

Table of Contents

How Should We Structure Our Organization As We Move To Agile (Development Manager)? - SSA	1
Premise	1
What is the Reporting Structure for the Development Manager?	2
What Other Kind of Structural Approaches Have You Seen?	3
What Kind of Responsibilities Does a Development Manager Have?	3
How Can the Development Manager Be Effective?	5
What does a Development Manager Do All Day?	5
What are the Characteristics of a Great Development Manager?	6
Want to Know More?	7
What are the Steps to Establish a Development Manager?	7
What are the Benefits of Establishing the Role of Development Manager?	8
How Do We Reward / Rank a Development Manager?	9
Do Managers Matter in an Organization?	9
Want to Know More?	10

How Should We Structure Our Organization As We Move To Agile (Development Manager)? - SSA

Or “What is the role of a Staff Development Manager?”

Or “How does the supervisors role change in agile?”

Premise

When you start working with an agile transformation one recommendation you will often hear is that roles such as Product Owner, Product Manager, Scrum Master, Release Train Engineer etc. should NOT be supervisors as well. There are a couple of reasons for this recommendation:

1. We want to ensure that the people in these named roles are able to dedicate time to the role, both initially to learn the role, but also longer term to completely support their role. Often organizations will see these key roles as part of a person’s “other duties as assigned” since they don’t understand what could possibly take so much time to do. This is a result of inexperience in the new system of delivery. The reality is that most agile transformations are replacing your current system of delivery with a new system of delivery. Many transformations fail because they have not actually worked the new system properly - if you have not allocated the capacity to the new roles, then you have not really embraced the new system. To support the new system, people should not be required to perform multiple roles.
2. We want everyone on the team(s) to be able to participate in the decisions being made without having to worry about what the Product Owner (for example) will do when it comes time to review / reward the person as part of whatever HR performance review process is in place. To put it simply this could create a conflict where, rather than speak up when there is an issue, a team member will hold their own council because of what they think the supervisor will think and do.
3. We want to reduce the number of people between the people that want something and the people that can actually provide that something. As much as possible we want direct communication. After all, every time someone is in the daisy chain of information, there is a risk that information gets lost or distorted, leading to misunderstanding of the need and therefore potential solutions.
4. We want to clearly change the process of working a prioritized backlog from working a push system (your job is to do this) to a pull system (we’ll take the next priority item when we have capacity). A manager can be seen to be pushing commitments rather than have the team make commitments.

In general, agile supports the idea that people on teams are nearly autonomous, self-organizing, and cross-functional. While I am sure there are some companies that are able to do this with the basic roles of agile, most organizations need to put in place some kind of structure to assist people with career development, provide advice, and other traditional management roles.

As you work with larger organizations, or organizations with a traditional managerial approach, or

organizations that maintain things like forced ranking systems, or organizations that don't have visibility to the customer, there is a need to do some level of reorganization, if only to ensure that both supervisors and team members don't have conflicting interests. The question is "what is the approach we can take to structure the organization?"

What is the Reporting Structure for the Development Manager?

For many organizations, the need for a Development Manager starts as they identify Agile Release Trains (team of teams) as part of a transformation to agile. Trains are structures with up to 125 people, so we need to make sure that we have defined reporting structure in place. The Train will affect people that are in supervisor roles and so we must be careful because a poor implementation will create a number of impediments to a successful program.

While there are a lot of variations, the basic pattern most organizations start with is:

- Program Manager or Business Owner
 - Release Train Engineer
 - Product Manager(s)
 - System Architect(s)
 - Development Manager 1
 - Team 1 (include Product Owner and Scrum Master)
 - Team 2
 - Etc to somewhere between 15-25 people
 - Development Manager 2
 - Team 5 (include Product Owner and Scrum Master)
 - Team 6
 - Etc to somewhere between 15-25 people
 - And so on

If you have an organization that is in multiple locations, you try to line up Development Manager responsibilities so that the people she is working with are "local".

With this type of structure, you are aligning all the interests of the people on the train to the program. This allows development of people to be focused on the outcomes the train is working toward, as well as the personal development of each person.

Also note that this kind of structure can repeat as you have more people. You end up with an "uber" Development Manager and "uber" Program Manager reporting into higher structures. One word of caution as you do this - try to make sure there are good paths of communication directly to people doing the work. The last thing we need is to put in place some kind of burdensome reporting structure.

What Other Kind of Structural Approaches Have You Seen?

Some organizations try to structure the Development Manager role by function. So, for example, if you have developers, they report up a development tree, QA up the QA tree, operations up the operations tree. For some organizations this is less of an impact to put in place because it already basically exists. The pros of this approach is that the Development Manager might have a better handle how to develop the skills in a particular discipline. The approach is more “skills family” oriented. The downside of this trade-off is that often the structure leads to a non-local Development Manager.

