

PRINCE2 Agile® and its Practical Implementation on Company "X" ¹

Irina Poghosyan

Introduction

This paper dives into the dynamic synergy between PRINCE2 Agile and agile project management principles, offering a comprehensive exploration of their integration within the unique operational context of Company "X." PRINCE2 Agile, recognised for its adaptability to diverse project scenarios, regardless of their size, complexity, or industry, is at the heart of this investigation. The primary objective is to separate how Company "X" effectively implements PRINCE2 Agile principles and to critically assess the suitability of this framework within their specific organisational environment.

In essence, this paper seeks to understand the extent to which PRINCE2 Agile can be tailored to meet the distinct needs of an organisation operating within the constraints of small project teams, minimal managerial hierarchy, and a fast-paced, ever-evolving business landscape. By scrutinising the practical application of PRINCE2 Agile principles within Company "X," this paper aims to provide insights into the framework's adaptability and effectiveness in addressing the challenges and opportunities unique to this organisational context.

Company "X"

Company "X" is a London-based data analytics and business intelligence company that specialises in providing data-driven solutions to businesses across various industries. With a team of skilled data scientists, analysts, and technology experts, Company "X" leverages cutting-edge technology and advanced analytics to help organisations make informed decisions, optimise operations, and drive growth.

The company offers a wide range of services, including data collection, data cleaning, data modelling, and data visualisation. They assist clients in transforming raw data into actionable insights, enabling them to identify trends, uncover opportunities, and mitigate risks effectively.

As a trusted partner for businesses seeking to harness the power of data, Company "X" is dedicated to delivering high-quality, customised data analytics services that contribute to enhanced decision-making and overall success.

¹ How to cite this article: Poghosyan, I. (2022). PRINCE2 Agile® and its Practical Implementation on Company "X", *PM World Journal*, Vol. XII, Issue XI, November.

Methodology

PRINCE2 Agile is a flexible project management framework that can be adapted to any project, regardless of size, complexity, or industry. It provides a set of principles and processes that can be tailored to meet the specific needs of the project team and the organisation.

To configure PRINCE2 Agile for agile projects, it is important to focus on the principles of flexibility, collaboration, and continuous improvement. The framework should be used to provide a basic structure for the project, but it should not be so rigid that it stifles innovation and creativity.

PRINCE2 Agile is not a traditional project management approach. It is designed to support a variety of working styles, including agile environments that emphasise informality, collaboration, and trust.

PRINCE2 does not require projects to be run in a waterfall manner. In fact, much of its guidance suggests otherwise.

Agile project management principles and practices have their roots in IT and software development, but PRINCE2 Agile is not exclusive to IT projects. It can be used to manage any project, in any industry.

Scrum is the most popular agile framework, but PRINCE2 Agile recognises that while Scrum is unquestionably agile, the reverse is not true: agile is not simply "using the Scrum framework." There are other frameworks available, and a framework is just one component of the agile way of working.

Agile is a broad term that encompasses a variety of frameworks and practices. The Scrum and Kanban frameworks are the most widely used agile frameworks, but there are many others.

Scrum and Kanban are not project management frameworks. They do not define a project manager role, and they cannot be used on their own to manage a project. However, they can be used within a project management framework, such as PRINCE2, to deliver products.

PRINCE2 is a comprehensive project management framework that is well-suited for high-level projects, while Agile is a more flexible approach that focuses on delivering projects in small, incremental stages.

PRINCE2 Agile offers a number of significant benefits, including:

Adaptability - The framework is generic enough that it can be applied to virtually any industry, sector, or location. That is not to say PRINCE2 Agile provides a simple to-do list that anyone can use without thinking. This framework does not prescribe a specific approach to project management. Instead, it identifies key factors and areas of focus that can be applied to any project, while still giving practitioners the flexibility to adapt it as needed.

Easy to incorporate - PRINCE2 is already the most widely used project management framework in the world, and Agile isn't exactly a new contender either! If your business already utilises one of them, upskilling your staff in PRINCE2 Agile can help you enjoy a more holistic project management style without having to start from scratch.

