

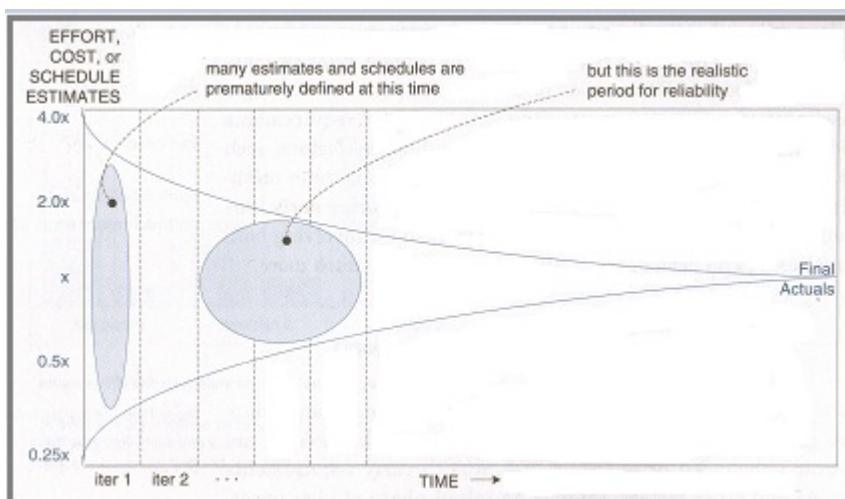
Agile Estimating, Planning and Scheduling Techniques relevant to Project Communication Management

According to PMI (2008), the Project Communications Management (PCM) knowledge area consists of 5 processes that are usually run in the following sequence:

1. Identify Stakeholders
2. Plan Communications
3. Distribute Information
4. Manage Stakeholder Expectations
5. Report Performance

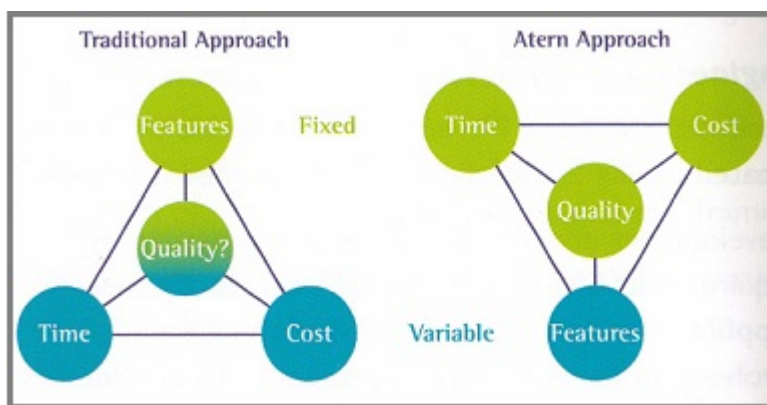
Agile estimating, planning and scheduling techniques as described in this blog are similar in principle to the above PCM processes, however, they tend to be less rigid and sequential as the PCM processes by PMI (2008). The overlap can be found especially in the Manage Stakeholders Expectations process where interpersonal, soft management & communications skills and techniques are widely used, which is typical of agile philosophy. This may indicate adoption of agile practices in the mainstream of project communication management as West & Grant (2010) suggest in the Forrester report. This report shows that large organizations have a tendency to mix and combine agile approaches with plan-driven methodologies. This was also supported by Pikkarainen et al (2008) who pointed to the fact that it is useful for an organization to use additional plan-driven practices so as the efficiency of external communication towards the outer actors is ensured.

Agile planning is carried out on agile projects throughout the duration of the project with an appropriate granularity for daily, iteration and release levels. According to Cohn (2006) and his Planning Onion concept, it is not useful or effective for the agile team to plan beyond the release plan level. As the initial phase tends to be the phase with most changes, Larman (2004) suggests that exploratory planning is more effective during the first iteration followed by adaptive planning rather than predictive planning in subsequent iterations. The initial phase is characterised by a high-level of uncertainty due to the lack of information and high frequency of early requirement changes typical for the beginning of the project, which is then lessened later in the project as information accumulates resulting in more accurate planning decisions. See the Cone of Uncertainty diagram depicted by McConnell (1998) below:



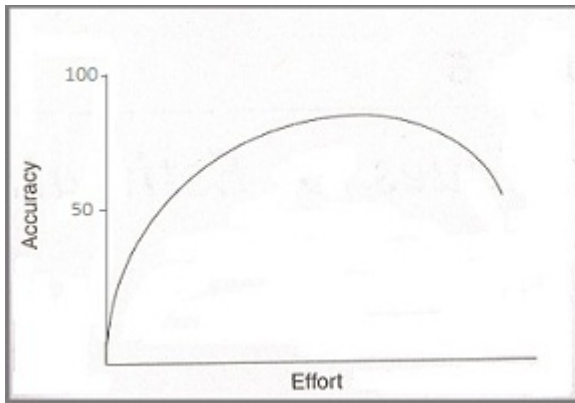
Each agile approach has slightly different length of iteration and planning procedures. For an example SCRUM is very precise about the iteration length, which is 30 days, Schwaber (2006). Scrum's evolutionary release cycle consists of 3 Scrum iterations called Sprints. Planning is vision-oriented approach with each stage starting with items prioritization. In collaboration with project the architect and developers, each item is assigned a tentative estimate. There is a rule that no item is larger than 3-person days of effort.

