

Module: Scaling Scrum with Distributed Teams

It is not the ship so much as the skillful sailing that assures the prosperous voyage.

— George William Curtis

7/27/2021 11:45 AM

vijaynathani.github.io

Page 1

Culture changes cause a drop in productivity. Beware of cultural differences.

=====

An American developer will say to the client, "This doesn't make sense, you idiot"

An Indian developer will say to someone other than the client, "This doesn't make sense, but this is how the client wants it"

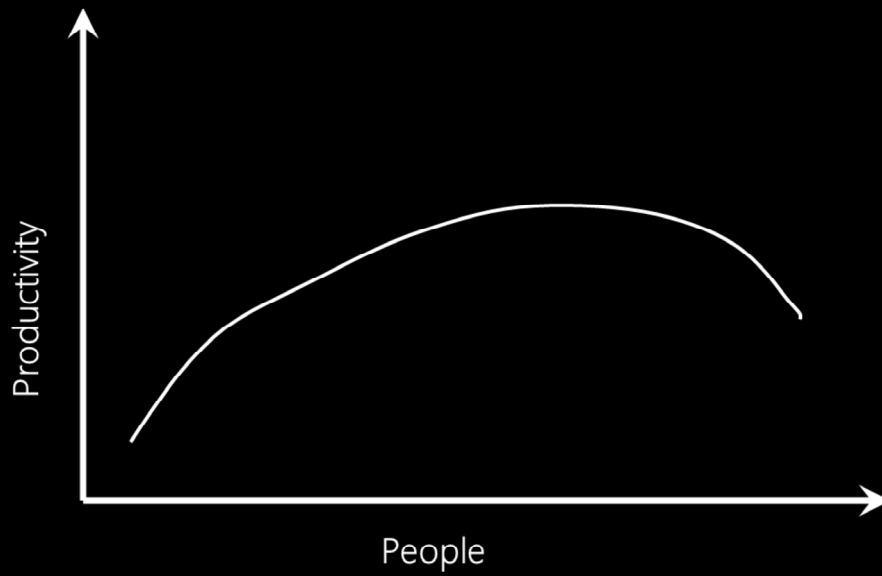
=====

An American developer says that this job is not possible in this time period

An Indian developer never says no. He either doesn't do the job or does a poor quality job.

=====

People vs Productivity

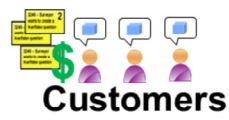


What is Scaled Scrum?

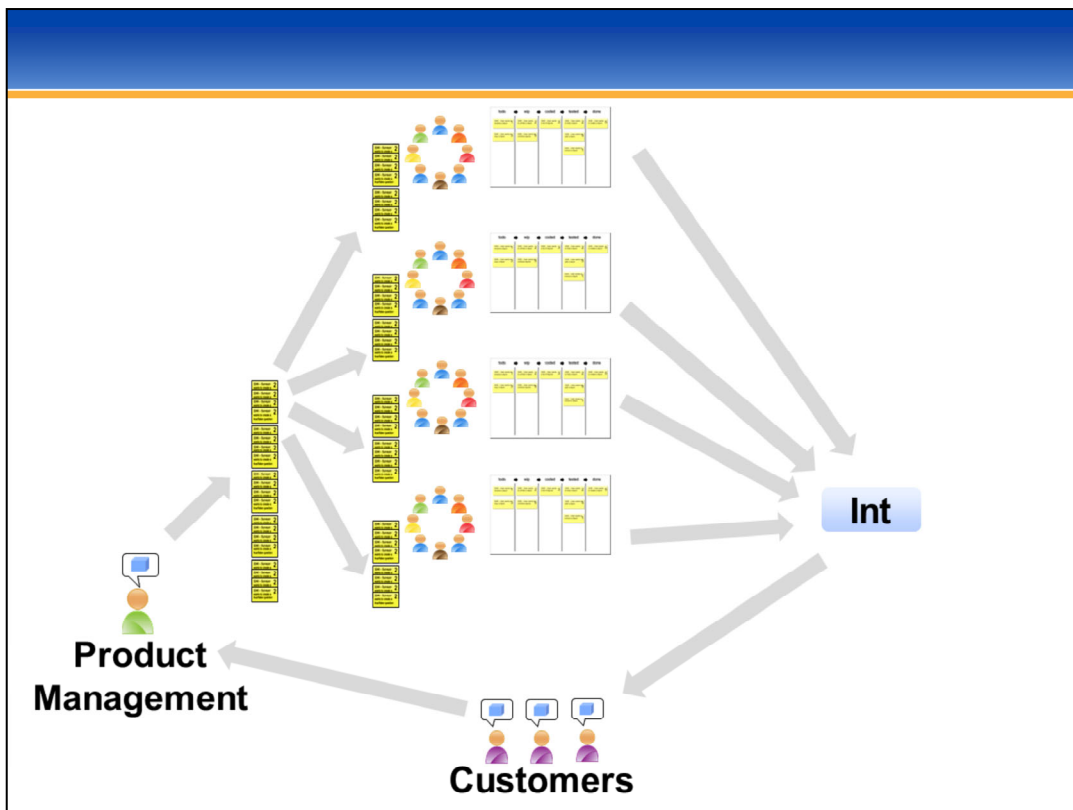
Multiple Scrum Teams building or enhancing

- One product
 - Multiple related products
- in one or more sprints.

