In defining a process framework founded on simple principles and humanist values, Scrum has played an influential role in agile IT software development. Consistent with many other agile methodologies, there is, however, a prevailing belief that simply being agile suffices to manage risk.

We not only propose that risk be explicitly managed, but that it be done so in an agile manner and are confident that Scrum already possesses many of the tools and techniques required to both mitigate threats (negative risks) and to exploit opportunities (positive risks).

This publication surveys the nature and perception of risk management in the Scrum community and provides a brief introductory overview of the principles and practices of agile risk management and how they relate to Scrum. Further information concerning this topic, together with a full calendar of IARM events, can be found at http://institute.agileriskmanagement.org/.

Dr. Alan Moran
Managing Director, IARM

2014
Table of Contents

Scrum Overview
Understanding Risk
Why Risk Management Matters
Agile Risk Management
Managing Risk in Scrum
Benefits
Further Information
Scrum Overview

Playing on sporting metaphors, the term "Scrum" is intended to evoke the collectivism of continual and incremental effort that contrasts with the “relay-race” of phased development approaches, which separate out and linearize the software development lifecycle. Scrum is best described as a process framework for the agile development of software products, rather than a methodology that prescribes specific tools and practices. Rooted in empiricism it is founded on notions of transparency (e.g., common standards), inspection (e.g., detection of variances) and adaptation (e.g., process correction and alignment) that express well the central tenets of agility (e.g., iterative and incremental development, feedback loops, self-organizing and cross-functional teams). Developed during the 1990s, it has today established itself as the most popular expression of agility within the IT industry.

The principles of Scrum capture both humanist and business values such as courage, openness and respect, as well as focus and commitment. Its set of official practices is limited to iterative development (referred to as sprinting), planning, daily scrum meetings and review and retrospective exercises. These are frequently augmented with other optional techniques such as Scrum-ban (essentially the use of a Kanban board) and burndown charting. Although iterative development and incremental delivery are not as clearly delineated as in other agile methodologies, it is clear from the descriptions of planning activities that there is at least an implicit division between the two.

Scrum teams are organized into the roles of Scrum Master, Product Owner and the Development Team. The Scrum Master is in essence both a process owner and a process manager who ensures that practices are understood and supports them through facilitation and coaching. The Product Owner is the sole custodian of the project requirements and is accountable for their clear expression and value alignment within the organisation. Finally the Development Team is responsible for transforming requirements into deliverables that constitute potentially releasable functionality. Typically the Development Team is a small, self-organizing and cross-functional unit that is jointly accountable for its work. It relies on neither titles nor substructures in order to operate.
Fixed length iterations, referred to as Sprints, constitute the primary arena for all Scrum events, each of which enables inspection and adaptation of the process. Chronologically, the first of these is the Sprint Planning event, during which deliverables and the work required to deliver them are determined. Thereafter the team meets on a daily basis (the Daily Sprint) to briefly (i.e., no more than fifteen minutes) discuss their contribution towards the Sprint goal, the planned activities for that day and to declare any impediments that lie in their path. At the end of a Sprint there is a Sprint Review meeting to determine the state of affairs of items on the Product Backlog and propose any necessary adaptations, gather feedback on the iteration itself and review other details related to the project (e.g., budget, timelines, next steps). Finally the Sprint Retrospective is an opportunity for continual process improvement (e.g., by taking a closer look at relationships within the team, use of processes and tools) and how it can be implemented in the next Sprint.

Requirements are derived from the Product Backlog (an open-ended list of well-defined and understood requirements, enhancements and fixes that are estimated and assigned value) and used to create the Sprint Backlog that communicates, forecasts and monitors activities in greater detail on a daily basis. Much of the work at this stage is focused on gaining consensus regarding understanding of the requirements and how they should be implemented (incl. design) and tested during the course of which a well-defined and coherent Sprint goal is determined and formulated.

Alternatively the Scrum process can be represented as an agile chart in which movement around each circle implies multiple iterations of its inner circle (see our website for more information about the agile charting technique). Thus in the figure below, which depicts a simplified version of the Scrum process, each release comprises of one or more Sprints each of which involves one or more days.

Agile charts are both an effective means of communicating intent, as well as soliciting feedback (e.g., by leaving the chart in a communal area and inviting team members to annotate it for discussion within the Sprint retrospective). Some activities in an agile chart are recurrent in nature (i.e., they occur in differing levels of granularity at multiple cycles) and can thus be contained within a slice (e.g., planning slice in the above figure).
Understanding Risk

Risk is uncertainty that has an impact on project objectives in either a positive or a negative manner. The fact that not all uncertainty is either relevant or negative, is summed up in the analogy, that whilst a horse race may well be uncertain, it only becomes a risk once a bet is placed, the outcome of which may result in losses (threat) or winnings (opportunity).

