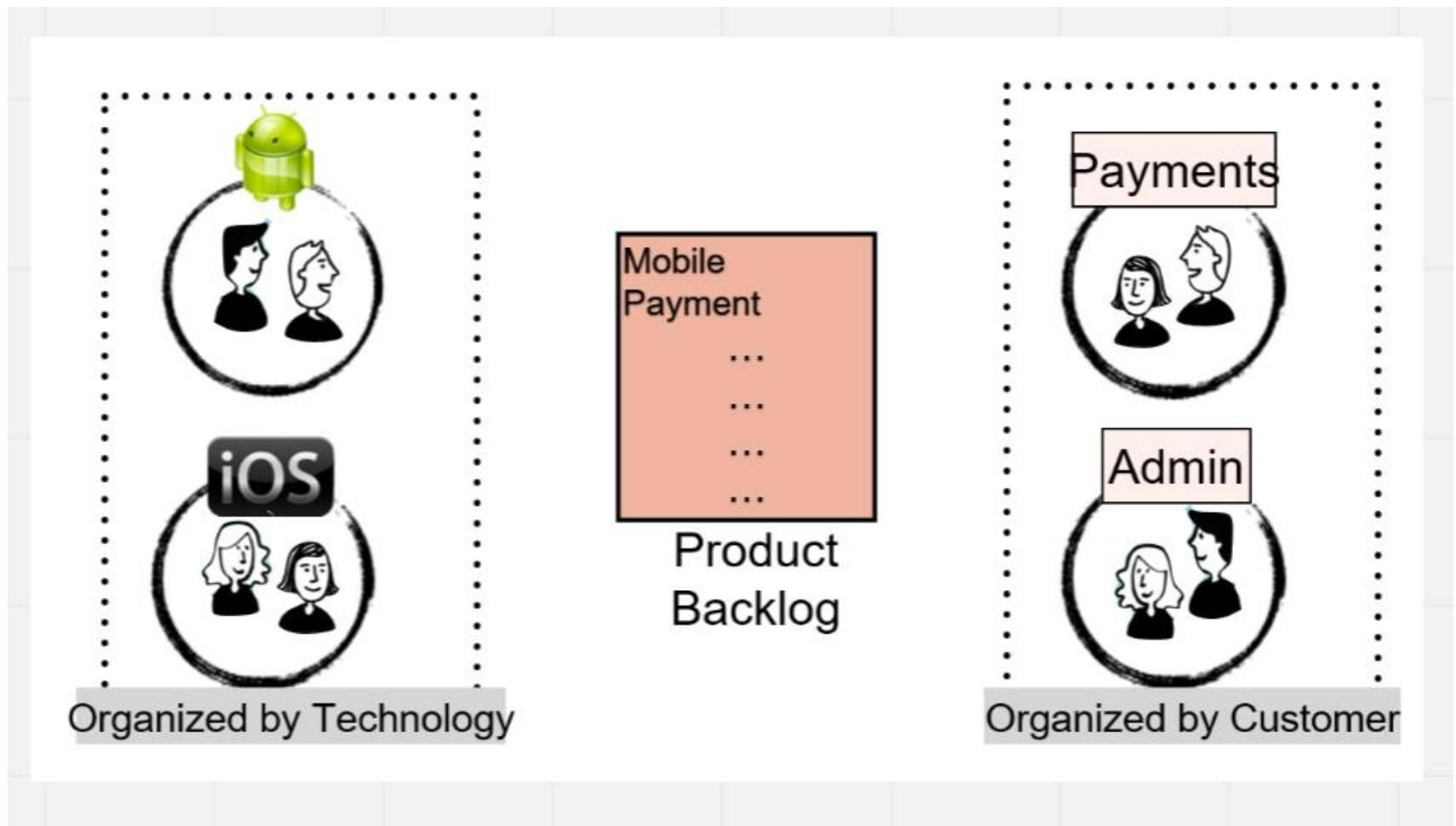
Sourced from creative commons repository of Certified LeSS Trainers: <https://less.works/courses/become-less-trainer>

How Do you Define Your Product?

Exercise: Bricks and Snakes



How Do you Define Your Product?



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



Exercise

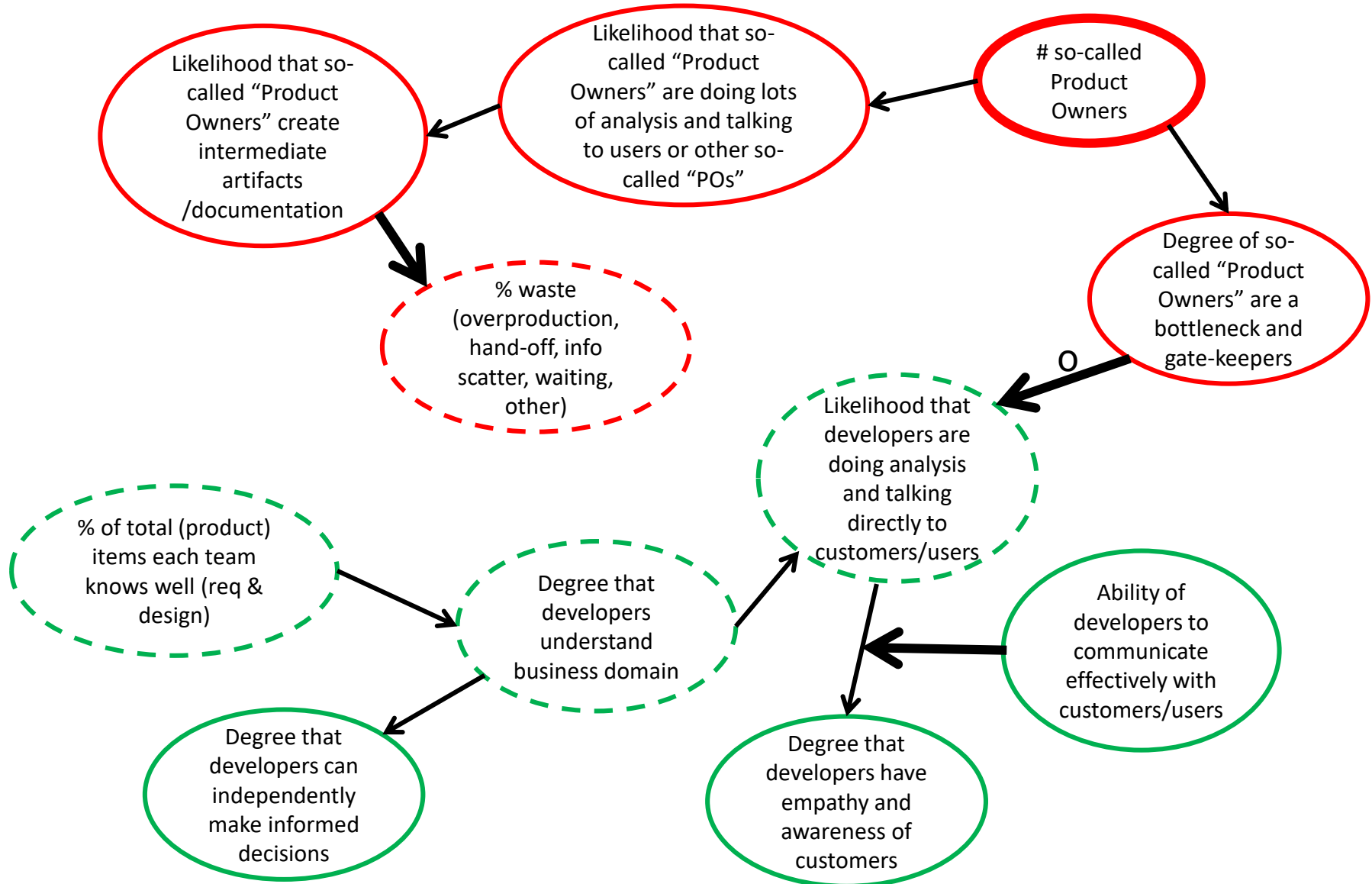
Duration: 10 min

Class: in-groups, brainstorm some of the most common examples of ***Local Optimization in Analysis and Design*** in your respective organizations. Work with provided system variables to create a model.

Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in **Analysis & Design**- Exercise

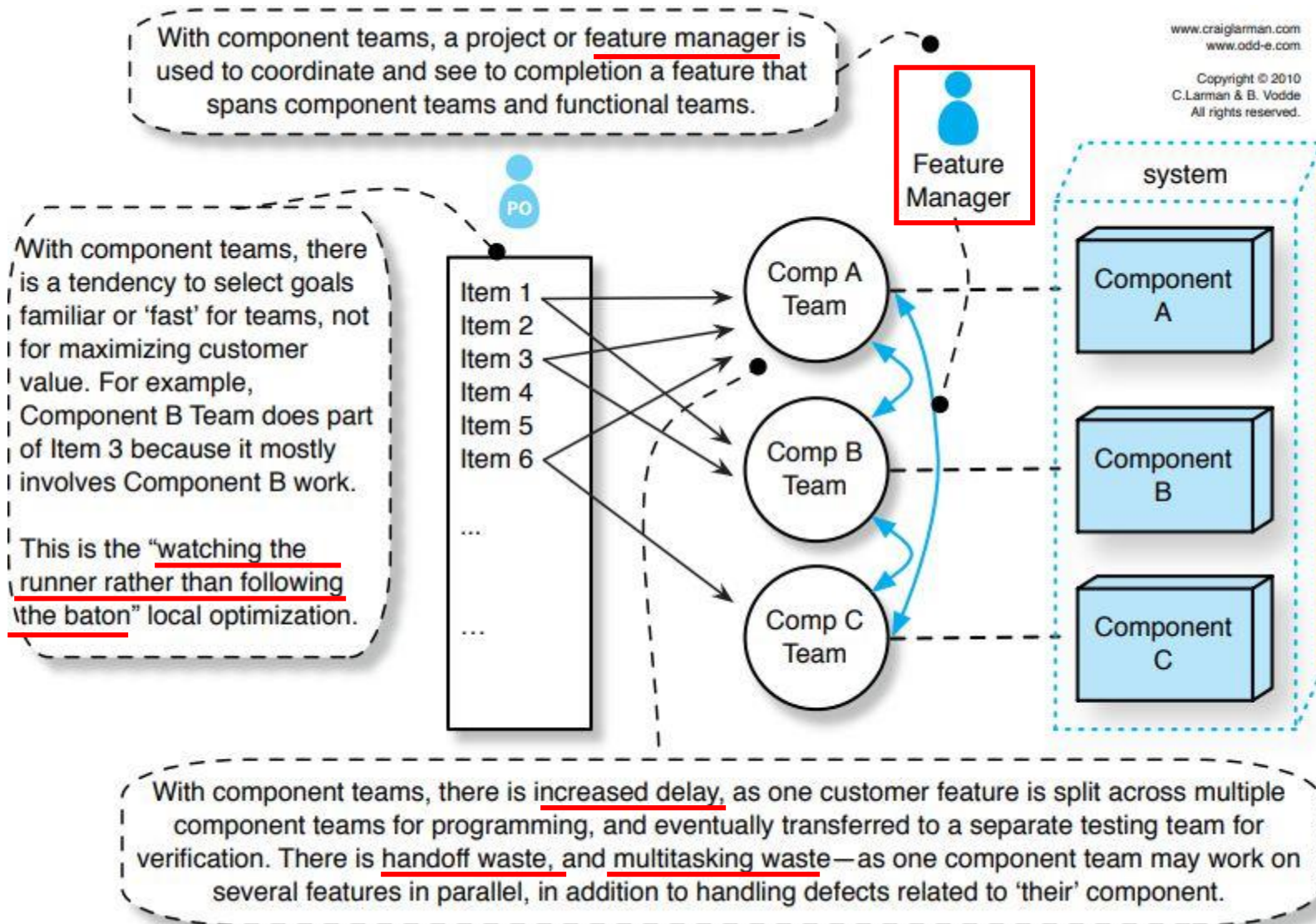


Feature Teams vs. Component Teams

Component Team

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

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

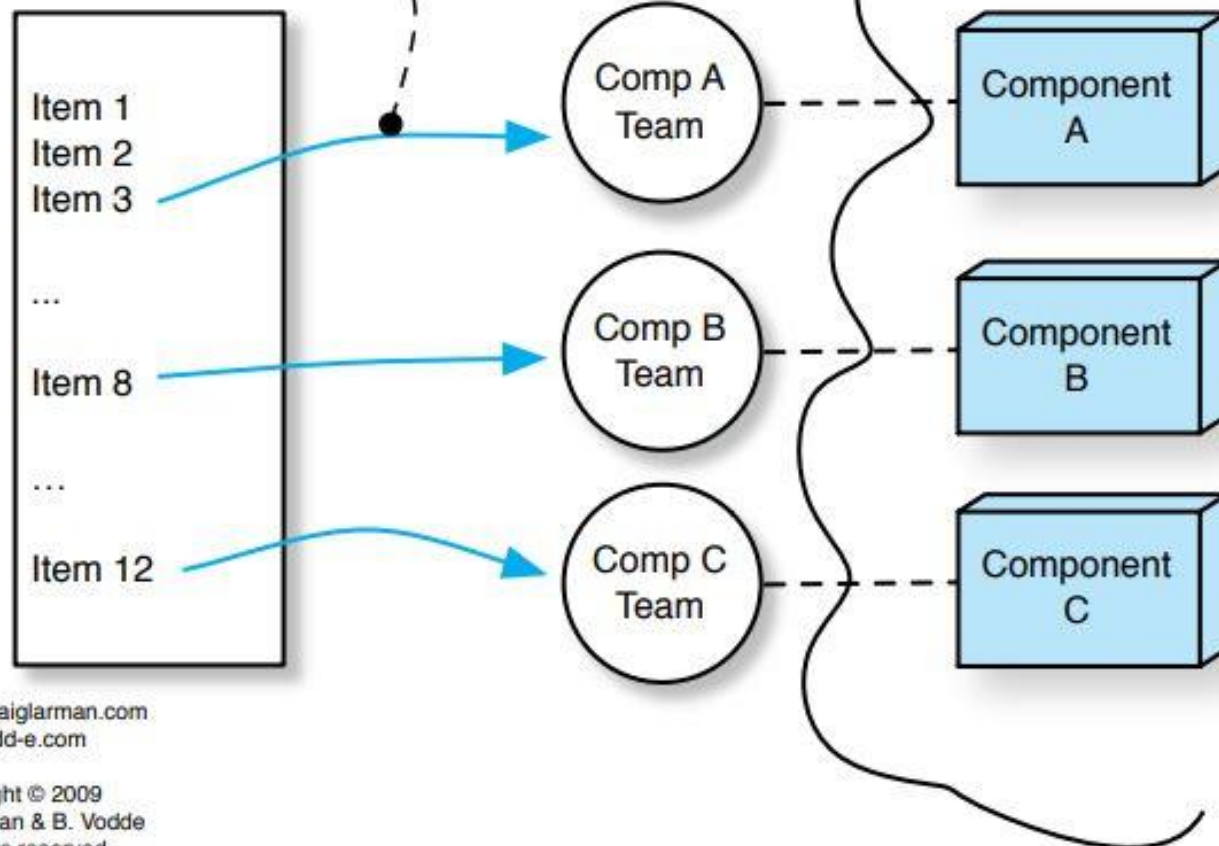


Sourced from: <https://less.works/resources/graphics/index.html>

Component Team

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

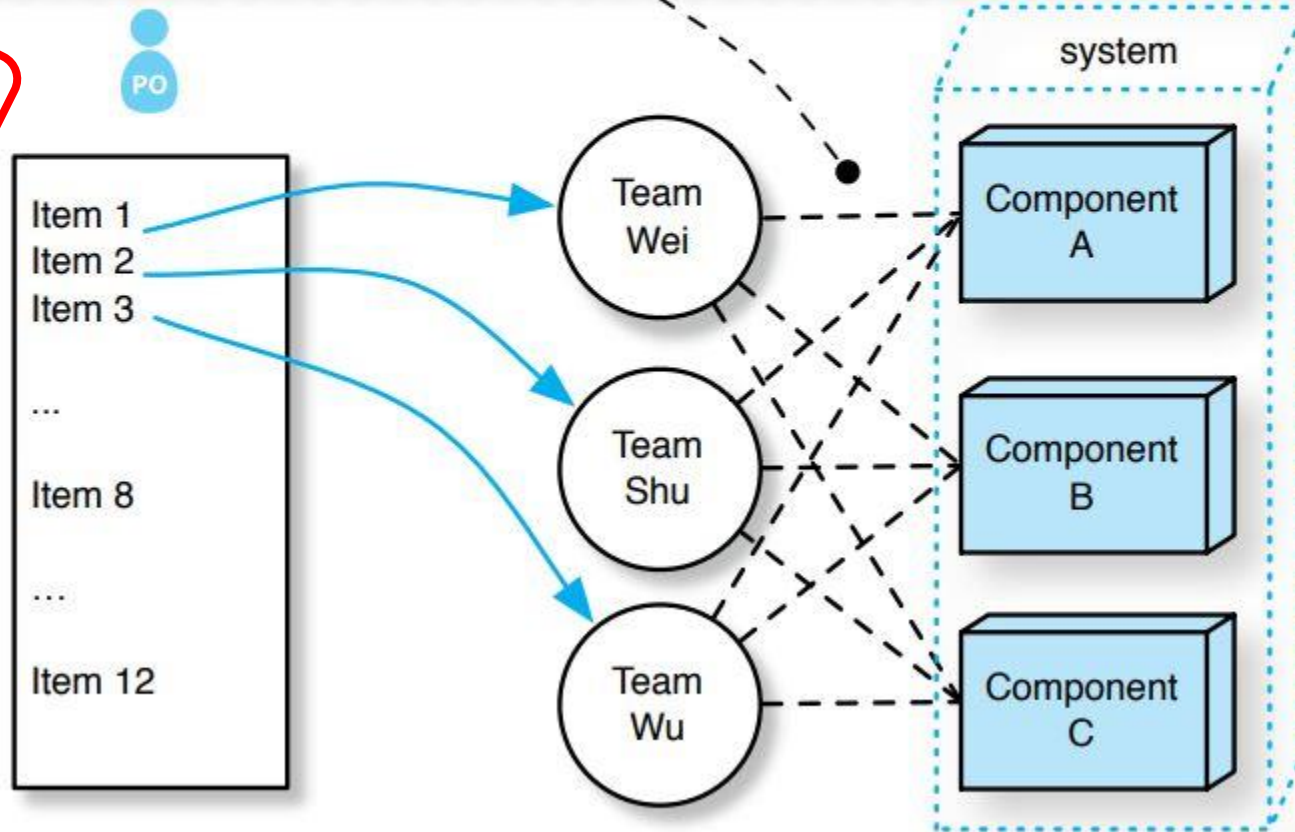
Copyright © 2009
C. Larman & B. Vodde
All rights reserved.