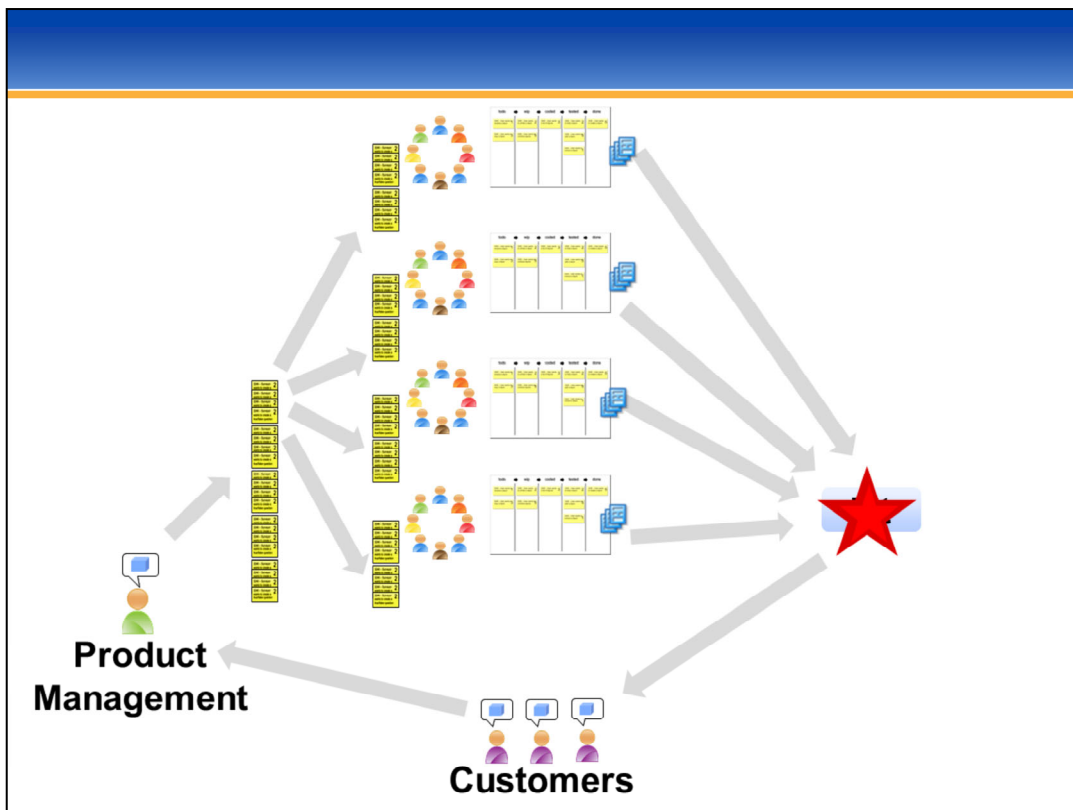
Providing Value as Fast as Possible



Do as little as possible with the highest possible value as fast as possible
What do we do that is like this today? Hot fixes! (patches)

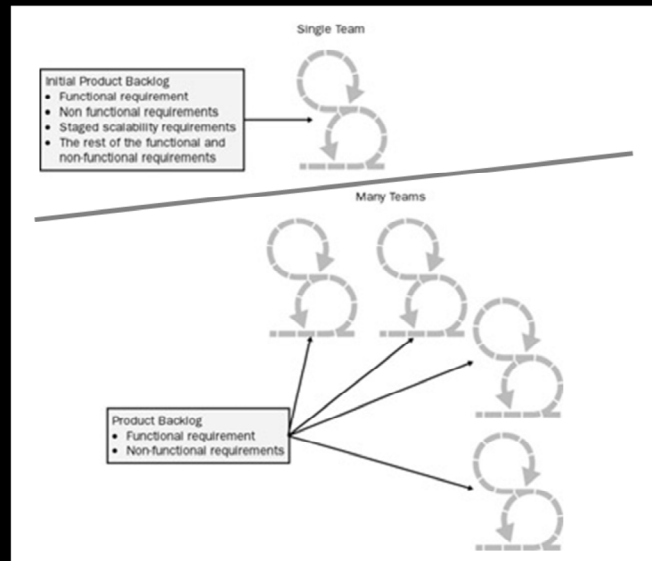


Scrum is fairly straight forward when you have just one team, the challenge starts to increase as you have multiple interdependent teams.