Some organizations take this to the next level. They have the Development Manager “tree” report to some time top level manager independent of program. This approach can be useful for organizations that need to signal a radical change (“no, we are serious, Product Owners are not also supervisors”) and / or establish the role (“we are going to have a number of Development Managers and need some degree of consistency”). Often this approach is set up initially with a view of moving to the more “Program” approach over time. Other times the approach stays and the organization becomes “talent management”.

Some organizations (e.g. Netflix) do a very different approach. They basically say to people “your career is your issue” to the point where they expect people to interview for jobs outside of their organization. The thinking here is not that their people are not valuable, but rather they treat their people like a professional sports team trying to figure out what they will need in 6 months down the road, and then working to ensure that future can be supported by the people they have. In this context some people are still needed; some people will need to be brought in to fill a skills gap; and some people will no longer have a role in the organization. In these situations, the Development Manager role will be more of an recruiting and planning role. While it is an interesting model, most large organizations with significant history are unable to put this kind of structure in place in the short term.

Some organizations don't make any changes. They typically believe or are in a situation where the problems raised above (lack of safety to raise issues, reward / ranking systems) are not an issue in their case. In a few cases this is true ...

What Kind of Responsibilities Does a Development Manager Have?

There are a number of variations that organizations can set up when thinking about the role of the Development Manager. The absolute basics are related to the development of people and their role in the success of teams:

- Develop a “skills” runway based on upcoming work
 - Up-skill people based on upcoming work

- If unable to up-skill, then work a recruiting plan (hire, contract)
- Based on overall strategic staffing model
- Develop succession plan to ensure continuity of value delivery
- Work issues as a result of corporate initiatives (eg transfer work from one site to another, improve reporting structure, etc)
- Help people develop their career
- Help people develop their skills
- Mentor people (both directly toward the role, and indirectly with regard to life)
- Help the organization attract and retain valuable people
- Help organization develop high performance teams
- Work with HR to establish fair, open, and simple (from the employee's perspective) policies
- Help identify opportunities for people in the organization to increase their contribution
- Evaluate performance of the individual, helping to identify and work items that contribute to participation in a high performing team (per organizational mandates)
- Serve as an agile coach and advisor to teams
- Provide enough space for teams and people to learn

Development Manager needs to treat her people like they are “volunteers” where the driving assumption is that if we don't create the right environment for them, they will leave. For knowledge workers if they are not happy, this will happen mentally if not physically. In many ways the attitude we are looking for a Development Manager is we want her to see her people as a “volunteer army”. We need to treat our people as if the only reason they are working for us is because they volunteer their services. This helps create the right atmosphere.

The Development Manager should also work to make the Program(s) successful:

- Help establish environment of sustainability.
- Help ensure the environment is “safe” for people to bring up problems.
- Help ensure transparency is maximized.
- Help ensure economic decision making, decentralized decisions, etc are part of the program environment.
- Help establish environment where people focus on problems and don't try to blame people for problems (“facts are friendly”, empirical approach).
- Assist in preparation for PI Planning event (eg message, themes) and the IP iteration (eg Training, hack-a-thon).
- Participate in various Program ceremonies such as the retrospective (I&A), planning (PI Planning), demonstrations (team and system demos), etc to identify impediments and opportunities for improvement.
- Help the people of the train to establish relationships with customers, stakeholders, and suppliers.
- Assist in working people related program and team impediments.
- Participate in Value Stream Mapping to help identify and work skills issues.
- Help establish core values and principles (eg quality, shift left, etc).
- Work with and support Communities of Practices to ensure discipline specific improvements of you people.

How Can the Development Manager Be Effective?

In particular, when it comes to reviews and / or ranking, how can the Development Manager have enough information to be fair. Being a Development Manager is hard and requires some basic actions to stay in touch with the people they are supervising to help them understand what is happening with that person. Some examples include:

- One-on-one meetings with people - say every 1 or 2 weeks - to talk about personal development and development of team behavior.
- Bi-weekly meetings with POs, SMs, teams to discuss people, needs of the team.
- Support (be the champion, especially for new CoPs) a Community of Practice aimed at developing skills associated with the practice, setting guardrails for the practice, and so on.
- Attend (spot) ceremonies - demonstrations, planning, dailies, etc and team retrospectives (by invitation.)
- Setup / make available training. This could be for a particular individual, or as a result of a need that the train has for common knowledge say in the IP iteration.

In the end, this is about proactively setting up feedback loops. As a Development Manager you cannot be everywhere all the time. Like anything else that is capacity limited, this means you have to prioritize your work to “what is important”. To be comfortable with this approach you have to be trust that when something important arises, you will be notified in time to be able to work the issue. This means there is a continuous flow of feedback.