Sourced from: <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

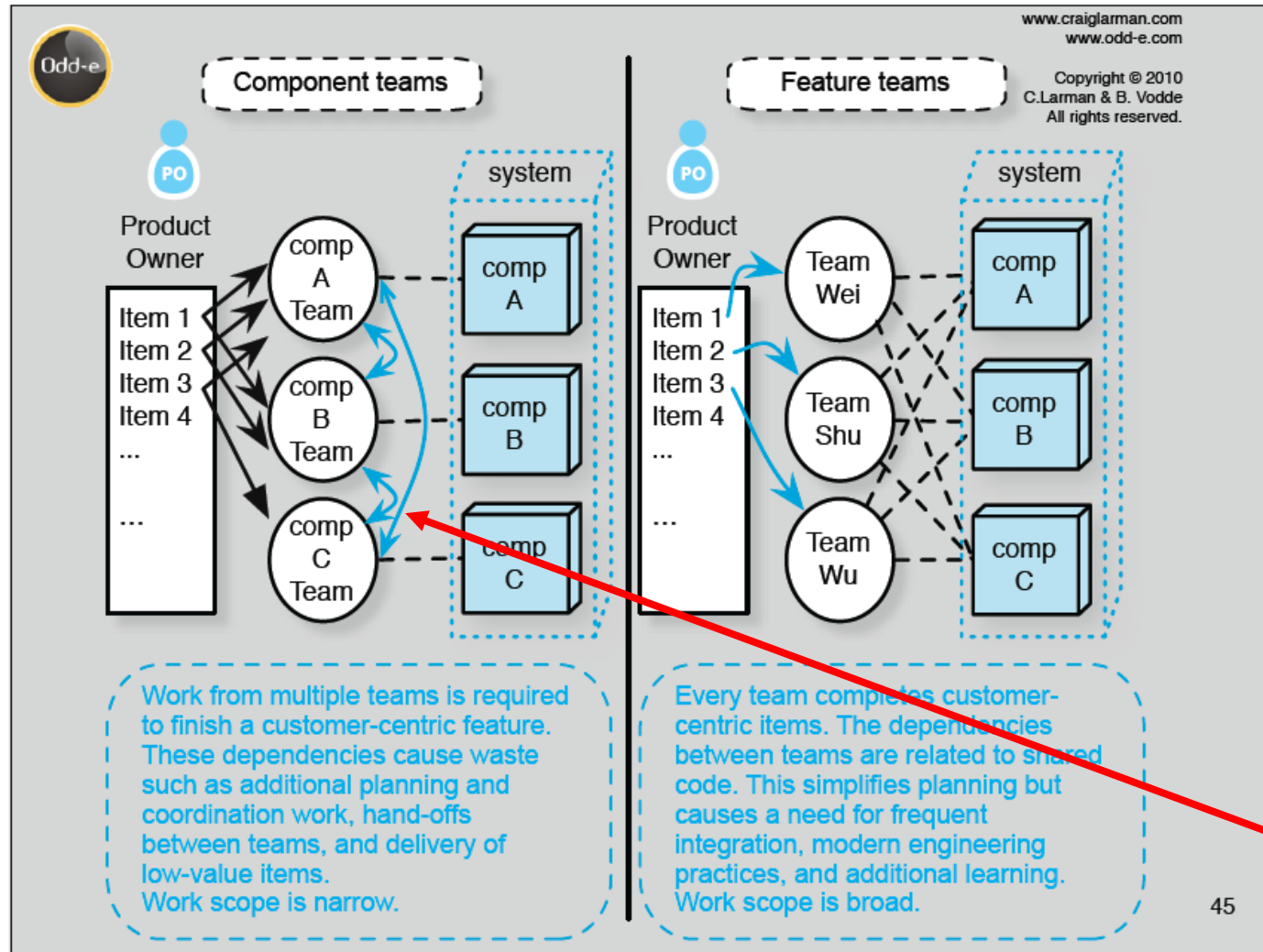


www.craiglarman.com
www.odd-e.com

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

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

Component & Feature Teams: Side by Side



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

No Team 'Tsars'

Component Teams

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



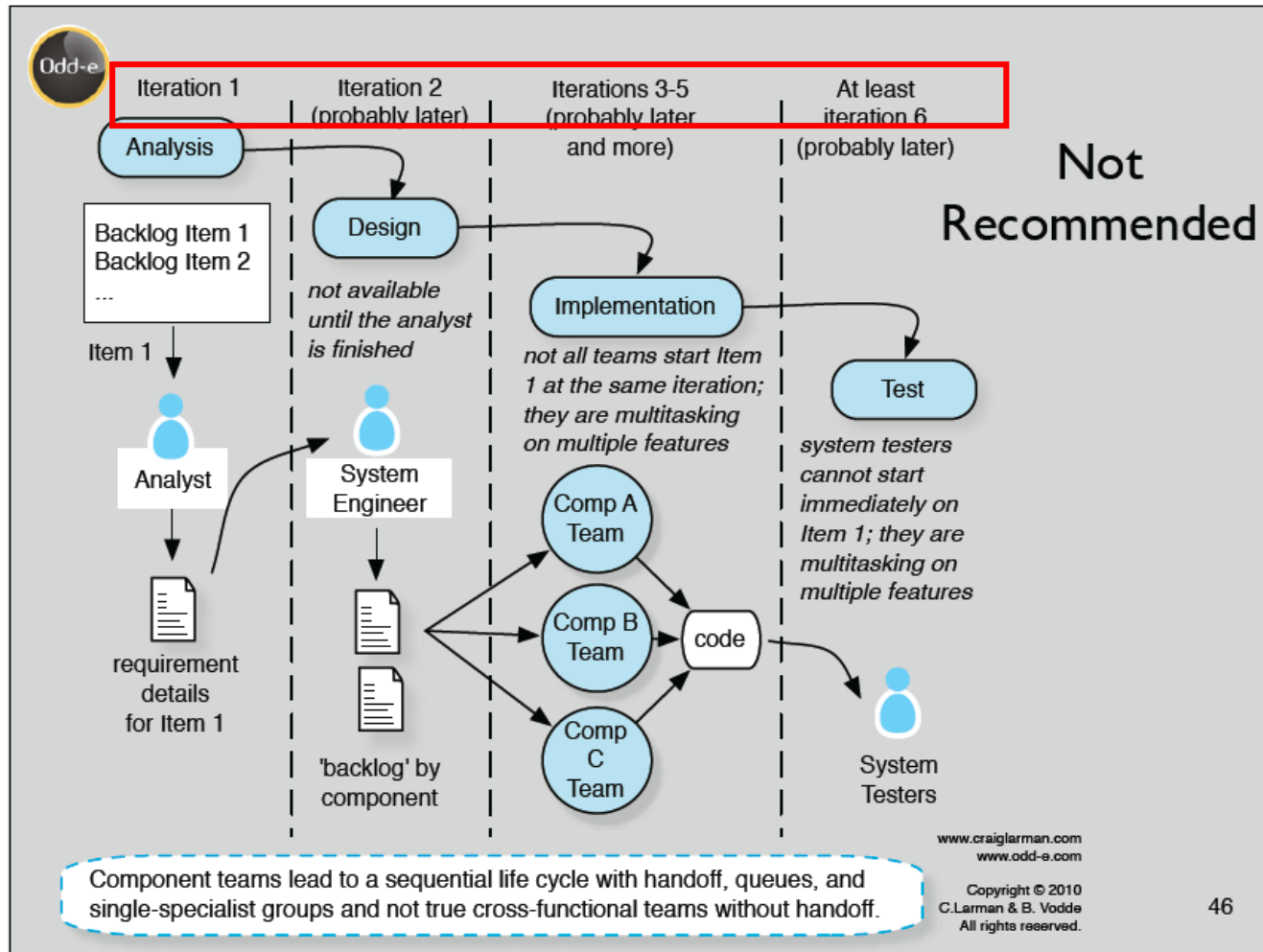
DBA

6 people



**Classic Example of
Local Optimization**

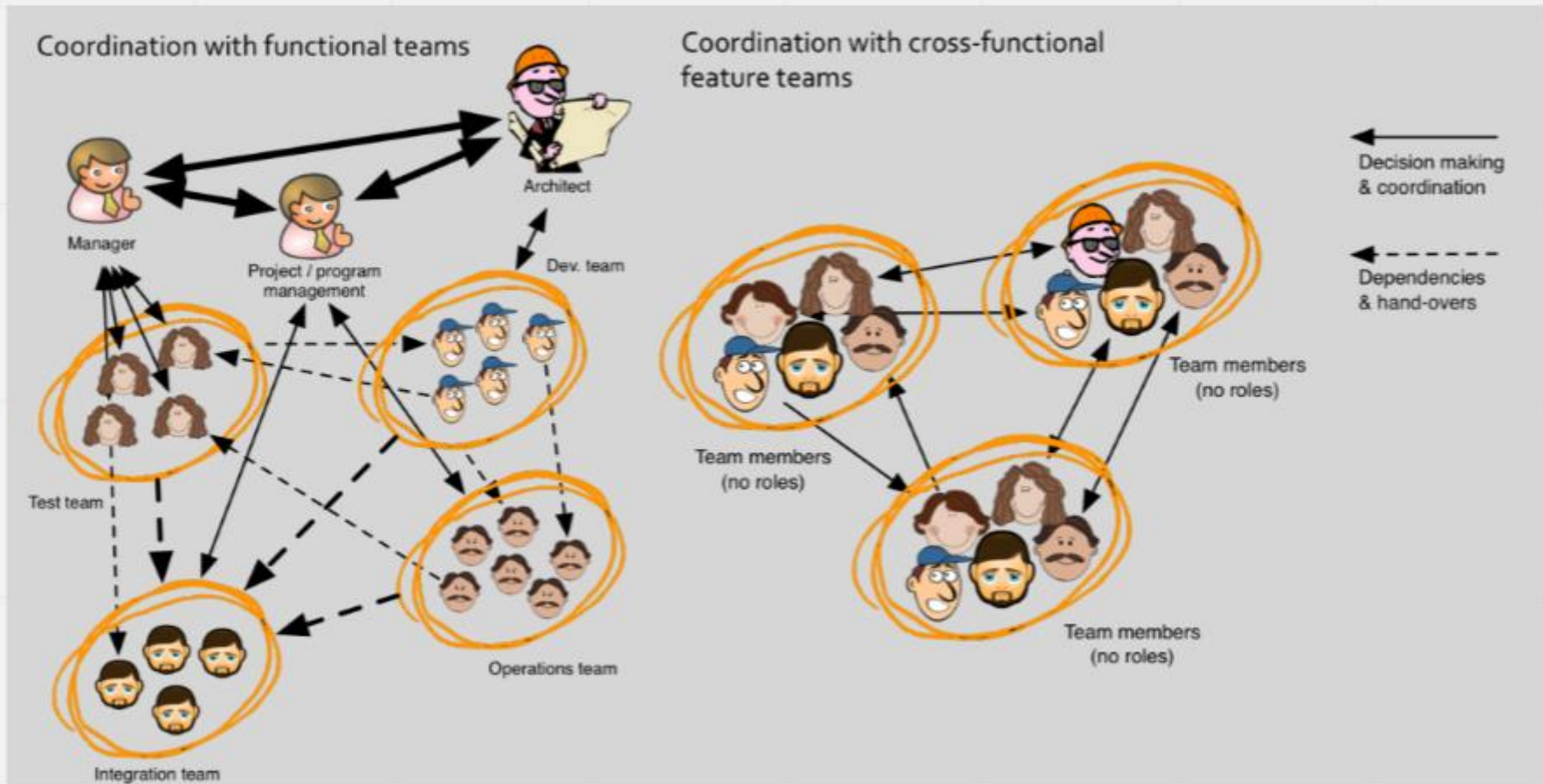
Beware of Mini-Waterfall in Scrum Clothes



- Fake Scrum
- “Wagile”
- Mini-waterfall

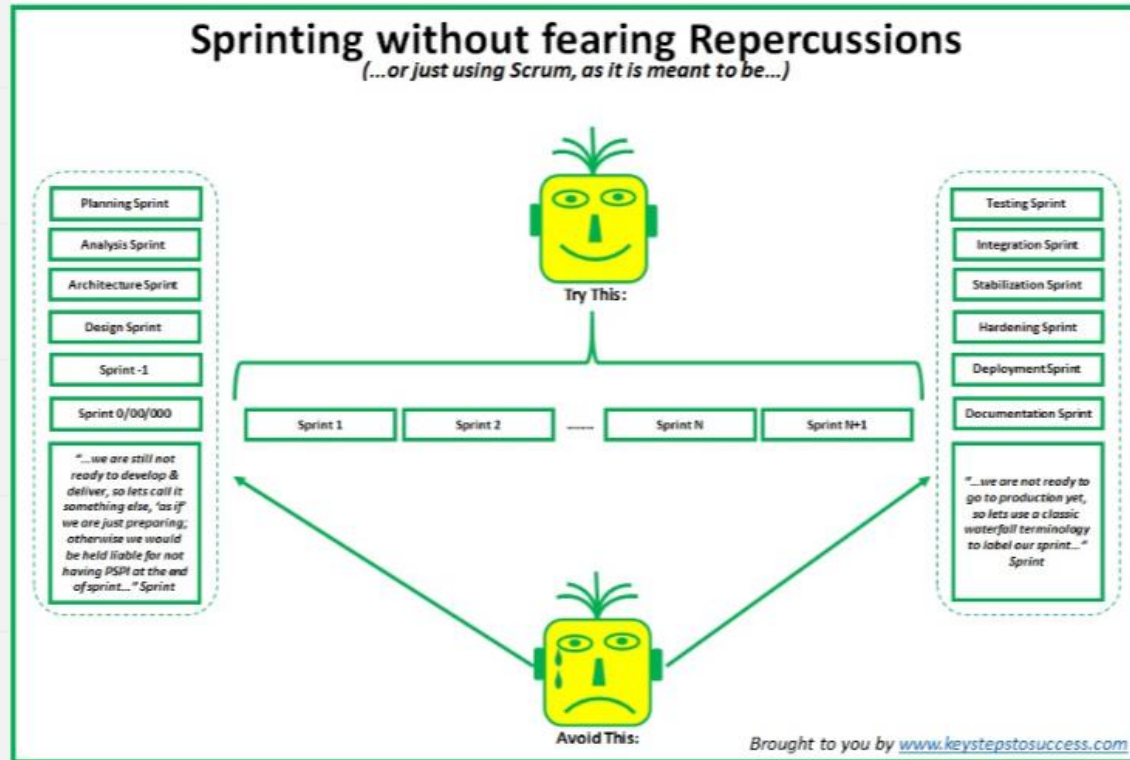
Sourced from creative commons repository of Certified LeSS Trainers: <https://less.works/courses/become-less-trainer>