The DSDM Atern Delivery Planning is based on iterative, incremental and time-boxed approach where the features constraint is subject to variation based on prioritization whereas the time, budget and quality constraints are fixed. On the other hand, the traditional project management tends to have the features fixed but the time, budget and quality constraints are not so fixed as many traditionally managed projects show in reality with projects delivered late and over budget – see the DSDM (2008) diagram below for the comparison between traditional and Atern approach to project management constraints:



According to Beck (2006), planning in Extreme Programming is based on the XP Planning game. The team that consists of one or two customers, developers and the project manager called monitor are going through planning iteratively and incrementally averaging estimates among group members. The agile approach to the team work is based on social cohesion where each member of the team is motivated to deliver his or her best based on the collaborative approach to work founded in the no-blame culture as described by Whitworth & Biddle (2007). Each user requirement called user-story in XP is assigned a number of points depending on its complexity to develop. Planning based on story points tends to be more accurate than planning based on man-days, Cohn (2004). Agile team's rate of progress called velocity is defined as the sum of all story points that are completed in a single iteration. As project moves ahead iteration by iteration, based on past progress against planning, new estimates for future iterations are recalculated/re-estimated.

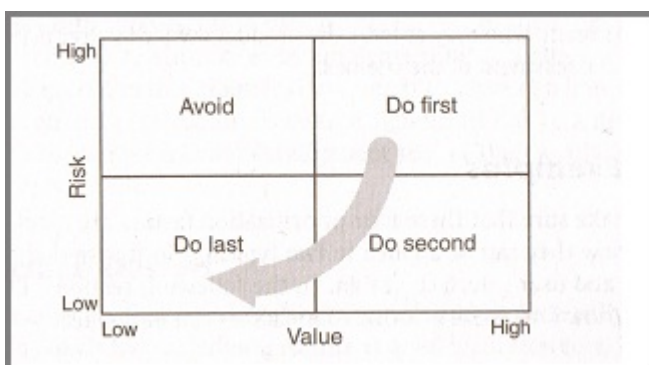
There are myriad techniques used for agile estimating, such as expert opinion, analogy, splitting complex stories, averaging of individual estimates to name a few that are the most used ones. The choice depends on the specific circumstances on each project. For an example, if there is an expert on the project, it is naturally useful and cost-effective to use his or her expert opinion, similarly if there is historical data available, it is highly effective to use it. Data splitting can be done many ways, such as across boundaries, operations or function/non functional requirements. Even though combining different estimating methods leads in principle to better estimates, it is suggested by Cohn (2006) that after a certain point, more effort may actually lead to decreasing accuracy – see the Cohn's Effort Accuracy diagram on the next page.



Due to the time, budget and quality constraints on all projects, it is necessary to prioritize. In order to do it effectively, user stories or features are grouped into themes which are then prioritized according to a number of factors. According to Cohn (2006), there are 4 main factors according to which estimating and planning can be prioritized:

1. Value
2. Cost
3. Learning
4. Risk

These factors can be combined during the prioritization process. Cohn (2006) suggests that high risk and high value features should be developed first, followed by low risk, high value and low risk, low value. The high risk, low value features should be left until the last to implement or completely avoided as depicted in Cohn's diagram below:



In summary, agile planning is based on open and transparent communication. In terms of visibility, both iteration and release plans are visible to everyone. Based on the principle of trust in agile teams, it is important that developers provide realistic estimates and product owners realistic dates. Agile planning is then conducted in an iterative and incremental way following the time-boxed approach to ensure that time, budget and quality constraints stay fixed whereas the features constraints is variable determined by the prioritization process.

References:

- Beck, K. (2006) Extreme Programming Explained: Embrace Change. 2nd edn. New York: Addison Wesley
- Cohn, M. (2006) Agile Estimating and Planning. New York: Prentice Hall
- Cohn, M. (2004) User Stories Applied for Agile Software Development. New York: Addison Wesley
- DSDM (2008) DSDM Atern. DSDM Consortium
- Larman, C. (2004) Agile & Iterative Development: A Manager's Guide. New York: Addison Wesley
- McConnell, S. (1998) Software Project Survival Guide. Microsoft Press
- Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J. (2008) 'The impact of agile practices on communication in software development.' Empir Software Eng 13, pp 303-337
- PMI (2008) Project Management Body of Knowledge: PMBOK Guide. 4th edn. PMI
- Schwaber, K. (2004) Agile Project Management with Scrum. Redmond: Microsoft Press
- West, D. & Grant, T. (2010) 'Agile Development: Mainstream Adoption Has Changed Agility'. Forrester
- Whitworth, E. & Biddle, R. (2007) 'Motivation and Cohesion in Agile Teams'. XP 2007, LNCS. 4536, pp. 62-69