

## **From agile software development to management agility:**

### **The way forward <sup>1, 2</sup>**

**Darren Dalcher, PhD**

The adoption of agile concepts, principles, methods and approaches in many sectors is rapidly transforming software development and management practice. Crucially, agile methods focus on managing — and speeding up — development activities and can therefore offer new ideas and significant improvements for management-in-the-small.

The aim of this article is to reflect on the impact that agile methods are likely to have on project management practice. Agile methods challenge some of the basic assumptions embedded in sound project management, and while they overcome some limitations, they introduce new constraints and ways of thinking. Their adoption requires intelligent adjustments to the way projects are perceived, prioritized, and managed.

#### **1. Agility is everywhere**

Agile software development emerged in response to widespread dissatisfaction with sequential approaches to developing software. Much like prototyping before it, agile development embraced the ideas of greater interest in users, capturing true requirements, and rapid delivery of working versions of the agreed content. While agile methods adopted the creative ideas of prototyping, they also placed a greater emphasis on controlling development through the use of timeboxing, thus offering a more structured framework for delivery.

Agile normally means nimble, quick-moving, or active. Agile methods aim to use such qualities in order to prosper in an environment of constant and unpredictable change. They integrate established software development practices and principles into organized approaches for responsive development that emphasize collaboration and communication. Short delivery cycles enable developers to respond more easily to changing demands from clients and a shifting marketplace and environment. Agile

---

<sup>1</sup> *Editor's note: Second Editions are previously published papers that have continued relevance in today's project management world, or which were originally published in conference proceedings or in a language other than English. Original publication acknowledged; authors retain copyright. This paper revisits an article (Dalcher, 2008) published as an invited contribution in the Cutter IT Journal in 2008. The current version includes a number of minor corrections and an improved diagram. For more recent articles and additional perspectives on agile, see (Dalcher, 2021; 2022). This paper is republished here with the author's permission.*

<sup>2</sup> How to cite this paper: Dalcher, D. (2023. 2008). From agile software development to management agility: The way forward; originally published in the Cutter IT Journal in 2008; republished in the *PM World Journal*, Vol. XII, Issue IV, April 2023.

development is typified by adaptive software built by small teams that focus on rapid feedback and continuous improvement.

Agile methods boast an ever-growing influence. Such methods, and the principles they embody, have played a key part in shaping research and practice in software development and engineering, bringing many new topics to the forefront. Moreover, the improved IT project success rate — with more IT projects delivered on time, on budget, and within requirements — is being attributed in part to the increased adoption of agile methods.

Meanwhile, the “agile movement” has spawned new terms, such as sprints, scrums, velocity, epics, sagas, user stories, retrospectives, huddles, and daily stand-up meetings, which have become part of the accepted software development and even project management jargon. The Agile Conference is probably the best-attended conference in software development and continues to grow in popularity. Many new conferences and communities have formed to discuss and agile ideas, and tracks in established conferences also highlight agile concepts. And this explosion of interest in all things agile has extended well beyond IT. Bookshelves in bookshops include many new titles focusing on agile investors, agile competitors, agile organizations, agile systems and enterprises, agile genes, and even the agile library.

Not surprisingly, the improved success rates in agile IT project delivery have also sparked an interest amongst a project management community looking to improve its own practices. Many of the ideas and emphases fostered in agile methods, such as value, benefit, relationships, feedback, and response to change, chime in with interests the project management community has explored. Given the shared focus on projects and project delivery, it is easy to see why importing new agile ideas into project management seems so attractive. However, it is worth noting that there is a fundamental difference between agile as a development method for organising small technical teams and seeking to employ agile as a project management method. This is not clear in the discourse surrounding agile methods and their application.

## **2. The experiment**

Anecdotal evidence suggests that the use of agile methods will translate into increased productivity, quality, and stakeholder satisfaction. Products developed using agile methods are often reported to be more creative and to result in higher user satisfaction levels. To date, however, there have not been many direct comparisons made between the different approaches in terms of the quality, functionality, and scope of the resulting products. Nor is there much information regarding the impact of various methods on projects, their intermediary products, and the development teams. Among the few examples is a case study (Layman, 2004) focusing on the effects of adopting Extreme Programming (XP) at IBM and Sabre Airlines. That study indicated productivity improvements of 46%-70% in lines of code (LOC) per person-month (PM).