Coordination



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

Coordination



Degree of
Local
Optimization
in teams'
structure

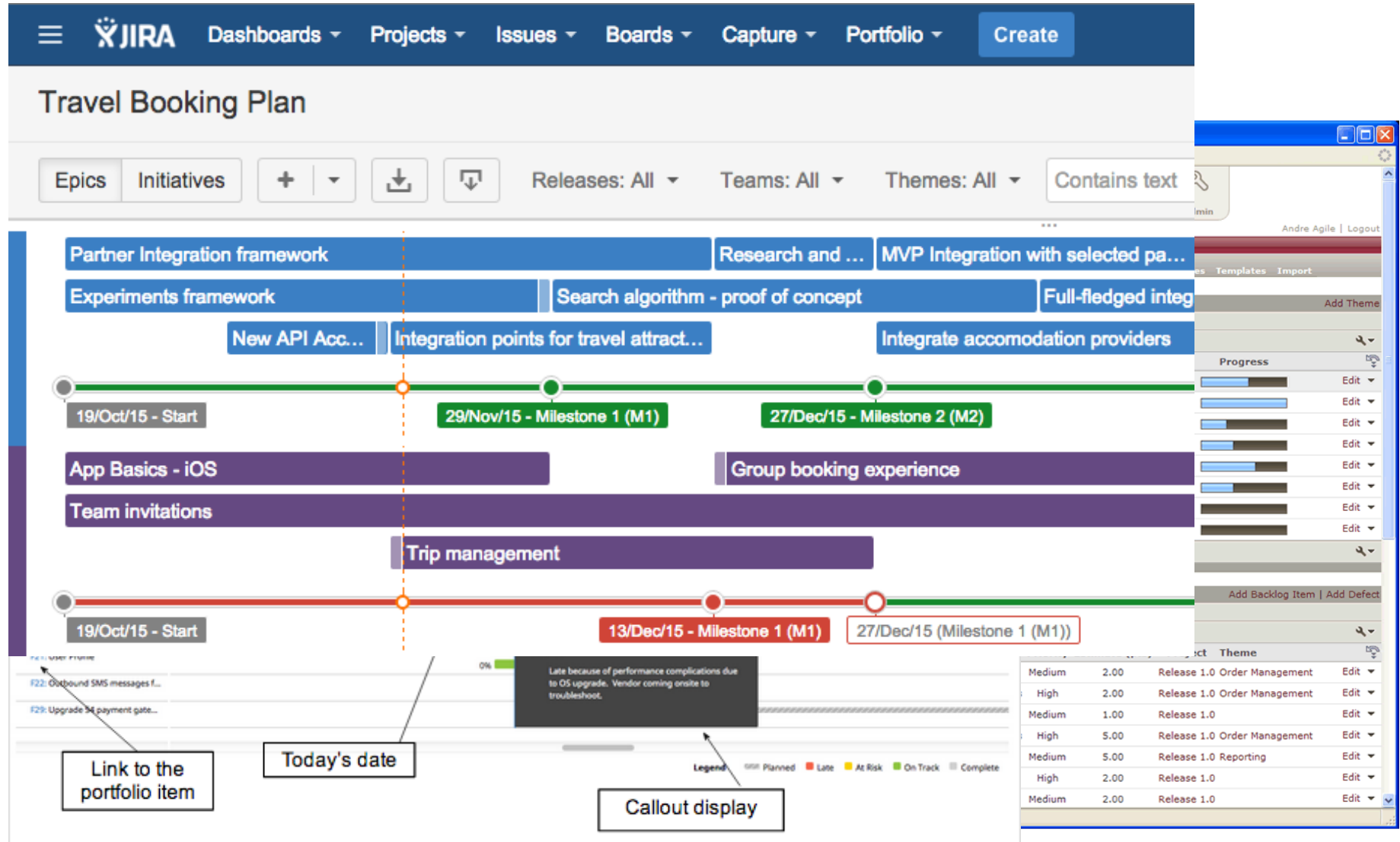
9

Degree of
Local
Optimization
in a backlog

10

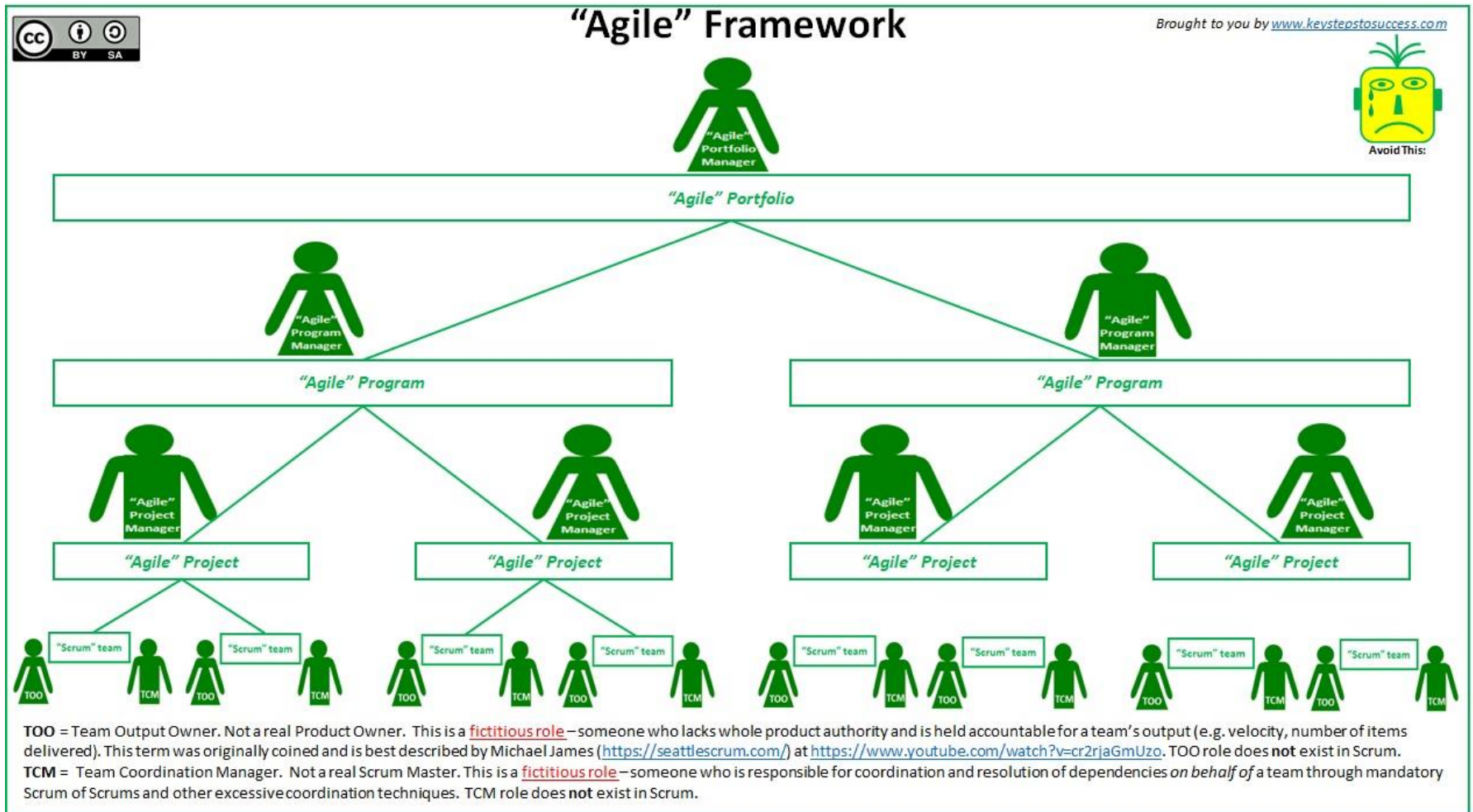
Fake Product Portfolio

Fake Product Portfolio



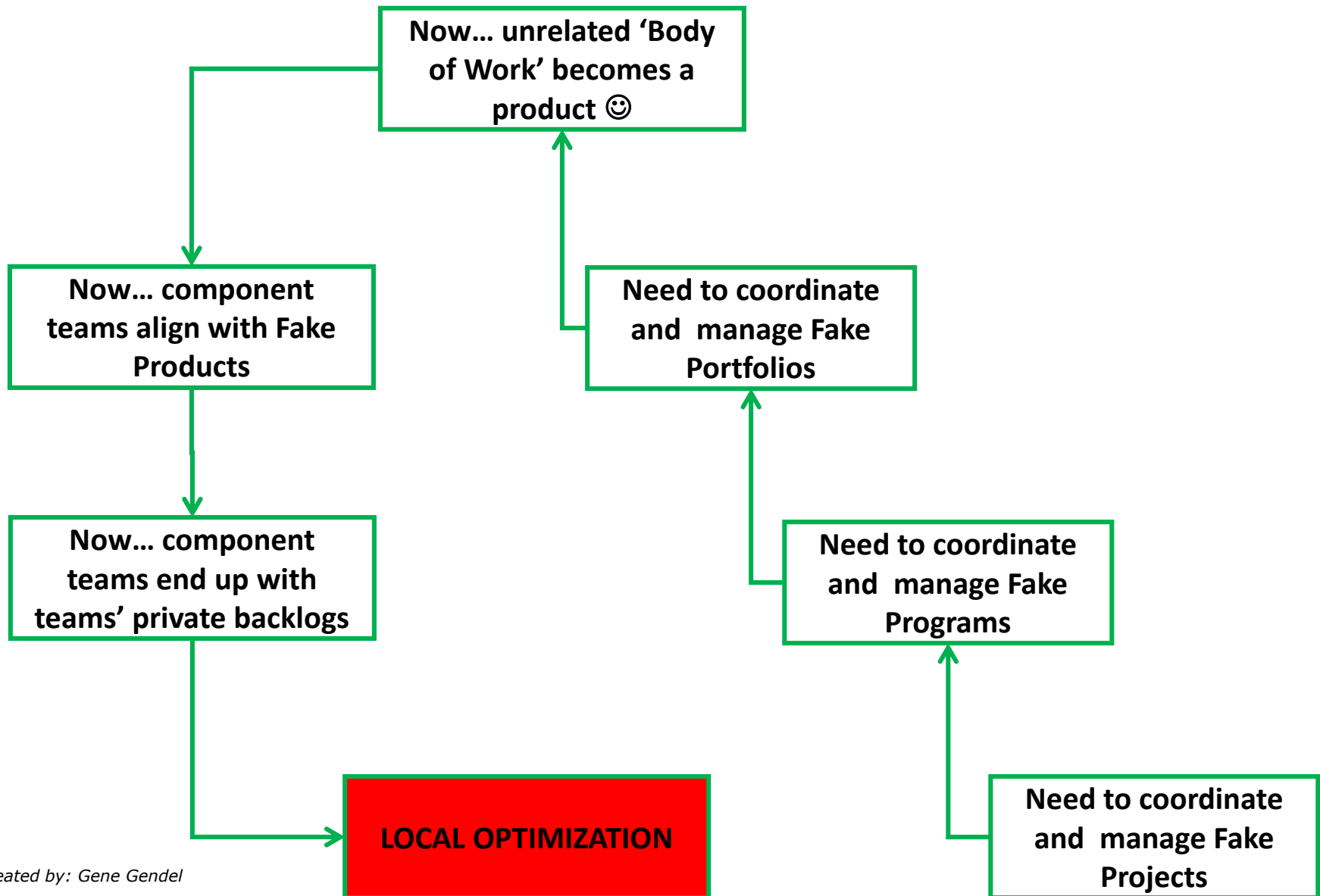
Sourced from: <https://www.atlassian.com/blog/jira-software/portfolio-for-jira-project-managers-edition>

Fake Product Portfolio



Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

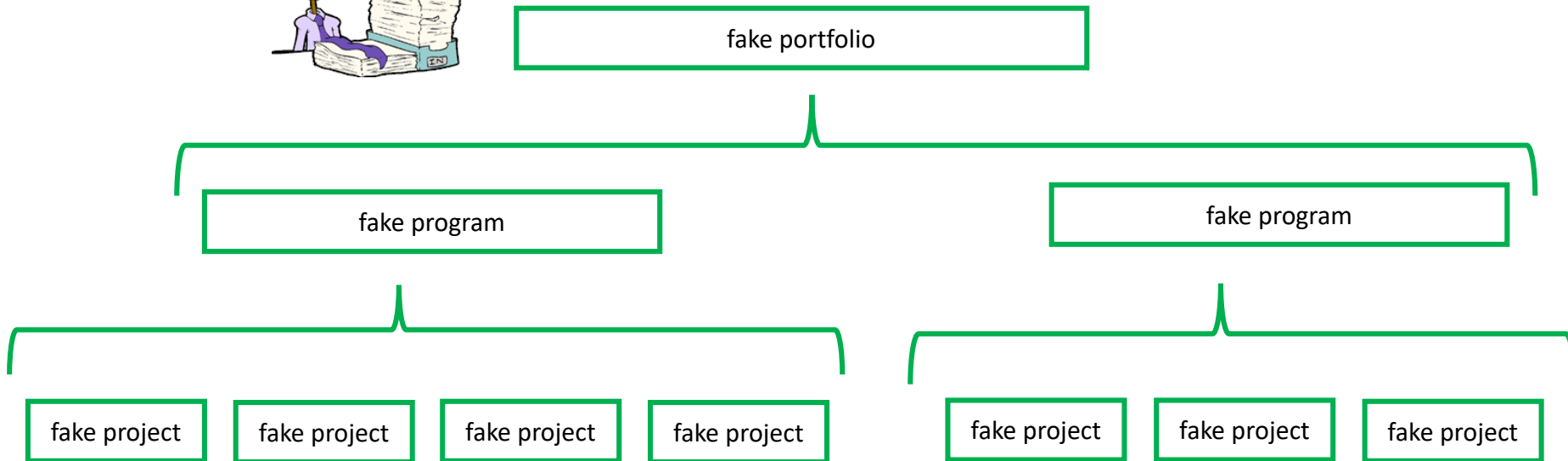
Fake Product Portfolio



Created by: Gene Gendel

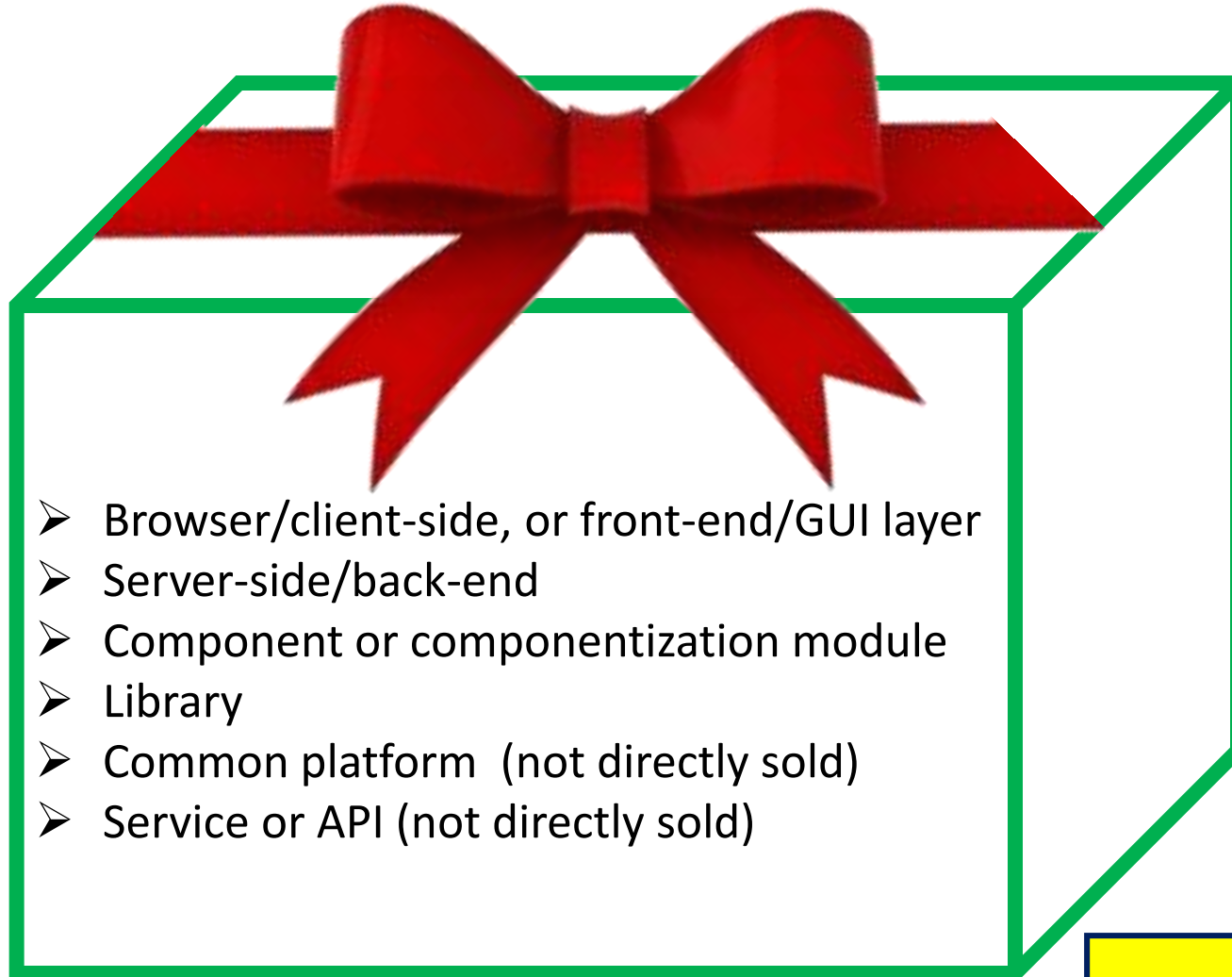
Self-Inflicted Wound: Local Optimization

*Requires complex
management and reporting
structure with multiple
translation layers of
management bureaucracy*



Created by: Gene Gendel

Moment of Self-Assessment



Q: Do you consider these as your 'products' or 'projects'?

Created by: Gene Gendel

Product Definition in LeSS

- Organizing by Business Value and Customer Centricity
- **Expanding** Product as wide as possible
- **Constraining** Product to keep it reasonably manageable
- Maintaining single Product Backlog (not multiple, private team backlogs)



PRODUCT

11 Degree
Local
Optimization
in BoW
Management

12
Ability to
identify
PRODUCT

Exercise

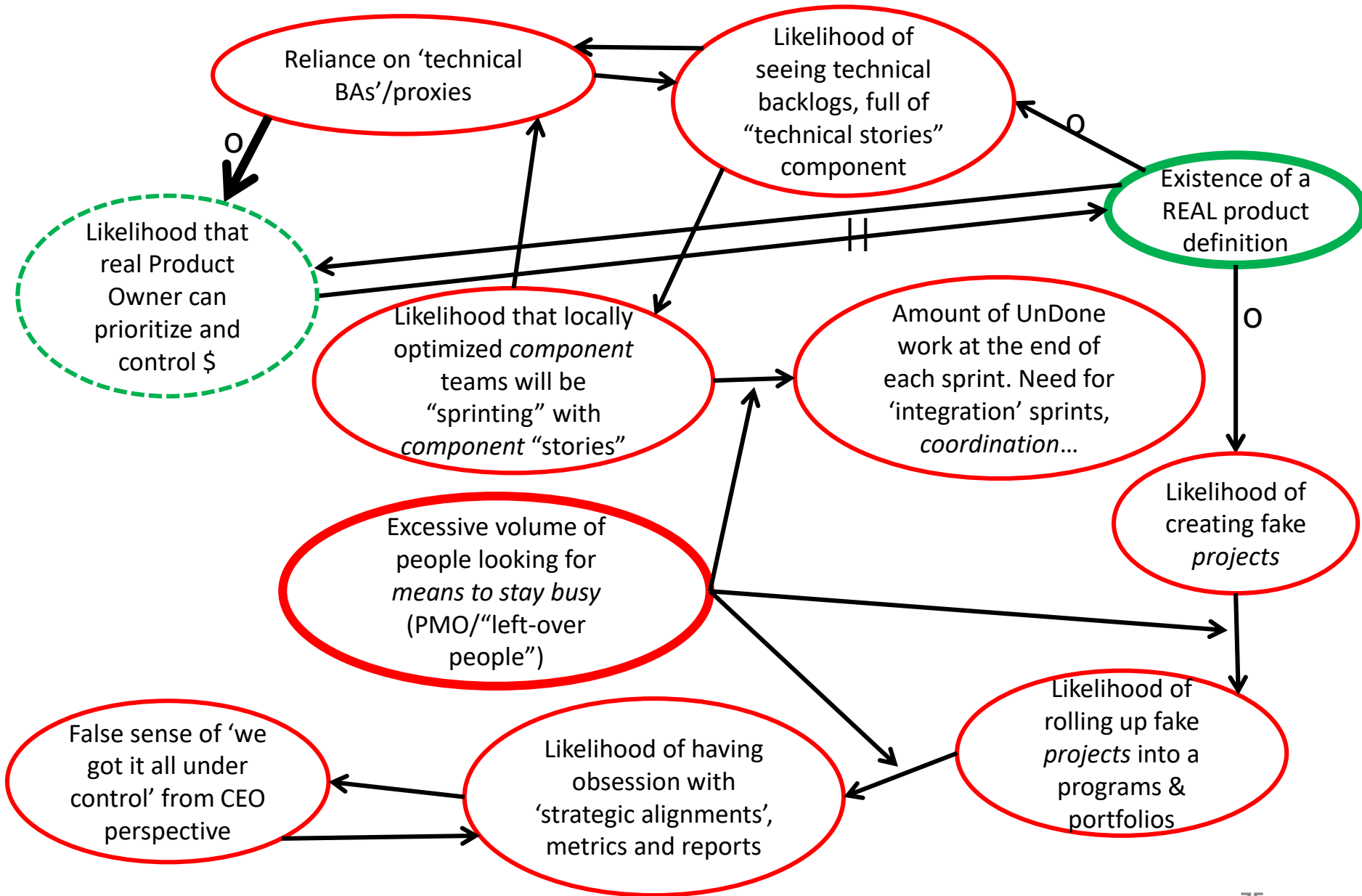
Duration: 10 min

Class: in-groups, brainstorm some of the most common pitfalls in defining a **product and problems with *complex WBS (projects, programs and portfolios) and redundant roles.*** Work with provided system variables to create a model.

Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in Roles & WBS - Exercise



LeSS Roles

LeSS Responsibilities

LeSS Relationships

Team



<http://less.works> 

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

Team Maturation

Types of teams


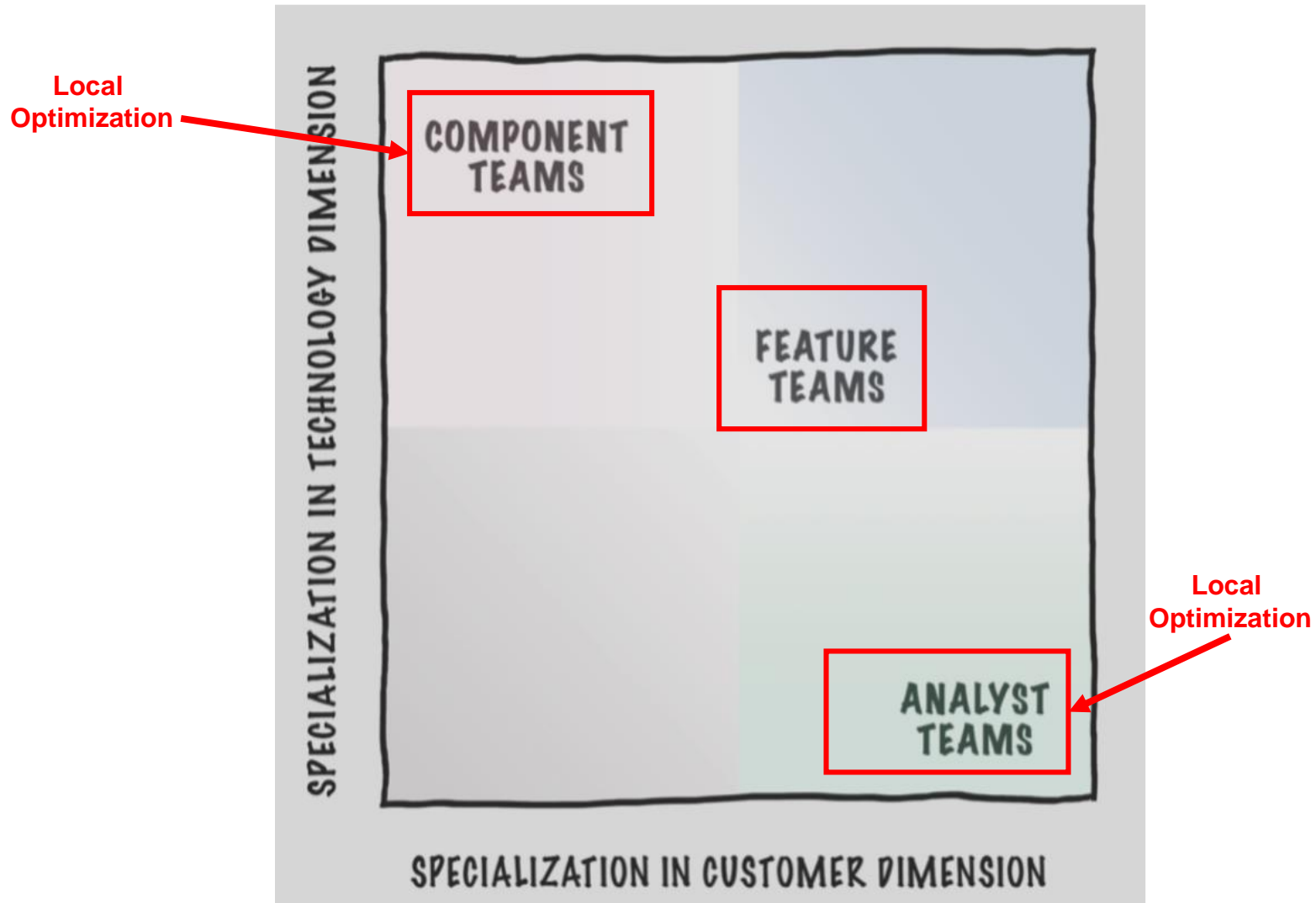
Setting overall direction				
Designing the team and its organizational context	Management Responsibility			
Monitoring and managing work process and progress		Team's Own Responsibility		
Executing the team task				
	Manager-led teams	Self-Managing teams	Self-Designing teams	Self-Governing teams

Figure 1. Types of Teams.

