



Development Manager: A Scrum Master's Best Friend!

ENDER YÜKSEL, SIMCORP

MARTIN SCHWALBE LOHMANN, ALM. BRAND

Development Manager is an agile leadership role in SimCorp focused on growing people. The Development Manager is not part of the day-to-day execution; instead the Development Manager's focus is on long-term individual and team development over short and medium-term gains. However, for the role to be successful constructive collaboration with other agile roles is paramount. Especially the interaction with the Scrum Masters is a potential hotspot. With the Scrum Master being responsible for not only team process, but also team climate, the interaction between the two roles has both the potential for futile power struggling as well as for releasing the synergies of two complementing roles pulling in the same direction. We have seen both types of outcomes, and in this report, we will share our success and failures regarding Development Manager and Scrum Master collaboration in a SAFe-organization.

1. INTRODUCTION

Most Agile frameworks prescribe specific roles to be responsible for each of the areas of product, process and development/technology. However, when it comes to areas such as developing people, creating a learning organization, and long-term team development according to organizational strategies, there are gaps or very high-level guidance [LeSS, SAFe]. In SimCorp, the Development Manager (DM) role was introduced into a Scaled Agile Framework (SAFe) setup to specifically address these areas and to complement the other SAFe roles.

A DM is responsible for three areas: 1) People, 2) Learning, and 3) Strategy. The DM typically covers four agile teams, including Scrum Masters (SMs) and Product Owners (POs), and takes care of individual and team development. The DM's focus is on the long-term; from that position she collaborates closely with other agile leadership roles, who typically have more focus on the short- and medium-term gains.

Whereas the entire scrum team is engaged in producing a potentially shippable increment sprint after sprint, the DM is focusing on what skills are needed within the team. This includes knowing SimCorp's strategy and understanding what the strategy requires from the team. Both SM and DM have focus and interest in developing high performance teams, and they are *the best allies*.

In this report, we will take you through our journey since the inception of the DM role in 2016, and explain how we have dealt with the people management aspect in an agile setup. We will share our accomplishments as well as challenges we have faced when introducing the DM role.

2. BACKGROUND

SimCorp has been developing investment management software for almost five decades, and embarked on an agile journey in 2015. SAFe was adopted by more than 550 people forming around 55 teams within 7 ARTs across 4 different locations in Europe, all working on the same product. SAFe successfully supported SimCorp's agile adoption in most areas, and whenever it didn't, SimCorp came up with their own solutions as compiled in a previous experience report, "What SAFe doesn't tell you"[Cook et al].

In the past, SimCorp managers resembled traditional engineering managers being involved in a variety of areas from product to process, and from technology to people management. Most managers had a background as a successful specialist, e.g. developer or tester, before becoming a manager of a specialist team. A Group Manager would typically manage a co-located component team, and a Unit Manager would be overlooking multiple teams, practically being a manager of several Group Managers.

SimCorp's agile journey scrapped the Group Manager and Unit Manager roles. Instead of automatically offering those former managers *new titles*, they were all asked to apply for the new agile roles, based on their interests. So, most of the managers went through a job application process including detailed interviews, based

[Ender Yüksel](mailto:ender.yuksel@gmail.com), Copenhagen, Denmark; email: ender.yuksel@gmail.com

[Martin Schwalbe Lohmann](mailto:martinlohmann@gmail.com), Copenhagen, Denmark; email: martinlohmann@gmail.com

Copyright 2020 is held by the author(s).

on their professional interests. For example, technology-oriented and hands-on managers preferred the System Architecture path; the managers who were more into the processes and execution went for the SM or Release Train Engineer (RTE) (a SAFe-specific role, resembling a high-level SM role) path; the ones with intensive knowledge of the product tried the Product Management or PO path, and so on. Not all of them would succeed in the process, and also some decided to leave the company, as they preferred to continue the traditional ways of working.

Scrapping two levels of line management was one of the necessary steps to kick-start the agile journey and to empower new agile teams (mostly scrum teams) and the new agile roles as specified by SAFe. However, there were still areas to be addressed, including, but not limited to employment, remuneration, building capabilities, growing individuals and teams, long-term focus and strategy. Neither the Scrum Guide, nor the scaling frameworks were helpful in addressing these areas.

