



ADAPTION OF SELECTED PMBOK PROCESSES TO FIT SCRUM DEVELOPMENTS

Philipp Rosenberger, FH Campus Wien;
József Tick, Obuda University

Abstract

Project managers managing agile developed IT projects often find themselves in difficult situations. Their frameworks, like PMBOK project management framework of PMI Organizations, demand a deep level of planning, control and active management. On the other side, agile development frameworks like SCRUM demand self-management, flexibility and appreciate change. This article proposes solutions for five PMBOK processes that have been identified as critical in SCRUM development environments in the previous publication Suitability of PMBOK 6th edition for agile-developed IT Projects, by Rosenberger and Tick. The process of “Manage project execution” is adapted by introducing Strike Events; “Work Breakdown Structure Plan creation” and “Scheduling” processes are changed by dividing large backlogs into phases and break down individual phases into Macro and Micro level planning; “Cost Estimation” processes uses velocity of development teams as planning reference; “Developing and Managing Teams” is adapted by introducing the project manager as SCRUM master and if needed apply again the Strike System in case of serious problems. These proposed solutions adapt the classical PMBOK project framework to cope with SCRUM developed project to an “Agile IT Project Management Framework”. These process specific solution results are based on literature research. The actual applicability in agile developed projects and adaptations will researched and applied in a following step of this research topic towards the way of creating an optimized, tailored agile IT project management framework.

Key words: *SCRUM, IT-Project Management, Agile, PMBOK*

JEL code: M15 (IT-Management)

Introduction

Published in 2001 the agile manifesto (*Agile Manifesto*, 2001) provided the basis for SCUM framework of agile development in IT projects. The goal was to make development processes more flexible and to achieve early results for customer feedback. But the SCRUM framework as defined in the SCRUM Guide (*SCRUM Guide*, 2017, Schwaber K.& Sutherland) describes only an agile process of software development. It was not meant to be seen as a project management approach.

But in reality, SCRUM is often used as a “agile project management” framework. By adopting agile tools and methods, or sometimes even just terminologies used in SCRUM organizations pretend to use agile project management approaches, without even deeply understanding the real nature of agile project management. However, these organizations are not be blamed. There is no real finalised “agile IT project management” framework existing at the moment. There are classical project management frameworks like PRINCE2 (*Prince 2 Handbook*, 2017, Axelos Global Best Practice) or PMBOK (*PMBOK-Guide*) – *Sixth version*, 2017, Project



Management Institute, Pennsylvania, USA) of PMI organization. And then there are agile development models like SCRUM, which are used in classic project environments.

So when agile IT project management is defined as classical project management, including an agile development approach, problems can develop due to the fact that these two frameworks focus sometimes on completely different values. This cultural inaptitude, often results in decreased overall project success, problems in communication and understanding of project participants.

Basis and approach for this research

This article sets up the basis for an adapted PMBOK project framework specially focussed on agile, with SCRUM, developed IT projects. PMI organisation already took a first step in this direction by adding an “agile guideline” document to its newest sixth version of the PMBOK framework. But this guideline is only an introduction in agility and agile methods and tools. It does not change the processes defined in PMBOK as such.

To now to completely redefine the PMBOK processes and make them suitable for SCRUM developed IT projects two steps need to be taken:

- 1) Critical areas of the PMBOK processes have to be defined.
- 2) Solutions regarding these areas have to be investigated, analysed and evaluated

The first step of identifying critical processes has already happened. In the IEEE publication “Suitability of PMBOK 6th edition for agile-developed IT Projects” (Rosenberger P. & Tick J, 2018) five processes have been identified to cause problems:

- Manage project execution
- Develop project structure plan
- Develop project schedule
- Estimate and define costs based on requirements
- Develop and manage team

This article now uses these identified critical areas as starting point and proposes approaches to be integrated into the existing PMBOK framework. These proposed solutions are based on existing tools and methods identified by literature research and followed by an assessment of applicability using a KPI evaluation. Please note, that the last step of proofing the applicability of the proposed solutions via an large scale online survey is yet not finalised and therefore not part of this article.

