Appropriate Process Approaches

Paul Oldfield

Mentors of Cally

Member of the Appropriate Process Movement

http://www.aptprocess.com

Abstract

This whitepaper summarises the approach for Appropriate Process.

It is Work In Progress.

The author will be happy to receive your comments.

Copyright Note

This document resides online at http://www.aptprocess.com/whitepapers/appropriateprocessapproaches.pdf and has been authored by Hüseyin Angay of Karabash Ltd. and the Appropriate Process Movement. It may be copied freely in part or in whole, with the restriction that anywhere using a copy of more than three paragraphs must include as reference the web address of its origin, as given above.

Copyright 2003, Appropriate Process Group

Contents

Abstract 1
Copyright Note 1
Contents 2
Appropriate Process – The Approaches 3
Getting Started – Characterise the Project 3
Criteria – Definitions and Scales 4
Criteria – Impact, Approaches, Interdependence 8

Copyright 2003, Appropriate Process Group

Appropriate Process – The Approaches

The overall guidelines for Appropriate Process is that

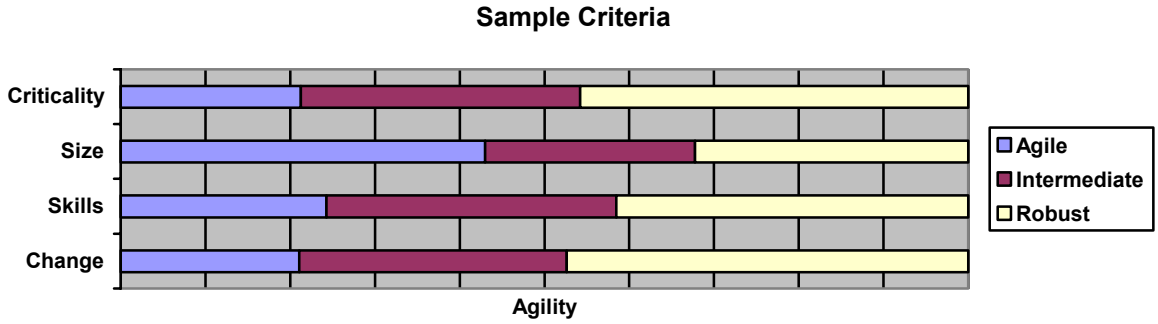
A process should be as agile as possible, and as robust as necessary

These are the conflicting forces that need to be balanced to find the 'sweet spot' for a process that is appropriate to a situation.

Getting Started – Characterise the Project

This activity is based on an approach described by Alistair Cockburn. Various other authors have variants of the approach but few have taken it to the depth of detail Alistair gives in his Crystal family of methodologies. There are several key criteria that characterise the project. Some of these are fixed; some are theoretically mutable. Some of the characteristics interact with each other. Some are more significant than others.

At one end of the scale of methodologies we have rigorous, heavyweight, robust, plan-driven methodologies. At the other end of the scale, we have minimalist, lightweight, agile methodologies. This is a simplistic picture, methodologies can to some large extent be constructed from techniques that themselves fit at different places on the agile – rigorous scale.



This table is just sample data – real criteria, real figures, will vary, and the type of rigour needed, or of agility enabled, will vary. The position on the scale for each criterion will vary between projects, and as a result, each project should have its own process. Each project will have its own ideal process, but the tailoring also needs to take into account attainability and cost vs. benefit of changing process.

Copyright 2003, Appropriate Process Group

One scale that tends to confuse some readers new to the concept of Agility is that of Discipline. An agile methodology can be high discipline without being high rigour. A rigorous process takes a disciplined approach to all aspects of development; everything is cross-checked. An agile process may choose to cut out a lot of checks and steps in the process, but demand discipline in the steps that remain. Thus discipline is orthogonal to agility while rigour is at the other end of the scale from agility. I prefer not to call it opposite to agility. Discipline attempts to ensure agile does not mean fragile. However, one of the keys to agility is the understanding that we can allow things to break and fix them when it happens; this can be a good approach where the cost of preventing the breakage is high, and fixing it low.

Criteria – Definitions and Scales

Definitions or descriptions of the key criteria are given, and scales against which they can be measured are indicated, where possible.

Criticality

The scale of criticality takes several range delimiters. Failure at the most lax (i.e. agile-friendly) end of the range would affect comfort only. In order of increasing criticality we have loss of discretionary funds, loss of critical funds, physical injury, loss of life, loss of multiple lives. One may wish to tweak the scale where loss of reputation is a factor. Some authors choose not to distinguish between loss of life and loss of multiple lives; I place either possibility as requiring maximum rigour.