Scrum is fairly straight forward when you have just one team, the challenge starts to increase as you have multiple interdependent teams.

Scaling Scrum for Large Projects



7/27/2021 11:45 AM

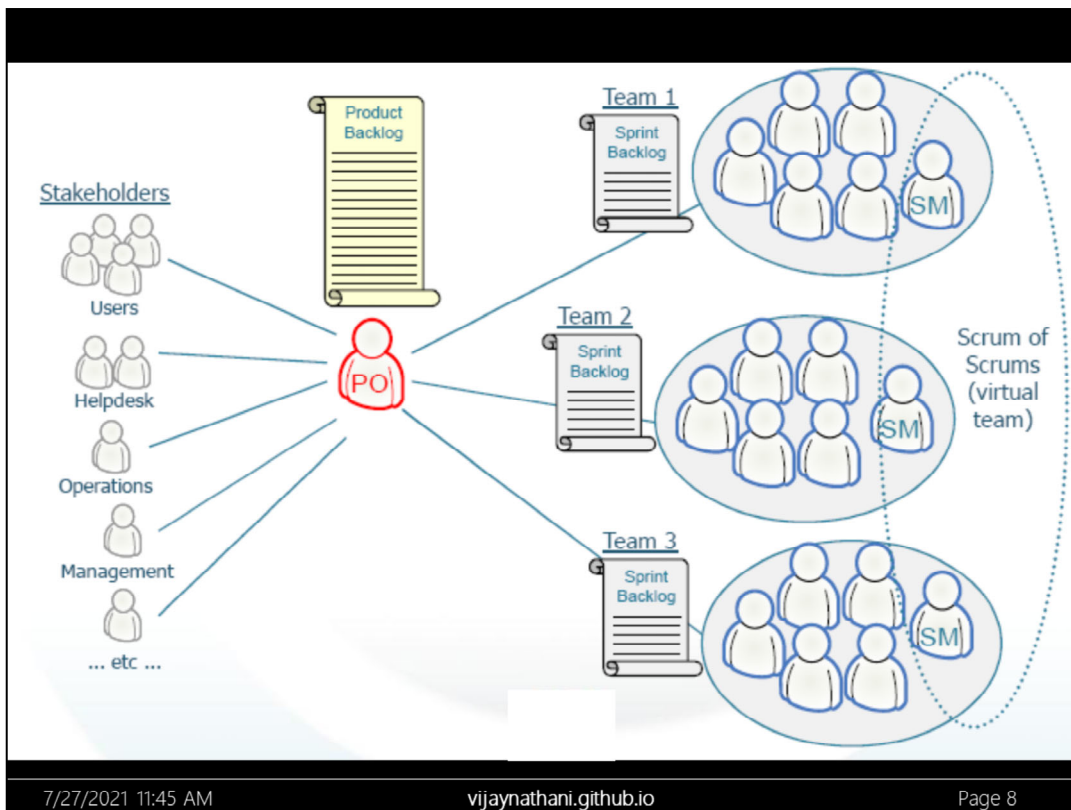
vijaynathani.github.io

Page 7

The scrum of scrums meeting is an important technique in scaling Scrum to large project teams. These meetings allow clusters of teams to discuss their work, focusing especially on areas of overlap and integration. Imagine a perfectly balanced project comprising seven teams each with seven team members. Each of the seven teams would conduct (simultaneously or sequentially) its own daily scrum meeting. Each team would then designate one person to also attend a scrum of scrums meeting.

The decision of who to send should belong to the team. Usually the person chosen should be a technical contributor on the team—a programmer, tester, database administrator, designer, and so on—rather than a product owner or ScrumMaster. Being chosen to attend the scrum of scrums meeting is not a life sentence. The attendees should change over the course of a typical project. The team should choose its representative based on who will be in the best position to understand and comment on the issues most likely to arise at that time during a project. For instance, early in a project the issues brought up at the scrum of scrums meeting may center mostly on technical issues or user experience design. Teams may opt to send someone strong in one of those areas to those early meetings. Later, issues may center around how to collaborate on testing, and so testers will be the more likely participants. If the number of teams participating is small, it may be acceptable for each team to send two representatives—a technical contributor, as described above, and a ScrumMaster—if the teams desire. I tend to do this only when there are four or fewer teams, which keeps the meeting size to eight or less.