Faster benefit realisation - Agile is about completing projects in increments, generating benefits that can be enjoyed at several stages, rather than just at the conclusion. This can be a great benefit for projects designed to empower rapid change management, as well as those in competitive industries where companies cannot wait for benefits to come all at once.

Greater control - Agile project managers are experts at delivering results, but their focus may be too narrow. To ensure project success at a strategic level, there needs to be an upper tier of management. This is where PRINCE2 excels.

Collaboration and communication - One of the benefits of using popular corporate standards and frameworks is that they make it easier for team members to collaborate and communicate, because everyone is using the same terminology and methodology. This can help to streamline projects and reduce the time it takes to complete them, without sacrificing quality. PRINCE2 Agile does this by clearly defining roles, responsibilities, and lines of communication at the start of each project. For this reason, many businesses choose to train multiple employees in PRINCE2 Agile at the same time.

Prince2 Agile is linked to five 'Targets':

- Be on time and hit deadlines
- Protect the level of quality
- Embrace change
- Keep teams stable
- Accept that the customer doesn't need everything

The PRINCE2 organisation structure for the project has three layers, namely “Directing” – the project board, “Managing” – the project manager and finally “Delivering” – the team manager and team members producing specialist products.

Processes in Agile Setting:

1. Starting up and initiating
2. Directing a project
3. Controlling a stage
4. Managing product delivery
5. Managing a stage boundary
6. Closing a project

Equally the PRINCE2 answers the questions such as:

Initiating a Project:

Commencing a project and its initiation involve the consolidation of initial tasks within the agile methodology. Furthermore, we explore the utilisation of the Cynefin framework to comprehend the complexity level.

Directing a Project:

Project direction plays a pivotal role in cultivating an environment conducive to the successful implementation of agile practices. The project board assumes a central role in this process.

Controlling a Stage:

During the stage control phase, the project manager's focus shifts toward feature delivery, daily stand-up meetings, and empirical forecasting. Additionally, retrospectives serve as a means of garnering lessons and facilitating continuous improvement.

Managing Product Delivery:

The management of product delivery constitutes a substantial portion of the agile methodology, exemplified by practices such as Scrum or Kanban.

Managing a Stage Boundary:

Distinguishing between stage management and release activities is crucial. In this context, the concluding stage assessment, which is less ceremonious than in the project direction phase, is highlighted.

Closing a Project:

Similar to stage boundary management, project closure takes on reduced significance in the presence of incremental delivery. However, it remains an essential phase, emphasising the importance of formal acceptance processes, potentially involving structured workshops.

The PM and the team

How to relate PM to the team?

An optimal option would be to leave the roles as they are but identify a single point of contact for the project manager. In a sense, someone would be acting as a quasi-team manager, possibly without knowing it. This may get around the thorny issue of renaming an existing Scrum Master, who should coach and facilitate, as a team manager, with all the potentially negative connotations of the word 'manager'.

Alternatively, considered ‘less agile’, we have the standard set up in PRINCE2 where a Team Manager role would be created in the delivery team. This might be appropriate where the team hasn’t worked together before and can appreciate that as part of a project, there will be additional organisational requirements.

Which one of these is considered most appropriate will depend on the circumstances around the project.

In terms of the delivery team structure itself, in PRINCE2, very little is said about this. All we have is a team manager, and team members, producing specialist products.

In other agile project frameworks such as AgilePM, the Development Team is expanded upon, critically with customer side involvement and not just supply side specialists.

