Reconciling Waterfall and Agile for Long Term Best Results

April 2015

Juha Kauhanen
Introduction and summary of key points

Agile project and product management approaches are becoming the norm rather than an exception. Traditional waterfall approaches, originating from pre-information economy, also still have valuable attributes that should not be overlooked. Combining different approaches to suit specific needs rather than fixatedly applying a single approach will yield the best long-term results.

Traditional and agile practices both have their strengths and instead of choosing between them I believe it is better to use the strengths of both and make them work and flourish together, creating an even stronger continually developing project culture. For traditional waterfall and newer agile practices to coexist and evolve together, companies should:

- Favor principles and a framework over detailed and strict one-size-fits all methods.
- Allow freedom to customize the practices within the boundaries of the overall framework and long-term corporate objectives.
- Initiating projects within the broader portfolio management and managing scope and contracts benefits from traditional approaches supporting agile practices.
- Consistent measurement of progress is required to sustain an overall view and retain ability to scale agile development efforts.
- People matter by far the most - over processes and tools. Rethink your project management roles. Cultivate a culture of continuous cross-fertilization of lessons learned and sharing of best practices.

Evolving methodologies

For the 20 past years or so, there has been “rivalry” of approaches in how knowledge worker projects should be managed - between the more traditional waterfall approaches and the more recent agile principles. Waterfall and agile represent the opposite ends of the spectrum in how time, cost and scope are viewed in projects and most other frameworks lean towards either end of the spectrum and are more predominantly influenced by their approaches. There are various adaptations of these frameworks such as PRinCe2 and PMBOK in waterfall and Scrum and XP (Extreme Programming) in agile.

What makes companies struggle most is when they start applying a single framework too dogmatically. One size-for-all methods do not typically fit or scale cross projects that differ in context, scope, complexity, duration, etc.

Many of the ideas in traditional (waterfall) project management ideas date back to the industrial revolution - to the development of factories, machines and assembly lines. Gantt charts, functional decomposition, localized labor force and tools like work breakdown structure (WBS). The latest revolution (where we are now), the information revolution, is focused on information and collaboration, rather than manufacturing and set processes with fixed procedures and pre-determined results of processes. Value is placed most on the ownership of knowledge and the ability to apply that knowledge in varying ways to create or improve products and services. Knowledge workers are the by far largest segment of workforce in modern economy. There are some key aspects in how industrial work and knowledge work differ, as shown in table 1 below.

Table 1. Industrial and knowledge work compared.

<table>
<thead>
<tr>
<th>Industrial work</th>
<th>Knowledge work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work is visible, stable</td>
<td>Work is often invisible, changing</td>
</tr>
<tr>
<td>More structure, less decisions</td>
<td>Less structure, more decisions</td>
</tr>
<tr>
<td>Focus on right answers (questions usually known)</td>
<td>Right questions (answers emerge)</td>
</tr>
<tr>
<td>Command and control</td>
<td>Autonomy</td>
</tr>
<tr>
<td>Strict standards</td>
<td>Principles, goals, continuous innovation and learning</td>
</tr>
<tr>
<td>Minimize cost of workers</td>
<td>Workers as assets</td>
</tr>
</tbody>
</table>

Knowledge worker projects (IT, engineering, science, law, medical, etc.) occur almost without exception in fast-moving and time constrained environments, where work is changing and emphasis is on changing things (vs. running things). If strict industrial era ideas and approaches are applied to knowledge worker projects, frustration and failures increase.
Waterfall versus agile

There can (and most often should) be a degree of waterfall and agile influences in each project. The emphasis which way to lean more is dependent on factors such as those listed in table 2.