The scrum of scrums meetings can be scaled up in a recursive manner. Imagine there are seven scrum of scrums meetings occurring in your organization. Each contains a representative from each of seven teams (as in the previous example). The work of the seven scrum of scrums meetings can be coordinated through an even higher level meeting: what might be called a scrum of scrum of scrums. (It isn't usually called this, though, because things start to sound a bit silly at some point. Scrum of scrums often suffices even for these higher levels of meeting.)

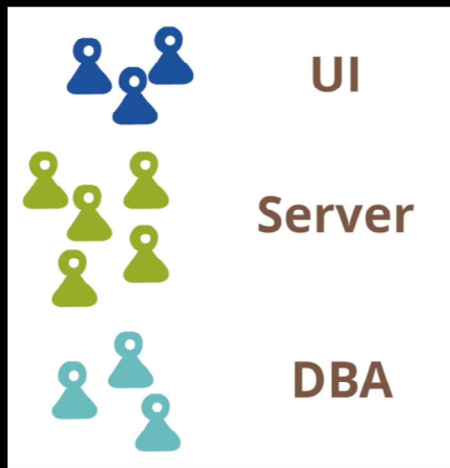


7/27/2021 11:45 AM

vijaynathani.github.io

Page 8

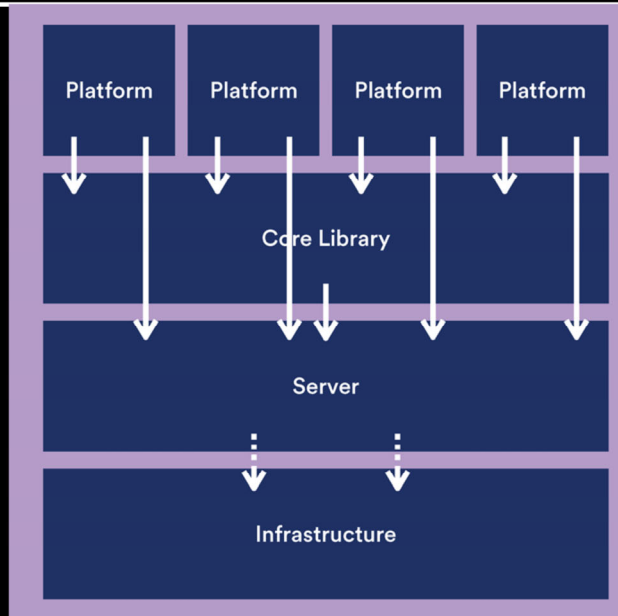
Team Organization



Which is better?



At Spotify – initial plan



7/27/2021 11:45 AM

[vijaynathani.github.io](https://github.com/vijaynathani)

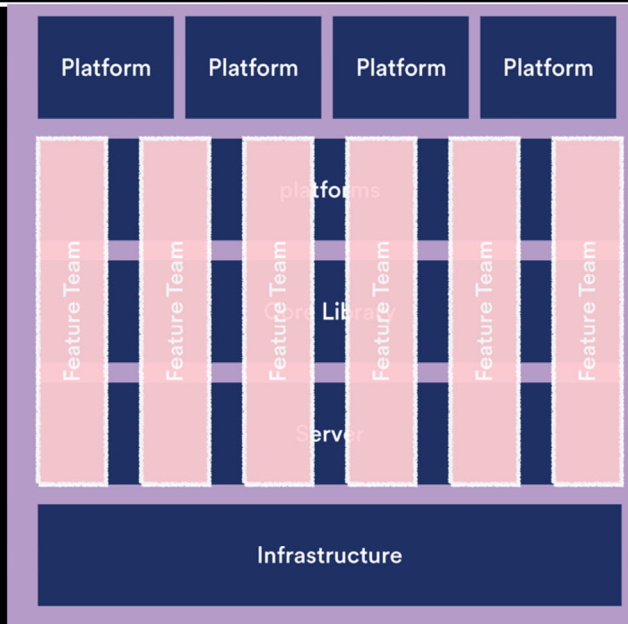
Page 10

Spotify is an online music company.

Initially, they were thinking of having one set of teams for core library, one set of teams for server, one or more team for each platform.

The idea was rejected, because it would result in lot of inter-team communication. New features will not be delivered fast.