In order to manage risk it is therefore necessary to not only understand the project objectives, but also the project context and risk environment. The project context determines how risk is to be interpreted and the primary drivers of risk (e.g., regulatory, technology, business), whereas the risk environment reflects what constitutes acceptable levels of risk (e.g., dispensations for research and development activities). Together these help assess the risk profile of a project in terms of the wider attitude towards, and tolerance of, risk within the organisation. This enables management to compare projects on a risk basis and to balance risk and reward.

Whilst the understanding of risk as process variability might well be unique to Scrum, much of its literature implicitly frames risk negatively and limits its scope to requirements change and matters of technical implementation. Moreover in keeping with most agile methodologies, there are no explicit measures to manage risk, relying instead on the implicit belief that “being agile” suffices.

Why Risk Management Matters

It is telling that whilst Scrum team members can often tell what it is they are working on (e.g., user stories), how important their tasks are and what constitutes completion (e.g., definition of done), they are seldom able to articulate the implications of their work with regard to managing risk. Recognising and assessing threats (negative risks) and opportunities (positive risks) and determining appropriate steps to treat them, are therefore essential in order to balance risk and reward within a project. Monitoring the effectiveness of risk management by tracking how risks are managed, ensures feedback into the process and allows all involved to learn from mistakes and to improve the overall risk management process.

Failure to adequately address risks within a project can lead to:

- Inability to make informed risk and reward decisions.
- Failure to identify appropriate risk response strategies based on risk exposure.
- Lack of oversight in risk monitoring leading to ineffective or inefficient treatment of risks.
- Poor understanding of when to engage in risk activities.

Social and cultural influences on risk management constitute an important, but often neglected, aspect of the analysis and treatment of risk within the organization. Given the humanistic values of Scrum it is therefore fitting that these influences are also taken into account.
Agile Risk Management

Agile risk management is concerned with the identification, assessment, prioritization, treatment and monitoring of project risks in a manner consistent with agile principles and practices. It considers not only threats (negative risks) but also opportunities (positive risks) in the context of the enterprise attitude towards risk and employs its own techniques and practices to inform decision making to balance risk and reward. Extending the agile mantra of “embrace change”, agile risk management encourages practitioners to “embrace risk”.

Agile risk management is founded on the following principles:

- **Transparency**: Make visible and accessible all risk artefacts used to understand, communicate and manage risks within the project.
- **Balance**: Establish clarity about the nature and distribution of risk and reward throughout the project.
- **Flow**: Ensure that risks do not inhibit the project and that the agile process itself is capable of withstanding perturbations arising from risk.

Application of the agile risk management process involves establishing an understanding of the nature of risk facing a project and adapting the existing agile process to better cater for risk management. The agile risk management process comprises of the following stages:

- **Understand project objectives, context and risk environment.** This stage is about achieving an understanding of the environment in which the project operates. This is a necessary prerequisite to frame risk management practices and assist in clarifying the relevance of risk within the project and in relation to the organization (e.g., the need for risk dispensations).
- **Risk Scoping. Identify Risk Drivers and Appetite.** By scoping the project risk drivers, the primary sources of risk and the institutional attitude towards and tolerance thereof can be established and communicated in a clear manner. This ensures the foundations for alignment of personal and institutional attitudes towards risk-reward behaviour.
- **Risk Tailoring. Embed Risk Management in Agile Process.** The dynamic view of agile process being employed is charted in order to determine the most appropriate positioning of risk management activities (e.g., risk identification workshops at the start of an iteration and risk retrospectives at the end). In light of the risks facing a project, additional measures may also be proposed (e.g., application of specific agile techniques to tackle risk).
- **Risk Management. Identify Analyse, Manage and Monitor.** This encompasses the operational aspects of risk management within the project albeit imbued with an agile character (e.g., use of risk modified Kanban boards to visualize the distribution of risk and reward, tagging of activities to indicate the application of an agile technique in order to treat risk and communal ownership of risk artefacts).
The result is a risk optimized agile process that empowers the project team to explicitly understand, articulate, manage and track risks as they arise in the project. The application of the agile risk management process to a variety of agile methodologies including Scrum, XP, DSDM, SAFe and DAD has been described in the literature.

Managing Risk in Scrum

The following picture describes an approach to agile risk management that suits Scrum teams using a Kanban board approach (also known as a Scrum-ban) in their work. Omitted are the details of risk scoping and tailoring which usually take place before start of the project.

Identifying Risks

Identification of risks is harder than one might imagine as the biggest problem is conflating uncertainties and effects. For example, the risk in a website migration lies not in the lack of availability of the site afterwards (which is the effect of an unsuccessful migration) but rather the uncertainty surrounding the circumstances that led to its unavailability (e.g., how to configure DNS). A simple but effective technique for risk identification is to brainstorm what might occur (effects) and then in each case, ask why it might occur (risks).
Once identified risks should be recorded in a risk log (e.g., description, inherent risk exposure) to which will be added further information (e.g., risk response strategy and treatment, residual risk exposure) later.