Expertise

There are many variants on the scale, I favour the Student – Journeyman – Master model. Cockburn uses Level 1, Level 2 and Level 3 practitioners that more or less correspond to these; Boehm and Turner split Level 1 into 1a and 1b. Another writer talks about a progression from Unconscious Ignorance through Conscious Ignorance and Conscious Competence to Unconscious Competence. As a rule of thumb, a Student has no more than 3 years experience since graduating and needs supervision even to follow a detailed written process. A Journeyman can usually be trusted to follow a process unsupervised, and can select appropriate variants of a process to suit the situation. He knows when it is safe to cut corners. A Master will have at least 12 years experience, but some people never achieve

Copyright 2003, Appropriate Process Group

Master status no matter how much experience they get. A Master doesn't follow process. He understands what needs to be done and decides what to do based on the situation. True masters are rare, and nobody has full mastery of the full range of skills. However, a Master knows what he doesn't know, and will know how to locate and apply the necessary expertise.

Again, putting numbers on the scale results in many different opinions. The skill criterion scale has a rather complicated set of range delimiters that refers to the proportion of each class of skill that exists in the workforce. The ones I give here are very approximate, giving a feeling for the problem rather than strict guidelines. If we take the percentage of 'Student' or Level 1 practitioners in the workforce, where it is assumed around 90% of the rest will be Journeymen, the remainder Masters, then agile methodologies will require as few as 40% Students; and a 90% Student workforce will need a detailed, cross-checked process at the rigorous end of the scale.

It should be noted that some agile methodologies are more tolerant of low levels of expertise than others. A low expertise team will make slower progress than a team of higher expertise when these 'tolerant' methodologies are employed. These will be high discipline methodologies.

Geographical Dispersion

Agile methodologies favour having the entire team in the same room. Range delimiters proceed through same building, same city, same country to same planet. At the most dispersed end, agility is not totally out of the question, it has just become considerably more difficult to achieve.

Size

The number of people working on a project is a significant factor in how much agility can be introduced into the process. Again, opinions vary on the range delimiters, but the shape of the 'graph' is fairly consistent between writers. A team of around 3 has the highest potential to be agile; range delimiters proceed through project sizes of 7, 18, 36, 70, 120 to 200 and larger. In larger teams there may be localized cells of agility, but it is unlikely that the benefits of having agility coherent across the whole project can be achieved.

Copyright 2003, Appropriate Process Group

Rate of Change

The rate of change of requirements is a determinant of how much agility is necessary on a project. One can choose to employ more agility than necessary, but the project will be at risk Boehm and Turner give their figures as a percentage of the requirements that change per month, though it is unclear what they would call a change in requirements. It is clear that some changes have more impact than others. Here is a rough sequence progressing from most impact to least; the sequence is a matter of opinion, others may differ. Most impact is new scope-changing requirements, followed by new architectural requirements, new major functional requirements, dropped scope requirements, dropped architectural requirements, dropped major functional requirements, reprioritised scope requirements, reprioritised major functional requirements, new detail requirements, and dropped detail requirements. The sequence of impact severity will vary with the chosen process. I assume architectural requirements are always high priority and cannot be reprioritised.

Boehm and Turner give the figures for range delimiters as 50% changing per month requiring extreme agility, through 30%, 10%, 5% to 1% change per month that can probably be handled by rigid, non-agile processes. Clearly a relatively heavyweight process should be able to handle high percentages of new detailed requirements per month, but we would not expect it to handle scope or architecture changing requirements at anything like so high a volume.

Culture

Boehm and Turner provide a culture scale with extremes at the agile end of "thriving on chaos", and at the rigorous end of "thriving on order". Jim Highsmith indicates that the most agile teams are those that tread the boundary between order and chaos, where any more order limits creativity and response to opportunity, while any more chaos will lead to good results disintegrating and slipping out of one's grasp. Perhaps the best measure to use is "tolerance to change in direction". A high tolerance to change in direction is necessary for agile approaches. Cultures that have a low tolerance to change in direction would suit more rigidly defined processes and would need to accept that they may be limited to projects where there will be little need to react to changes in direction, the product will be that which was intended at the start of the project. It would be hard to put

Copyright 2003, Appropriate Process Group

meaningful range delimiters on this axis, I have named the extremes and shall let that stand.

Priorities

Cockburn distinguishes different goals for the development process, Highsmith indicates that though multiple goals are possible, a single goal should take overriding priority. The Agile Alliance places delivery of working software as the highest goal. There should be other goals that occasionally take higher priority. In safety critical projects, delivery of working software is not of much use unless it is known that the software is safe. In such circumstances, traceability and audit compliance may become higher priorities. Again, it is hard to put figures on the axis, but it is clear that altered overall priorities will impact process and the degree of rigour required.

System Character

A system may be characterised as data-rich, behaviour-rich or control-rich. In addition it may have additional constraints such as stringent performance needs, real-time behaviour and/or embedded constraints. The early off-the-shelf agile methodologies were unsuitable for these special needs projects, being tailored for data-rich projects. However, there are now add-in techniques such as those promoted by Agile Modeling that can add the right amount of rigour in the right places. Again, it would be hard to put figures on the axis, but the agile end would be a straightforward data-rich project, the rigorous end would be a control-rich real-time embedded project. Note that even the rigorous end of the scale can be performed in an agile manner by adding no more rigour than is necessary.