At Spotify - actual



7/27/2021 11:45 AM

vijaynathani.github.io

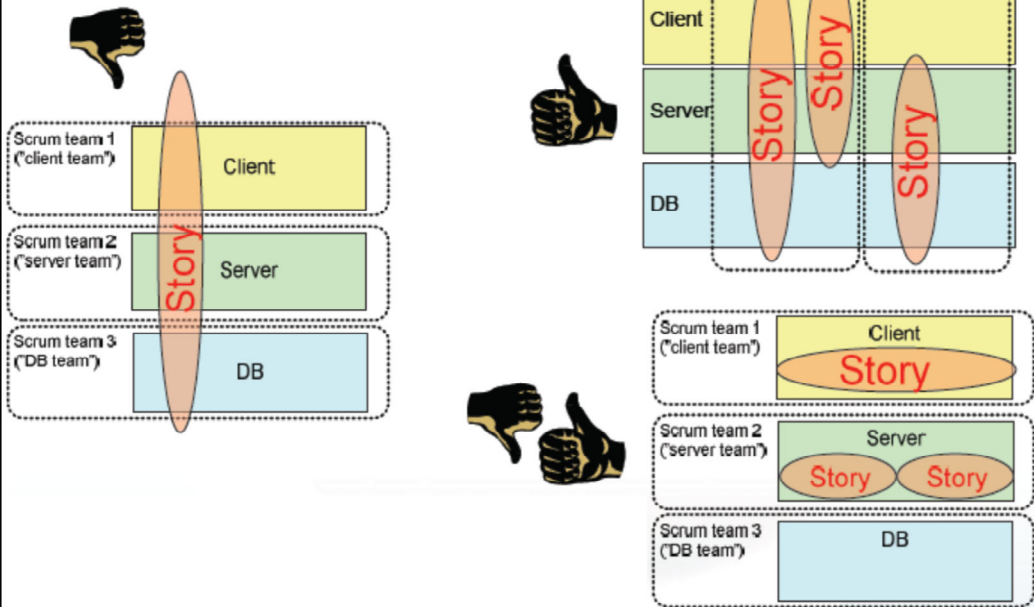
Page 11

To reduce cycle time, Spotify has feature teams. Each Feature team manages a group of micro services. Each team has all skills to convert feature to production grade.

Each team is also responsible for deployment of its microservice.

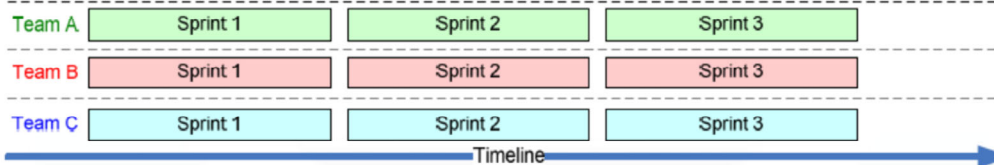
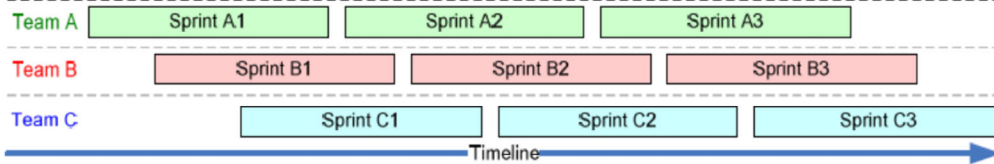
Scaling - multiple teams

Avoid splitting a single story between multiple teams

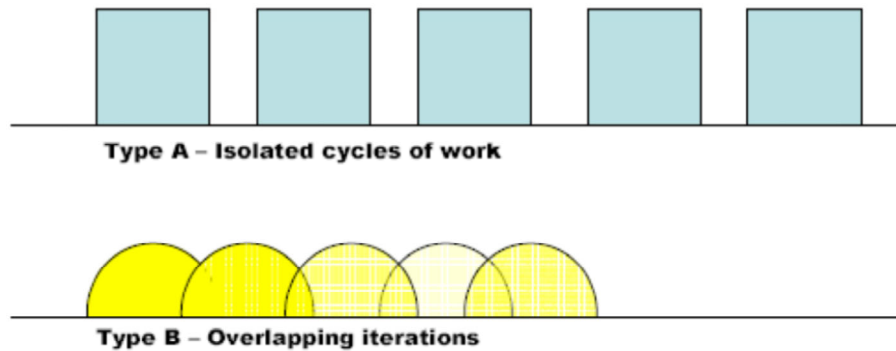


Coordinated Sprints

Scaling - simultaneous sprints



Types of Scrum



7/27/2021 11:46 AM

vijaynathani.github.io

Page 14

Type C Sprint, often called Continuous Sprint, can be envisioned as pipelining Sprints by running multiple overlapping Sprints through the same Scrum team. Sprints themselves are overlapped within the same teams. This requires experienced Scrum teams, well designed product architecture, and automation of product and Sprint backlogs. Throughput can be enhanced to deliver dozens of new releases of enterprise software annually. Competitors can be overwhelmed and market dominance achieved.

Use Type A -> Use Type B -> Consider going to Type C.

Examples of Type C:

- Client is not sure whether to use HTML browser-based or Swing based front end.
- Camera manufacturer has 2 algorithms for red-eye reduction. One algorithm will take finish before the marketing date of the camera but does not give very good results. The second algorithm gives very good results but may not finish before the marketing date of the camera.