Intrigued by the overall lack of evidence, Professors Oddur Benediktsson and Helgi Thorbergsson and I decided to set up a comparative experiment involving 15 one-year IT projects, which were completed in parallel (albeit in a semi academic environment). The experiment was conducted at the University of Iceland in 2004 (Benediktsson et al., 2006). The comparable products to be developed were interactive packages centred on a database system for home or family usage with a Web-based user interface. Four of the project teams used the V-model, a variation on the traditional waterfall approach; four used incremental development; four applied an evolutionary approach; and three utilised Extreme Programming (XP), an agile method.

Our goal was to examine and compare resource utilisation and efficiency in projects developed using sequential, incremental, evolutionary, and XP approaches. The experiment appears to be the largest comparative study conducted to date. While there were no measurable differences in quality, there were astonishing differences in terms of productivity and a matching increase in the satisfaction levels of clients. The greatest difference in productivity was between V-model teams and XP teams, with the XP teams being, on average, 337% more productive than the V-model teams.

### 3. Results

The XP teams produced on average 3.5 times more LOC than those using the V-type model, 2.7 times more LOC than those using the incremental model, and 2.2 times more LOC than the evolutionary teams. This significant trend is also reflected in terms of the number of LOC produced in Java and in XML. Table 1 offers a size comparison between the results of the different groups. We can easily see that the XP teams developed the greatest number of lines. The increased code production by the XP teams also translated into a greater productivity rate over the course of the project (see Figure 1).

**Table 1: Code Production in Lines of Code (LOCs)**

	<b>Java LOC</b>	<b>Total LOC</b>
<b>V-model</b>	1032	2233
<b>Incremental</b>	1803	2776
<b>Evolutionary</b>	1477	3417
<b>XP</b>	4836	7740

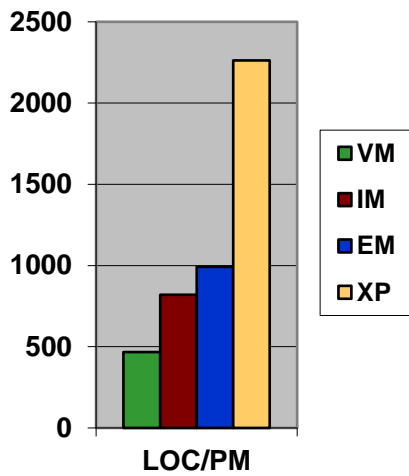


Figure 1: Monthly productivity

Figure 2 offers a size and effort comparison between the different groups. It is easy to see that the XP teams developed the greatest number of lines. The figure also offers a productivity clustering by method. Back in 1984, Barry Boehm, Terence Gray, and Thomas Seewaldt showed that with traditional development and with prototyping, development effort is generally proportional to the size of the developed product. They further asserted that this relationship will apply to most development methods. While this principle still appears to roughly apply to most groups in our experiment irrespective of the method they applied, two of the XP teams in our study achieved significantly greater productivity as measured by size in LOC. Another way of viewing this is to say that if we drew a line from the bottom left of Figure 2 to the bottom third on the right, all the methods with the exception of XP will cluster around it (as Boehm et al. initially predicted). In comparison with more traditional development approaches, then, XP appears to defy the relationship posited by Boehm and his colleagues.