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

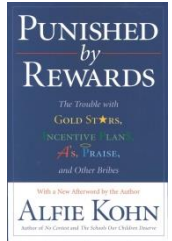
Feature Team Specialization



Sourced from: <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"



13 Degree of team maturity

14 Likelihood that Scrum Master role has impact

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

ScrumMaster



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

Scrum Master: “Bad HR SQL”



Bad SQL (in HR Database)

*It is amazing if your company, creates the following important titles (marked in **red**) in its HR database, to support agile roles and careers. However, it is a sign of a deep systemic dysfunction and ‘agile masquerade’, if the following SQL queries are ran, as a batch job, overnight... ☹.*

```
UPDATE table_employee  
    SET table_employee.title = ‘Senior Agile Coach’  
WHERE table_employee.title = ‘Senior Project Manager’
```

```
UPDATE table_employee  
    SET table_employee.title = ‘Scrum Master’  
WHERE table_employee.title = ‘Junior Project Manager’
```

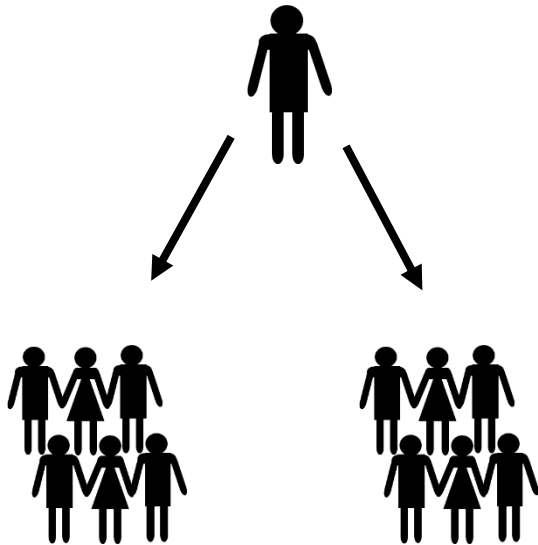
```
UPDATE table_employee  
    SET table_employee.title = ‘Product Owner’  
WHERE table_employee.title = ‘Business Analyst’
```

Brought to you by www.keystepstosuccess.com

Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

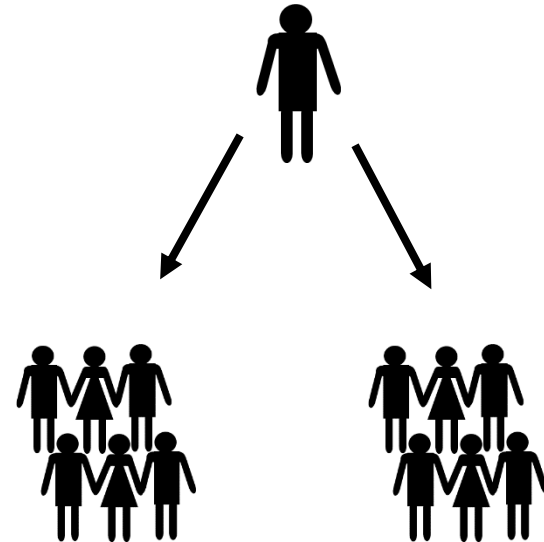
Scrum Master: Style of Engagement

Project Manager



Coordination

Scrum Master



Facilitation



Scrum Master: What is Misunderstood?

Misunderstood

The ScrumMaster role is a new one and often not understood by teams and organizations who are adopting Scrum. A frequent response is to make the “leftover people” the ScrumMasters. They might be nice people but often lack the right skills, motivation, and Scrum knowledge to be effective ScrumMasters. They morph the role into something else which then becomes accepted within the organization as the way a ScrumMaster should be. After all, the ScrumMaster should know if they’re doing things correctly, right? Sometimes well-meaning ScrumMasters who are new to Scrum or not a good fit for the ScrumMaster role cause things to happen that are actually counter to Scrum and detrimental to Scrum adoption, thereby transforming them into anti-ScrumMasters.

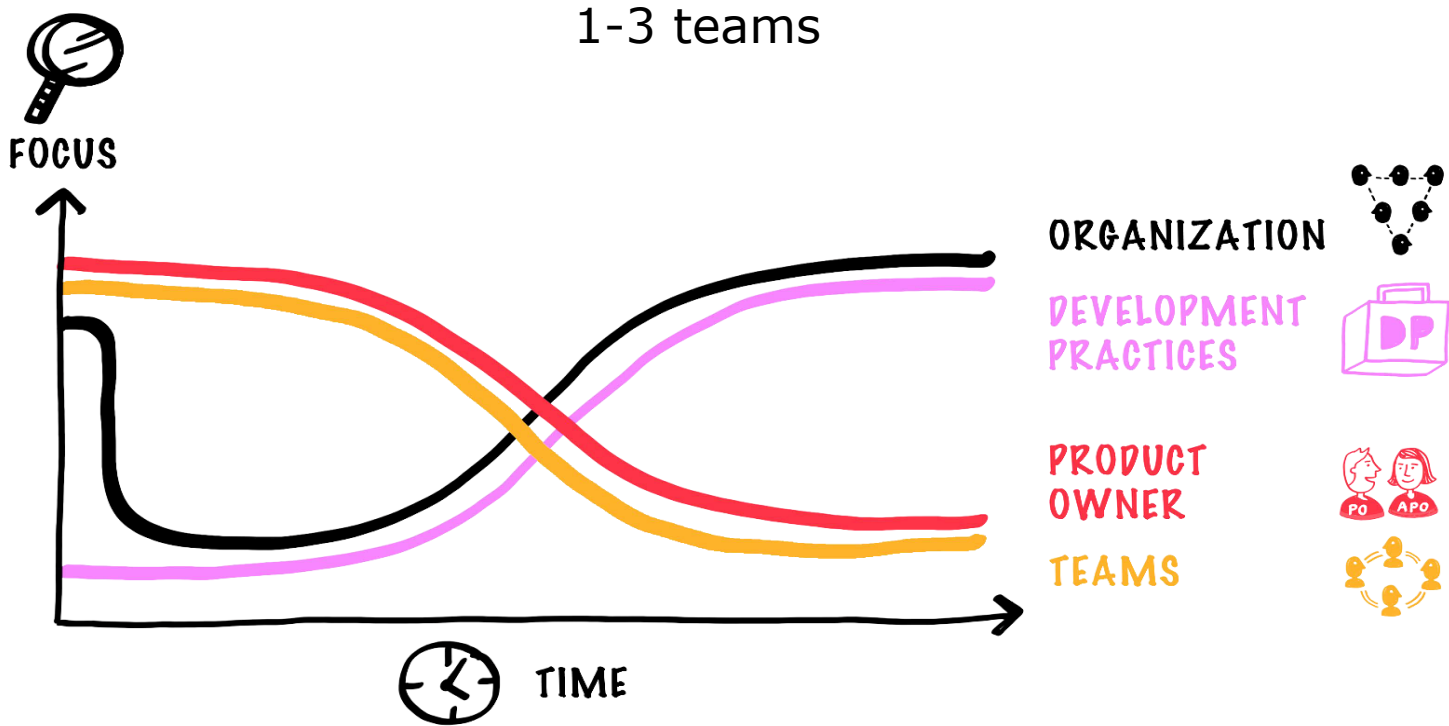
Q: Is there HR Support for Scrum Master role?

Dedicated full-time role

In LeSS, the ScrumMaster role is vital. We’ve seen many organizations try part-time ScrumMasters, which usually leads to no ScrumMasters at all. This then affects the LeSS adoption enormously. In LeSS the ScrumMaster is a dedicated, full-time role in the same way that being a Scrum Team member is a dedicated, full-time role. Having said that, it is possible for one full-time ScrumMaster fill the role for up to three teams, depending on any number of factors.

Sourced from less.works

Scrum Master: Focus



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

<http://less.works> 

14 Likelihood
that Scrum
Master role
has impact

Exercise

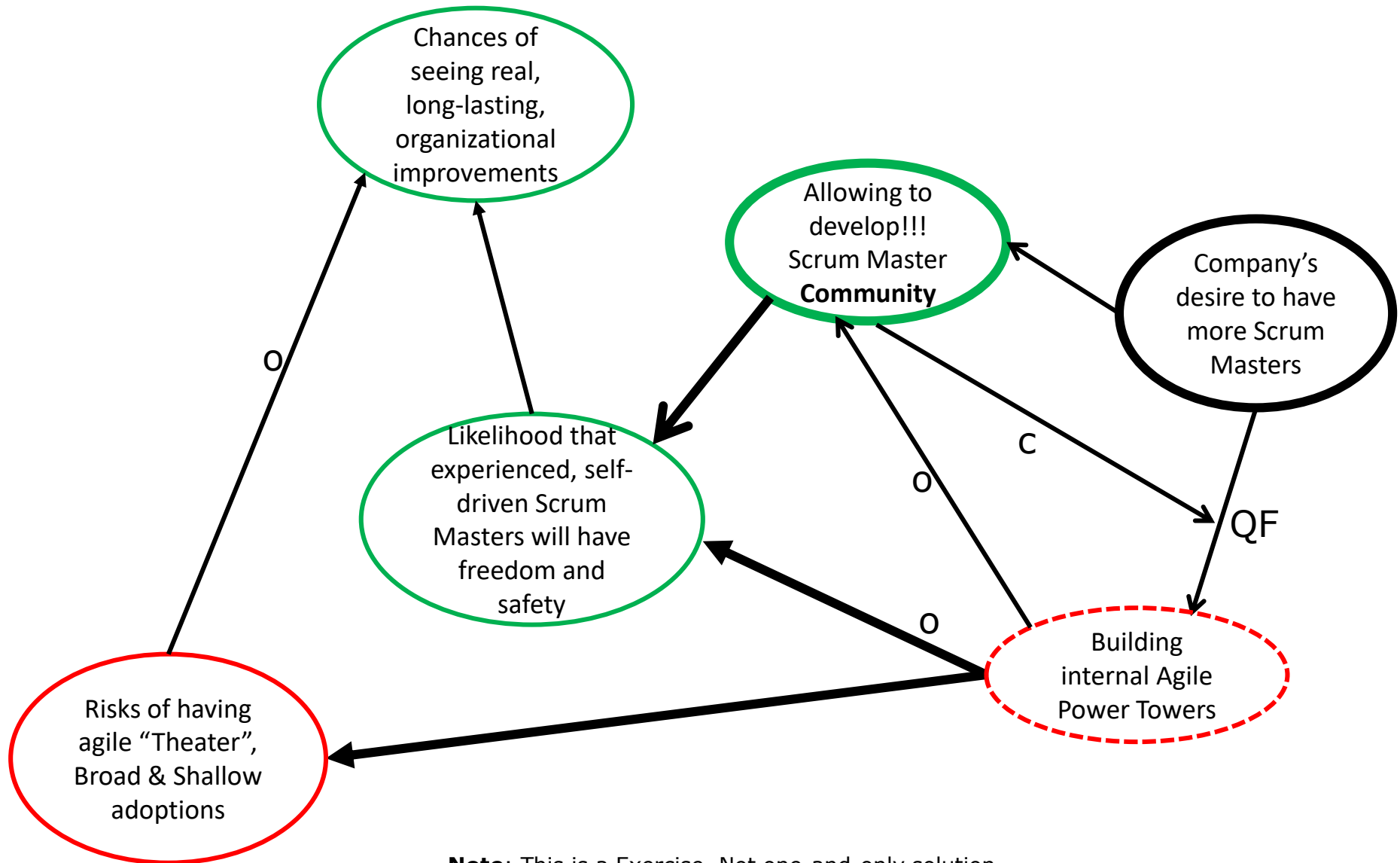
Duration: 10 min

Class: in-groups, brainstorm typical anti-patterns associated with misunderstanding **Scrum Master** profession ; use post-it notes to capture discoveries

Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in Scrum Master Role - Exercise



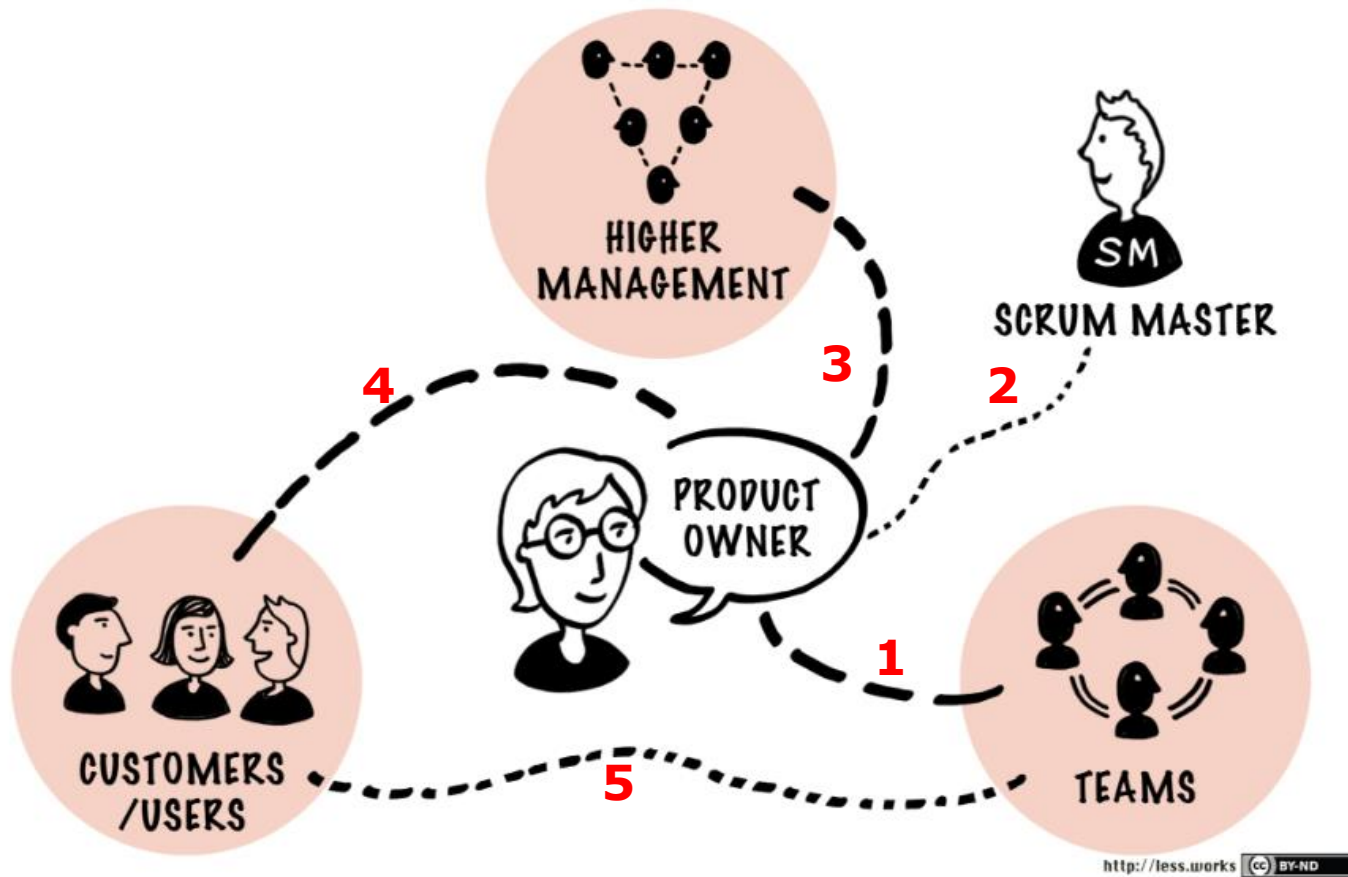
Note: This is a Exercise. Not one-and-only solution.

Product Owner



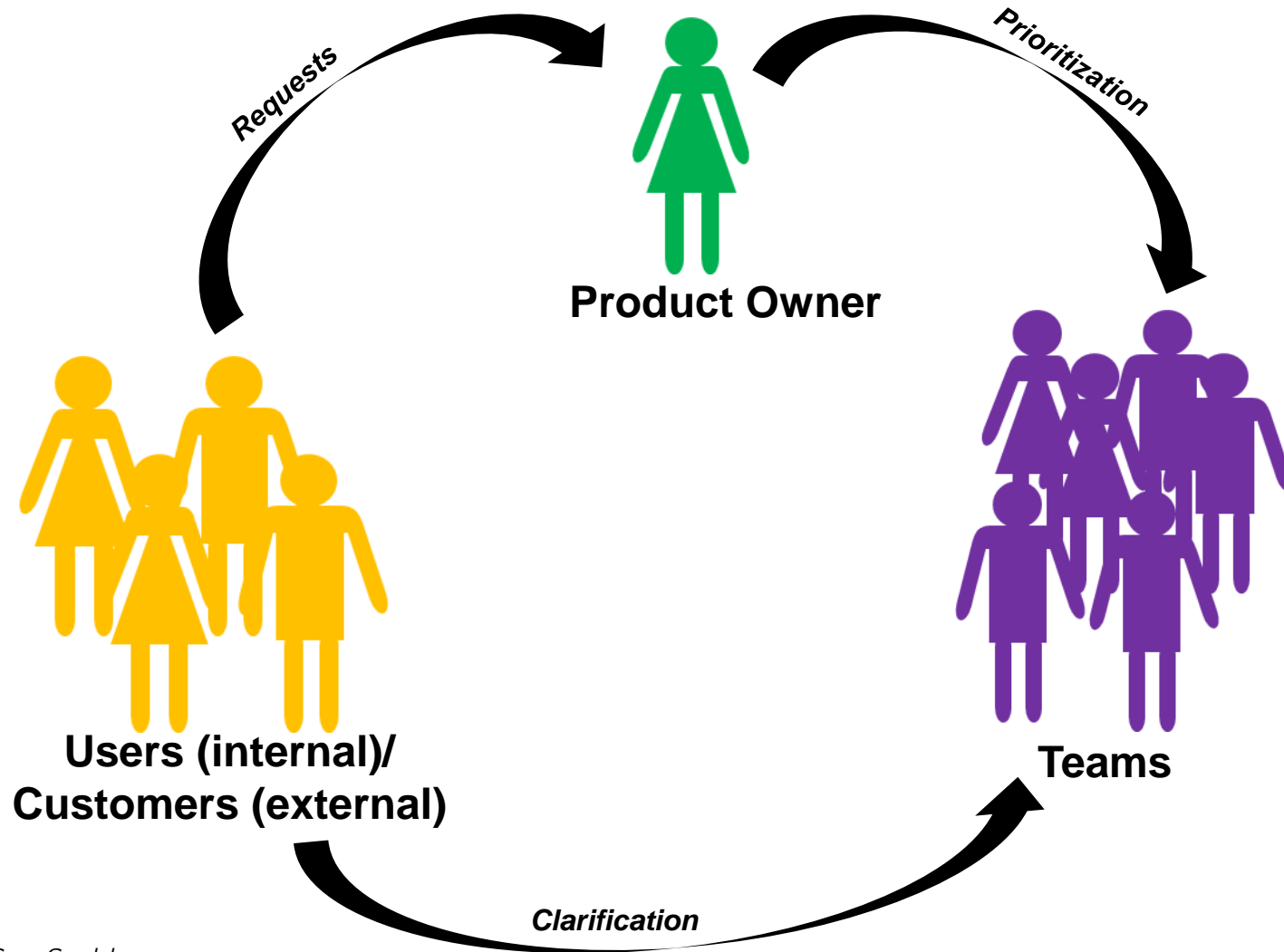
Sourced from: <https://less.works/resources/graphics/index.html>