Table 2. Waterfall and agile compared – characteristics of suitability for projects

<table>
<thead>
<tr>
<th>Waterfall pull</th>
<th>Agile pull</th>
</tr>
</thead>
<tbody>
<tr>
<td>• A sustainable scope can be clearly defined upfront; e.g. construction project, hardware installation.</td>
<td>• A sustainable scope cannot be clearly defined, or only parts of it; the product will gradually emerge</td>
</tr>
<tr>
<td>• Clear product description available upfront</td>
<td>Requirements change frequently</td>
</tr>
<tr>
<td>• Sustainable requirements can be defined up-front</td>
<td>• Customer learns more about what they want as the project goes on</td>
</tr>
<tr>
<td>• Activities can be well defined upfront</td>
<td>• Activities cannot be well defined upfront Estimating (planning) is difficult</td>
</tr>
<tr>
<td>• Estimating is possible and reliable</td>
<td>• Time and resources are fixed per delivery cycle</td>
</tr>
<tr>
<td>• Similar projects done before</td>
<td>• Use of contingency buffers less typical</td>
</tr>
<tr>
<td>• Few changes are expected during the project and-or not expected to change much</td>
<td>• Project can be complex and it’s duration long</td>
</tr>
<tr>
<td>• Time and resources usually cannot be reliably fixed per project (inability to reliably estimate scope upfront)</td>
<td>• Development process is iterative, multiple delivery cycles</td>
</tr>
<tr>
<td>• Use of contingency buffers typical</td>
<td>• Each cycle builds on the previous ones, adding to the product being developed</td>
</tr>
<tr>
<td>• Project can be a large and complex, in which case it is likely better managed and a program split into phases and with sub-projects</td>
<td>• Incremental usable part deliveries preferred over having to wait until the whole or majority of the project work is done</td>
</tr>
<tr>
<td>• Users cannot typically start using the product until the project is complete</td>
<td>• For projects with defined tasks and phases that can and should be completed in specific sequence (agile practices can be employed within the phases)</td>
</tr>
</tbody>
</table>

Projects, especially large and complex, typically should have both waterfall and agile characteristics and elements to them. A combination of elements from both approaches usually works best for any given project, regardless of context, complexity, size or any other factor.

Today is the dumbest day of the rest of our project

The mindset that should be applied in modern knowledge worker projects is to accept the uncertainties and that the best results will emerge when all stakeholders work together towards the same goals with as much transparency and collaboration as possible. Each day everyone will be wiser and better equipped to reach the next minor sub-goal, adding on to reaching the end goal of the project.

Major software suppliers, such as SAP, Oracle, Microsoft and Salesforce are all applying agile practices and ideas, revamping and modernizing their software projects management methodologies. They do not abandon waterfall practices; rather combine them with the newer ideas.

Another basis of thinking that should be adopted in today’s projects is a product mindset; organizations and people do not use projects they use products, which evolve and adapt continually through ongoing development rather than one-off projects (with another project after to fix the previous ones faults). Instead of jumping to large-scale improvement initiatives, small-scale continuous improvement is usually less risky and offers best longer-term results. To enable this, scalable and open development platforms are required and this is the direction most software vendors have moved over the past 10 years. However, this mindset is not very common among organizations overall yet (as a collectively adopted norm within the organization), and many are stuck with the more industrial era thinking; work assigned as pre-planned projects with ‘definite’ upfront specifications.

Methodologists may roll complex processes asserting these as silver-bullets to fix issues in project management. No methodology can fix anything. All methodologies need to be applied and adjusted by people using common sense and with end goals, surrounding context and environmental constraints and opportunities in mind. Pick

Reconciling Waterfall and Agile for Long Term Best Results
the things that work and ditch the ones that do not. Adjust as you learn more what works, and log and share the lessons learned so that the next projects have less to experiment in tested areas.

Solid frameworks such as Sofigate project model seek to simplify processes, roles and tools. While the model presents an overall waterfall based phased approach, it advocates use off agile practices within the phases. It also asserts that at the beginning of each project there should be a coordinated effort to evaluate whether and how to customize parts of the model to suit the environment and constraints of a particular project.