Solution proposal for PMBOK process “Manage project execution”

In traditional project management according PMI, a project manager is responsible for managing the project team and its execution (*PMBOK-Guide*) – *Sixth version*, 2017, Project Management Institute, Pennsylvania, USA). SCRUM, as a contrast, demands strict self-management of the development team (*SCRUM Guide*, 2017, Schwaber K.& Sutherland). Only the team itself takes care about delivering quality results. Often these self-managing development teams are even protected from any disturbance or influence by a SCRUM Master. This difference shows the gap between the two frameworks. The agile project manager has to take overall responsibility of the project and the product, but is not allowed to actively manage the execution in regards of programming done by the agile development team.

Three different solution approaches have been investigated and will later be evaluated using suitable KPIs:



1) *Strike System*

Lewthwaite (Lewthwaite, J., 2006) defines a “Strike” as a proactive intervention of a project manager overruling the self-management of SCRUM development teams. This overruling once started lasts the rest of the ongoing sprint. Trigger for such shifts in responsibilities need to be substantial because strikes completely undermine agile culture of self-management and trust. Trigger of such strike events need to be defined in detail to create a common understanding and avoid negative personal feelings as much as possible. Strikes could for example be triggered by:

- SCRUM Master intervention
- Danger of non-deliverable increments at the end of a sprint
- Extreme delay visualized in burndown charts
- Extreme bottle necks visualized on KANBAN boards,
- Great changes in effort estimations of user stories during a sprint in comparison to estimations in sprint planning meetings.

The strike system is therefore a kind of “Management by Exception” methodology.

2) *Indirect Management by Backlog*

Lewthwaite (Lewthwaite, J., 2006) mentioned regarding large scale IT projects that roll like a product manager or project manager acting outside a SCRUM development team can manage project execution indirectly by influencing the product backlog. These roles can change the completeness of user stories and priorities of user stories. With these tools, they can indirectly decide what will be developed next.

3) *Traditional project management of chosen SCRUM artefacts*

Pichler (Pichler, R., 2007.) suggests allowing switching specific SCRUM artefacts or epics from an agile towards a traditional project approach like waterfall. By doing so, a project manager can actively and directly manage the execution of this artefact. It needs to be mentioned, that keeping a well-functioning agile culture alive could get much harder by such interferences. Additional agile tools and methods like daily stand-ups and retrospectives can and should be kept alive, even in these “traditional managed islands”.

After presenting three possible solutions based on literature research a most suitable solution needs to be chosen by application of a pointing system measuring PMBOK in regards of several related success criteria. Points will be assigned based on the applicability of success criteria in regards to PMBOK integration of solution approaches in such way:

- Easy to be integrated into PMBOK processes: 2 points
- Possible but not easily to be integrated into PMBOK processes: 1 point
- Hardly to be integrated into PMBOK processes: 0 points

Note, that this approach of solution selection will be used on all 5 processes



Table 1

Success Criteria	Strike System	Indirect Mgmt.	Traditional Artefacts
Direct PM influence of project execution enabled?	2	0	2
PM influence of project execution measurable?	2	0	2
Usable for all project scales?	2	1 (only large scale)	2
Extensive communication required?	2	1	1
SUM	8	1	7

Conclusion:

After comparison of three potential solutions enabling IT project managers in agile developed projects to manage project execution without disturbing SCRUM processes or culture, it shows that the Strike system is potentially the best candidate to be integrated in this particular PMBOK process. It acts without disturbing SCRUM development at all in most of the time. Only when high involvement of a project manager is needed in projects, it is used to solve issues that the self-managing development team was not able to solve on their own. This management by exception approach combines traditional project management methods and agile frameworks in the least conflicting way.