**Risk Analysis, Prioritization and Treatment**

The purpose of risk analysis is to determine a course of action and prioritize it accordingly. Risks should be assessed in terms of likelihood and impact (together these are known as risk exposure) for which T-shirt sizing (i.e., S, M, L) usually suffices. As a first step, the risk as originally encountered (inherent risk) should be estimated and later reassessed once a treatment has been determined (residual risk). Sometimes the treatment of risk introduces entirely new risks (secondary risks) and thus risks in practice are linked in a complex web of causality. Bear in mind, that range estimates of risk exposure components are perfectly acceptable if these serve as the basis for discussion within the team.

It is important to understand the limitations of risk assessment techniques such as asking people (e.g., hidden agendas, confirmation bias), using past data (e.g., might not be indicative of future trends) or probability models (e.g., hidden assumptions) so whenever an assessment is made, it should be challenged.

It is a common misconception that high risk must imply high reward. In fact, what should really be asked is whether or not the reward implied by a story or task warrants the level of risk it entails. Or in other words: is it possible to achieve the same level of reward for less risk? This helps better prioritize user stories on the backlog as risk becomes an influencing factor in prioritization though it is never the primary determinant of ordering on the backlog.
The following six risk response strategies are commonly used in risk management to determine the nature of risk treatment:

- **Accept** Undertake no action to manage the risk, but instead have a contingency plan in place in the event that the risk is realized.
- **Exploit/Reduce** Enact measures to increase/decrease either the likelihood or the impact of the risk.
- **Share/Transfer** Endeavour to share/transfer the risk to other parties in exchange for a share in the rewards or a fee for assuming the risks.
- **Avoid** Refrain from taking part in the task that gave rise to the risk.

Strategies can be chosen largely on the basis of risk exposure which helps mitigate some of the social and cultural influences on risk management. The following chart can assist in determining which strategies are most appropriate though it should always be used as a basis for discussion owing to the uncertainties inherent in risk analysis (e.g., range estimates).

Once a strategy has been determined the next step is to determine concrete measures to treat the risk. The following options are available:

- **Do nothing (but plan)** Accept that the risk might occur and think about what would need to be done if it were realized. This becomes an optional task on the Kanban which might never be needed.
- **Risk Tasking** Create a task that deals with the risk (e.g., exploit, reduce, share or transfer it). These tasks are just like any other task in Sprint planning. It is very helpful to colour code such tasks (e.g., red for reduction, green for exploitation) so that the distribution of risk and reward can be visualized on the Kanban board.
- **Risk Tagging** This refers to the selection of an agile technique specifically chosen to cater for a risk (e.g., pair programming) and which is applied to a class of activities (e.g., all GUI related tasks). Tagging involves placing a mark next to each affected task to remind the team of the technique to be applied.
- **Task Dropping** Remove the task from the Kanban that is giving rise to the risk.
Risk Monitoring

Risk monitoring provides a visual cue of what is being done to tackle risk whilst at the same time acknowledging the systemic nature of risk within the project.

Risk monitoring requires the assigning of scores to risk exposure bands when assessing both inherent and residual risks. For example, the inner region of the risk response strategies chart might be assigned two points, the middle four and outer six. The amount of risk mitigated thus reflects the difference between inherent and residual risk scores and serves as the basis of a risk burndown for the Sprint.

Whilst this coarse approach may serve most purposes it is worth noting a couple of details. First, not all risk can be mitigated and thus a certain amount of iteration residual risk remains despite the risk treatment efforts. Secondly, risk may go up as well as down as the impact of secondary risks become apparent during risk management.
Benefits

The benefits of agile risk management include the following:

- Improved capacity to manage project uncertainties that would otherwise threaten revenues or impede efforts to exploit opportunities that arise within projects.
- Enhanced communication of the nature and sources of risks facing projects within the organization, leading to an improved culture of awareness and understanding of the need to balance risk and reward.
- Better alignment of project and enterprise risks that promote the transfer of lessons learned between projects of similar scope and complexity.
- Empowerment of project teams, enabling them to accept responsibility for the identification analysis, prioritization, treatment and monitoring of risks within their projects.
- Appreciation for the social and cultural influences on risk management, (e.g., risk compensation effects) particularly in environments where these can play a dominant role (e.g., geographically dispersed teams).
- Ability to extend and enhance existing investments in Scrum by tailoring and embedding risk management practices into daily activities.

Further Information

Agile Risk Management (Springer Verlag)

This work is the definitive guide for IT managers and agile practitioners. It elucidates the principles of agile risk management and how these relate to individual projects. Explained in clear and concise terms, this synthesis of project risk management and agile techniques is illustrated using the major methodologies such as XP, Scrum and DSDM. Available on Amazon and through all book stores.

For further information concerning the principles and practices of agile risk management, including a full calendar of training events, please contact the Institute for Agile Risk Management:

http://institute.agileriskmanagement.org/
info@institute.agileriskmanagement.org/
The Institute for Agile Risk Management (IARM) is a Swiss based institution that exists to promote the principles and practices of agile risk management in the context of agile project management and the agile enterprise. It undertakes research and provides training in association with its network of third parties in the agile and academic communities as well as in the private sector. For further information concerning the institute please refer http://institute.agileriskmanagement.org/