Product Owner: 5 Relationships



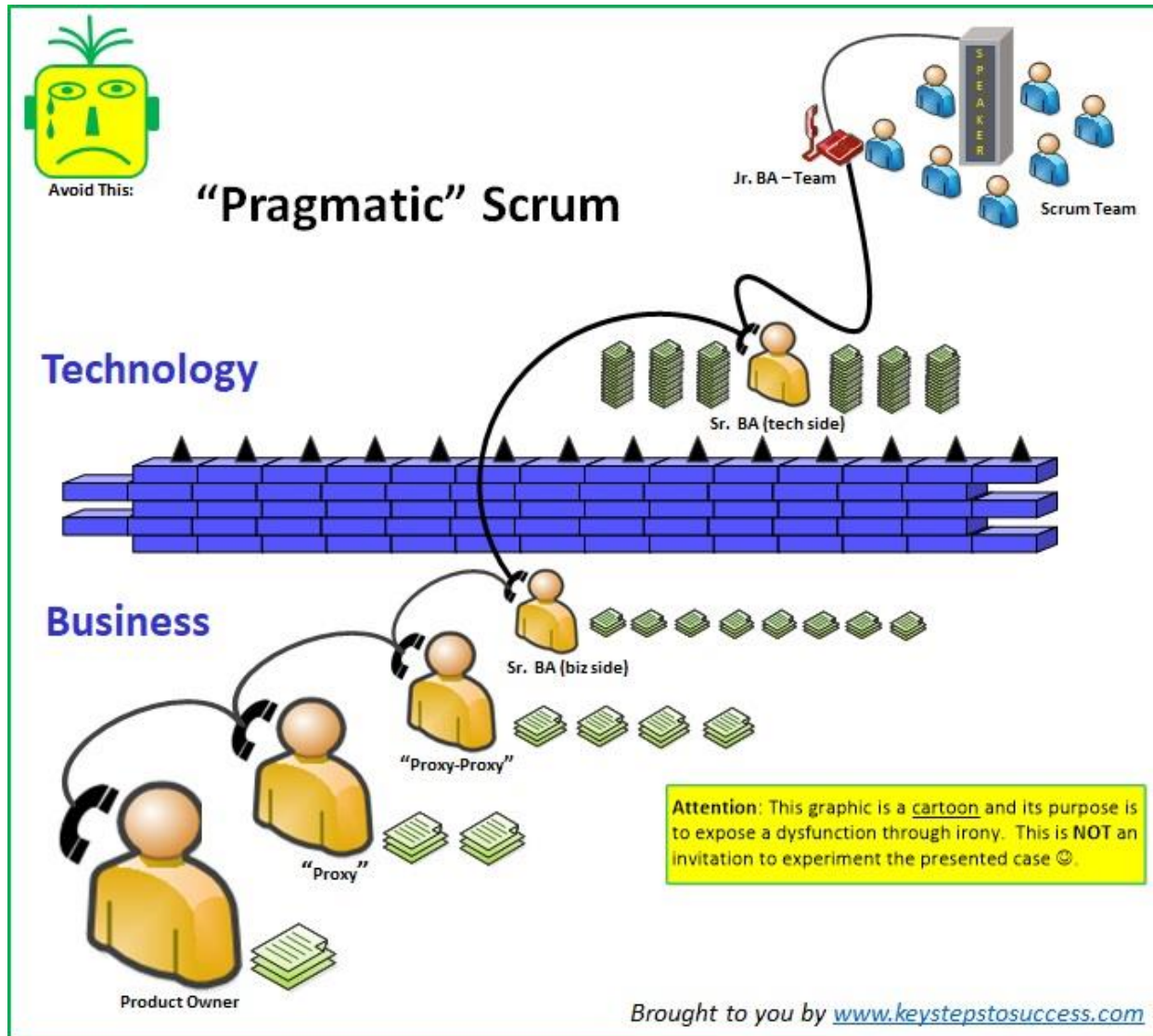
Sourced from: <https://less.works/resources/graphics/index.html>

Product Owner: Prioritization vs. Clarification



Created by: Gene Gendel

Product Owner: Avoid This Dysfunction



Sourced from: <https://www.keystepstosuccess.com/agile-anti-patterns-with-irony-and-satire/>

No Team Output Owner (“TOO”)



0:05 / 9:42



Why "Scrum" Isn't Making Your Organization Agile: Harmful Misconceptions About Product Owner Role

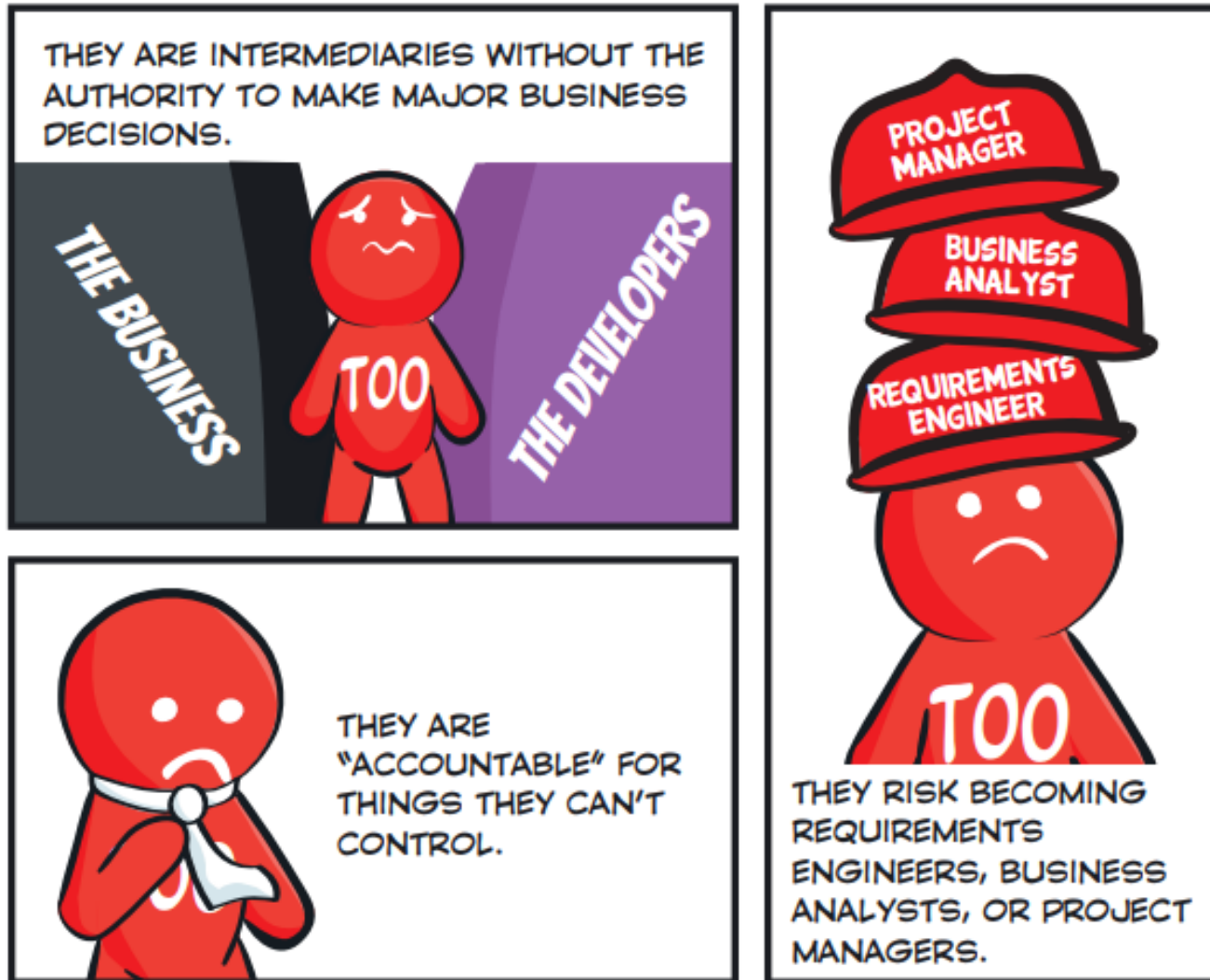


MJ (MICHAEL JAMES)
has written many things,
including the Scrum Master Checklist at
<http://ScrumMasterChecklist.org>
Email him at mj@seattlescrum.com
or tweet him at [@michaeldotjames](https://twitter.com/michaeldotjames)

Sourced from: <https://www.youtube.com/watch?v=cr2rjaGmUzo>

Sourced from: <https://seattlescrum.com/downloads/Why-Scrum-Isnt-Making-Your-Company-Very-Agile-v2.pdf>

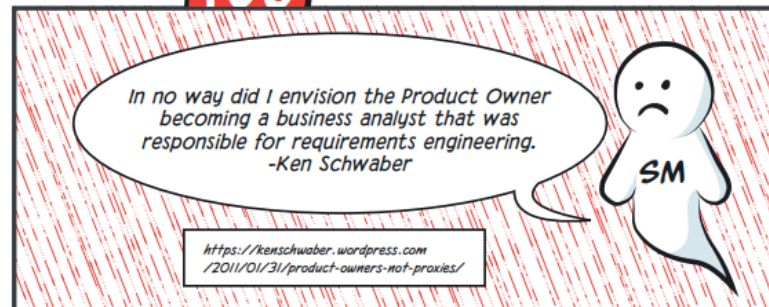
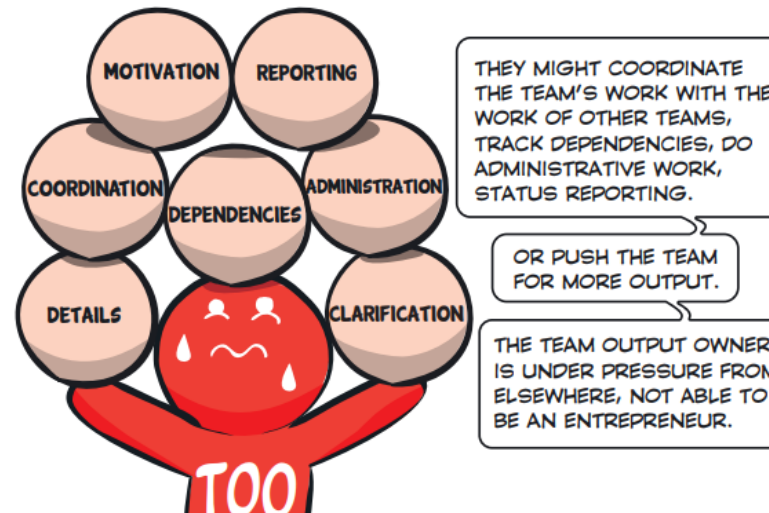
Team Output Owner (TOO): “Accountability”



(C) 2019 Michael James (mj). All rights reserved.

18

Team Output Owner (TOO) Local Optimization



19

(C) 2019 Michael James (mj). All rights reserved.

15

Likelihood
that Product
Owner role
has impact

Exercise

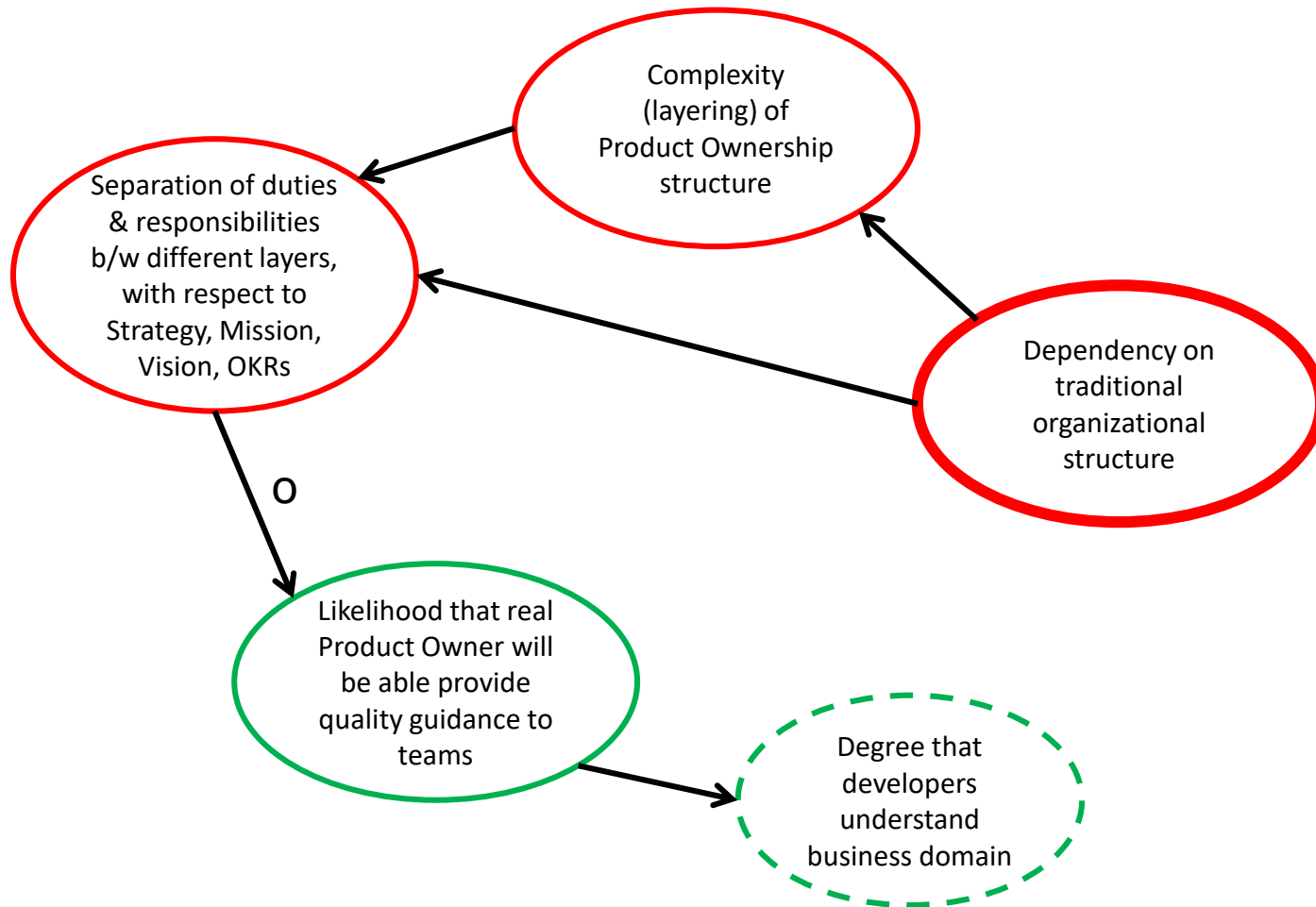
Duration: 10 min

Class: in-groups, brainstorm some of the most common examples of ***Local Optimization in PO-ship Structure***, in your respective organizations. Work with provided system variables to create a model.