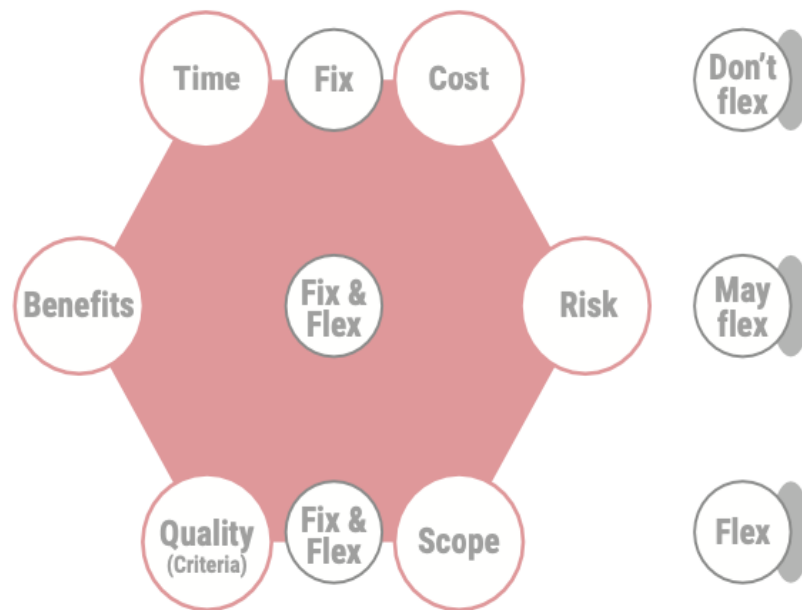
PRINCE2 Agile advises it is best to have:

- One person to lead the team – not quite a direct match for the Scrum
- Someone from the customer side – this is needed as we get into the detail of the requirements within the timebox
- Someone to assure the quality of the product - something is only ‘done’ in agile when it has been tested and verifiably fit for purpose
- Someone to coach the team – generally and in agile. In Scrum this would be the Scrum Master.

In some industries, same where Company “X” operates, the project managers are individuals working almost always alone in one or multiple short time projects at once. Especially if it operates as a start-up. Therefore, there is no team to manage, and the delivery team is PMs itself. On some rare occasions there might be a project which requires a side help from other teams, for example overseas teams or contractors. In those cases, the PM is required to manage them accordingly depending on the project needs.

What we can Fix and Flex in Prince2 Agile

PRINCE2 Agile offers a ‘Hexagon Model’ that mixes clarity and control with flexibility. It has six points of focus: ‘Time’ and ‘Cost’, which should not be varied during the project management lifecycle, ‘Benefits’ and ‘Risk’, which may be flexible, and ‘Quality’ and ‘Scope’, which are highly flexible.



Looking at the hexagon from the top down we see first of all time and cost.

‘Fix’ is mentioned here, as is ‘Don’t flex’. What we are saying then, is that in an agile setting, the tolerance on time and cost is zero. If the delivery team forecast to exceed their timebox duration or the project manager forecasts the release cannot be done in the release timebox, this will immediately trigger an exception. Fixing time and cost, goes for all three levels of plan.

Moving down the diagram we have Benefit and risk. For benefits, the reason why it says ‘may flex’ is that we may need to fix a level of benefit realisation that ensures the ‘minimum viability’ of the business case. In a sense, the worst-case scenario. Thereafter tolerances could be set for anything above that. For instance, an over-realisation of benefits, which far exceeds the expectation, might cause big problems for small businesses who have limited warehousing, logistics and customer service resources to draw upon. They would actually prefer to raise prices to choke off demand, rather than try and cope with a huge volume of sales that is not sustainable.

Next, in the middle part of the hexagon we have Risk. This is listed as ‘may flex’. Generically, risk tolerance in PRINCE2 is at what point you need to escalate something that may happen up to the next level. Another name for this is a risk threshold. This aspect, and the tolerance associated with it, doesn’t really relate to agile or non-agile settings. We just need to note that some risks are connected with the fact that we are using an agile approach. Contrary to what some think in the agile community, using agile does not remove all risk!

Finally, we come to the things we would flex in an agile setting – the lower part of the hexagon. First, we have the idea of quality flexing. Let’s be clear here though, what we are NOT saying is that the overall level of quality will flex. In essence, our must have customer quality expectations

and associated ‘must have’ acceptance criteria are fixed, as detailed in the project product description. They are non-negotiable.

By flex, what we are referring to is flexing the quality criteria for individual products. For example, a surveillance system used by the military must work up to -10 degrees, ‘should’ work up to -20 degrees, and ‘could’ work up to -30 degrees.

These should have and could have criteria could then be dropped if needs be within the timebox. Finally, we come to the final aspect of project performance that we can flex: scope.

If you think back to the triangle – with time, cost and quality being at each apex. What we have said is that in an agile setting, these are fixed, remembering that with respect to quality, we are referring here to the overall level.

