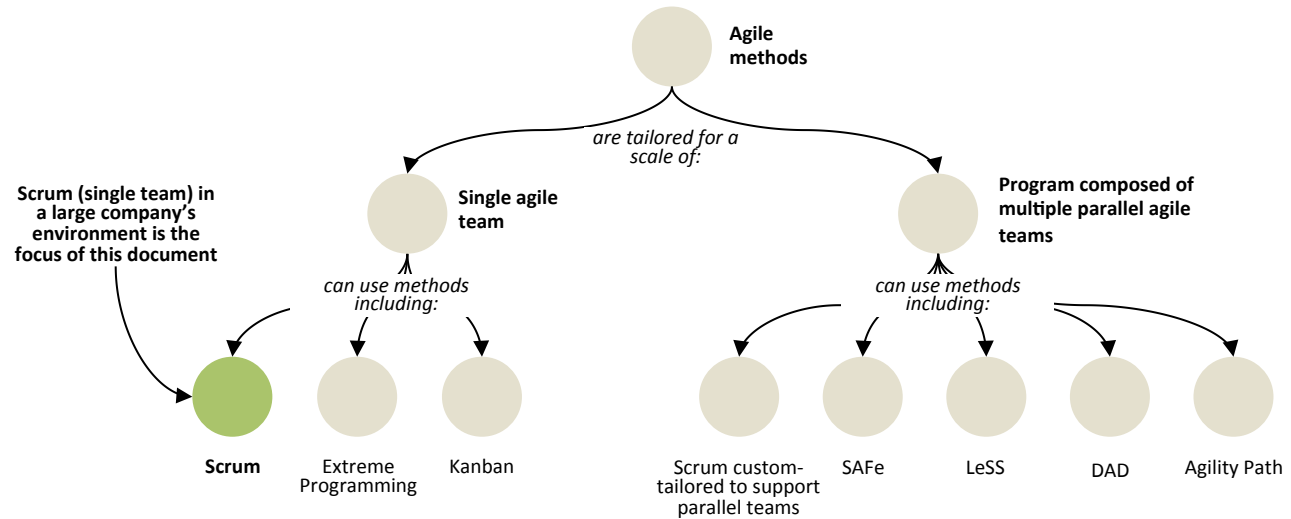


Scrum in large companies concept map

scope of this document

We are writing about Scrum in large organizations.

- Scrum is just one of agile methods that can be used to organize development projects. We focus on Scrum because of its popularity; however, if you are using other agile method, much of the document's content will probably still apply
- Scrum is not a method of managing whole programs – if you wish to manage a program of multiple agile projects linked through interdependencies, you must either make custom modifications to scale Scrum (some public case studies exist, e.g. Spotify), or use one of the already defined frameworks such as SAFe – they are not the topic of this document
- Some practices that we describe result from requirements of large company's environment and are not part of Scrum (e.g. practical need of making and tracking estimates for budgeting)



Is agile scalable?

- In order to manage parallel teams effectively, adjustments need to be made to address obvious non-scalable elements of a single team Scrum (or other method)
- Among other difficulties, large scale programs have more than one Product Owner for the whole solution, teams need to work together on interdependencies and program progress should be tracked as a whole
- The simplest case of scaling beyond one team is a single Product Owner managing single backlog developed by more than one team, with Scrum Master tasked with managing interdependencies

- In order to scale further, one approach is to organically accrue adjustments to scale the method that had already been proven on a smaller scale, while another would be to reuse one of the scaled agile frameworks listed on the map above
- Typical measures to scale Scrum organically include introducing new practices to coordinate product development (e.g. Scrum of Scrums), and assigning new roles to ensure one point of ownership of certain areas (e.g. system owners) to coordinate changes to respective applications

SAFe

- SAFe is a quite elaborate framework, which on one hand might feel appropriate in some corporate environments, but on the other hand, some agile practitioners feel it moves too far from agile principles (e.g. self-organization) to appease management's need for control
- The framework has three levels:
 - Team, which is organized similarly to Scrum teams
 - Program, which collects increments into a common release called Train
 - Portfolio, which is responsible for strategic vision, high-level requirements, architecture and budget allocation

LeSS

- On the other hand LeSS (Large-Scale Scrum) is a framework considered relatively lightweight compared to SAFe
- The authors try to keep adjustments required to scale Scrum only to a necessary minimum, so as not to compromise the spirit of agile development
- There are two variations of the framework – for smaller programs (up to 10 Scrum teams), and for even larger initiatives
- Marketing and formal framework (certification) are at the moment of writing not well developed

Scrum in large companies readiness checklist

Based on our experience in working for large IT organizations we have collected a list of checkpoints that we understood as critical for successful Scrum introduction.

The checklist cannot capture complexity of every organizational context, but it is useful for a quick check to avoid common pitfalls.

why is it more difficult..

..to implement Scrum in a large company than in a small one?

- executives are used to working in silos and to **finger-pointing at counterparts** (IT vs. business) when something goes wrong, rather than working together
- people are reluctant to take on tasks '**out of their job description**', whereas in Scrum team members are multidisciplinary
- navigating organizational structure and parallel projects is **often more complex than the product itself** and requires effective communication rather than technical skills
- programmers must work with **legacy frameworks and practices** adopted by corporate IT organization, which adds to the learning curve and slows down the development

1. team composition

not just programmers: get the right mix of profiles.