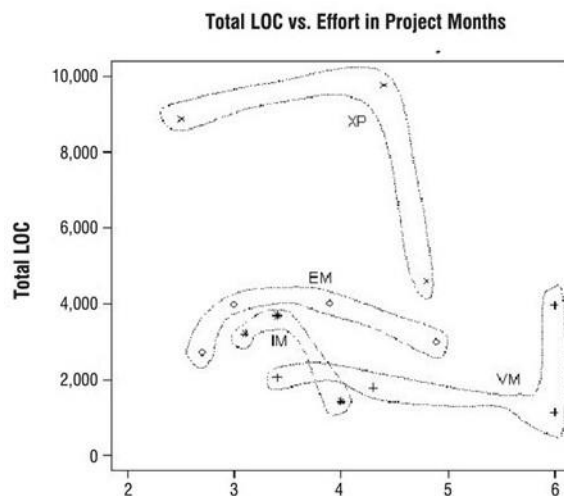


Figure 2: Size as a function of effort

So, the case for the adoption of agile methods is powerful. In addition to increased productivity, the use of agile methods in our experiment resulted in increased acceptance of the delivered product as well as other business benefits with the potential to improve the bottom line. Given the same amount of time for the overall undertaking, XP teams were able to build on the feedback from small and frequent releases and produce a medium-sized product (compared to the smaller products delivered by the other teams). Significantly, of all the systems delivered in the experiment, users viewed the XP-developed products as the most comprehensive solutions and the most satisfactory products. It seems that involving the users and delivering on a more regular basis yielded products that were not only larger, but also more significant in terms of functionality.

#### **4. The impact of agile management (or can we get rid of traditional project management?)**

Agile development uses well-established principles and effective methods to deliver value more rapidly through increments. It emphasises the role of learning, feedback, and people in evolutionary cycles. The approach encourages team cohesion and emergence and fosters commitment and knowledge sharing. Agile projects take dedicated project teams and focus on short-term delivery in which nonessential activities are stripped away from the process, thus resulting in improved productivity.

Agile development offers a powerful tool for micro-planning, controlling increments, and demonstrating progress. In essence, agile methods organise technical development by dividing it into microprojects. This enables teams to measure their progress closely, thus providing unparalleled visibility. Frequent delivery cycles also offer improved control over the entire effort. In situations where the deadline and the available resource level are fixed, the use of multiple increments gives project managers significant leverage (as demonstrated by, Benediktsson & Dalcher, 2004; Dalcher & Benediktsson 2006). In such cases, greater productivity gains and enhanced flexibility will result from organising the development effort into short, frequent increments, as we saw in our experiment. All 15 teams had the same overall time boundary and resource level, yet the XP teams delivered larger products, thereby displaying a significant increase in productivity over the entire 12-month period.

Complex project environments with high rates of change are ideal for the employment of rapid increments embedded within agile methods (ibid.). But the true focus in agile methods is on the technical tasks and their organisation and not on the project per se. In traditional project management, projects have a clear beginning and a definite end. Agile methods, on the other hand, are designed to help teams demonstrate business value more quickly and more often. Therefore, agile development has a more continuous sense and may extend beyond more conventional projects. Agile increments continue to bunch together segments of functionality, and project work thus becomes an ongoing operation, a service that can continue to deliver. The continuous nature of change in a turbulent environment opens up opportunities, whilst also introducing new challenges.

## 5. A typical agile failure story?

One of my research interests involves the collection of failure stories. Over the last 20 years, I have collected stories of hundreds of IT project failures. While the bulk of these relate to traditional approaches, I am now seeing a growing number of agile failure stories. Many of them have certain common features that were not typical of more “traditional” failures. In fact, a new kind of project failure may have been made possible by the ways agile methods are defined and applied — a project in which each increment is a success but the overall project is a flop. The ultimate failure often results from a missing key system or ignored integration, proving that value accumulation is not always an adequate replacement for overall project coordination.

In the case I will discuss, the Candy Corporation (CC)<sup>3</sup> wanted to develop a state-of-the-art Web interface to showcase its candy. In a highly competitive environment, with the potential for international competition, senior management decided that the Web display was essential to capturing — possibly even cornering — the market. As the project involved a new concept in this sector, management agreed to adopt an agile approach and involve a wide range of stakeholders in order to ensure the integrated system satisfied all concerned parties. Agile development was selected so that changes could be incorporated more easily and business value demonstrated more efficiently. The new system was to be implemented by Halloween a year hence.

As the teams focused on each current development cycle, more users were brought into the process and many novel features were incorporated in the Web display. Thus the new system kept growing and improving, with each new increment adding value and offering a more impressive system. Six weeks before Halloween, a visiting senior manager was being shown the almost complete Web ordering interface. When he asked about the volume of orders, it quickly emerged that no one had built the capability to fulfil orders into the system. While each project increment had delivered business value, the final system was missing the key component needed for business operation.