**Framework over method**

Methods are useful tools, not sacred processes that must not be touched. If calling a method a ‘donut’, or a ‘pumpkin pie’ and wearing silly t-shirts helps to deliver a better product and project that makes customer, sponsors and all shareholders happy, then that is good.

When choosing a new practice to manage projects, care should be taken to check that the environment in which the practice was originally employed was not considerably different to our own. Tailoring will be required more often than not. Culture dominates over processes, and people’s perceptions and expectations need to be understood and acceptance needs to be gained. Culture of the organization eats strategy for lunch, as Peter Drucker said.

A less risky path compared to trying to roll-out a complex and detailed method is to start with a broad framework for providing an overall business objective driven direction, guidelines to identify and handle things that are the most critical (e.g. bottleneck activities and prioritization or effort). Simple, yet effective tools to help streamline activities and leave room to adjust and improve based on experience and increased context specific knowledge are then added to augment the framework.

The methods and practices are fine-tuned by projects, recording and sharing lessons learned, continually making the framework more robust and adding to the standard core practices. Again, even after years of developing a solid framework, if something in the framework does not seem to suit a particular occasion, a return to the main principles and leeway for adjustment should be encouraged and reminded.

These frameworks should pick the most fitting parts from both traditional (waterfall) and agile. Waterfall methods provide sound and proven techniques for setting overall blueprints and align initiatives with corporate objectives via portfolio management. Agile practices should be embraced after an overall groundwork has been laid by the traditional approaches. Design, development and transition activities should be iterative and incremental. They should also feed back into the planning processes and elaborate the results provided by waterfall upfront definitions.

Waterfall approach is good for setting overall goals and defining milestones and checkpoints where overall progress and alignment with portfolio objectives is be measured, results compared with other projects and outcomes evaluated against long term organizational objectives. Agile is good in delivering fit for purpose and fit-for-use solutions and in avoiding unneeded functionality which only adds to maintenance costs and risks.

Therefore, instead of trying to use waterfall to plan everything upfront and then hope it all goes to plan, space should be left for empiricism, experiment, trial and error. This does not need to mean larger buffers for potential failure - the opposite. Giving more space to experiment and innovation with a broad group of diverse stakeholders will ensure faster incremental usable deliveries and more sustainable and successful products in the long run.

**Initiating projects - managing scope and contracts**

Steve Jobs is quoted as saying, “Innovation is not about saying yes to everything. It is about saying no to all but the most crucial features. Exciting as they may be, big-bang development efforts tend to consume lots of time and money and exhibit a high risk of failure. The results are often over-engineered, expensive products that are not user friendly or easy to maintain. A big-bang effort can make it difficult to evolve the product based on later customer and user feedback, as much functionality and architecture is predetermined (leading to significant technical debt).

Reconciling Waterfall and Agile for Long Term Best Results
One main difference between waterfall and agile is that in waterfall scope usually is relatively fixed, leading to time and cost typically flexing (even if intention is to keep them also fixed). In agile, time and cost are fixed via time-boxed development efforts (iterations) with fixed resources, but the overall scope is unfixed, in the sense that it emerges during the project and in every iteration of development.

Instead of rigidly fixing the scope upfront, set a broad estimate and freeze the potentially identified must have features only. This will be the initial project backlog, which will have space to evolve as the project progresses, estimations become more precise and requirements more detailed. You can also set a fixed buffer and budget for the amount the scope can flex, if required by organizations financial e.g. policies. In effect, there is a project budget with a contingency buffer reserved to include emergent scope extensions. This is not much different from traditional risk management approaches of waterfall methods and their use of contingency reserves to account for uncertainties.

Have an empirical mindset and learn as you progress. Use the increased knowledge to improve both the product being developed and the project itself each new day. Resist the temptation to overspecify. More functionality will be discovered and captured naturally as the project progresses. Creating a smaller (feature-wise) product provides a number of advantages: Time to market is reduced; the product/project is developed at a lower cost and generates a faster (thus higher) return on investment. Cash flow and learning is accelerated.