Solution proposal for PMBOK process “Develop Project Structure Plan” and “Project Scheduling”

Note: Due to the strong relation between PMBOK Processes Structure Plan Development and Project Scheduling”, these two processes are analysed together

Traditional project management structures and schedules the whole project in the initial planning phase. This regards all work packages. Even work packages that are still far away in the future and very uncertain. There is no difference in the level of planning between certain and uncertain work packages accepting that uncertain packages may change in the future causing the project schedule to be adapted. SCRUM totally avoids this restructuring and re-planning by just focusing on the next sprint. This gap in the two approaches can result in major conflicts between agile developments and traditional project managers. Three different solution approaches are now proposed and will later be evaluated using suitable KPIs:

Hybrid Macro and Micro Planning of Project Schedule and Structure

A hybrid approach could differentiate between a macro and micro structuring level - also separating the two cultures. The project manager keeps the overall scope and focus by structuring the whole project like usually with a project structure plan, but only on a macro level. Accepting, not knowing definite responsibilities and durations. But for example only T-Shirt size estimations on an epic instead of a user story level.

But during actual development, in development sprints, micro planning in form of planning poker story point estimations can be used in sprint planning meetings to get into details.

After several sprints, a factor between actual effort and rough T-Shirt size can be postulated. So with experience in project delivery, a project manager could even get quite



detailed effort and structural estimations enabling him to even develop an understanding about longer term planning (Wendt R., 2016).

1) Project Phase Specific Backlogs

This approach does not change the structural planning in the initial project phases at all. A project manager, will create a work breakdown structure and will define project phases and major milestones based on a basic specification in a traditional way. All these major project phases are then seen as “mini-agile-projects” within a traditional project. Each phase has its own specific backlog, SCRUM team and goal. With such an approach, the two cultures can easily coexist. On a big level, managed by a project manager in a traditional way, on the small level in a purely agile SCRUM based approach with minimal project management interference. (The Project Group, *TPG Phase Method, 2019*).

2) Extrapolation and continuous adjustment

When an agile development framework like SCRUM should not be changed, adapted or disturbed at all, a project manager could accept not planning and structuring before the start at all. Just starting the development, when the first user stories are ready for development in the product backlog. Based on a comparison of story points associated before the sprints to user stories and the actual needed time and cost consumption extrapolation about the open efforts and timelines with the currently existing product backlog can be made, fulfilling the PMBOK need to plan, structure and schedule. These extrapolations will get more and more accurate and refined, when more and more sprints have been finalised and learnings from these sprints are available for adjustment of extrapolation. Due to a product backlog that is typically constantly changing in agile projects, the planning and work breakdown structures will also be affected by these changes and the project manager has to constantly keep them up to date. In such a role, the project manager is essentially just “documenting change” and not really managing change, to apply to PMBOK process requirements.

Table2

Success Criteria	Macro/ Micro	Phase specific backlogs	Extrapolation
Effort of planning and work breakdown structure creation and maintenance	2	2	0
Accuracy of planning	1	2	1
Agility and flexibility of planning and structure	2	1	2
SUM	5	5	3

Conclusion:

Planning project work structures and based on those, planning the timing of projects is challenging in agile developed projects, because planning is based on fixed assumptions and agility is based on flexibility. However, the analysis of potential solutions and their applicability shows, that two approaches are especially usable to close this gap between traditional project management and agile development. Phase specific backlogs divide large projects into phases enabling maximal flexibility within the phases themselves. In these phases accepting only macro level planning in the beginning and micro level planning only on sprint level can help traditional project managers to fulfil PMBOK requirements.



Solution proposal for PMBOK process “Estimate and define costs based on requirements”

Cost estimation in traditional projects normally consists about manpower related costs and material related costs. IT projects, basically sharing these concepts with all other projects, often develop most of their costs manpower related. Often the actual time and effort invested by people is much more significant than investments in hardware or other material. Based on this understanding, the cost estimation can also be split in two parts:

- **Material related costs:**
This part of costs are untouched by agile development frameworks
- **People related costs:**
These costs are difficult to estimate and define, because complete and traditional requirements are missing in SCRUM developed IT projects, due to constant backlog changes.