Obviously, this was a problem that other organizations also faced. They tried approaches such as assigning a line manager per *role* (e.g. a manager for developers, another manager for SMs), per *geographical location*, and per *Agile Release Train* (ART). There were also less costly approaches such as keeping the Group Managers to manage teams, and keeping the Unit Managers to manage ARTs [Ljung]. There were even cases where one of the agile roles, such as SM or Product Owner (PO), also wore the line manager hat.

For us, this was not merely a problem of line management. We needed *agile leaders* to lead the change, have a long-term focus, and to grow people—our most important asset!

3. OUR STORY

Ender and Martin both worked as SMs in SimCorp, where each of them served multiple scrum teams and interacted with multiple DMs for several years. In addition to that, Ender has been practicing the DM role since mid-2019. Over these years, we have figured out that DM-SM collaboration is key to success in an agile organization, and these two roles can be best friends!

3.1 How do we define the DM role in SimCorp?

A DM is responsible for growing people and agile teams as a whole. Competency development, performance enablement, and talent management all fall into the DM area of responsibility. Creating the right environment for team collaboration, helping with identification and removal of impediments, and driving and executing business improvement strategy are also parts of the formal definition of the DM responsibilities in SimCorp.

Practical duties of a DM include building high performance teams, establishing mission and purpose for individuals and teams, performing career counseling, and listening and supporting teams in problem identification, root cause analysis and decision making. There are also other responsibilities that may overlap with other agile roles, for instance serving as an agile coach and advisor to agile teams (and ARTs which are agile teams of teams). Our motto is, remaining close enough to the team to add value and to be a competent manager; staying far enough away to let them problem-solve on their own.

The DM role has been subject to continuous improvement over time such as:

- The concept of a learning organization where the DM is supported by learning areas in order to give the best support and learning opportunities to different roles within the agile teams (e.g. PO, SM, Development Team Member, Technical Writer, User Experience Designer). More information about this concept is available in a previous experience report [Cook et al]
- Loosening the affiliation with the *ART leadership* in order to avoid local optimization favoring the ART and improve collaborations between ARTs towards reaching overarching goals
- *Rotations* in order to spread experiences between ARTs and learn about the challenges and strengths in different ARTs
- Possibility and encouragement to work with *multiple ARTs at the same time*
- Improving adaptability by moving whole scrum teams from one ART to another—*together with their DMs*—in order to adapt to changing business needs and support the company level goals.

In order to make it simple, we will use the following practical explanation for the DM role, as described by Daniela Lück, who is leading our Development Managers in Denmark and Germany:

"In SimCorp, responsibilities are basically divided into four areas: Product, Process, Technology, and People as in Figure 1. In the SAFe setup, product leadership is with POs, Product Managers, and Product Portfolio Managers (PPMs); process leadership is with SMs, RTEs, and Agile Coaches; technology leadership is with Tech Leads, System Architects (SAs), and Enterprise Architects. Neither SAFe nor Scrum is prescribing the People leadership. On the other hand, without people leadership, people side would always be on the losing side when it comes to short-term

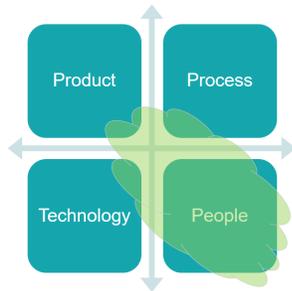


Figure 1: Separation of Responsibilities

decisions around deliveries. DM role is covering this important gap, and bridging with the Product, Process, and Technology areas.

Traditional managers would be responsible for all the aforementioned areas, and having the DM role is a significant change for both the former managers and the employees. Former managers would tend to think that their role is degraded, and employees would tend to think that their manager cannot manage them anymore. The reality is, People area is already a huge area and it did not have sufficient focus before, mostly due to managers prioritizing other areas. Besides, now every employee has a dedicated DM whose focus is growing her rather than anything else. Therefore,

both DMs and employees are gaining a lot from the presence of the DM role, rather than losing something.

I would like to think of the DM role as a triangle of people, learning, and strategy, as in Figure 2, where the intention is to create awesome teams. This triangle is exactly the area that the SM also has room and possibility to work with a team. DM and SM need to be very well aligned to be able to do their job the best."