Method/Tool: System Modeling with CLD

Instructor: review with Class

Local Optimization in **PO-ship Structure**- Exercise

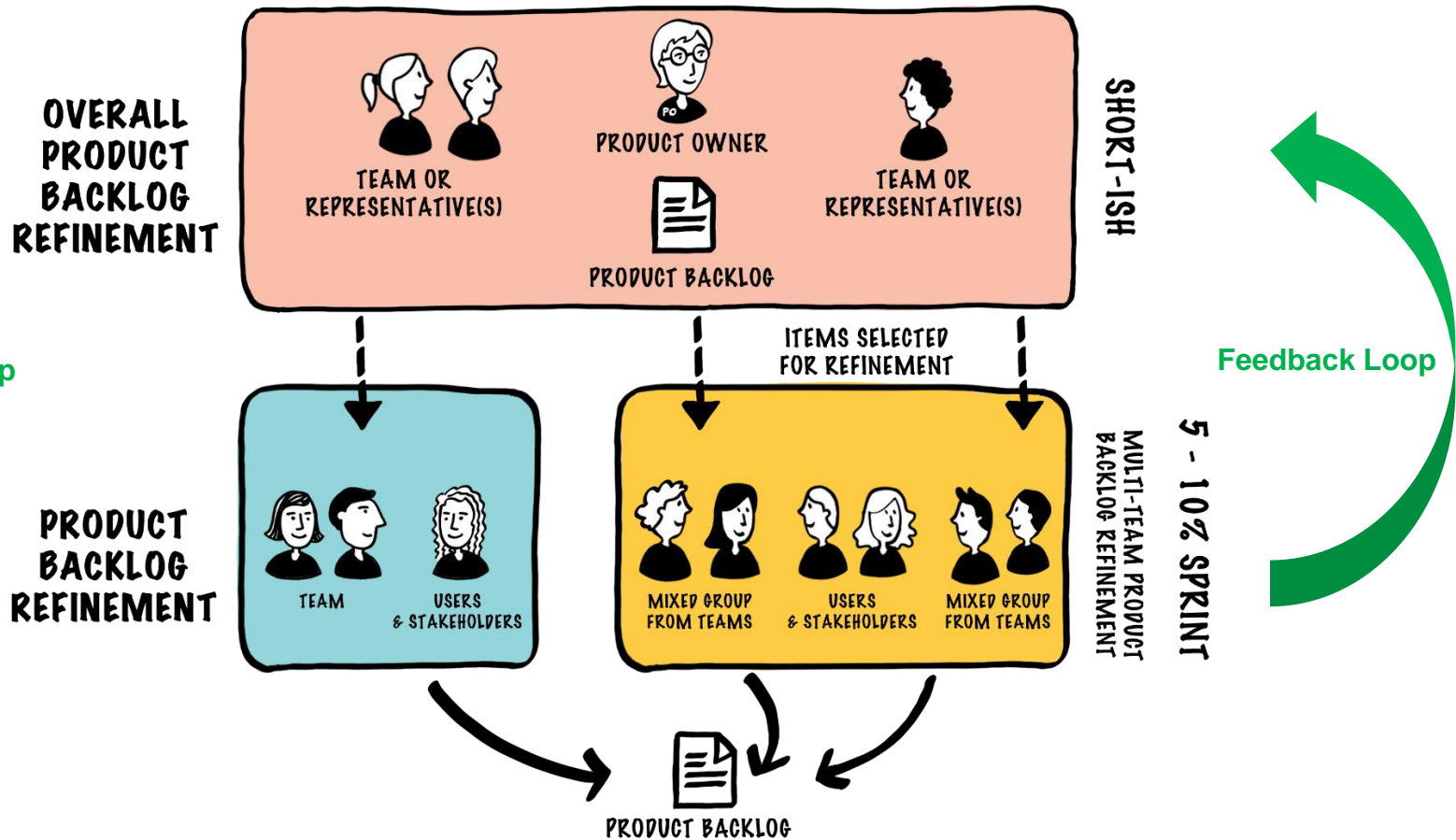




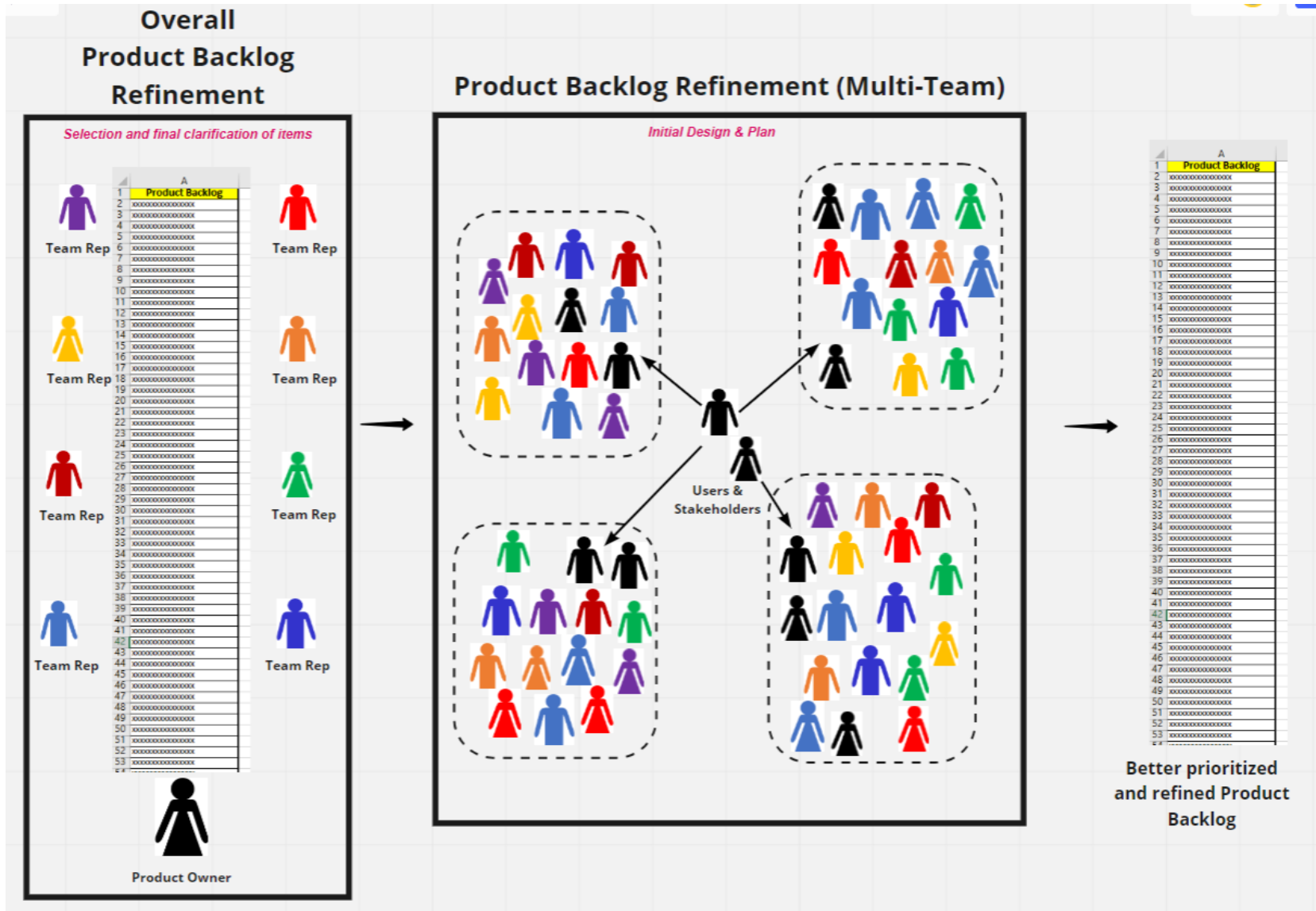
LeSS Events

LeSS Events

LeSS PRODUCT BACKLOG REFINEMENT



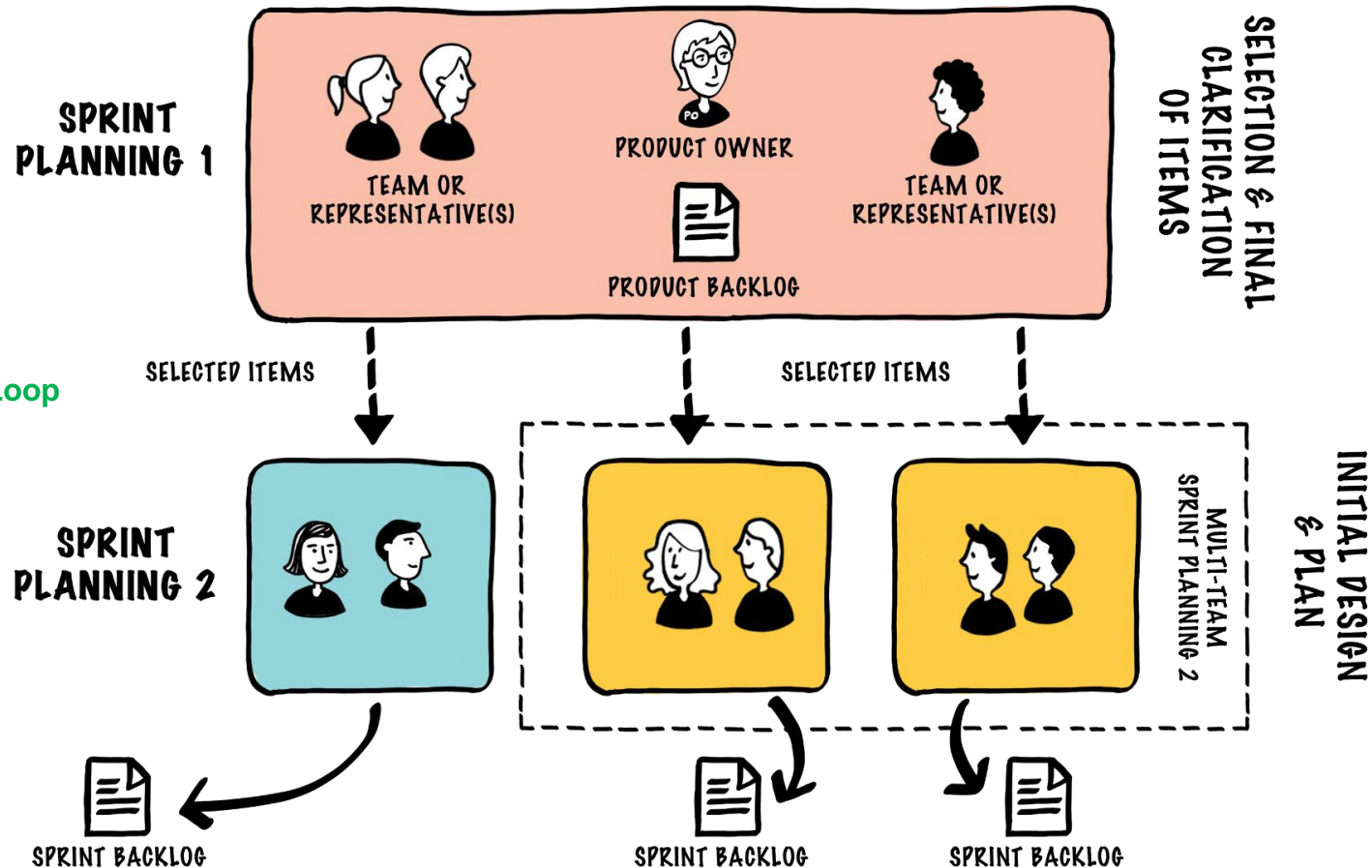
LeSS Events



Created by: Gene Gendel

LeSS Events

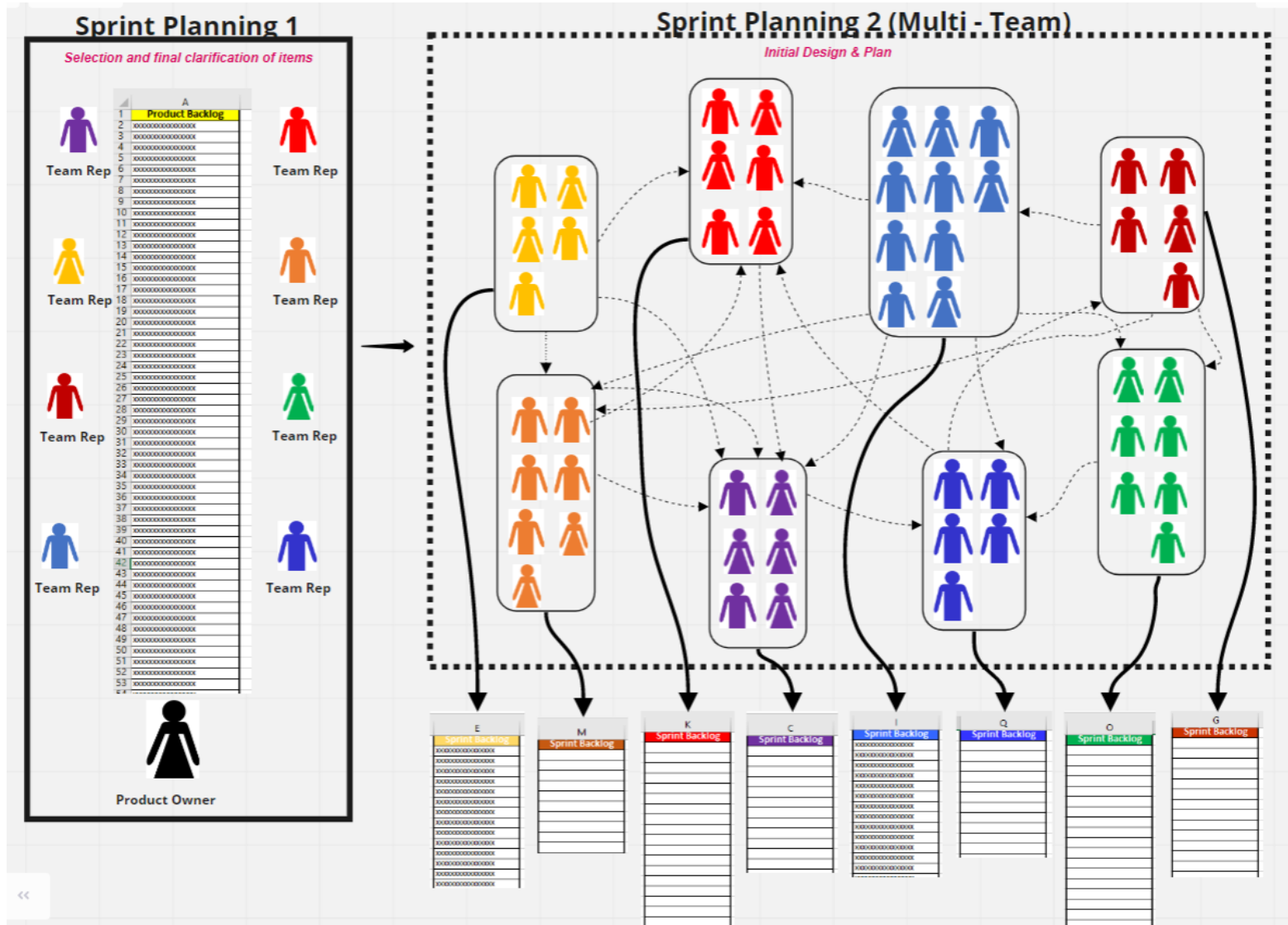
LeSS SPRINT PLANNING



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

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

LeSS Events

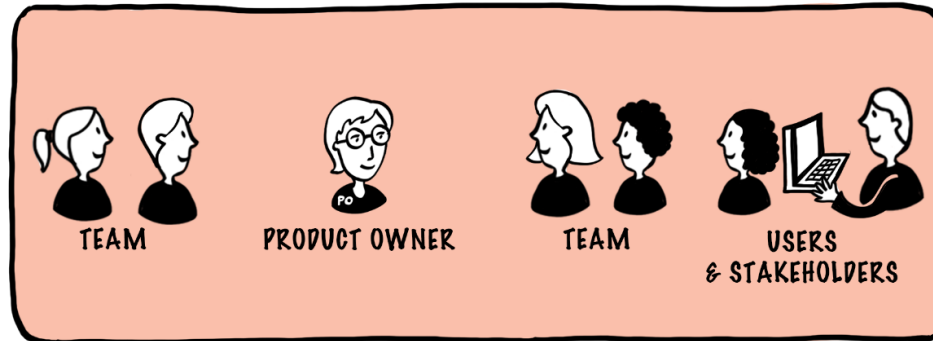


Created by: Gene Gendel

LeSS Events

LeSS SPRINT REVIEW & RETROSPECTIVE

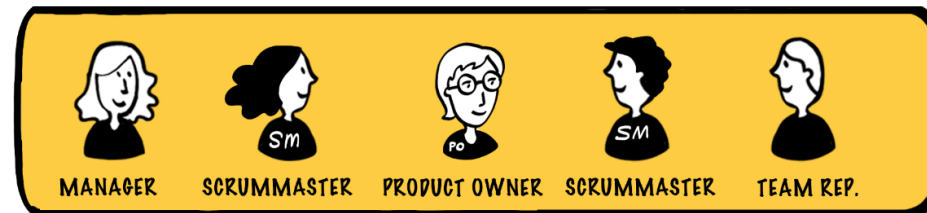
SPRINT REVIEW



TEAM RETROSPECTIVE



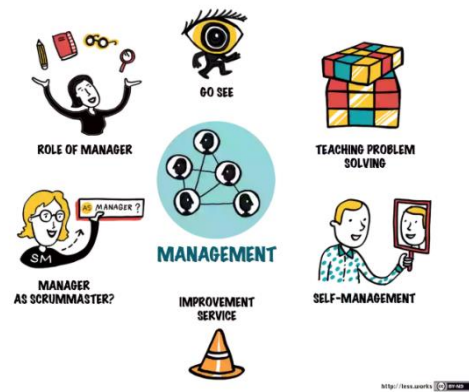
OVERALL RETROSPECTIVE



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

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

Management In LeSS



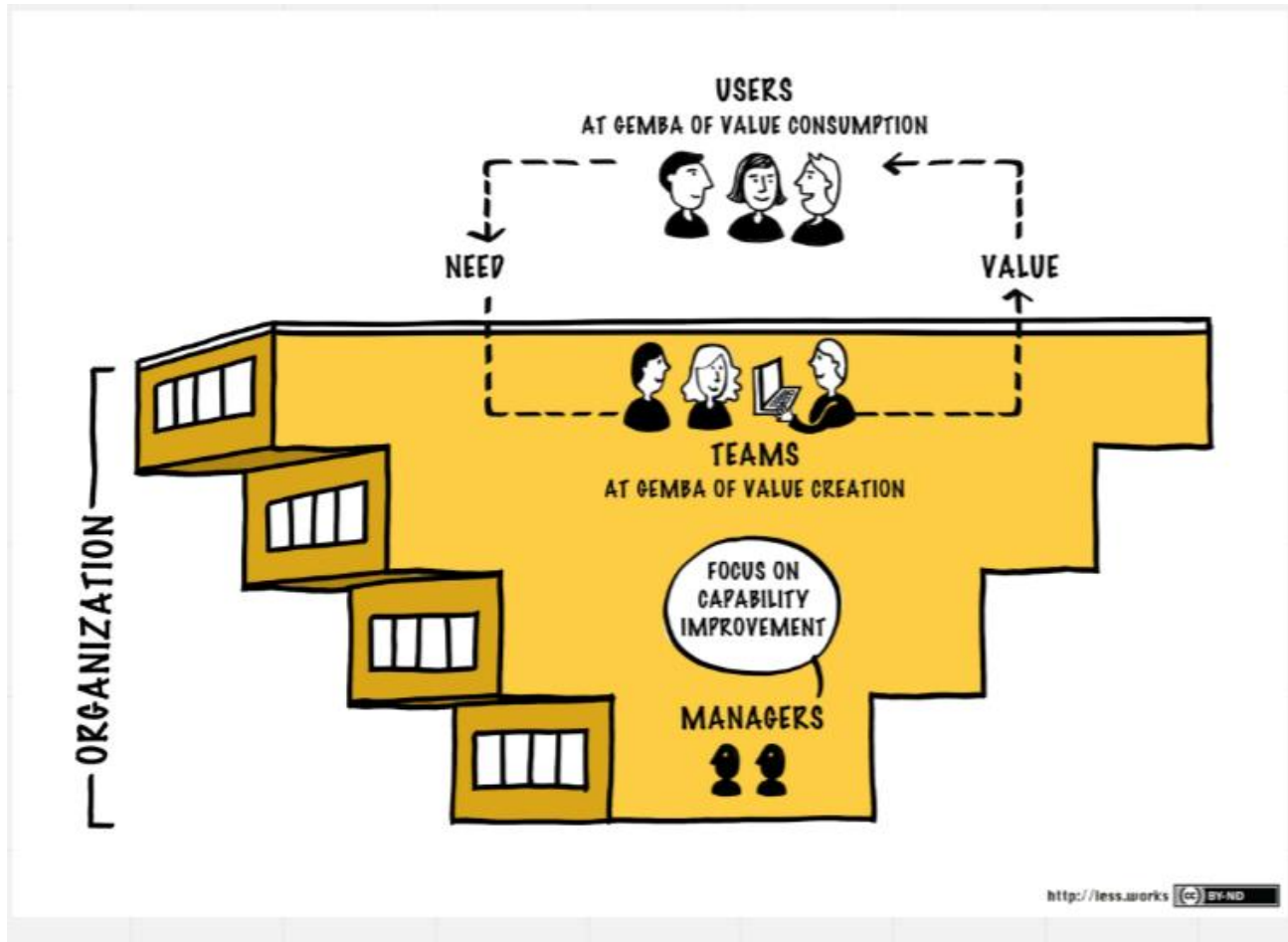
Sourced from: <https://less.works/resources/graphics/index.html>

LeSS Management



Sourced from: <https://www.youtube.com/watch?v=Wdroj6F3VIQ>

LeSS Management



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

LeSS Management

Frederick W. Taylor



Frederick Taylor is known today as the father of scientific management. One of his many contributions to modern management is the common practice of giving employees rest breaks throughout the day.

Henry Fayol



Pioneered the study of principles and functions of management. Developed management functions:

- Planning
- Coordinating
- Organizing -Controlling
- Commanding



'Theory X'

Management

Theory X - authoritarian, repressive style, tight control, no development, produces limited, depressed culture

staff

'Theory Y'

Staff

Theory Y - liberating and developmental, control, achievement and continuous improvement achieved by enabling, empowering and giving responsibility

management



LeSS Management

A common problem with Scrum adoption occurs when managers do not give up these responsibilities, thereby causing an organizational conflict between the Team, Product Owner, Scrum Master, and managers.

Job & Salary Safety
but not
Role Safety

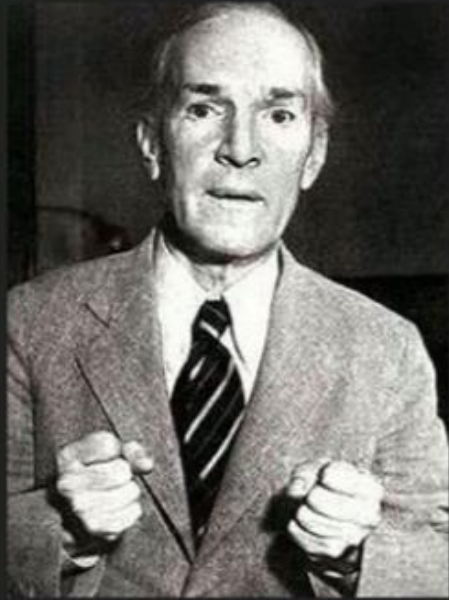
LeSS Management

It is difficult to get a man to understand something when his job depends on not understanding it. —Upton Sinclair

Who is going to strive for continuous improvement when the likely outcome is losing a job? Nobody. In a LeSS adoption, it is vitally important to establish the policy that nobody is going to lose employment. At least not due to position or role eliminations from the structural changes caused by the LeSS adoption. Communicate this clearly and repeatedly.

Sourced from: 2009 - Scaling Lean and Agile Development - Craig Larman, Bas Vodde

LeSS Management



It is difficult to get a man to understand something when his salary depends upon his not understanding it.

(Upton Sinclair)

izquotes.com

Sourced from: https://www.reddit.com/r/anticapitalism/comments/eivy6c/it_is_difficult_to_get_a_man_to_understand/

LeSS Management

Larman's Laws of Organizational Behavior

([Spanish translation](#))([Russian translation](#)) ([Japanese translation](#))([Persian/Farsi translation](#))([Polish translation](#))([Portuguese translation](#))

After decades of observation and organizational consulting, here are *Larman's Laws of Organizational Behavior*. These are observations rather than laws to follow ;)

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 redefining or overloading 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”, “revolutionary”, “religion”, and “needing pragmatic customization for local concerns” — which deflects from addressing weaknesses and manager/specialist status quo.
4. As a corollary to (1), if after *changing the change* some managers and single-specialists are still displaced, they become “coaches/trainers” for the change, frequently reinforcing (2) and (3).
5. (in large established orgs) *Culture follows structure*. And in tiny young orgs, structure follows culture.