Different branches e.g. for different models of Nokia is not type C scrum.

Scrum in a Distributed Environment

- Is Distributed Development using Scrum even possible?
- Yes, but...

7/27/2021 11:46 AM

vijaynathani.github.io

Page 15

"If you are going to hire a group of developers halfway around the world in a time zone 12 hours away, how should you work with them? Should you give them a huge requirements document and wait to see what they give you in six months? Or should you ask for something every week?" When it's difficult to communicate, you need to shorten the feedback cycle and measure results frequently."

Scrum emphasizes face-to-face communication, physical proximity, minimizing documentation, cross-functional teams...

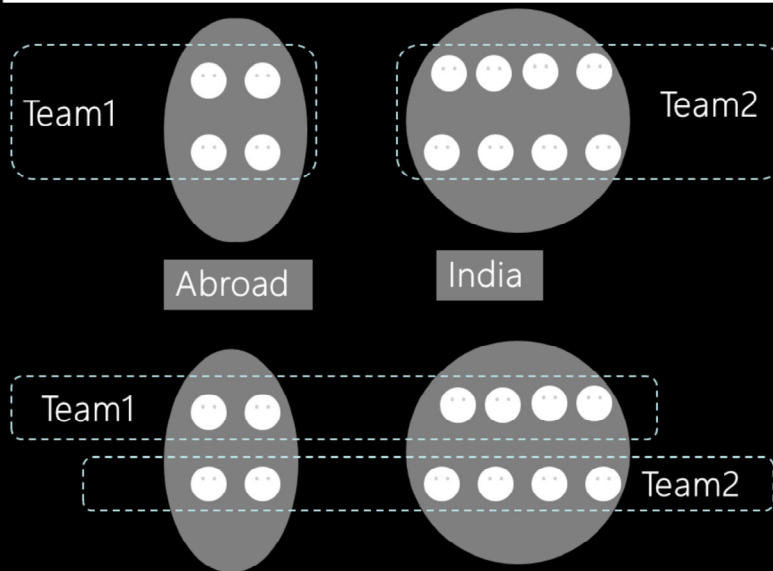
Distributed development seems to attract plan-driven approach, detailed requirements and "defined" plans, separation by function (i.e., analysts onshore, developers offshore, etc.)

Scrum can be used in a distributed environment, but

Distributed Scrum is not as good as Colocated Scrum

However, it's still better than Distributed Waterfall

Which is better Global Team?



7/27/2021 11:46 AM

vijaynathani.github.io

Page 16

The first will cause significant communication gap between two sites.

The second will reduce the gaps.

Reference: Distributed Scrum: Agile Project Management with outsourced Development teams at HICSS 2007 by Jeff Sutherland, et al.

Best Practices for Software

- Ideal: Single codebase with Cruise Control running builds and tests
- Discipline required for success

7/27/2021 11:46 AM

vijaynathani.github.io

Page 17

Single Codebase:

Everyone is kept close to the mainline

Continuous integration and automated testing identifies problems quickly and enables fixes early

Discipline:

When the build breaks, it must be fixed immediately

If you commit changes to mainline, don't go home until Cruise Control emails you that build was successful.

Developing Team Relationships

- One of the reasons a relationship of trust and openness are required is that remote teams need to be able to communicate honestly
- Establish known hours, with as much overlap as possible

Speak up, disagree, raise problems, express concern, voice dissatisfaction

Developing Team Relationships

- Telephone
- Instant Messaging
- Videoconferencing
- Wikis

7/27/2021 11:46 AM

vijaynathani.github.io

Page 19

Telephone:

Make sure people can pick up the phone easily; have a speakerphone in the work area with speed dial to remote developers

Demand an excellent speakerphone

Start conf calls with chit chat, helps reinforce human element

Take turns picking times of calls, and be aware of time of call issues

Instant Messaging

IM is useful, particular because it shows whether the person is there
may have to configure to make the status refresh more often

VideoConferencing

Just seeing someone's face can make a big different – if you don't have professional videoconferencing equipment, buy webcams for both sides

