

# LeSS (Large Scale Scrum) Rules

from <http://less.works> as of April 2016

## LeSS Structure

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

### Teams

- Structure the organization using real teams as the basic organizational building block.
- Each team is (1) self-managing, (2) cross-functional, (3) co-located, and (4) long-lived.
- The majority of the teams are customer-focused feature teams.

### Scrum Masters

- Scrum Masters are responsible for a well-working LeSS adoption. Their focus is towards the Teams, Product Owner, organization, and development practices. A Scrum Master does not focus on just one team but on the overall organizational system.
- A Scrum Master is a dedicated full-time role.
- One Scrum Master can serve 1-3 teams.

### Managers

- In LeSS, managers are optional, but if managers do exist their role is likely to change. Their focus shifts from managing the day-to-day product work to improving the value-delivering capability of the product development system.
- Managers' role is to improve the product development system by practicing Go See, encouraging Stop & Fix, and “experiments over conformance.”

### Adoption

- For the product group, establish the complete LeSS structure “at the start”; this is vital for a LeSS adoption.
- For the larger organization beyond the product group, adopt LeSS evolutionarily using Go and See to create an organization where experimentation and improvement is the norm.

## LeSS Product

There is one Product Owner and one Product Backlog for the complete shippable product.

The Product Owner shouldn't work alone on Product Backlog refinement; he is supported by the multiple Teams working directly with customers/users and other stakeholders.

All prioritization goes through the Product Owner, but clarification is as much as possible directly between the Teams and customer/users and other stakeholders.

The definition of product should be as broad and end-user/customer centric as is practical. Over time, the definition of product might expand. Broader definitions are preferred.

One Definition of Done for the whole product common for all teams.

Each team can have their own stronger Definition of Done by expanding the common one.

The perfection goal is to improve the Definition of Done so that it results in a shippable product each Sprint (or even more frequently).

## LeSS Sprint

There is one product-level Sprint, not a different Sprint for each Team. Each Team starts and ends the Sprint at the same time. Each Sprint results in an integrated whole product.

### Sprint Planning

- Sprint Planning consists of two parts: Sprint Planning One is common for all teams while Sprint Planning Two is usually done separately for each team. Do multi-team Sprint Planning Two in a shared space for closely related items.
- Sprint Planning One is attended by the Product Owner and Teams or Team representatives. They together tentatively select the items that each team will work on the that Sprint. The Teams identify opportunities to work together and final questions are clarified.
- Each Team has their own Sprint Backlog.
- Sprint Planning Two is for Teams to decide how they will do the selected items. This usually involves design and the creation of their Sprint Backlogs.

### Sprint Execution

- Each Team has their own Daily Scrum.
- Cross-team coordination is decided by the teams. Prefer decentralized and informal coordination over centralized coordination. Emphasize Just Talk and informal networks via communicate in code, cross-team meetings, component mentors, travelers, scouts, and open spaces.

### Backlog Refinement

- Product Backlog Refinement (PBR) is done per team for the items they are likely going to do in the future. Do multi-team and/or overall PBR to increase shared understanding and exploiting coordination opportunities when having closely related items or a need for broader input/learning

### Sprint Review

- There is one product Sprint Review; it is common for all teams. Ensure that suitable stakeholders join to contribute the information needed for effective inspection and adaptation.

### Sprint Retrospective

- Each Team has their own Sprint Retrospective.
- An Overall Retrospective is held after the Team Retrospectives to discuss cross-team and system-wide issues, and create improvement experiments. This is attended by Product Owner, Scrum Masters, Team representatives, and managers (if any).

# LeSS Huge Rules

from <http://less.works> as of April 2016

# Why I Like LeSS

by Michael James, Scrum Alliance CST

## LeSS Huge Structure

LeSS Huge applies to products with “8+” teams. Avoid applying LeSS Huge for smaller product groups as it will result in more overhead and local optimizations.

All LeSS rules apply to LeSS Huge, unless otherwise stated. Each Requirement Area acts like the basic LeSS framework.

### Requirements Areas

- Customer requirements that are strongly related from a customer perspective are grouped in Requirement Areas.
- Each Team specializes in one Requirement Area. Teams stay in one area for a long time. When there is more value in other areas, teams might change Requirement Area
- Each Requirement Area has one Area Product Owner.
- Each Requirement Area has between “4-8” teams. Avoid violating this range.

### Incremental Adoption

- LeSS Huge adoptions, including the structural changes, are done with an evolutionary incremental approach.
- Remember each day: LeSS Huge adoptions take months or years, infinite patience, and sense of humor.

## LeSS Huge Product

### Product Owner

- Each Requirement Area has one Area Product Owner.
- One (overall) Product Owner is responsible for product-wide prioritization and deciding which teams work in which Area. He works closely with Area Product Owners.
- Area Product Owners act as Product Owners towards their teams.

### Product Backlog

- There is one Product Backlog; every item in it belongs to exactly one Requirement Area.
- There is one Area Product Backlog per Requirement Area. This backlog is conceptually a more granular view onto the one Product Backlog.

## LeSS Huge Sprint

There is one product-level Sprint, not a different Sprint for each Requirement Area. It ends in one integrated whole product.

The Product Owner and Area Product Owners synchronize frequently. Before Sprint Planning they ensure the Teams work on the most valuable items. After the Sprint Review, they further enable product-level adaptations.

## De-Scale Your Organization

Scrum was relatively easy for my first team in a small company, but I've usually been disappointed with the Scrum implementations I've seen in larger, more established organizations. They fall short of what the people in the organizations are capable of because the far-reaching implications of Scrum and Agility aren't grasped.

If your organization has more than 12 people, optimizing it to continuously adapt to reality is going to be a lot of difficult work with discouraging setbacks. There are two fundamentally different ways I've seen Scrum used in organizations. The first is the most common. The second is what LeSS tries to achieve:

### Pretending To Be Agile

Scrum is the “implementation” layer underneath a hierarchy of defined roles, which may loosely correspond to the original org structure (despite different names). There are potentially many people called “Product Owner,” often middle managers who would have been called some flavor of PM the week before. Various methods resembling traditional project/program management are used to synchronize and align them. While lip service is paid to empowerment, coordination is mostly outside the teams and self organization is mostly restricted within teams. I hear language like “my Jira ticket.” “Scrum Masters” are not expected to transform the outer organization, but they might act as coordinators since they were probably PMs before anyway. Coaches try in vain to “change the culture” of the organization.

### Being Agile

A real Product Owner sets priorities (in \*one\* Product Backlog, per the Scrum Guide) and vision, and it's team self organization underneath that, even if there are multiple development teams. Previously unthinkable responsibilities eventually move into the realm of team self organization. The teams strive to be truly cross functional feature teams, as connected to real customers as possible. TDD skills, colocation, and collaboration are paramount. Teams of teams self organize and collaborate across team boundaries in one Sprint, with one integrated product at the end of each Sprint. Teams own their processes, methods, and tools.



**LeSS Rule:** A Scrum Master does not focus on just one team but on the overall organizational system.