Elaboration:

Sourced from: https://www.craiglarman.com/wiki/index.php?title=Larman%27s_Laws_of_Organizational_Behavior

LeSS Management



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Current events](#)
[Random article](#)
[About Wikipedia](#)
[Contact us](#)
[Donate](#)

[Contribute](#)
[Help](#)
[Learn to edit](#)
[Community portal](#)
[Recent changes](#)
[Upload file](#)

Not logged in

Article

Talk

Read

Edit

View history

Informed consent

From Wikipedia, the free encyclopedia

This article is about consent to medical or research procedures. For consent in other contexts, see [Consent](#). For the [House of Commons](#), see [House of Commons](#).

Informed consent is a process for getting permission before conducting a healthcare intervention on a person, for conducting some form of research on a person, or for disclosing a person's information. A [health care provider](#) may ask a patient to [consent](#) to receive [therapy](#) before providing it, a [clinical researcher](#) may ask a [research participant](#) before enrolling that person into a [clinical trial](#), and a researcher may ask a research participant before starting some form of [controlled experiment](#). Informed consent is collected according to guidelines from the fields of [medical ethics](#) and [research ethics](#).

An informed consent can be said to have been given based upon a clear appreciation and understanding of the facts, implications, and consequences of an action. Adequate informed consent is rooted in respecting a person's dignity.^[1] To give informed consent, the individual concerned must have adequate reasoning faculties and be in possession of all relevant facts. Impairments to reasoning and judgment that may prevent informed consent include basic intellectual or emotional immaturity, high levels of stress such as [posttraumatic stress disorder](#) (PTSD) or a severe [intellectual disability](#), severe [mental disorder](#), [intoxication](#), severe [sleep deprivation](#), [Alzheimer's disease](#), or being in a [coma](#).

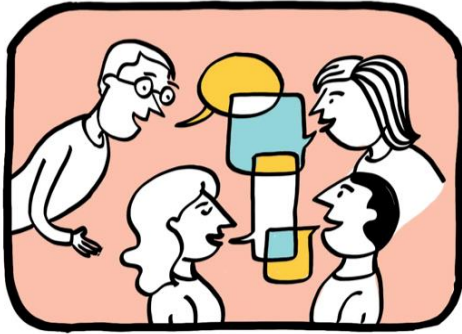
Sourced from: https://en.wikipedia.org/wiki/Informed_consent

Degree of sr.
management
support (real
gembu, not "in
spirit only")

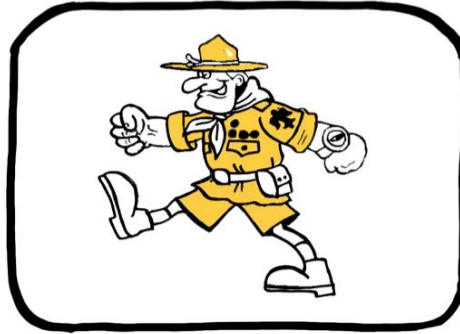
5

Coordination in LeSS (Outside of LeSS events)

Knowledge Flow in LeSS



JUST TALK



SCOUT



OPEN SPACE



TRAVELER



COMMUNITIES



COMPONENT MENTOR

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

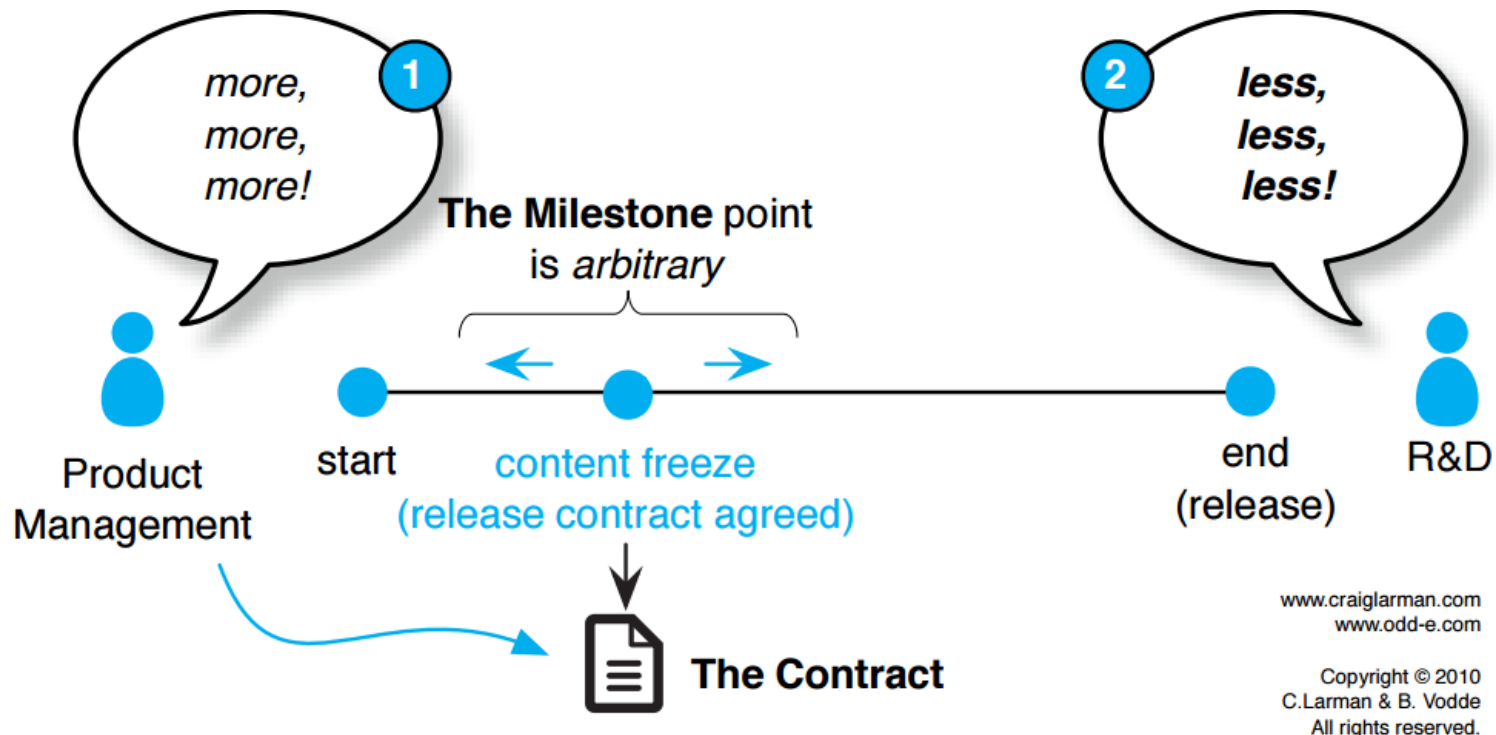
18

Quality of
vertical and
horizontal
learning in
LeSS

<http://less.works>



The Contract Game



[The “Contract Game” \(22-44min\)](#), by Craig Larman

Sourced from: <https://less.works/resources/graphics/book-images>

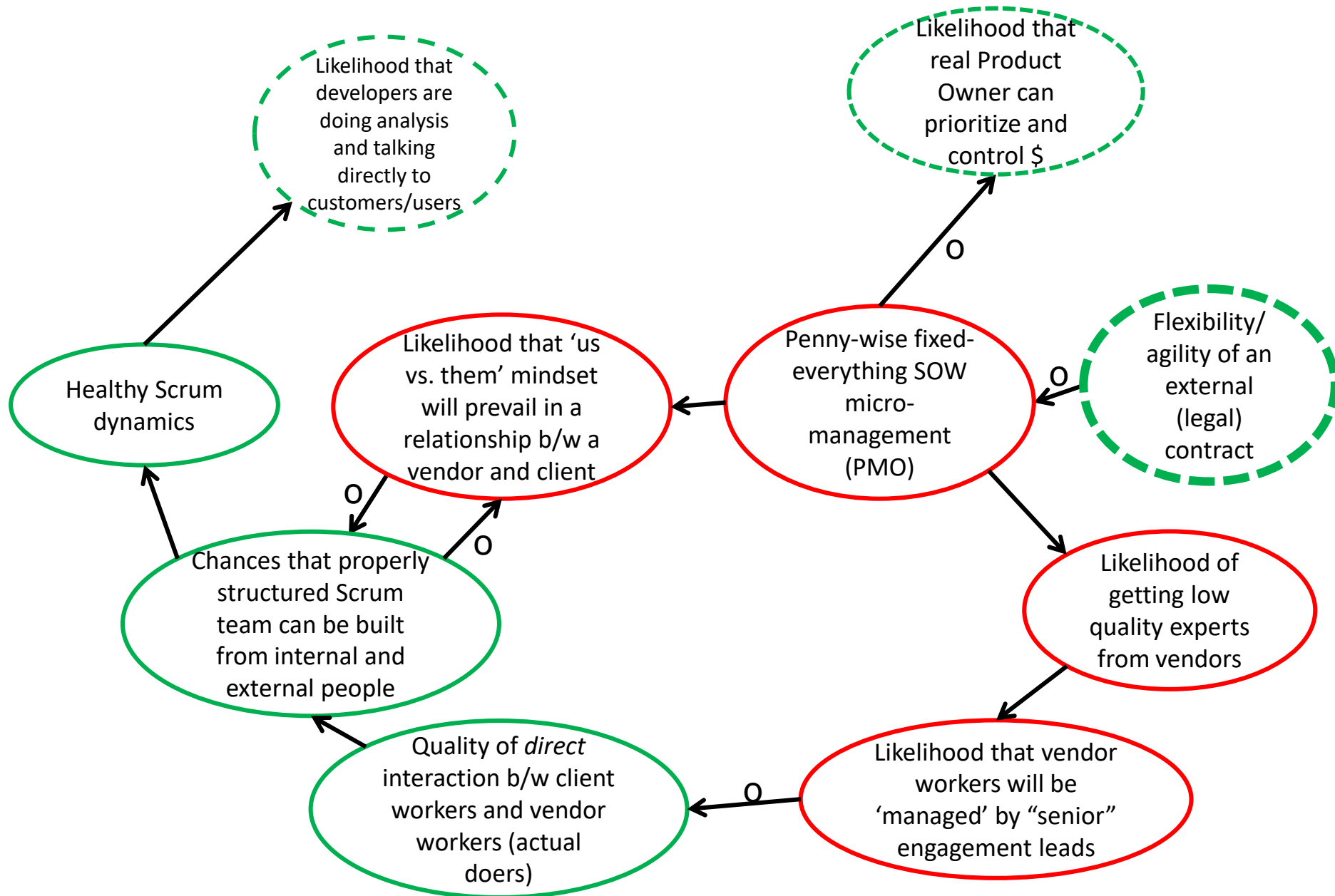
Exercise

Class: In-groups, think of how (legal) contracts, if 'translated' into an **internal contracts**, could lead to **local optimization**. Work with provided system variables to create a model.

Instructor: Give short feedback. Offer a recommended solution.