In practice, a typical DM works with around four agile teams, and has around thirty people in her line. These numbers can seem high at first, yet the lack of operational responsibilities and good collaboration with SMs provide the sufficient time for DMs to take good care of people. In our experience, traditional managers had around half the number of people, though the delivery deadlines, product support cases, talking to the clients, and many other operational tasks refrained them from securing the time for individuals' growth.

The DM is the line manager of the whole agile team, including the SM and PO on the scrum teams. However, for various reasons, we do have some teams with more than one DM. In our experience, the best DM-team and DM-SM collaborations occur for co-located teams that have only one DM. In the teams that have multiple DMs, we have observed cases where teams received misaligned or even contradictory guidance, delays on DM actions, and complications with communication.

A DM also acts as a Learning Owner of a specific learning area (e.g. software engineering, product management, cloud, agile) where she works on providing the best learning opportunities and supports other DMs in terms of growing people in related roles.

A DM is also a member of a forum that defines the strategy (for product development) and hence actively works on strategy. Besides, DMs are active in executing the strategy as expressed by Objectives and Key Results (OKR) and Conversation-Feedback-Recognition (CFR) as a tool.

A DM does not only work on a team level, but also on a program (ART) level together with the so-called ART leadership, composed of RTE, SA, and PPM. For the DM to succeed, input from and close collaboration with the ART leadership is crucial.

During the last four years, the DMs have supported SimCorp's agile journey in many different areas including:

- *scrapping individual bonuses*, raising the base salaries instead;
- *scrapping annual performance reviews*, moving to a scheme where ongoing feedback and quarterly reviews cover both individual and team performance;
- *replacing the traditional goal setting mechanism* where the manager defines the goals, with new mechanisms where the employee defines her goals, and where achievements are not directly related to monetary compensation;
- *removing time registrations*, as they do not add value;
- *helping with the organization of program increment planning events* where the whole ART comes together;



Figure 2: DM Triangle, aiming for awesome teams.

- *organizing world class trainings* with known thought-leaders in agile, product management, quality, remote collaboration;
- *involving teams in recruitment and onboarding*, finding better fits for the teams and providing better onboarding for new employees;
- *providing and supporting different learning initiatives* such as brown bag sessions, book clubs, sharing-is-caring sessions; and
- *revising value streams*, resulting in restructuring of ARTs.

3.2 Can the Development Manager be the Scrum Master's best friend?

In our pre-SimCorp experience, traditional managers often got in the way of SMs, e.g. through micromanagement, lack of trust in the team, addiction to command and control, and general violation of the principles behind the Agile Manifesto¹. The DM role, however, was explicitly designed to have nothing to do with execution, and is generally fulfilled by professionals that understand and exhibit an agile mindset.

DM and SM collaborate in many areas that affect an agile team's success. Below we have listed some areas where DM/SM collaboration have proven to be particularly beneficial to SimCorp:

Team formation: Formally this is the responsibility of the DM, yet a SM could have valuable knowledge regarding group dynamics. In our experience, the best decisions around team formations and adjustments are founded on close SM and DM partnerships, where the SM provides input regarding needs or issues, and the DM acts by considering these inputs, and eventually involves the SM and the team members. We have seen numerous cases where valuable input from SMs has helped the DM, for instance: adjusting team size, replacing vacancies, adjusting skill (e.g. testing, technical writing) ratios within a team, and team member rotation. Moreover, SMs can be very influential in facilitating a team self-selection process as described in a previous experience report co-authored by Martin, "Is team self-selection the obvious choice?" [Harre and Lohmann].

Recruitment: This is another formal DM responsibility area, however, SM involvement can bring significant benefits to the recruitment process. Before SimCorp's agile transformation the recruitment process was typically run almost entirely by a manager who would decide on which specific profile to look for, then she would conduct the interviews, and finalize the recruitment process. In our experience, it is better to involve the SM and representatives from the team in the whole recruitment process, from the identification of what specific profile to look for in terms of skills and the team's preferences in what to look for in a new colleague, to the final team interview, where the DM is not present. The final interview is typically facilitated by the SM.