Without the ability to harvest orders, CC tried to survive Halloween but ultimately folded within six months. In this case, agile methods were adopted without a full understanding of the implications. This lack of understanding is made worse when the concept of a project is abandoned in favour of an ongoing operation that fails to define the key parameters. Some of the agile failures I have looked at occurred because project teams were concentrating on a single increment at a time while ignoring the overall picture. When projects are not viewed as complete undertakings, certain aspects can slide under the radar. The delivery of value in each increment means that content is optimized per delivery cycle. Unless sound methods are adopted, essential features and functions can be forgotten — and sometimes it is too late to recover.

---

<sup>3</sup> To hide the identity of the organizations involved, the case presented is a composite of two such failures.



Agile practitioners have noted that, occasionally, customer involvement in work planning can lead to scope creep. Even prioritization by the business product owner is no guarantee that the entire project perspective has been accounted for. CC had a string of successful projects, each of which delivered benefits. But when they were all combined, they added up to ultimate business failure.

## 6. The shifting triangle

The missing aspect for CC was the visibility that comes with the complete project view. An altogether different level of scrutiny is required for managing agile projects as agile projects need to be viewed in an entirely new way. The application of agile development as if it were a management approach means that we need to consider the impact of this new approach on constraints and on how projects are prioritised and managed.

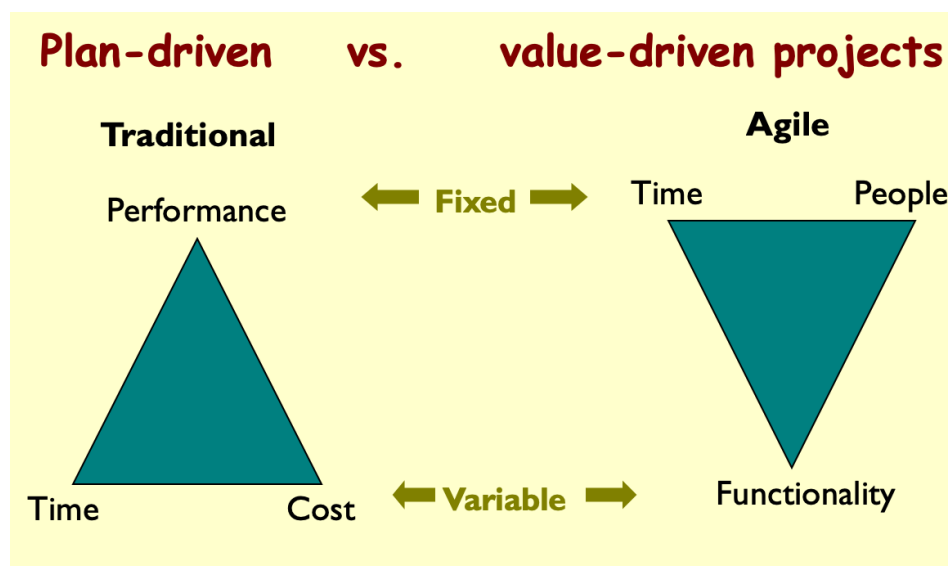
Traditional project management is predicated on the idea of a project — a distinct entity that must be completed to make the ambition come true. The vision of the project drives the tradeoffs between the constraints of time, cost and performance in order to ensure that a project delivers as intended. In many areas, a project manager may encounter the need to manage two lifecycle systems, one revolving around the technical tasks and a second concerned with the project activities. Agile development blurs the boundaries between the two, as it seems to span both domains. Agile methods feature timeboxing as a way of managing work elements. The focus on work content has led some project managers to view agile development, and the emerging ideas of agile project management, as the management of work packages or tasks.

The truth is that agile methods adopt a different perspective. Rather than focus on the management perspective and the project concept, agile methods are concerned with the work that can be completed over a given period of time with the already agreed resources. If we depict traditional management as a triangle in which the project idea drives decisions about resource levels and the timing available to deliver the project vision in a plan-driven fashion, agile development would turn the triangle on its head creating a value-driven perspective (see Figure 3). In agile projects, the time available is fixed by the size of the increment, or the timebox, and so is the size of the team. What needs to be determined is the content that will be delivered as part of that increment. This shift has the potential to transform the notion of the project as we know it.