Note that responsibility for career development, ranking etc. is not just something Development Managers have to do. Like most commitments, there is a two-way commitment between the Development Manager and the person being supervised:

- Development Manager has a responsibility to reach out and find out what is going on with the people she is supervising.
- The Supervised has a responsibility to reach out to the Development Manager to help that person be effective.

What does a Development Manager Do All Day?

When this role is first discussed, many will ask “what does a Development Manager do all day”. The thinking here is that the traditional role of management has been to assign work to people and hold them accountable for that work. If they are not doing this, what does the Development Manager do? The reality is that if you think about what a manager should be doing (career development, solving problems, working with people toward results, etc) then the reality is that we have not been doing a “complete” job

as a Manager.

From the responsibilities and the role it is clear there is a lot to do. One big change is that the role of the Development Manager is an active role where they are out with the people they are managing, seeing the work, and working to solve problems. To give you an idea of the role, let's work some basics assuming the Development Manager has 20 people she is working with.

If we look at a **2 week** period (or 80 hours), for a normal week time could be allocated as follows:

Activity	Hours	Comments
One-on-one meeting	20	20 people, 1 hour each
Attend program meetings	10	4 x SoS, 2 x PO Sync plus preparation and follow-up
Spot attend Team Dailies, Planning, Demos	10	10 meetings of 60 minutes each
Champion Community of Practice	4	1 hour meeting with 3 hours prep and followup
Development Managers Community of Practice	1	Collaboration and improvement
Buffer per normal agile planning (eg email, meeting, etc)	20	25% of capacity
Total:	65	Every 2 weeks

This leaves 15 hours in the two weeks, or 7.5 hours per week, for all the other responsibilities.

What are the Characteristics of a Great Development Manager?

Clearly selecting the right people to be a Development Manager is critical. What kinds of people should we look for? Great Development Managers exhibit the following characteristics. They are:

- **Lean-agile Leader:** The Development Manager must “walk” the role of a lean-agile servant leader. The summary from SAFe help understand this. Development Managers must be seen to be leading the change, to emphasize lifelong learning, to help inspire and align with mission, to help minimize the number of constraints in the organization that get in the way of delivering value (especially the ones that have built up over time), to help decentralize decision making, and to unlock the intrinsic motivation of knowledge workers.
- **Trusted:** Have you ever worked with someone who you have implicitly trusted? You know that in everything they say and do, no matter how small, they are consistent, open, and honest, in their approach and are deserving of the trust you place in them. To be effective, Development Managers must to be worthy of trust.
- **Inspiring:** In many ways Development Managers need to treat their people like they are a group of volunteers. To keep a volunteer group in place usually requires a common, understandable and relatable vision, and an engagement approach that truly respects the value of the knowledge worker's opinions.

- **Likable:** We need our people to want to discuss things with their Development Managers so they can be effective. This means there should be no personal barriers to having a discussion which means that the Development Manager must be likable. You want people to go out of their way to have a discussion with the Development Manager, not just because it is useful, or part of the process, but because they want to.
- **All about their people:** The Development Manager needs to put the interests of their people first and, in particular, their people's interests over the interests of the Development Manager.
- **Properly blunt:** The Development Manager will sometimes need to discuss difficult subjects with their people. Perhaps there is something that needs to be addressed to help a team, or perhaps there are issues associated with individual behaviors that need to be addressed, but whatever the case, sometimes things need to be said.
- **Local:** The Development Manager needs to be physically located with the people who report to them as much as possible. Let's face it, whereas traditional managers handled task assignment to their subordinates and so should have a pretty good handle on what the person is doing, this is not the case with the Development Manager and so they should be co-located as much as possible. This will also help the Development Manager work cultural issues in multi-national organizations, and work local issues (e.g. expectations, legal issues, and soon).

Want to Know More?

- Part of a series:
 - [What Are The Characteristics of a Great Product Owner?](#)
 - [What Are The Characteristics of a Great Scrum Master?](#)
 - [What Are The Characteristics of a Great Manager?](#)
 - [What Are The Characteristics of a Great Development Manager?](#)
 - [What Are The Characteristics of a Great Agile Coach?](#)

What are the Steps to Establish a Development Manager?

If you don't have the role of a Development Manager in your organization already, or there is not common understanding of the role, you need to do more than identify the Development Manager and announce the new reporting structure.

1. Identify potential Development Managers. Review the role, the characteristics of the role and try to find people you think will be a good fit.
2. Create an interest in the role of Development Manager with the people identified. The new role is critical to the success of the agile transformation and so you want people to be passionate about it. Many people will see this as a boring, administrative HR role, so there is a degree of "selling the role" involved. In the final analysis you really want people to volunteer for the role. You should not just tell them, that this is their role ("surprise!")