Wiki

Use wikis to share as much project-related information as possible

Tooling

Excel
Word

Pivotal
Tracker

JIRA

VersionOne
Rally

Less

More

Ceremony

7/27/2021 11:46 AM

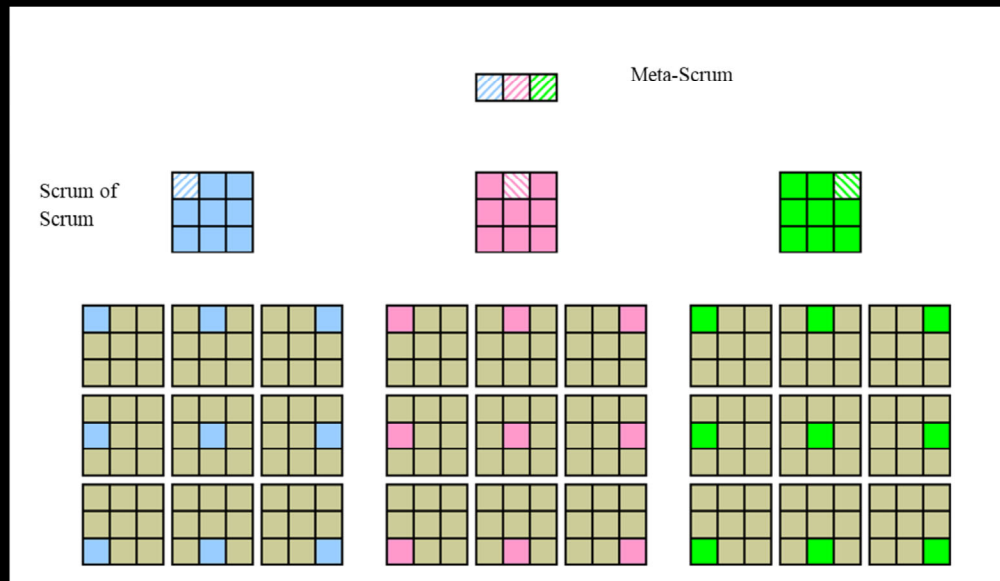
vijaynathani.github.io

Page 20

Use the simplest tool that will work for you.

A complex tool will cause overheads. Also usually distracts people from goal.

Scrum of Scrums / Meta-Scrum



7/27/2021 11:46 AM

vijaynathani.github.io

Page 21

Informal co-ordination techniques

1. Scrum of Scrum: This method has been used by Mike Cohn for co-ordination with 500 people. Scrum of scrum is held twice a week. Meta-scrum is held one a week.
2. Not all Scrum teams follow "Scrum of Scrum"
Netobjectives (a consulting company) recommends that one person on the team be used full time to interact with other teams. This person will keep changing from iteration to iteration, so that the people (responsible for interacting) do not become ivory tower.
3. Google AdWords has 5 teams distributed world wide. They don't follow this procedure. On a need basis, the team members interact with other teams. Self-organization with Inspect & Adapt.

Formal Techniques:

1. LeSS (Large Scale Scrum) by Craig Larman. <http://less.works/>
2. NEXUS – Ken Schwaber of Scrum at scrum.org.
3. SFAFe – Scaled Agile Framework. Most prescriptive of all formal techniques.

If a formal technique is needed prefer Nexus because LeSS is not popular and SFAFe is too heavy-weight.

Frequency of Scrum of Scrum

The frequency for scrum of scrums meetings should be determined by the team. Ken Schwaber has suggested that these meetings should occur daily, just like the daily standup or daily scrum. He also suggests timeboxing the meetings to last no more than fifteen minutes. My preference is to hold potentially longer meetings less frequently. I find that two or three times a week is often sufficient. This makes a Tuesday-Thursday or Monday-Wednesday-Friday schedule appropriate. While a scrum of scrums meeting will often be completed in fifteen minutes as Schwaber suggests, I recommend blocking thirty or sixty minutes for them on the calendar. Here's why: A common rule about daily scrum meetings is that they are not for problem solving; if a problem is identified it is usually addressed after the daily scrum (often immediately after). This rule doesn't apply to scrum of scrums meetings. If a problem is identified and the right people to address that problem are together, they should address it then and there. A problem that has risen to the attention of the scrum of scrums meeting participants is often a significant problem that could be affecting the work of up to 100 people. It deserves to be addressed and, if possible, resolved in that meeting. Therefore, while many scrum of scrums meetings will be over in fifteen minutes, always budget more time to address potential problems.