When in doubt about options or alternative features, consider excluding or postponing the requirement. Once a product is released without any of the options, requests for new in-demand features will start to come in and a decision can be made based on tested market feedback. One of the secrets behind the success of iPhone when it launched in 2007 was the narrow initial set of customer needs selected and deliberately leaving out at the time common and expected functionality (like copy/paste). These limitations did not hinder success and paring down functionality allowed development and shipping within a significantly more competitive time frame.

Initiation and planning stages in traditional waterfall model should be kept light, and extended to continue through all the way to the end the project. Initiation activities should cover all aspects of the project, integrating it with the portfolio management and financial management of the organization, creating interfaces and working relationship with all stakeholders and the project management office. Initiation and planning activities should then be repeated all the way to the end, as scaled down versions at the start of development iterations.

Many tools used traditionally in waterfall methods can be very valuable in an agile setting to manage scope. Risks, time, resources and scope within the iterations need to be managed. The self-organizing team delivering and managing the iteration can use these tools to their advantage and it certainly does not hurt if the team members are well averse with concepts and tools of traditional risk, scope, time and resource management such as those advocated by PMBOK and Prince2.

Start and ship with the core set of features, resisting a big bang approach if possible and have a product and service mindset instead of a project mindset (more about product mindset in Managing portfolios section). Employ tested traditional tools to your advantage. Many tools used in agile projects and waterfall projects are actually very similar and agile tools and techniques are often modifications of the traditional tools employing same concepts.

Avoid complex technical tools, though, which may produce impressive looking reports and graphs, but which often only serve to confuse the stakeholders and decision makers as the workings behind the figures and graphs are difficult to comprehend and lost in translation. The coordination of military battles is still often performed by manipulating physical tokens for boats and units despite budgets for computer models, because the physical tokens engage the participants better, leading to less - or ideally no - confusion. Keep things as simple as possible and take on bite-sized pieces.

Projects with suppliers on fixed price contracts require standard project management mechanisms augmenting the purely iterative agile perspectives to allow fit the project to a (for the most part) fixed-price contract structure by

- Defining the core set of deliverables and their nature as clearly as possible to form the fixed price part of the contract, while acknowledging functional variability as noted above and leaving a space for re-estimations. Dividing the scope in logical portions, each with an option for re-estimation (say + 15 %) may be helpful.

Reconciling Waterfall and Agile for Long Term Best Results
Waterfall stage gates then serve as checkpoints where these estimates are reviewed and options used if needed.

- Planning the overall project from an end-to-end and from the portfolio perspective, acknowledging and accepting the challenge of estimating the overall work, duration and dependencies from day one. If the estimates start with ‘poorly’ detailed requirements, focus on the critical core features – the bottlenecks.

- Agile reporting practices often should be harmonized with the reporting requirements of the organizations traditional financial and portfolio management practices and solid reporting and steering mechanism in a more traditional fashion are needed - not merely the agile burn-down reports per iteration. Tracking overall progress against an end-to-end plan is essential for product, portfolio and system (organization) view.

- Considering mutual termination clauses carefully in project initiation to protect customer from continuing to spend inappropriate amounts on a supplier who repeatedly fails to achieve iteration goals and to protect the supplier from inappropriate amounts of scope changes.

**Running projects - measuring and steering**

As described in previous section, iterative perspectives often require standard (waterfall) steering mechanisms in place to manage scope and contracts for a better peace of mind for all stakeholders. Assessing, planning, delivering and confirming value happen in an ‘agifall’ projects in the traditional waterfall stage gate reviews and at the start and end and within each agile iteration. In effect, all these elements of coordination and control are more enhanced than if one method was chosen over the other to be used solely.

In an agile environment, the need to monitor and track progress becomes ever more vital to make sure the project stays on track to deliver what is (being) agreed. This progress also needs to be communicated to all stakeholders in ways that fulfill their needs and is understandable to them. In an ‘agifall’ project, there will actually be more planning and control than in traditional waterfall project, but it all happens more effectively and efficiently, due to the very nature of empiricism and continuous collaboration that is added to the traditional approach with an agile mindset.