Unlike the traditional view of project management, agile development is not necessarily concerned with having a defined project and delivering parts of that vision. Agile teams often operate within what I have termed a “fixed resource envelope” with fixed time and predetermined resource levels. This has a huge impact on how content is selected and agreed upon, as the traditional triangle of project management has shifted in emphasis. Agile projects use a variety of methods to allocate stories, backlogs, or requirements into these envelopes, thereby populating them with content, or work. Developers determine and rearrange the content of their envelope to ensure they deliver useful and productive

functionality which equates with delivering value for the users. Moreover, the content is optimised internally by the team rather than on a project-wide basis. Given the brief timescale of increments, each agile project has a short-term concern for optimising what may be delivered in that time. Without further scrutiny from some level of management, critical requirements can be missed or dropped as the team progresses. Interactions and dependencies can also get missed through local prioritisation. Organizations such as CC, which are used to focusing on a more traditional project vision, can be led astray by such “short- sightedness” or “functional blindness” and be lured into pursuing a series of small incremental steps that lead to failure.

To reiterate, an essential difference between the two approaches is that traditional project management has developed around the delivery of projects, while agile development is concerned with delivering value. Project management has recently learned from program management that every project should deliver benefits to the organisation and can also focus on the delivery of outcomes. While the delivery of benefits is a key focus, project management is not used to dealing with the constant delivery of value that may or may not add up to an old-style project, especially when additional chunks of operational value are added to the stack of requirements.



**Figure 3: Traditional vs. agile projects**

## 7. Making agile projects succeed

Managing agile projects requires some adjustments. Traditional project managers who try to manage streams of fixed resource envelopes face fundamental challenges. For example, traditional project management has developed two general approaches to scheduling. One assumes unlimited resources within a fixed project timeframe, and the other assumes limited resources within a flexible project timeframe. Agile project management requires a new mindset, as it assumes fixed resources and a fixed timeframe,



rendering both traditional scheduling approaches somewhat irrelevant. It also has the added bonus of avoiding the need for smoothing resource levels in the project, as resource size is a fixed commodity.

Agile development offers a new approach for dealing with change in turbulent environments. Many project managers are keen to understand and harness the potential of this approach. As agile projects replace traditionally managed project, many project managers find themselves in unfamiliar territory outside the project team. In agile development, content is selected by autonomous teams. Teams create and monitor iteration plans, dividing the tasks and fitting them into time-boxes and iterations; customers prioritise features based on their perceived business value.

A number the agile approaches contend that there is no need for project management. However, some of the failures reveal that there is still a need for a higher form of project management activities, not least for prioritising across agile increments; looking at how work content is selected, allocated, and revised; and considering the integration of different elements and agile projects who may be subject to shape shifting as they progress. The guiding vision embedded in traditional project management needs to be preserved and utilised to support decisions and perform tradeoffs. Project managers are trained to develop, control, and monitor plans to ensure the achievement of project objectives and the integrity of the project concept. They need to retain the overall view and perspective rather than get bogged down in detail.

Coexistence requires a middle ground, especially when both sides have so much to offer. Traditional project managers retain a helicopter view, or architectural perspective of the complete project. Agile managers offer insights into reducing uncertainty and dealing with inherent complexity through collaboration and rapid iterations leading to new insights and opportunities. We can seize the middle ground by utilising agile management approaches, focusing on speeding up microproject increments, whilst also employing programme management techniques to look after the overall project. Programme management will enable the coordination of streams of projects to ensure the successful delivery of outcomes. The focus on outcomes and the delivery of relevant benefits to users can mitigate against some of the new failures.

While agile methods occasionally cross the boundary between technical and project management, programme management is ideally suited to sit alongside them, as it can comfortably deal with collections or clusters of smaller projects managed in a coordinated way. Programmes operate in inherently ambiguous, uncertain, and less well-understood environments typified by change and turbulence. They also link with strategy achievement and entail an understanding of internal culture and politics, making them an ideal partner for agile projects.

In such a powerful arrangement, project management in-the-small can be done within autonomous agile teams. The vision and architecture can be maintained by technical and management experts ensuring the integrity of the overall concept of the project and over-seeing its decomposition, architectural design, and ultimate integration. Moving up a level

will mean that project managers will become the new programme managers, with a more strategic role and a new orientation that involves leading people rather than managing, and guiding rather than directing. Ultimately, they will also ensure that strategy materialises and is constantly served by rapidly executed change initiatives that deliver organizational benefits.