Agenda of Scrum of Scrum

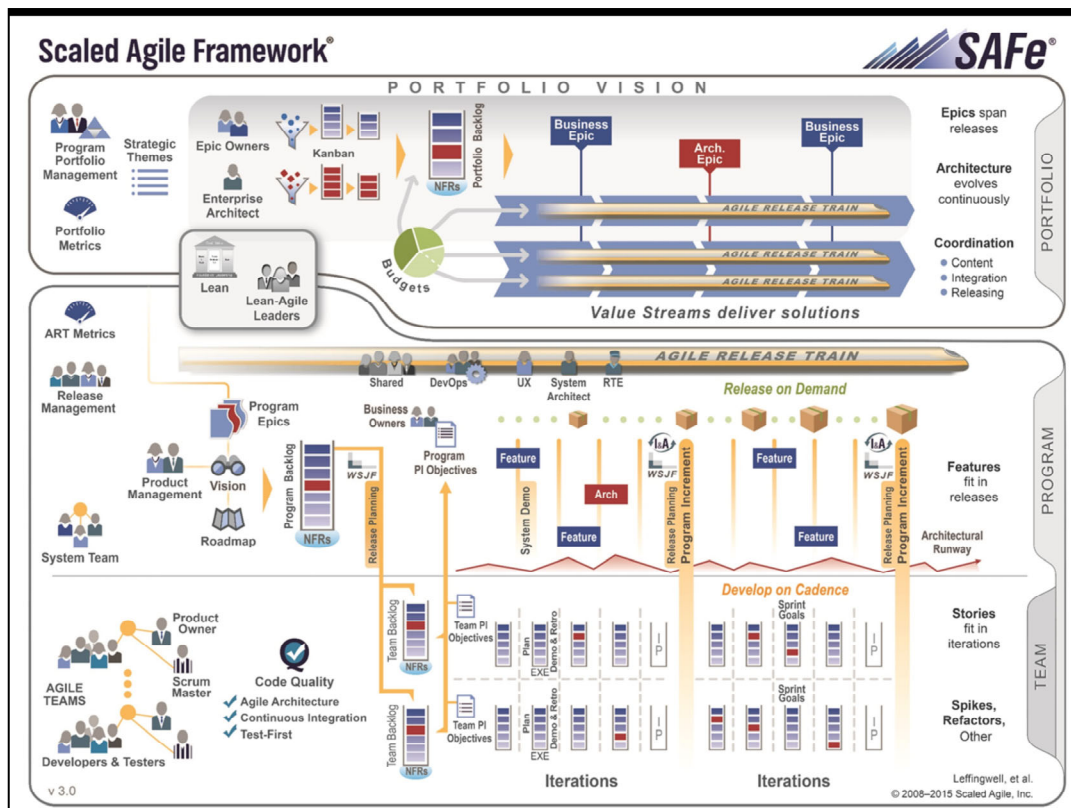
Because the scrum of scrums meetings may not be daily and because one person is there representing his or her entire team, these three questions need to be rephrased a bit. I also find it beneficial to add a fourth question:

1. What has your team done since we last met?
2. What will your team do before we meet again?
3. Is anything slowing your team down or getting in their way?
4. Are you about to put something in another team's way?

This last question can be extremely helpful when coordinating the work of multiple teams. Common answers are things like, "We're about to do a major check-in of the payroll processing code. We had to restructure the version control repository to do that, which made us rewrite a big part of the build script. But we've thoroughly tested everything and don't expect this check-in to break anything." Well, we all know how that story ends. Having advance notice of potential impediments like this can be very helpful. The scrum of scrums meeting starts with each participant answering these four questions. Like the daily scrum, this part of the meeting is meant to be fast-paced and fairly short. One technique I've found helpful in achieving this is adopting a rule that no names can be used. There are two reasons for this. First, leaving out names keeps the discussion at the appropriate level of detail. While attending the meeting, I want to hear about each team, not about each person on each team. Second, too many people equate importance with how long they talk during a meeting. Removing the ability to describe the activities and plans of each person on a team goes a long way to keeping this part of the meeting short. During this part of the meeting problems can and should be raised, but solutions should not be discussed and considered until after everyone has had a chance to answer these four questions.

After everyone has answered the initial questions, the focus of the meeting shifts to Resolving problems and discuss issues on team backlog. Participants address any issues, problems, or challenges that were raised during the initial discussion or previously identified and maintained on a scrum of scrums backlog. This backlog is analogous to what might have been called an issues list on a traditional project. It is a simple list of outstanding issues that participants in the scrum of scrums meeting either feel responsible for addressing or that they are tracking for some other reason. (For example, the issue is being addressed in another scrum of scrums meeting but this team needs to know the resolution.) Often a simple, low-tech tracking mechanism is adequate for this backlog. Many teams use a large piece of paper hanging in a team room. Some also use a spreadsheet or wiki.

One last difference between the daily scrum and the scrum of scrums meetings is that while most scrum of scrums meetings maintain a backlog of issues and problems to address, very few conduct formal iteration planning and iteration reviews analogous to what the individual teams are doing. Participants in the scrum of scrums meetings are first and foremost individual contributors on their teams. The higher-level scrum of scrums is a more transient group; each iteration has the potential to bring a new set of attendees. The iteration planning and commitments that drive a project forward belong, for the most part, at the individual team level. Some scrum of scrums may conduct iteration planning meetings, but if they do, these are usually much less formal than what the individual teams do and consist of general goals for an iteration such as "we'll address this issue, that issue, and resolve that other problem."



SAFe – Scaled Agile Framework.

e added at the end for pronunciation purposes only.

Very Good Marketing. Better than “EXTREME” or “Rugby game technique”

Nexus

Nexus™ Framework, exoskeleton of scaled Scrum