In traditional methods, earned value analysis (EVA) and earned value management (EVM) were created to set a solid measurement and reporting framework. PMBOK, among others, bases its progress measurement principles on these. Earned value combines spending and schedule data to produce comprehensive set of measures and progress metrics. It compares actual project performance to planned performance at a particular point in time. The mathematics work in agile setting as well but since the planned performance for the whole project is defined inherently differently in waterfall and agile (waterfall more, agile less defined) care should be taken about what numbers the measurements are based against. The quality of the baseline is critical in using earned value. In agifall projects there needs to be acknowledgement that initial plans are likely to change, and that the basis for EVM needs to be kept in line with the changes and evolving plans. In addition, earned value only covers cost and time aspects, not addressing the quality aspects and additional measures should be agreed for these. Quality, of course, is based on meeting the agreed functional and performance requirements, which are set for each project and product differently.

Earned value has its benefits in it being forward looking and providing leading indicators. It is also visual and simple. Traditional earned value and agile earned value can easily be reconciled. The basic leading indicators are formed as follows: SPI = completed features (story points)/planned features and CPI = Earned value (cost of planned features)/actual costs (cost of completed features). These indicators when plotted on a graphs provide the basis for reporting framework that should in most cases be good enough starting point to suit most stakeholder’s needs.

Agile and traditional project progress reporting practices can be used concurrently without adding overhead to the process. Agile burn-down charts and cumulative flow down diagrams can be converted to traditional earned value analyses for financial management and consolidated for portfolio management needs. What is required is consistency in agreed practices across projects.

Consistent measurement and reporting practices is a requirement for managing project portfolios – an area that sometimes lacks discipline and consistency in agile environments when scaled outside and beyond a single project.
Managing portfolios

Product and service mindset shifts the thinking from looking at projects as one off, big bang initiatives that have a definite beginning and end (really contradicting with the very classic definition of a project, but so be it) to continuous, rolling development, where projects merge into each other, forming a continual work as usual practice, where products and services are developed continually. Projects and peoples roles as members of projects get weaved into their usual business-as-usual roles. Project work becomes second nature to people. Agility becomes second nature. No more do we need ceremonial trainings and kick-offs (except occasionally, for more special cases perhaps and for team building purposes) to get the project team and stakeholders somewhere near the same playing field.

Organizations are then executing improvements to the ways the organization executes its operations and achieves its mission, focusing on removing the constraints between reaching the organizational goals. They are not executing projects for projects sake: Projects have not intrinsic value, only the (properly set) goals the projects are aiming for do.

Extend the idea of having and product/service mindset to having a system mindset. System mindset when applied to managing portfolios of projects call for attention on the following aspects and principles:

- A portfolio should be seen as more than the sum of its constituent projects to leverage dependencies and synergies. Flow of information between projects is essential, even between initially unrelated ones.
- Portfolio structure is the source of portfolio behavior - how you set up and manage the portfolio reveals itself in the behavior of the portfolio and the projects.
  - Portfolios evolve from the bottom up. Purpose of the upper layers of any system is to serve the purposes of the bottom layers, not vice versa.
  - Portfolio needs to be managed not only for productivity or stability, but also for resilience (changes)
  - Ensure feedback loops for balance and to reinforce positive momentum.
- Look for and focus on fixing the bottlenecks and constraints that affect the parts and the portfolio as a whole the most.
- There will always be limits to growth – internalize the externalities and balance the focus on long term and short-term gains.

I will focus on only one of the points above here. Others are covered in my other articles more in depth. The one in focus here are the few critical bottlenecks in the system – the constraints that manifest themselves as the restrictions on the overall portfolio. Being able to identify and predict where the bottlenecks are or are likely to emerge and then fixing them will have much more significant beneficial effects on the system performance, than on focusing on any non-bottleneck activity. The portfolio can be made up of any level and scope of projects, consisting of divisional or enterprise wide initiatives.