In an increasingly global environment, as we move toward smaller projects in a flatter society, we need to employ higher-level approaches to guarantee the value we deliver represents coherent and complete content. Our ability to continue to prosper in an environment of constant change will depend on the mastery of these techniques.

### References:

Benediktsson, O., & Dalcher, D. (2004). New insights into effort estimation for incremental software development projects. *Project Management Journal*, 35(2), 5-12.

Benediktsson, O., Dalcher, D., & Thorbergsson, H. (2006). Comparison of software development life cycles: a multiproject experiment. *IEE Proceedings-Software*, 153(3), 87-101.

Boehm, B. W., Gray, T. E., & Seewaldt, T. (1984). Prototyping versus specifying: a multiproject experiment. *IEEE transactions on Software Engineering*, (3), 290-303.

Dalcher, D., & Benediktsson, O. (2006). Managing software development project size: Overcoming the effort-boxing constraint. *Project Management Journal*, 37(2), 51-58.

Dalcher, D. (2008). Beyond agile project management: the way forward. *Cutter IT Journal*, 21(5), 28-34.

Dalcher, D. (2021). Scaling up to business agility. *PM World Journal*, 10(5), 8pp.  
<https://pmworldlibrary.net/wp-content/uploads/2021/05/pmwj105-May2021-Dalcher-scaling-up-to-business-agile.pdf>

Dalcher, D. (2022). Is it time to rethink agile? *PM World Journal*, 11(12), 11pp.  
<https://pmworldlibrary.net/wp-content/uploads/2022/12/pmwj124-Dec2022-Dalcher-is-it-time-to-rethink-agile.pdf>

Layman, L. (2004). Empirical investigation of the impact of extreme programming practices on software projects. in *Proceedings of OOPSLA '04*. 2004. Vancouver, British Columbia: ACM Press.

## About the Author



### **Darren Dalcher, PhD**

Author, Professor, Series Editor  
Director, National Centre for Project Management  
Lancaster University Management School, UK



**Darren Dalcher**, Ph.D., HonFAPM, FRSA, FBCS, CITP, FCMI, SMIEEE, SFHEA, MINCOSE is Professor in Strategic Project Management at Lancaster University, and founder and Director of the National Centre for Project Management (NCPM) in the UK. He has been named by the Association for Project Management (APM) as one of the top 10 “movers and shapers” in project management and was voted Project Magazine’s “Academic of the Year” for his contribution in “integrating and weaving academic work with practice”. Following industrial and consultancy experience in managing IT projects, Professor Dalcher gained his PhD in Software Engineering from King's College, University of London.

Professor Dalcher has written over 300 papers and book chapters on project management and software engineering. He is Editor-in-Chief of *Journal of Software: Evolution and Process*, a leading international software engineering journal. He is the editor of the book series, *Advances in Project Management*, published by Routledge and of the companion series *Fundamentals of Project Management*. Heavily involved in a variety of research projects and subjects, Professor Dalcher has built a reputation as leader and innovator in the areas of practice-based education and reflection in project management. He works with many major industrial and commercial organisations and government bodies.

Darren is an Honorary Fellow of the APM, a Chartered Fellow of the British Computer Society, a Fellow of the Chartered Management Institute, and the Royal Society of Arts, a Senior Member of the Institute of Electrical and Electronic Engineers, a Senior Fellow of the Higher Education Academy and a Member of the Project Management Institute (PMI), the British Academy of Management and the International Council on Systems Engineering. He is a Chartered IT Practitioner. He sits on numerous senior research and professional boards, including The PMI Academic Insight Team, the CMI Academic Council and the APM Group Ethics and Standards Governance Board as well as the British Library Management Book of the Year Panel. He is the Academic Advisor, author and co-Editor of the highly influential 7<sup>th</sup> edition of the APM Body of Knowledge. Prof Dalcher is an academic advisor for the *PM World Journal*. He can be contacted at [d.dalcher@lancaster.ac.uk](mailto:d.dalcher@lancaster.ac.uk).

To view other works by Prof Darren Dalcher, visit his author showcase in the *PM World Library* at <http://pmworldlibrary.net/authors/darren-dalcher/>.