The only thing that can flex is what is in the middle of the triangle, namely scope. We flex what’s in, and by definition, what’s out.

If you are running out of time or running out of money, or both, the first thing you would try and drop is a “could have” requirement. In a sense something that is nice to have, but not essential. Next up, you would drop a should have requirement – this is more serious as we would do our very best to deliver these. Dropping should have caused pain, but the product being developed will still work without it.

This links to the idea of so-called ‘feature-bloat’ or ‘YAGNI’ – you ain’t gonna need it. Think of a washing machine for example. It has a dial with many different wash cycles, then it has buttons for pre-wash, intensive and so on, and finally, we have the different maximum spin settings. The permutations and combinations for the washing machine are into the hundreds, whereas most users will only use a hand full of settings - normal, woollens, sports clothing for example. We didn’t need all the additional cycles that are rarely used.

Agile would say it is better delivering fewer features, but delivering all the most important things, on time, on budget and thoroughly tested, rather than trying to deliver a washing machine with all the cycles, late, over budget and not tested.

In Company “X” 's context of fix and flex features we can clearly say that both time, cost and scope can be easily flexed. Quality is probably the only aspect where we have almost 0 tolerance in flexing. While starting up a project the appropriate team (here: Business Development department) prepares the project initiation document, engagement letter or project proposal. All the necessary details about the scope of the project, budget, timing and requirements from both parties are depicted there. After starting up a project within a certain amount of time passed either client or the team can negotiate or request a deadline extension to deliver the project. Therefore, time is flexible.

Equally, the client might change the scope of the project (i.e., change the target audience, require changes in the logic of the project, request quotas per population or disqualify some data points, etc). The cost of the project might be changed due to increasing or decreasing some requirements: “must”s and “should”s and/or requesting/offering discounts. Nevertheless, the Quality of the

project must always be the same, which is highly dependent on the market research, skills of the PM and some background information available.

Daily stand up or Daily Scrum

The daily stand up or Daily Scrum is a key time for the delivery team to think about risk. Traditionally, these are short meetings, lasting 15 minutes maximum, when the team gather around their team board, go around each team member and asks them to state:

- What did I do yesterday?
- What am I going to do today?
- What blockers, or impediments, do I have at the moment?

The third question is the most important for risk. Another name for impediment is ‘issue’, and one source of issues are risks that have materialised. This third question should then perhaps say... What blockers or impediments do I have at the moment or may I have in the future? Such risks could be recorded in a very simple form of risk register, perhaps just an extra few columns on the team board.

Company “X”: Currently there have been implemented the daily stand-up catch ups which are called “daily calls”. They usually last more than 15 minutes, since the team discuss more than 1 project and it takes time. Usually, the call has the following agenda:

- Discussing all the live projects
- Discussing the next steps needed to implement for that day
- Discussing the problems and solutions
- Presenting the daily engagement summary
- Allocating PMs for the starting soon projects

Retrospectives

Retrospectives are meetings that help to learn from experience. They are frequent meetings designed for inspecting the work process and trying to improve it for the next cycle.

Retrospectives can be held at the end of each iteration and each release in PRINCE2 Agile. The output can be used for capturing lessons learned, and identifying issues and risks.

Company “X”: Since the projects are short, having retrospectives after every project closure will be extensive. It might be beneficial to have a monthly feedback session discussing the performance of every project, lessons learned and the room for improvement. This can be combined with the overall performance feedback of the PM.

Once three months or once six months the PMs should receive detailed feedback from the Team Coordinator or Team Lead on their KPIs.

Feedback Loops

Talking about feedback, down at the detail level feedbacks are extremely important. The agile behaviour is explorational, meaning that feedback gathered from the customer/client as soon as possible is a gift in agile. It is the only way the customer gets what they actually want, not what they thought they might want at the start of a project. We should define the feedback be it good or bad, and if we are on the right track, we'll start to get quick wins of perhaps increased sales or publicity as a result. If the feedback is not what we were expecting, this is very useful. It could enable us to fail or learn fast, and potentially not pursue the project any further, thus saving valuable resources to be deployed elsewhere.