Duration: 10 min

Local Optimization in **Internal Contracts** - Exercise



Technical Excellence LeSS

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

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

Technical Excellence

Unfinished Work != Undone Work

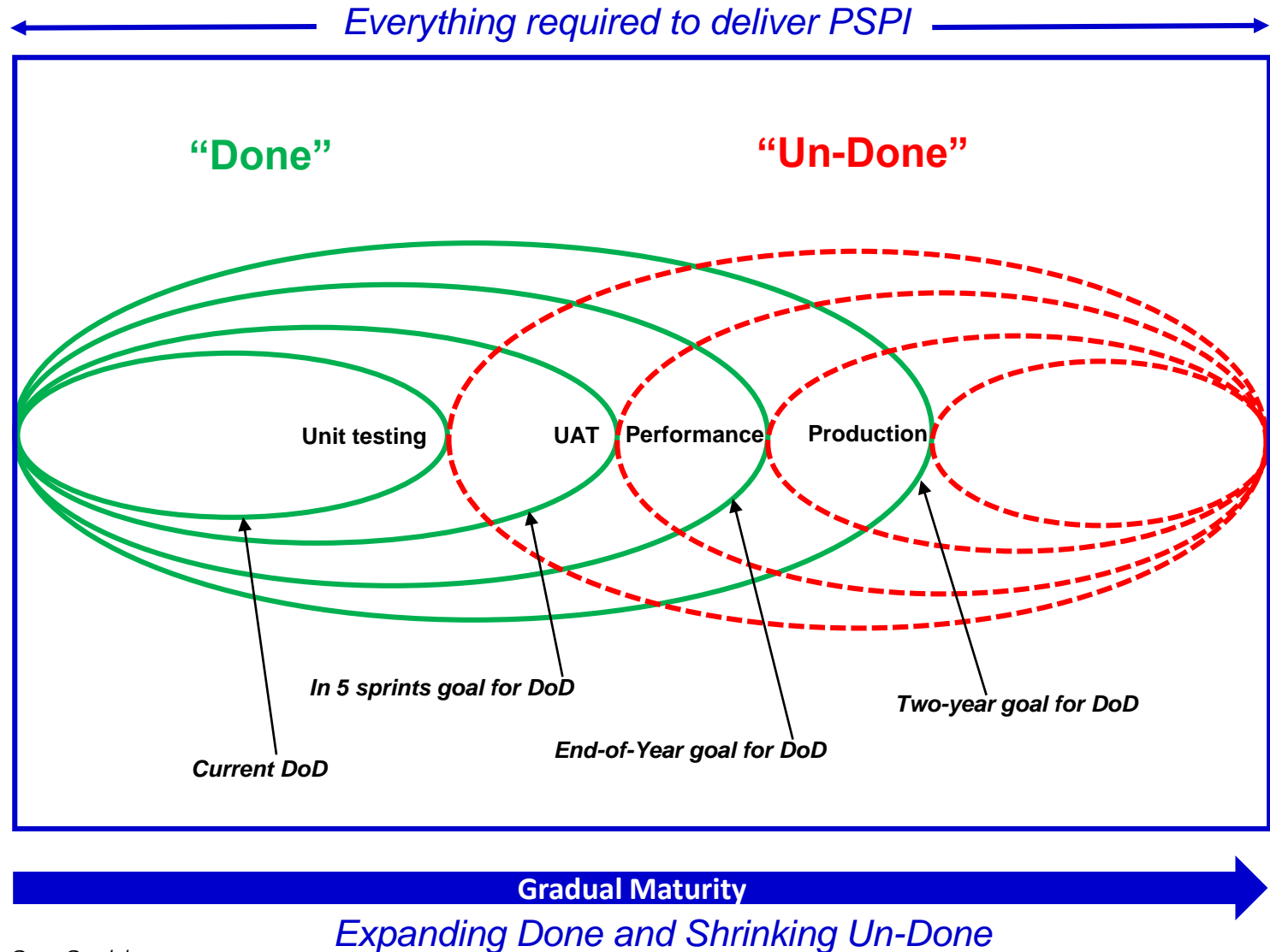
Usually, Team's issue



Usually, Organizational issue



Technical Excellence



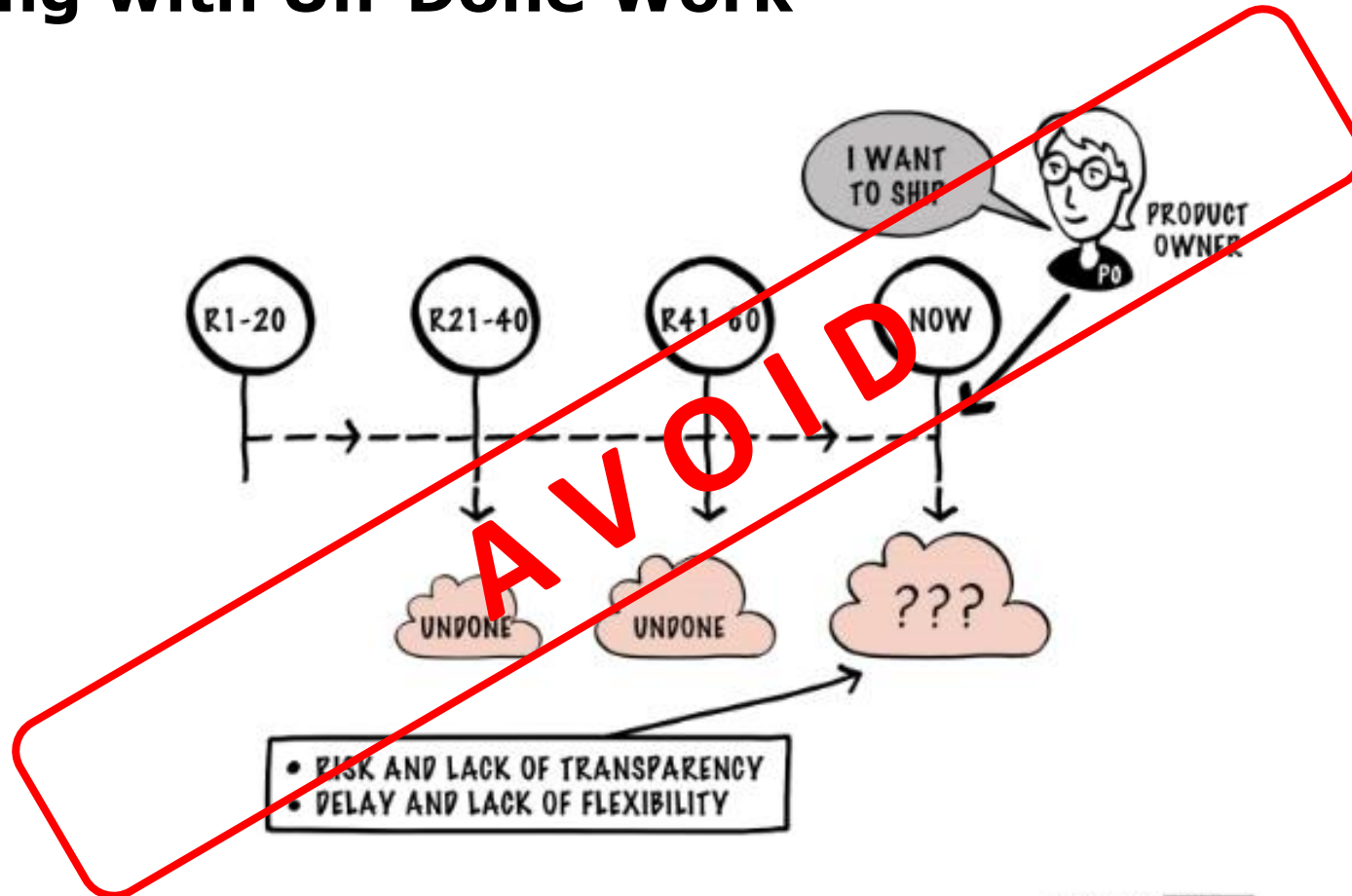
Created by: Gene Gendel

Technical Excellence

**REMOVE
UNDONE
“DEPARTMENT”**

Technical Excellence

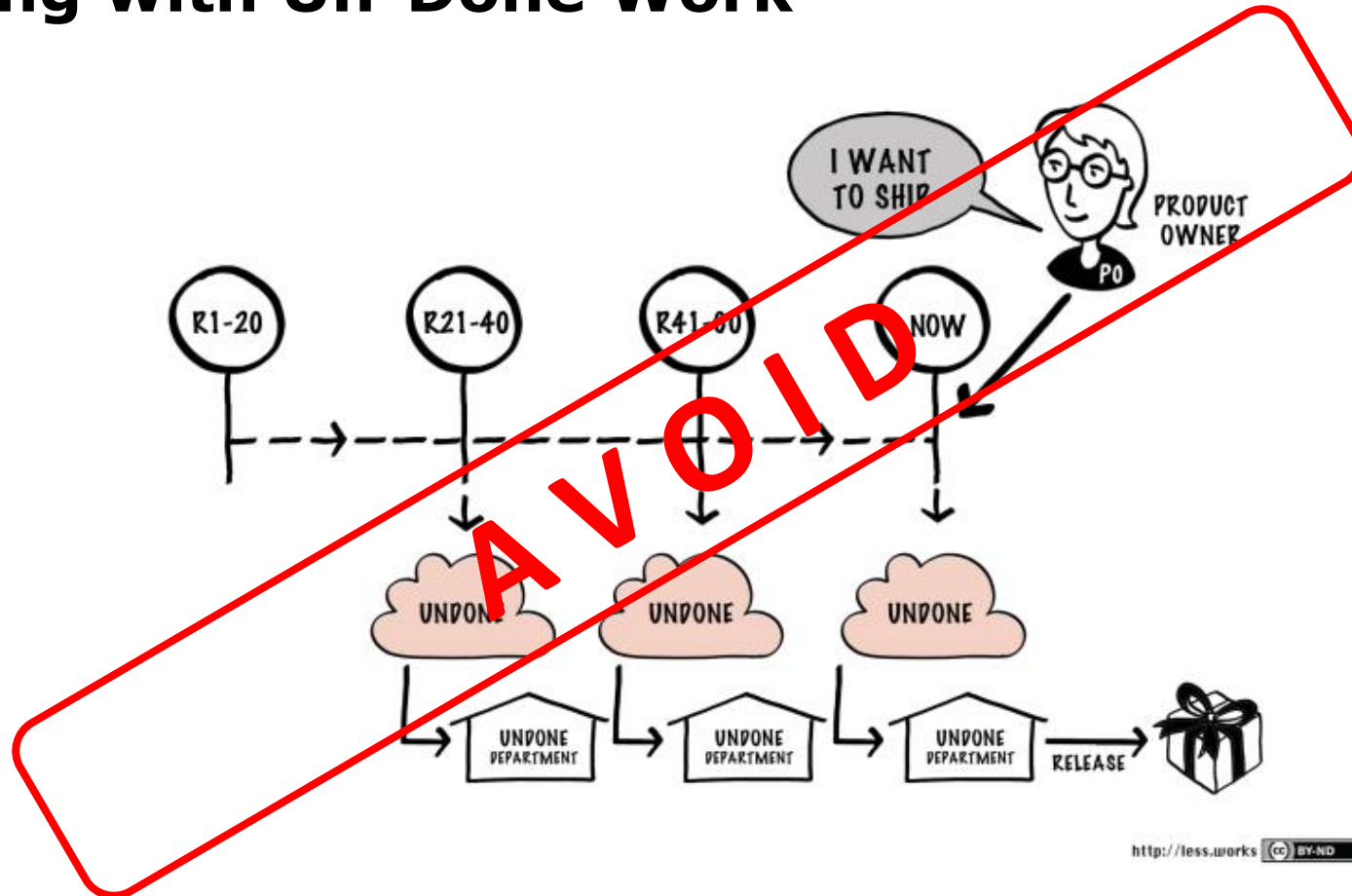
Dealing with Un-Done Work



Release Sprints

Technical Excellence

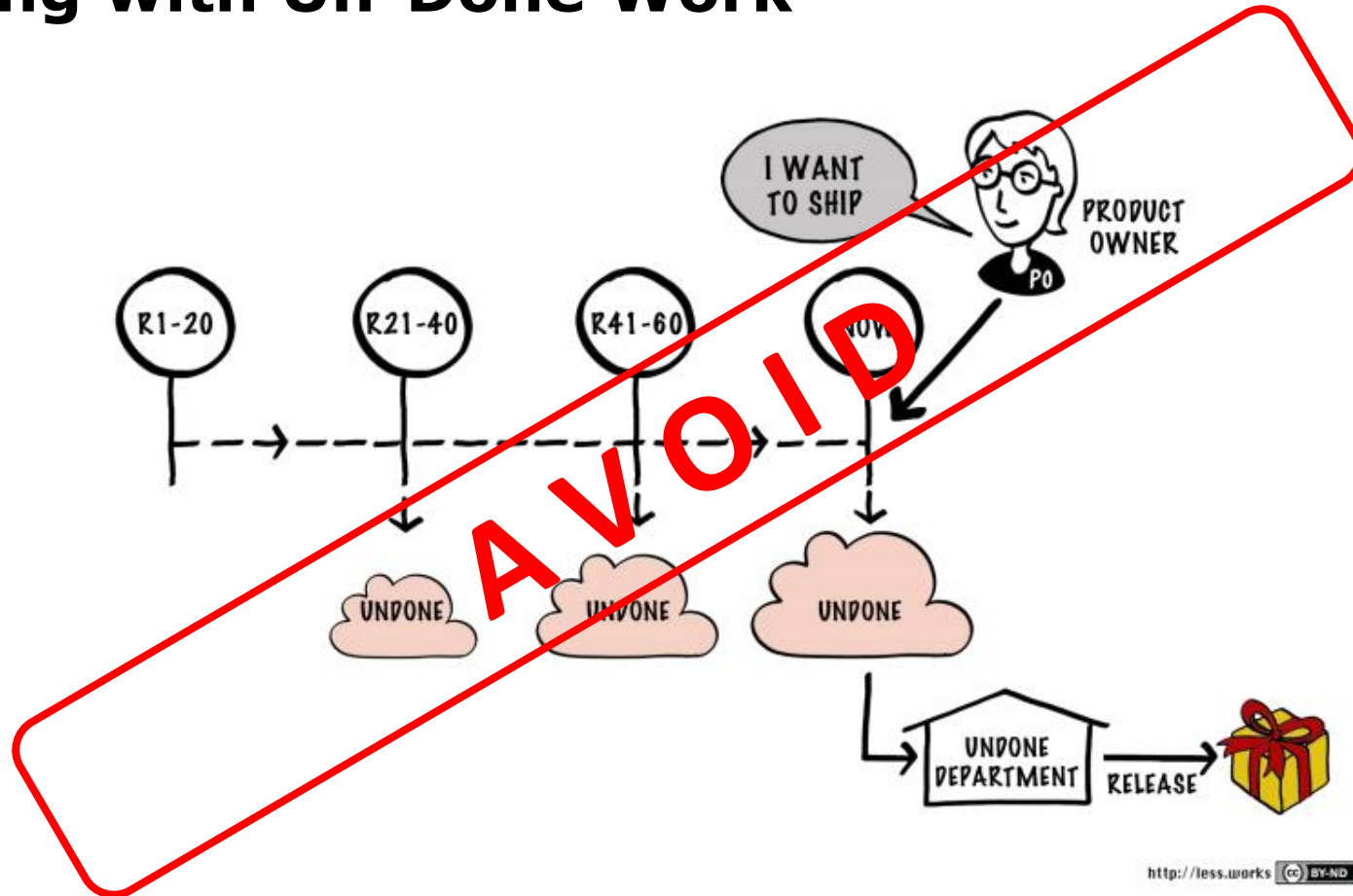
Dealing with Un-Done Work



Pipelining to Undone Department

Technical Excellence

Dealing with Un-Done Work



Undone Department Finalizes

Technical Excellence

**REMOVE
UNDONE
“DEPARTMENT”**

Managers—While there is an imperfect Definition of Done, the Definition of Done is *the major tool* for monitoring and managing organizational change. Expanding the Definition of Done leads to organizational changes and strategic decisions and are usually the responsibility of managers.

Scrum Masters—Not expanding the Definition of Done is a sign of not improving. Scrum Masters are responsible for building teams that are self-managing and continuously improving, and Scrum Masters are responsible for helping the organization to improve.

Maturity
of DoD

19

Customers
/ Users
Happiness

20

[Sourced from: 2009 - Scaling Lean and Agile Development - Craig Larman, Bas Vodde](#)

Exercise

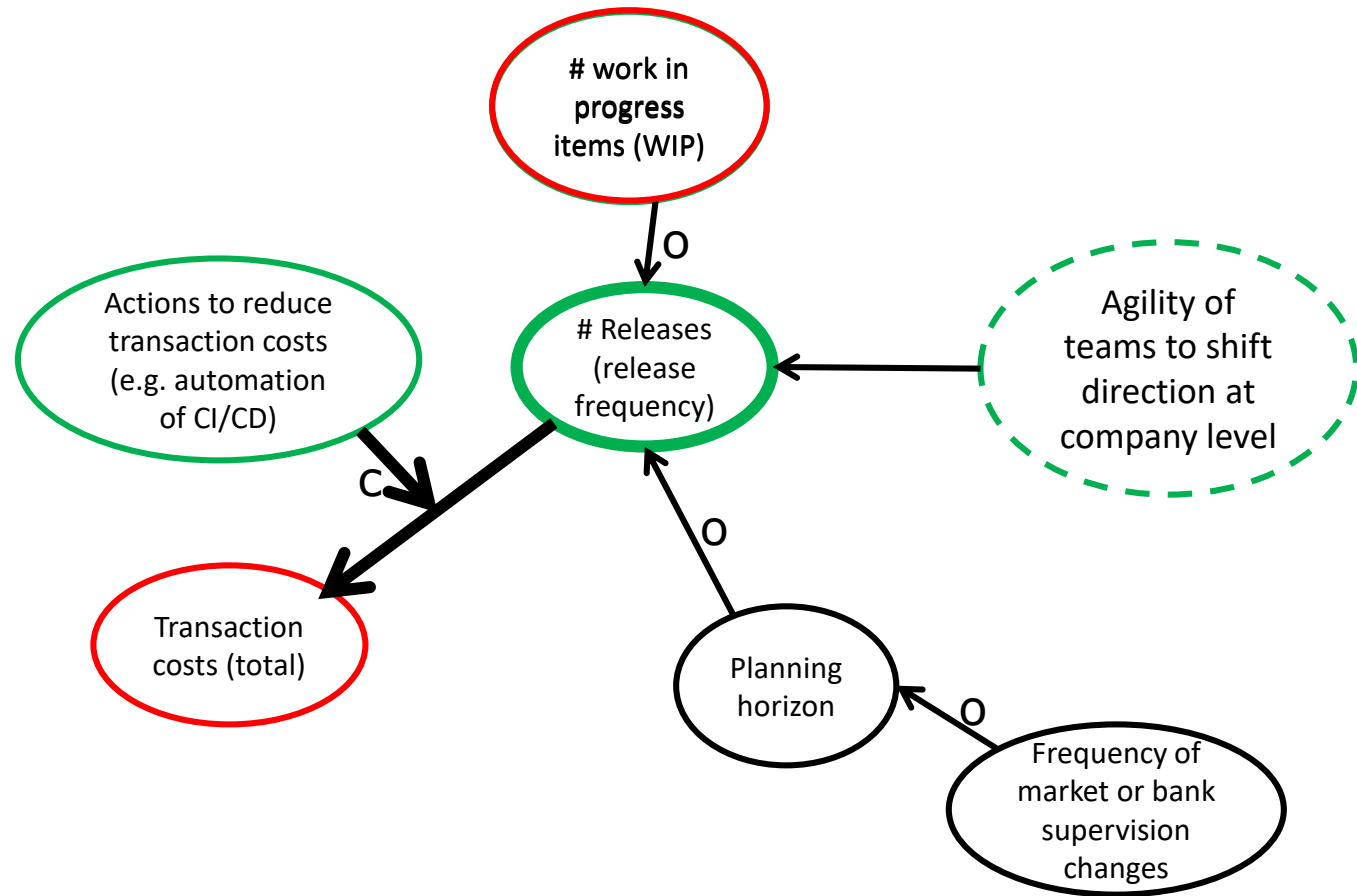
Duration: 10 min

Class: in-groups, brainstorm some of the most common examples of ***Local Optimization in Releasing*** in your respective organizations. Work with provided system variables to create a model.

Method/Tool: System Modeling with CLD

Instructor: review with Class

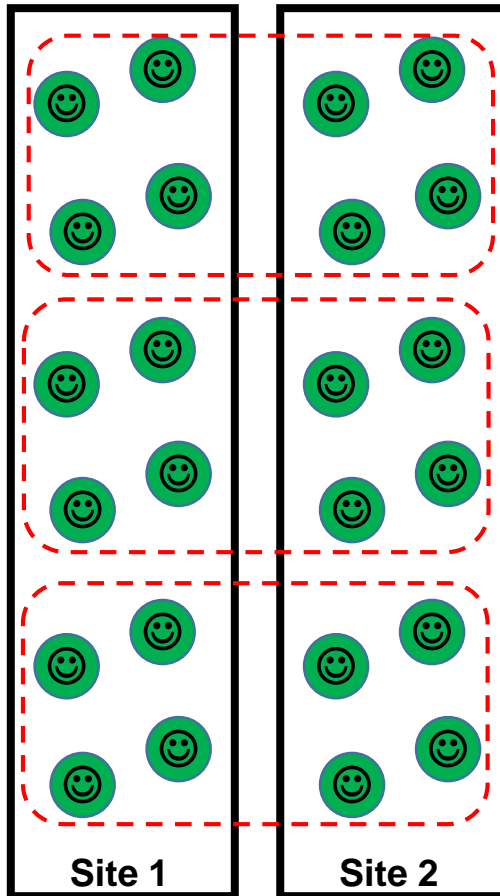
Local Optimization in **Releasing** - Exercise



Multi-Site /Time Zone

Team Location in LeSS

Distribution inside team

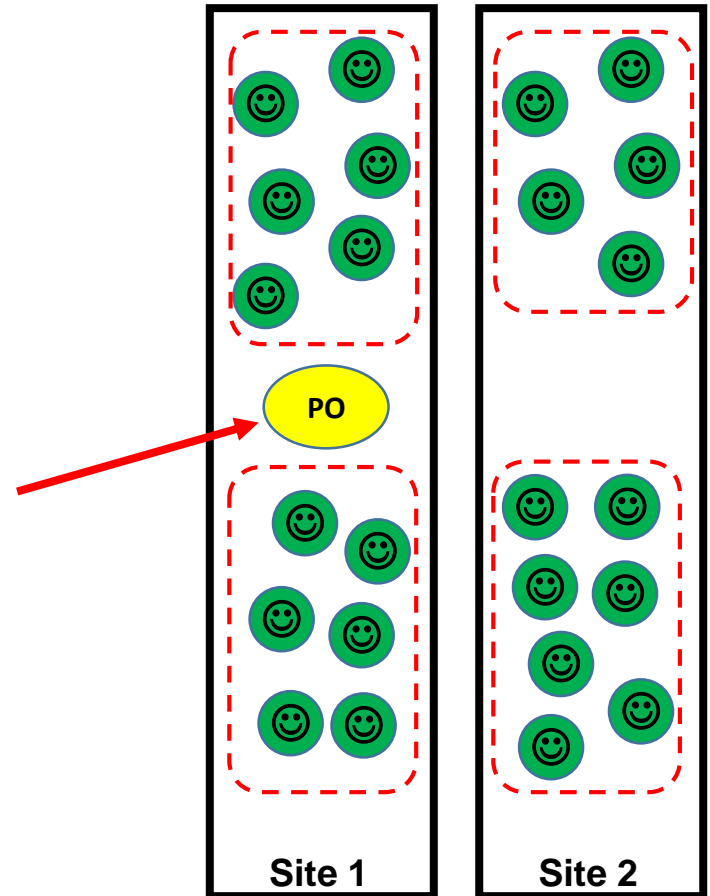


“All good Java developers are in Boston”...“The best SQL experts are in Bangalore”

Really? 😊

AVOID

Team colocation within site



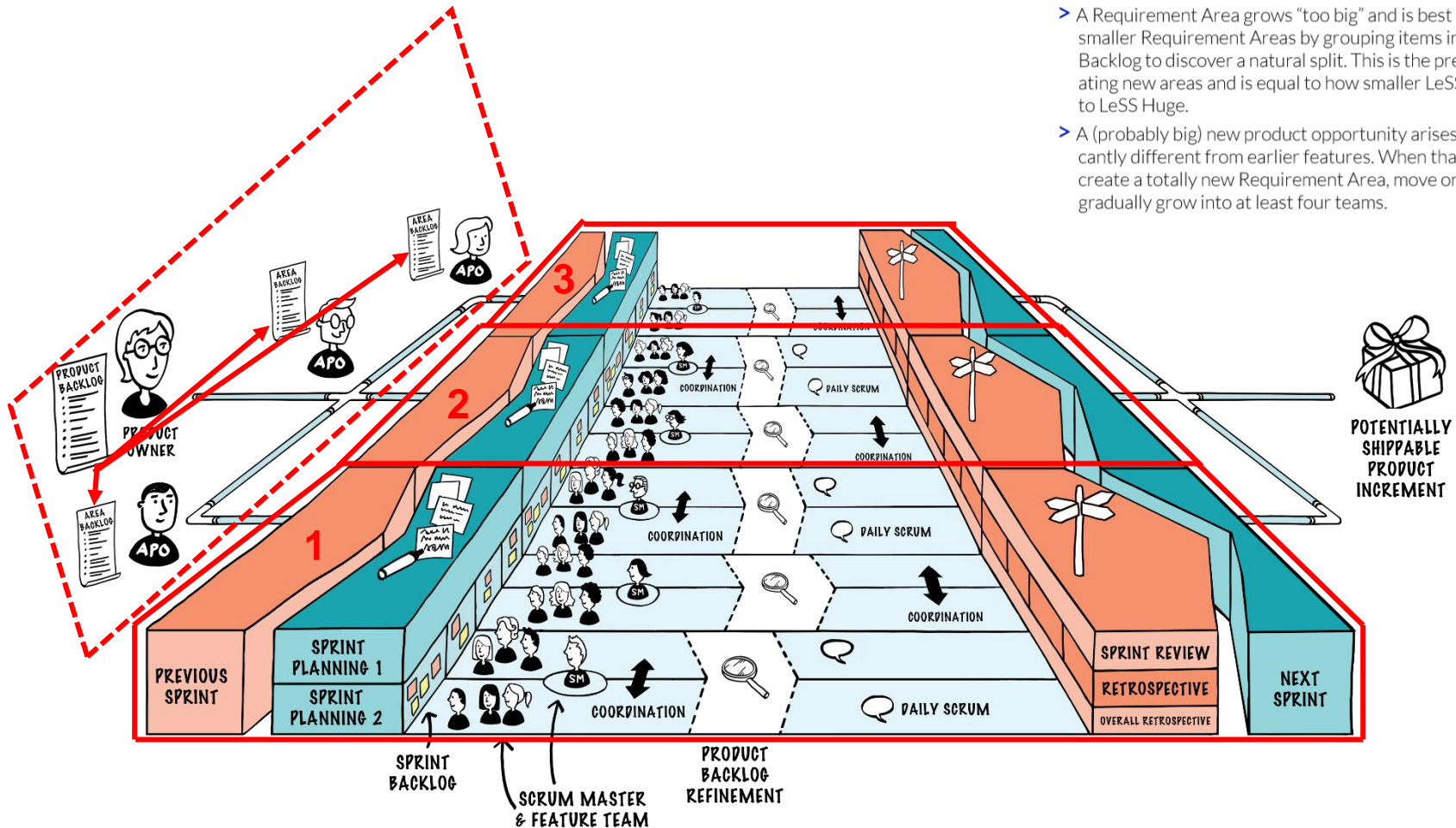
TRY

LeSS Huge

LeSS Huge

Birth—There are **two ways in which new Requirement Areas are born**:

- > A Requirement Area grows “too big” and is best split into two smaller Requirement Areas by grouping items inside the Area Backlog to discover a natural split. This is the preferred way of creating new areas and is equal to how smaller LeSS adoptions grow to LeSS Huge.
- > A (probably big) new product opportunity arises⁷ that is significantly different from earlier features. When that happens, we can create a totally new Requirement Area, move one team into it and gradually grow into at least four teams.



<http://less.works>

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

- Avoid having multiple Requirement Areas for as long as possible
- Any product backlog item belongs to one Product Area only

Where To Start?

Getting Started

The three adoption principles include that it's best to start with one product adoption. How can you increase the likelihood of its success?

1. educate everyone
2. define 'product'
3. define 'done'
4. have appropriately-structured teams
5. only the Product Owner gives work to the teams
6. keep project managers away from the teams

<https://less.works/less/adoption/getting-started>

Q & A

What Is Next?

What is Next?