- a. there is one clearly defined Product Owner
- b. Product Owner is empowered to make all decisions regarding scope and budget
- c. Product Owner is either dedicated to the project as much as it is required, or there is relevant support from proxy Product Owner (e.g. business analyst)
- d. there is a Scrum Master fully committed to the project
- e. there is at least one team member with significant experience in agile projects to provide knowledge and coaching
- f. development team is fully dedicated to the project and membership does not change unless it is truly required

2. team preparation

familiarize all members with their project roles.

- a. all team members, including Product Owner, had basic Scrum training
- b. Product Owner was made aware that initial reviews would be far from the final target solution
- c. Product Owner was trained to continuously monitor the project budget and timeline
- d. both IT and business team members describe project as 'our' rather than 'their'
- e. all team members are encouraged to contribute in all types of tasks (analysis, development, tests)
- f. Scrum Master is not appointed as project manager, but as a team member whose role is to remove project obstacles

3. project preparation

in complex environment, some things are better prepared in advance.

- a. all team members are on site and if not, efficient technical means exist for telepresence
- b. there is a separate project room for the team on site
- c. product backlog is created before the first sprint
- d. there is a high-level concept/design/architecture of the solution that is about to be developed
- e. software environments, including the one for sprint reviews, are set up before start of the project
- f. there is a dedicated tool to manage product backlog available to all team members, including Product Owner

4. project delivery

ensure that crucial practices are observed.

- a. product backlog is continuously reviewed by the Product Owner
- b. every sprint ends with sprint review
- c. Product Owner participates in sprint planning and reviews
- d. during sprint planning all user stories are given estimates
- e. estimates are compared against actuals: all team members log their time spent
- f. either Product Owner or proxy Product Owner (whichever is applicable) takes part in backlog refinements to prepare the team for the next sprint planning

Scrum in large companies troubleshooting

Below are some of the typical problems that may arise during Scrum introduction, along with illustrative mitigation measures.

troubleshooting

As for general rules for troubleshooting Scrum introduction:

- introducing Scrum is more about a culture change than new methodology, therefore to solve problems first look to soft skills, avoid urge to hard-wire processes and complicate tracking tools
- remember that you introduce Scrum in order to achieve specific goals, therefore keep an eye on the goal rather than compliance with each and every Scrum guideline, and feel free to adjust its practices to your project context
- invest in intensive coaching of your team in the early stages of Scrum introduction – team attitude and commitment is the most important success factor

1. Product Owner does not fulfill his or her role – business decisions are delayed or changed, the product is not reviewed regularly

- if Product Owner cannot commit as much of his or her time as is necessary (or if it is not nearly enough even if he can dedicate 100%), delegate proxy Product Owner with capacity to provide 80% input required by developers, spare Product Owner time for critical decisions and reviews
- make sure the team contains members with business analysis background, so Product Owner is not overloaded with tasks requiring extensive analysis

2. Developers cannot get hold of knowledge and documentation (e.g. interfaces) of other systems used in the organization and are wasting time on communicating and waiting around for input

- investigate if Scrum Master is effective: his or her job is exactly to remove obstacles in front of developers, so that they can be productive; maybe Scrum Master is not 100% assigned to the project, is busy with other tasks or has not enough experience to quickly address blocking issues
- ensure that at least some team members are on site physically close to key experts outside of the team, so they can reach them quicker and obtain information in a non formal way
- verify that some team members have business analysis background, so that they can navigate organization and gain access to knowledge quicker than developers

3. Developed software is of poor technical quality, which manifests itself in common occurrence of regression bugs

- verify that high-level design of the solution as well as coding guidelines exist and are known to the team members
- make sure refactoring is included in the estimates
- make sure unit tests are included in the estimates
- if team is geographically distributed, consider delegating senior team member to work on site where there are problems with delivery
- organize resources so that manual testing can be continuous and parallel to development, as opposed to be limited to the last days before a review or release

4. Sprint reviews do not happen, either because 'there are not enough new things' in the solution or there are organizational issues (environment is not working, Product Owner is not available)

- ensure the team has split business requirements into user stories small enough they take maximum one sprint to produce
- check that there exists a stable software environment to show the work that has been finished
- verify if the Product Owner is in close collaboration with developers, meaning he or she takes active part in solution review

[all 13 issues are available in the full edition, contact us for a complimentary copy.](#)

Scrum in large companies [contact us](#)

get full edition

if you are introducing Scrum in your company, email us to receive complimentary copy of the full edition:

contact@goldenberry.eu

Full edition contains among others:

- remaining troubleshooting items
- Sprint 0 description
- Scrum Master preferred profile
- Product Owner preferred profile
- comparison of Scrum elements in large vs. small company
- examples of tools to be used with Scrum

about goldenberry

Goldenberry is a management and IT consulting company.

We specialize in supporting large organizations in their most challenging initiatives, including IT transformation.

Need additional assistance for your Scrum project? Contact us to talk with one of our experts and discuss how we could help.

authors

Dina Dąbrowska associate

dina.dabrowska@goldenberry.eu

Bartłomiej Owczarek partner

bartlomiej.owczarek@goldenberry.eu

document version 1.1, October 2014

Metropolitan
Pl. Piłsudskiego 3
00-078 Warsaw

tel. +48 22 449 00 50

Berkeley Square House, 2nd floor
Berkeley Square
W1J 6BD London

tel. +44 20 7887 6067

www.goldenberry.eu