Essentially most feedback loops are a variation on Plan, Do and Review. We deliver something, it gets used or they test it, then it creates feedback which then drives further decisions and deliveries. In Scrum for instance, this is called Sprint or release review.

Ideally feedback loops will get shorter and shorter, and they indicate that we are heading in the right direction, or not as the case may be.

The most famous feedback loop is that of plan, do, check, act. This is similar to Plan, delegate, monitor and control in PRINCE2. Another is the so-called OODA loop from John Boyd. Boyd, a Colonel in the US Air Force, used to train fighter pilots to get inside the OODA loops of their adversaries. Observe, orientate, decide and act.

If you would like to read more about OODA loop, please visit the following website:

<https://onlinepmcourses.com/the-ooda-loop-take-control-of-your-project-with-this-powerful-idea/>

Feedback remembered is a 'gift'.

Company “X”: Ideally, the team asks for some feedback about the quality of the results and their expectations after 2-3 days of the start of the fieldwork of the project. It is essential to understand that the project is on the right track and the results provided are in line with the clients expectations. Before starting a project, the PMs usually set up a call with the client to understand the initial requirements and to get any necessary information about the market, trends, etc. Although challenging the information is acceptable and sometimes commendable, relying on the internal information only might be biased and not accurate.

Rich communications

Because of agile's love of transparency, iterative development, feedback and the need for quick decisions, face to face would always be a preferred communication channel. It has words, tone of voice and body language.

It's important to say though that while face to face, video conferencing and so on are favoured, there is no one perfect channel.

Email might be great to attach the PID (Project Initiation Document) - i.e. a Proposal, if sending to the project board but would not be the best channel to use in a quick 15 min daily stand up

meeting. Video conferencing might be great, but sometimes it's best to jump in the car and see the team in person.

The Project manager, and indeed team manager, need to make sure the team, and those above them, are using the right channels at the right time.

In addition, the project manager must make sure the documentation used on the project is clear and tries to utilise visualisation techniques such as information radiators. A picture paints a thousand words. This is clearer for the project manager to see evidence of, as opposed to reliance on emails which is harder to verify.

It is worth noting however that there is no documentation, as might be the case in some very basic agile environments. We need to ensure we have adequate controls but be mindful of PRINCE2's mantra – **we need information, not necessarily documentation, and decisions, not necessarily meetings.**

Company “X”: The world of communication and workstyle has been changed a lot. Nowadays it is not mandatory to be at the office (sometimes not even in the same country) to work in groups and deliver the desired results. Currently the operations team works mainly remotely. The daily stand ups or morning calls take place online. Moreover, even if the PMs and the whole team is at the office, the meetings are being held online in any case.

Having a healthy relationship within the team is more important than counting how many times we had a workshop together to bond the cultures, values, interests ,etc. Quality over quantity.

In more formal communications, the appropriate team member (usually the business development manager or account manager) updates the visual and colourful dashboard with all the necessary information to initiate the projects. Those actions are being done automatically without notifying the right person.

User stories

User stories are often described as a ‘token’ for a conversation, a starting point, and not even a partial description of a requirement. The whole point is that they are vague and promote the need for the specialists in the delivery team to further understand the customer's needs.

The format for a user story is: role, function, benefit or who, what and why. The card typically then features:

- As a <role>
- I want <function>
- So that <benefit>

For example, as a Walmart call centre manager, I want customers to get order updates with indicative delivery times and linked items, so that we reduce the number of calls coming in. Remember though, the ‘who’ element implies the requirement can be viewed from different perspectives.

For example, as a Walmart marketing manager, I want customers to get order updates with indicative delivery times and linked items, so that we can get additional purchases.

Finally, as a Walmart customer, I want to get order updates with indicative delivery times and linked items, so that I am reassured about when I will get my item and can explore similar items.

Company “X”: Here at Company “X” PMs do not get involved in user stories. The Business Development team on the other hand deals with customers (i.e. clients) and addresses their needs and pain points. To look at it from another perspective PMs try to understand the client’s expectations via research. As an example, if a client requests some information about certain products or markets, they usually do their own research and have their own hypothesis which “should” match the information PMs are about to give. Therefore, conducting market research to get an idea of what is the end result expected from the client is crucial.