Duration

The expected length of a project is a significant criterion when judging agility. However, it must be noted that agile projects can chop long duration projects into prioritised increments that require considerably less rigour. There remain a few projects where unless the project goals are all addressed, there is no real benefit to the customer. These are special cases that require higher rigour approaches to deal with the problems of extended duration. A rule of thumb is that many projects of over 2 years duration fail, and most projects of over 3 years duration fail. The conclusion, however, is ambiguous; a longer duration project either needs extra agility or extra specific rigour.

Copyright 2003, Appropriate Process Group

Criteria – Impact, Approaches, Interdependence

Criticality

Criticality addresses the impact of failure to deliver working software, or failure to deliver adequate quality of software. What do we need to do for high criticality projects? If the project is optional, extra care must be taken before committing to each stage in the project. In all cases, extra care must be taken to ensure the development process does not introduce any faults into the product. High rigour, high discipline and adequate expertise are required in cases of high criticality.

Expertise

In all cases, all projects can benefit from adequate skills transfer techniques. These are useful even where all project members are expert, because there will be new knowledge gathered as the project progresses, and this knowledge should be disseminated. Collaboration is always a good transfer technique, people learn as they work together. Where expertise is thin on the ground, techniques to use the available expertise to best effect are needed. More detail will need to be added to the techniques, and more cross checking or feedback to ensure they have been applied correctly. Ideally, care should be taken that people know not only what they need to do, but also why it needs to be done.

Self-organizing teams are ideal for the less strictly disciplined forms of agility, but teams need a threshold level of expertise before they can organize themselves effectively.

Low expertise and high criticality are not a good mix. Low expertise tends to increase the number of people required on a team, sometimes as much as four-fold. Alternatively, a team with low expertise will take longer than a team of the same size with higher expertise; duration is impacted. There is a rule of thumb that in an individual expertise can increase productivity 20-fold but doesn't increase wages more than 5-fold. Nurture and buy in good, broad expertise, it usually pays its way. Choose those with a deep understanding of why things are done in the way they are done, if agility is needed.

Geographical Dispersion

Copyright 2003, Appropriate Process Group

Geographical distribution impacts on communication and collaboration. Any distributed project will need some measures to allow communication and collaboration. This is particularly necessary for agile methodologies that, in general, require higher levels of communication and collaboration. Tools and techniques such as videoconferencing and computer-supported co-operative work can ameliorate the situation, but co-location is always preferable.

Size

Expertise and agility can reduce the number of people needed on a project by some considerable amount, yet some projects remain large. There is a limit of approximately eight, ideally fewer, people whereby all information can be shared by all people. Any larger, and the project needs to deal with partitioning of information and making it available as necessary. In larger teams, more effort needs to be expended in co-ordination. Careful approaches to separation of concerns can make the adoption of larger sizes possible without excessive pain. Where possible, one should consider more agility and more expertise before bringing in more people.

Rate of Change

The rate of change of requirements gives an idea of the minimum degree of agility necessary. Added rigour means increased amounts of re-work needed when needs change. The term 'agile' reflects the commitment to embracing change, seen as the Achilles heel of heavyweight methodologies. All agile methodologies cope well with change. There are agile techniques both to reduce the amount of re-work when change becomes necessary, and to reduce the cost of that re-work. These techniques require the specific expertise and perform best when the project size is small (ideally no more than 12 people, becoming cumbersome in teams much larger). However, larger projects can obtain reasonable benefits from the techniques, more so when dividing the work between teams. This work separation needs to be done with care.

Culture

It is unlikely that agile techniques will perform well in a culture that thrives on order. However, introduction of a high discipline agile methodology may pave the way to culture change. Agility is often perceived as lacking discipline, a misconception that can be put right.

Copyright 2003, Appropriate Process Group

Priorities

Individual priorities will impact the process in their individual ways and have their own interdependencies with other key criteria. Consult a good practitioner of Appropriate Process to obtain guidance on specific priorities.

System Character

The character of the system will indicate specific areas of expertise that the project will need to have available to the team. Any additional technique necessary to cope with the specific system character will add to the rigour of the process. Behaviour-rich systems will probably need more object-oriented analysis with class and collaboration diagrams or CRC card sessions. Control rich systems will probably require state charts. Similarly, real-time, embedded, fault-tolerant and other special needs all have their own specific techniques. Agile Modeling allows these to be used in an agile manner, adding no more weight to the process than is necessary.

Duration

The chief problems with projects of long duration is that by time of completion the original goals are irrelevant, the project patron has moved on, other key personnel have moved on, technology has moved on or the opportunity has disappeared. Various of the agile methodologies allow for tracking of changing goals and ensure incremental delivery of working software based on prioritised requirements. In the few cases where incremental delivery can deliver no real business value, rapid development and response to changing goals can reduce the risk of developing irrelevant software.

Copyright 2003, Appropriate Process Group