Onboarding: Our agile onboarding motto in SimCorp is *the team knows the best*. We refer to the knowledge or learnings a new team member needs to acquire to work effectively within a new team as *onboarding items*. We typically identify the onboarding items in two sessions. The first being a workshop with the agile team and the DM where the team identifies the onboarding items they believe are important for the newcomer. At a second workshop, the newcomer reviews the identified onboarding items and removes the ones that she believes are not necessary and adds new ones that she thinks she needs. In the end, the agreed onboarding items are prioritized and added to the product backlog, just like any other work item (e.g. a user story). The newcomer, together with other team members, pulls onboarding items (pairing) into an iteration at sprint planning. Onboarding items are demoed on iteration reviews as any other work item completed during the sprint. The process is initiated by the DM, but obviously the execution requires involvement of the SM. In our experience, mature SMs can run the whole onboarding with minimal involvement of the DM.

Coaching: Coaching is mostly associated with the SM role, yet the DM's support and help is also crucial. When both SM and DM coach individuals and teams in agile ways of working and mindset, from their different perspectives, then the journey to high performance for the team will go faster. For instance, in SimCorp, DMs coaching and teaching areas include, but are not limited to, learning culture, continuous improvement, flow, and feedback culture. However, DM collaboration with the SM results in more diverse perspectives and better results, as we have recently demonstrated with flow workshops where we (SM and DM) coached teams in lean principles [Yuksel].

Strategy: Aligning teams to the overall strategy of the company is important, especially when the ART's (short-term) strategy results in team members forgetting about the overarching long-term goals of the company. In SimCorp we use OKR (and CFR) as a tool to execute the company strategy. DMs are involved with strategy definition and execution, yet SMs are very influential in executing the strategy on a team level. OKR

¹ Especially, "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done." and "The best architectures, requirements, and designs emerge from self-organizing teams."

workshops and progress in strategic and transformational work relies on sound collaboration between DM and SM.

Long-term perspective: Different roles and areas would naturally have different interests, and their approaches to daily challenges can jeopardize long-term interests. For example, with all the best intentions, product or technology leaders might suggest solutions that would require removing/adding individuals to certain teams in order to speed up delivery of a concrete feature. DMs and SMs, who are aware of the negative effects caused by changing team composition, can explain the long-term damages in contrast to minor short-term gains.

3.3 Cases: The Good, the Bad, and the Ugly

In this section, we would like to provide more real cases and shed light on, not only the positive parts, but also the parts where we are challenged. We have chosen to keep the cases at a high level, refraining from going into personal details.

3.3.1 The Good

Fortunately, we have way more cases that fall into this category, compared to others.

DM at Team Retrospective: Generally, DMs do not attend team retrospectives. But in this case a newly hired DM was asking the SM to attend team retrospectives to learn more about his new teams. The SM was reluctant to invite the DM to the team retrospectives out of fear that this would prevent team members from speaking their minds and discussing difficult issues. Furthermore, the SM was also concerned that the DM would use retrospectives to assert his authority and prove how he could solve problems for the teams and thus undermine the SM role. The DM, on the other hand argued: “How can I support the team if I am not part of an important meeting such as the retrospective?” The SM shared his concerns, and said he would like to ask the teams first how they felt about having their DM at their retrospectives. It turned out that the teams were happy to have the new DM present so he could learn about the issues they were struggling with. After that, the DM and SM agreed that the DM could attend the retrospective, but the team should know in advance when the DM was planning on participating, a rule which later relaxed as trust was built between the SM, DM, and the teams. This set-up proved to work for both parties. It allowed the SM to ask for feedback from the DM, for instance after a difficult subject had been discussed at a retrospective. There are a number of factors which made this work: 1) the team was generally outspoken no matter who was in the room; 2) the DM understood his role as being mainly an observer staying in the background letting the SM facilitate the meetings; 3) the DM only participated occasionally; 4) the DM had a solid agile mindset and knew how to be a servant leader; and 5) the SM and DM established a high level of trust.