So focussing on people related costs, the following two approaches could be used be integrated into PMBOK processes:

1) Cost estimation based on Development Velocity

Velocity is a key performance indicator of agile development teams, describing the amount of Story Points being developed in each sprint in average. It's the speed, the SCRUM teams are developing with. Often this measurement is also used in portfolio management of agile developed project portfolios (Rouse, M., 2013). Knowing and tracking the velocity of development teams can enable an agile project manager to estimate project costs. Knowing the developers involved and their internal and external hourly rates, the project manager can summarize the cost of one story point, or one average user story based on the amount of user stories developed by the team in one sprint. So relying on the planning of work breakdown structure and project scheduling and knowing the development teams and their costs and velocity the project manager can simply multiply planned development effort with velocity related cost factors and therefor develop a cost planning in the same way as a scheduling. It is important to mention, that the velocity can change and therefor the basis of the cost factor can change. The project manager needs to keep constant track of this factor.

2) Fixed Minimal Viable Product and unplanned ongoing feature development costs

A minimal viable product (MVP) is often used as basic concept of so called “hybrid” IT projects. This MVP is the smallest, fastest and most simple set of features providing desired functionality, without taking care of usability, design, safety, reliability and all other necessary factors of a quality system. In hybrid IT projects the development of such MVPs is planned and executed in a classical waterfall approach, which is easily manageable with PMBOK processes due to the high level of planning activities and rigid structure. After finalization of the MVP increment, features and “quality” is added to the system in a strictly agile way. This hybrid approach of splitting MVPs and agile feature integration can also be used to solve cost estimation gaps in agile projects. Classical cost estimations are used to define MVP costs, and no cost estimation is used at all for agile feature integration later on. This allows strict separation of agile and classical frameworks, avoiding problems (Sharma, S, 2017).

Success Criteria	Velocity based	MVP based
Applicability for different project	2	1



categories and sizes		
Accuracy of cost estimation	1	2
Effort of cost estimation	2	1
SUM	5	4

Conclusion:

Velocity based cost estimation present itself as a usable solution for agile developed projects by building up on planning regarding schedule and project structure and just applying people-related costs. In addition to this solution classical cost estimation methods can be used, when the definition of a Minimal Viable Product makes sense or is defined in a hybrid project management environment. In this case, only the agile part of the hybrid project should use velocity based cost estimations. It is important to mention, that MVPs minimize the agile culture and advantages to a great extent, and the decision to use them should be taken carefully and with intensive communication effort to all project stakeholders.

Solution proposal for PMBOK process “Develop and Manage Team”

According PMBOK(*PMBOK-Guide*) – Sixth version, 2017, Project Management Institute, Pennsylvania, USA), the project manager is responsible for organizing and managing resources, including human resources, for the project. The organizational part is not as critical. A project manager can and will set up a project team and including development teams in the initial project phase. As soon as the development team is set up, it demands self-management. Meaning, that there should not be active management and controlling from outside. This characteristic is a very strong one in SCRUM. The teams share work and task internally and are even “protected by a SCRUM master from outside disturbances. So as soon as a development team is set-up by a project manager the management tasks are taken away from the project manager and are transferred to the team itself. This shift in responsibility can cause trouble in a project and challenge a traditional PMI project manager who needs to take all-over project responsibility.

Two solutions approaches have been identified to close this gap:

1) *Adaptation of Strike System for team management*

As described in the first process, Lewthwaite (Lewthwaite, J., 2006) mentions a strike system as a potential compromise to share responsibility between self-managing SCRUM teams and outside project managers. This approach can not only be used in project execution, but also in processes of team-management. Potential trigger of Strike-Situations, in which the project manager will pause self-management of the team and take over, could be retrospective meetings, in which problems within the development team are discussed. It is important to define clear situations within that retrospective to start a strike-action. Otherwise, development teams will always hesitate to solve problems within the retrospective meeting in fear of a potential loss of self-management.

2) *Project Manager takes role as SCRUM Master*

If a project manager is comfortable in only being “inside” of a self-managing development team but accepting their demands of self-organization, he or she could take the roll as SCRUM master. Within this role the project manager can actively trigger team-problem-solving in retrospective events or even on a daily-stand-up basis. This realization of critical situation and the start of a problem solving process is often more than enough to keep projects



and team structures productive, even without acting as authority and directly manage and decide changes.