Empiricism

One fundamental principle upon which most Agile’s methods are based is the concept of empiricism.

To put it simply, empiricism is planning on the basis of what is happening or has actually happened. If you got 7 user stories done last week for instance, then plan for 7 or possibly more user stories to get done next week. Agile uses ‘yesterday’s weather’ to forecast forwards.

This contrasts to rationalism, which uses reasoning or logic to make predictions and plan what should happen.

Using empiricism, in agile, based on what happened previously you are trying to increase the amount you get done in each subsequent timebox. This is why retrospectives and the concept of the Kaizen are so important – by having a stable team, with fixed timebox durations providing cadence and rhythm, we can increase the rate or progress – the so-called velocity.

Servant leadership

Next, we come to the agile concept of servant leadership as alluded to in frameworks such as Scrum, with its Scrum Master role, or accountability.

Starting with its origins in ancient history, the servant leadership concept was outlined in essays by Robert Greenleaf. Put simply, the best way to lead a team is to be its servant. You put the needs of the team above your own, and ensure the team is looked after and gets what it needs. The theory being- a happier team is a more productive one.

This concept was then adapted for agile, with this being the basis of not needing a ‘manager’ as such. This is then potentially problematic in a PRINCE2 environment.

There is much debate in the agile community on this concept, with some believing there is too much emphasis on the 'servant', and not so much on the 'leadership'. For this reason, the Scrum Guide of November 2020 refers to a 'leader who serves'. If we think to sport, football teams always need a manager, and a team captain; both of which need to display leadership on and off the pitch. For this reason, some prefer the term 'Facilitative leadership'.

Another comment to make is that servant leadership, which implies collective accountability up to a point, doesn't necessarily square with a single point of accountability in PRINCE2. Here the Executive is accountable for the success or failure of the project for instance.

That said, the PRINCE2 style of leadership is not defined, and in an agile context the hope is that the delivery team feel empowered enough to take ownership for the products they are producing.

Company "X": Within an organisational context Servant leadership becomes a bit complex to implement. The PMs are not fully independent in controlling the project and tools. Nevertheless, taking ownership of the project is one of the main drivers of success. The micromanagement is totally unacceptable in both agile and Prince2 context, equally is the full management. The key is to find the most effective leadership model based on the needs of the management.

Conclusion

In summary, this paper has explored the dynamic relationship between PRINCE2 Agile and agile project management principles, with a spotlight on Company "X." While PRINCE2 Agile's adaptability makes it versatile for various projects, Company "X" presents an interesting case.

Operating in a context where project managers often work solo or in small teams, Company "X" has integrated PRINCE2 Agile principles effectively. However, it's essential to note that their success highlights certain limitations.

Company "X" appears to have adopted PRINCE2 Agile in a way that aligns with their unique operational needs. Yet, this approach may not be universally applicable, and the paper raises questions about whether their specific circumstances might require different adaptations for other organisations.

In conclusion, PRINCE2 Agile offers a valuable framework, but its effectiveness can vary based on the organisational context. Company "X" provides an intriguing case study, showing both the benefits and potential limitations of PRINCE2 Agile when applied in distinct operational settings.

About the Author



Irina Poghosyan

London, UK



Irina Poghosyan is a Prince2 Agile certified project manager, currently serving as a Senior Project Manager at GrapeData, a prominent data analytics firm in London, England. For over 2 years she has demonstrated leadership and project management skills at this firm, her dedication and expertise have contributed significantly to the success of the company.

Her commitment to personal and professional growth is evident in her educational achievements. Irina holds a Master's degree (MSc) with Distinction in International Business and Management from the University of Westminster, where she received the prestigious Gold Westminster Award. Her academic journey began at the American University of Armenia, where she earned a BA in Business Administration and earned Best Capstone Performance Award for handling the business simulation with exceptional results.

Irina is not only a skilled project manager but also fluent in multiple languages, including English, Armenian, French, and Russian. In addition to her professional achievements, Irina has several publications to her name. Her research includes topics such as the world economic crisis, debt burden analysis, and the implementation of corporate social responsibility policies.

Irina can be contacted at theirinapoghosian@gmail.com