SM and DM working on organizational change: In this case three Scrum teams were working together in a small ART. The work performed within the teams had many uncertainties, which made forecasting what the teams would be working on more than a couple of sprints ahead highly inaccurate. Furthermore, there were almost no dependencies between the teams. All of this caused frustration with the whole SAFe-ART setup, which created much bureaucracy without helping the teams. After consulting with the DM, the SM scrapped the SAFe setup and had the three teams doing one-team scrum with a little extra coordination borrowed from LeSS. The DM also helped by backing the SM’s decision with upper management.

SM and DM working on dysfunctional teams: Turning the ship around is not trivial when it comes to dysfunctional teams. However, we observed success after the SM and DM worked together for a long period of time and heavily invested in coaching and team workshops. We also had cases where the DM had to make difficult decisions, such as off-boarding people. The insights of the SM, due to being closer to the team, helped the DM in taking the right steps from a team perspective. This included finding a better match with respect to the type of work or department for the challenged employees—or dismissal from the company.

Grooming internal staff for new positions: Spotting internal talent and providing the best support in terms of growing the talent into a new role (SM in this case) is not trivial. We had a case where SM and DM collaboration resulted in a developer successfully transitioning to the role of SM within the organization. This collaboration included DM securing participation in Agile Coach Camp and external workshops while the SM provided on-the-job training and support.

SMs and DMs collaborate on providing internal training to new employees: For more than a year we have both SMs and DMs collaborate on an internal agile course offered to new employees. Both roles are part of a volunteer team that is responsible for course design and delivery.

Moving an entire team to another ART: Quoting Craig Larman, agile is about turning on a dime for a dime. On the other hand, this should not mean breaking well-functioning teams, or creating new problems. Similar to the Brent character in the Phoenix Project [Kim et al], our organization also suffered from internal bottlenecks and dependencies on superstar employees that could solve any issue and deliver results in no time. As a result, these employees would be shifted from department to department depending on the business-level urgency and priority. A case that we are especially proud of is being able to move entire agile teams and their DMs to another ART (value stream) due to new business priorities. Within the last year, we have done so while realizing positive returns in the sense of deliveries as well as technical growth of the teams. This again happened in close collaboration between DMs and SMs where both roles were focusing on optimizing for adaptability and avoiding local optimizations.

SM and DM working together to clarify roles and expectations: SimCorp has had a strong specialist culture for decades and for some senior developers the concept of cross-functionality has been hard to come to terms with. This has resulted in a few cases where a team member would simply refuse to work outside his or her main skill area and to do any pair work or knowledge sharing, no matter what teaching and coaching the SM could come up with. This kind of resistance was often rooted in a belief that cross-functionality was only important to the SM and that management was still favoring and rewarding the old specialist culture and/or fear of losing the status and job security that comes with being the only person who can maintain a complex and business-critical system. In these cases the DMs have supported the SM by clarifying what is expected from a developer in the new agile setup. In most cases this has removed or reduced the resistance against agile ways of working.

SM-DM Coordination: How much and how often should DM and SM meet to coordinate? There is no general rule for this in SimCorp. In general, it varies from team to team depending on SM experiences, team issues, and personalities. In most cases SM and DM finds the right level of DM involvement where the DM can support the entire team while leaving plenty of space for the SM for continuous growth.

3.3.2 The Bad

Misalignment between DM and SM: We have had cases where the DM and SM were not aligned with respect to coaching of individuals and the team as a whole. These cases tend to be related to DM and/or SM lacking agile knowledge and/or experience. As the DM is formally the SM's boss, such misalignment can put the SM in an awkward position especially when it is the DM who is lacking knowledge and/or experience. In our experience the most successful DMs have solid agile mindsets and understand the dynamics within an agile team.

Multiple DMs for one team: It is hard to generalize since we have had mixed experiences with having multiple DMs for one team. When a team has two or more DMs, it is mostly due to the team being geographically distributed across multiple countries. One challenge with this setup is misalignment, where the SM and team may get different (and at times conflicting) guidance and advice depending on the DM they talk to. In the first real case, DMs had conflicting understanding of the needs and performance of the same team. In the second real case, DMs had different assessments of the capabilities of the same individuals. The third case was different approaches to onboarding using very different processes and different degrees of SM involvement. In all cases, the confusion the DMs generated caused frustration for both individuals and teams. Another challenge with this setup is conflict resolution. Most conflicts in a healthy agile team are solved within the team itself, often facilitated by the SM. However, sometimes conflicts escalate to a point where the DM, as the people manager, must get involved. In these situations the DM, as the manager of all the involved parties, can be a great help to the SM. On the other hand, we have also seen that when conflict resolution involves multiple DMs they tend to, with the best of intentions, stand up for their people, which is often not very helpful and may only result in escalating the conflict further. In general, we have found that we get the most out of the DM role when there is a single DM per team.