The principle is to prioritize projects based on their impact on current and future performance. Prioritizing means focusing efforts and resources on the most valuable projects, and acting when projects are deemed not necessary by either halting or cancelling them. How to prioritize is up to the organization, but the above system principles should be kept in mind when doing it.

Roles

Agile practices call for different roles to those employed in waterfall world. Main difference with roles between traditional and waterfall is, again, in the mindset people take and the culture of the organization (that is created by the collective majority mindset).

The culture in agile organization needs to be collaborative, open, and self-organizing. The roles take on a different shape. In agile the emphasis is in more autonomy compared to command and control emphasis in waterfall. Sometimes command and control can work, but in knowledge worker projects, collaboration and autonomy are primary keys to success. Top layers of the organization are servants to the lower layers, not vice
versa. The key roles that should be integrated in an organization that mixes traditional waterfall and agile practices in are simplified in table 3.

Table 3. Integrating waterfall and agile roles.

<table>
<thead>
<tr>
<th>Waterfall role</th>
<th>Meet half way</th>
<th>Agile role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sponsor</td>
<td>Same</td>
<td>Sponsor</td>
</tr>
<tr>
<td>Product manager</td>
<td>Product manager would typically be accountable for the whole product or product line, and oversee potentially many product owners, or act as a product owner herself.</td>
<td>Product owner</td>
</tr>
<tr>
<td>Project manager</td>
<td>Project management would be elevated to oversee programs and dependencies of projects, leaving teams to manage more autonomously.</td>
<td>Development team</td>
</tr>
<tr>
<td>Project methodology champion (if one exists)</td>
<td>Same idea here, if either practices are applied, care should be taken to apply with true understanding of why they are applied to not dilute the intent and effect.</td>
<td>Scrum master (Scrum being by far the most widely adopted agile framework, hence this role used here)</td>
</tr>
<tr>
<td>End users</td>
<td>Active involvement from the beginning to the end. Traditional stage gates with wider presentation, key roles from core stakeholder groups involved continually.</td>
<td>End users</td>
</tr>
<tr>
<td>Other stakeholders</td>
<td>Address and respect the product owner.</td>
<td>Other stakeholders</td>
</tr>
</tbody>
</table>

Estimating knowledge worker projects is difficult because of all the inherent risks involved in developing solutions for new business problems that tend to have high rates of change. Despite this difficulty, organizations need a way to bid on projects and estimate their likely budgets. One popular estimation model called COCOMO was created by reverse engineering the inputs from thousands of projects that had a known exact cost. The idea was to assess the correlation between project input variables (assets and resources required to complete the project) and the final cost of the project, to then be able to use this for estimating future projects. The method has been a success and it is at the heart of many commercial estimation systems.

The weighed factors tell a clear message: People factors are by far the most significant cost factor with 33% overall weight, compared to 10% for product and 3% for tools & processes factors. Impact of people factors such as skills and management & leadership practices - imposed on the project via roles – are 10 times more significant than the tools and processes used in the project. A fool with a tool is still a fool, but skilled professionals with a clear purpose, agreed well-functioning roles and following fitting processes can make wonders happen.

About the author

Juha Kauhanen works as a Senior Advisor at Sofigate, since 2013. Before that, he worked for Deloitte, based in Finland and Australia. Over the past 15 years, Juha has served a wide range of clients in various industries. His focus areas are IT governance, change management, enterprise architectures and project, program and portfolio management. Juha holds an MSc in cultural geography and PMI-PMP, PMI-ACP, ITIL Expert, PSM, PSPO, TOGAF 9 Certified, Lean Six Sigma and ISO 20000 Consultant Manager certificates.

Juha.kauhanen@sofigate.com
Juhaelias@me.com

Reconciling Waterfall and Agile for Long Term Best Results