Success Criteria	Strike System	PM as SCRUM Master
Applicability for different project categories and sizes	2	2
Interference with agile culture and methods	1	2
Amount of influence by PM if necessary	2	1
SUM	5	5

Conclusion:

Comparing both solution proposes with each other and assessing their applicability with SCRUM methods as well as their ability to be integrated into PMBPOK processes, the KPIs show no favourite solution. So a potential approach combines both solutions. A project manager acting as SCRUM Master in non-critical project phases, triggering team communication and problem solving when necessary, but at the same time, being able to take corrective measures, using authority by applying the strike system.

PMBOK Integration of Proposed Solutions

This article searches for solutions regarding PMBOK processes not suitable for SCRUM developed IT projects. In the article “Suitability of PMBOK 6th edition for agile-developed IT Projects” (Rosenberger P. & Tick J ,2018) critical PMBOK processes have been identified. Based on literature research and a KPI based evaluation of possible solutions, the following methods can be used for PMBOK integration:

- **Manage project execution**
Strike Systematic enables the project manager to take full responsibility, via management by exception only when necessary, so the agile culture of self-management and personal responsibility is not disturbed at all as long as not necessary.
- **Develop project structure plan and Develop project schedule**
Phase specific backlogs divide large projects into phases which can be easily planned by an experienced project manager. Within these phases, only macro level planning at the phase beginning and micro level planning and structuring on a sprint level is necessary.
- **Estimate and define costs based on requirements**
Cost estimation based on agile structure planning and scheduling using velocity estimations of development teams and their manpower related costs can be used by project managers to handle work-related costs. No changes are necessary in estimating and defining material related costs.
- **Develop and manage team**
Project Manager acts as SCRUM Master passively influencing team related problem solving and using temporary Strike System to stop classical SCRUM rolls and approaches and using management authority to manage projects as long as necessary.

Please note that these results are only based on detailed literature research and KPI assessments, but not proven to be effective in agile environment. In a next step of research, the effectiveness



of these solutions will be measured by optimizing the PMBOK processes including the new solutions.

References

- Agile Manifesto*, 2001, [Online] Available at: <https://agilemanifesto.org/> [Accessed 11.03.2019]
- Lewthwaite, J., 2006. *Managing People for the First Time: Gaining Commitment and Improving Performance*. Thorogood Publishing
- Project Management Institute, *A Guide To The Project Management Body Of Knowledge (PMBOK-Guide) – Sixth version*, 2017, Project Management Institute, Pennsylvania, USA
- Pichler, R., 2007. *Scrum - Agiles Projektmanagement erfolgreich einsetzen*, 1 edition. ed. dpunkt.verlag, Heidelberg
- Prince 2 Handbook*, 2017, Axelos Global Best Practice, TSO The Stationary Office
- Rosenberger P. & Tick J., *Suitability of PMBOK 6th edition for agile-developed IT Projects*, 2018, CINTI 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics, Budapest Hungary
- Rouse, M., 2013, *Agile velocity* [Online]. Available at <http://whatis.techtarget.com/definition/Agile-velocity>. [Accessed 11.03.2019]
- SCRUM Guide*, 2017, Schwaber K.& Sutherland J., [Online] Available at: <https://www.scrumguides.org/index.html> [Accessed 11.03.2019]
- Sharma, S., 2017, *The DevOps Adoption Playbook. A Guide to Adopting DevOps in a Multi-Speed IT Enterprise*. Wiley, s.l.
- The Project Group, *TPG Phase Method*, [Online] Available at: <https://www.theprojectgroup.com/en/services/project-management-consulting/ms-project-implementation.html> [Accessed 11.03.2019]
- Wendt R., 2016, *Hybrides Projektmanagement für agile Projekte in mittelständischen und großen Unternehmen*, [Online] Available at: http://www.masventa.eu/fileadmin/user_upload/media/pdf_de/PMI/PMI-SG-Live_2016-12__1_.pdf, [Accessed 11.03.2019]