3.3.3 And the Ugly

The Puppet SM: This is fortunately a rare case, but we have seen the SM role implemented by a few DMs as what we call a puppet SM. In these cases, the SM could not make any decision without first asking the DM for permission. In practice, the DM was micromanaging the team through the SM. Both SM and DM had missed the points of their roles and caused unintentional confusion to their teams. The teams failed to grow into self-organising teams. These cases are generally caused by DMs with a poor knowledge and understanding of what it means to be an agile leader.

The Dictator DM: Another ugly, but not uncommon case is DMs resorting to pre-agile command and control habits especially when under stress or when overwhelmed with work, resulting in dictation of solutions and generally ignoring the agile manifesto. Some cases that we have observed: setting a work-in-progress limit for the teams; dictating a support scheme without involving the insights and inputs of the teams; and deciding to grow or shrink teams ignoring the insights of the SM. The impact of all these three cases was degraded trust between the scrum teams and the DM, and damaged efficiency of the DM-SM partnership.

4. WHAT WE LEARNED

We have listed our joint learnings below.

For the DM to successfully complement the SM it is important that:

- The DM is detached from operational responsibilities so she can focus solely on developing people without the pressure from short-term deadlines and obligations; and
- That the DM, being the manager for all roles within the team, is supported by the learning owner structure in order to successfully coach all roles with regard to professional growth, as every role has different needs.

Developing people and teams can best be done through DM and SM collaboration, rather than either of the roles doing this on their own.

The transition from traditional line manager to DM can be very difficult when it comes to unlearning old ways of managing. On the other hand, new DMs with no history of command and control tend to adapt much more easily to the DM role including establishing frictionless DM-SM collaborations.

Explaining the DM role to the organization is not to be underestimated. If team members do not know what to expect from their DM, the benefits of the role are greatly diminished.

DM-SM collaboration has been most beneficial to SimCorp in the areas of team formation, recruitment, onboarding, strategy, and long-term perspective.

In general we have learned that sound agile knowledge and a genuine agile mindset is a prerequisite for being a successful DM and to establish a powerful SM-DM tandem.

5. ACKNOWLEDGEMENTS

We would like to start with thanking our shepherd Bonnie Aumann, for their constant support and guidance. We would also like to thank the track co-chairs Rebecca Wirfs-Brock and Frank Olsen, for their support and patience. Special thanks go to Daniela Lück for sharing her vision and perspectives with us. Last but not least, we would like to thank all our former managers, we have learned a lot from each of them.

REFERENCES

- [Cook et al] Neil Cook, Piet Syhler, and Frank Olsen, 2018. [What SAFe doesn't tell you](#). Experience Report, XP 2018.
- [Harre and Lohmann] Niels Harre and Martin Lohmann, 2019, [Is team self-selection the obvious choice?](#) Experience Report, XP 2019.
- [Less] Large-Scale Scrum. 2019. *Role of Manager*. [ONLINE] Available at: <https://less.works/less/management/role-of-manager.html>. [Accessed September 2019].
- [Ljung] Anna Ljung and Johanna Udesen, 2019. [The role of the first line manager in a Scaled Agile organization](#). Master's Thesis, Chalmers University of Technology.
- [Safe] Scaled Agile Framework. 2019. *Lean-Agile Leadership*. [ONLINE] Available at: <http://www.scaledagileframework.com/lean-agile-leadership/>. [Accessed September 2019].
- [Yuksel] Medium.com. 2019. *What can development teams learn from Copenhagen buses?*. [ONLINE] Available at: <https://medium.com/@ender.yuksel/what-can-development-teams-learn-from-copenhagen-buses-5b254896550b>. [Accessed September 2019].
- [Kim et al] Gene Kim and Kevin Behr and George Spafford, 2013. *The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win*. IT Revolution Press.