3. Development Managers need training. They need are in fact the “ultimate servant leader” and this is not something you can do unless you have training. The training needs to be “lean agile leadership” oriented.
4. To establish the new role, significant organizational support will be required. Make sure you have a conversation about how you will support the Development Manager especially during the early part of the transition to agile and the new role.
5. Determine the new organizational structure, the reporting structure.
6. Make sure each of the people effected by the new reporting structure understand how it will work. We need to be clear when people should reach out to the Development Manager versus the Product Manager or Release Train Engineer, for example.
7. You might want to kick things off with an all-hands meeting with your people so that everyone hears the same message about the new role and its a fit with the rest of the changes going on. Side note on this - you cannot over-communicate this change especially in the early days.
8. Setup a cadence of 1 on 1 meetings with each of your people. Initial agenda should not just be “what are your career goals?” but rather aimed at establishing the trust you will need going forward. This will mean doing more listening to current issues of each of the people, and then actually starting to work those issues.
9. Ensure Development Management has access to the ceremonies they need to be successful. For example, if the portfolio of the organization requires the development of new skills in people or new people with skills in a particular area, then the Development Management needs to be directly involved in the relevant portfolio level ceremonies.

What are the Benefits of Establishing the Role of Development Manager?

The following benefits are typically seen using this approach:

1. It means that POs SMs etc are not supervisors and so we reduce chance of a number of anti-patterns (eg P.O. bossing team) and ensure there is enough capacity for these roles to actually do the job.
2. It means we have a far flatter structure as each Development Manager supports a lot more people. This is a step in the right direction.
3. With Development Manager and train members both reporting into Program Manager type role we are aligning interests of people with interests of train and therefore the value stream. Everyone is oriented toward “success of the program”.
4. Most existing HR organization can support this type of structure without significant change to their operation. Some HR organizations will object to the idea that one person can “handle” 20-40 people, but when the role is explained, this is typically not an issue.

How Do We Reward / Rank a Development Manager?

In shops where review and ranking is a requirement, there is an obvious need to also review / rank the Development Manager. What should you look for when ranking a Development Manager:

1. The most obvious result of the work of a good Development Manager is that the organization (trains, teams) are seeing good development of people in response to the demands being seen.
2. Impediments associated with people Development, many not considered “safe” discussions in the traditional organization, are being worked and resolved.
3. People want to work in the organization and with that particular Development Manager.

Do Managers Matter in an Organization?

There seems to be a general discussion in agile circles about whether managers matter. Early agile discussions seemed to be actively hostile to management, stating that you do not need anything like a management - just have self-empowered Teams.

What was interesting to me was that every agile transformation I was involved in was large enough that there was significant management structure in place, and that structure could either be an activist participant in the success of the transformation, or an active impediment to adoption. From my perspective it was always easier to engage with management to ensure good outcomes.

It turns out that others thought that managers might not be needed in an organization. “Google hasn’t always properly appreciated management. In 2002, Google ran an uncontrolled “experiment” by simply getting rid of all managers. It didn’t go well. So in 2008 a team of researchers set out to prove what some at Google suspected - that managers don’t matter. But very quickly the team discovered quite the opposite. Managers matter a lot.” More ideas can be found at [Google’s Research on the Role of Management (Project Oxygen)](<https://rework.withgoogle.com/blog/the-evolution-of-project-oxygen/>)

While this research is applicable to Google’s context, this research helped confirm for me that there is a role for management in general. The Development Manager approach is one potential approach. The research also identified “Good Manager Behaviors” at Google:

1. Is a good coach
2. Empowers the Team and does not micro-manage
3. Creates an inclusive Team environment, showing concern for success and well-being
4. Is productive and results oriented
5. Is a good communicator - listens and shares information
6. Supports career development and discusses performance
7. Has a clear vision / strategy for the Team
8. Has key technical skills to help advise the Team

9. Collaborates across Google
10. Is a strong decision maker

Although some do not apply (eg “collaborate across Google”) these behaviors line up well against the characteristics we’ve identified for Development Managers.

Want to Know More?

- Scaled Agile Framework (SAFe) (calls the role “Development Manager”) reference to [lean agile leaders](#)
- Scaled Agile Framework (SAFe) view of [evolving role of manager](#)
- Large Scale Scrum (LeSS) on [changing role of managers](#)
- Google’s research on the role of management
<https://rework.withgoogle.com/blog/the-evolution-of-project-oxygen/>

[FAQ](#), [SubjectSpecificArticle](#), [organization](#), [managers](#), [structure](#), [people](#), [executive](#), [developmentmanager](#)

From: <https://www.hanssamios.com/dokuwiki/> - **Hans Samios' Personal Lean-Agile Knowledge Base**

Permanent link: https://www.hanssamios.com/dokuwiki/how_should_we_structure_our_organization_as_we_move_to_agile_ssa

Last update: **2022/02/23 07:02**

