Whitepaper describing options for aligning PRINCE2 2009 and Agile Development practices in a Supplier context

By S.E.Brittain

June 2011

Overview

PRINCE2 (Projects in Controlled Environment) is a structured project management method based on best practice and offers a tried and tested approach to all aspects of project management. It is defined and controlled by the Office of Government Commerce (OGC) and is widely employed within the UK public sector although it is also used in public and private enterprises around the globe. Even if suppliers do not internally use PRINCE2 it is likely any work for organisations which have adopted PRINCE2 will require an appreciation of it. This paper assumes the reader understands the basics of PRINCE2 and understands the benefits brought about by Agile development.

Agile Development is a collection of different delivery frameworks, practices and techniques, some of which are interrelated, complementary, others contradictory or even fundamentally at odds with others. Although Agile should be considered a spectrum rather than a clearly defined categorisation, 4 common principles define the collection (known as the Agile Manifesto):

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan.

This paper addresses the question of whether a management methodology built around defined processes and products can fit with Agile practices, and if so how. Specially, this paper assumes the view point of a supplier delivering a capability into a customer – however more general conclusions regarding the fit of Agile with PRINCE2 can be drawn.

In the public domain, and recognised by PRINCE2, is declared compatibility with DSDM Atern. Being one of the more complete frameworks (arguably less "agile" on the spectrum), DSDM fits well owing to clearly defined roles and responsibilities, predefined phases and stages, and product descriptions. DSDM is complete in its own right and does not explicitly need PRINCE2, however the structures defined in DSDM are compatible and extend to a lower level than PRINCE2. Given the generally accepted compatibilities, this paper will concentrate on other methodologies which may prove more challenging.

Concerns

PRINCE2 has a reputation for being bureaucratic, overbearing and inflexible – designed upon the premise of Big Up Front Design (BUFD) (e.g. Waterfall). Whilst this reputation may have some foundation, it is almost certainly as a result of the type of organisation employing it, the type of projects it is being employed on, and a lack of tailoring to ensure it is fit for purpose. Specifically with the latter point, PRINCE2 has taken steps to ensure tailoring is understood as a core PRINCE2 Principal. Another point to consider is who is performing the tailoring. Normally this will be done by a Project Manager who may or may not

appropriate the nuances of Agile development.

Principles

PRINCE2 is application neutral but with a focus on upfront planning, agreeing scope, and quality requirements; it is a more natural fit with more traditional/ less agile BUFD frameworks. However, PRINCE2 defining principles, fit Agile surprisingly well, specifically:

Continued business justification – a more important consideration in agile than BUFD

Learn from experience – embedded into all agile frameworks by virtue of their iterative nature

Defined roles and responsibilities – Agile has no issue with defined roles (something that many agile practices themselves recommend/have standards for). Note that PRINCE2 defines roles and not jobs, hence roles can be shared and/or transferred (within some sensibly defined boundaries)

Manage by Stages – Again, this concept is akin to many agile development frameworks (e.g. SCRUM, RUP, DSDM, FDD etc.)

Manage by exception – this is about getting on with the job and removing bureaucracy; in line with Agile Principles

Focus on products – Not quite fully aligned with Agile, as PRINCE2 is about defined outputs (products) to agreed criteria whilst Agile is more about delivering business value, however they are not intrinsically at odds (discussed later on).

First Steps

PRINCE2 can be tailored to a lesser or greater extent depending on the characteristics of the project.

PRINCE2's tailoring approach is based on reducing formality, frequency of reporting and aggregation of roles. PRINCE2 does not support selective adoption – i.e. you either adopt PRINCE2 and all it entails, or you do not claim to be running a PRINCE2 project. This in itself is not an issue, however it is in tension with most agile principles which would prefer to see removal of non "productive" processes. When compared with frameworks such as XP, comparisons can be made; is XP still XP if Test Driven Development is removed?

Given that you should not selectively use PRINCE2, as a supplier two approaches are available; whether to stand up a full "PRINCE2 project", or to position the project as a Work Package within a Customer's PRINCE2 project (or both).

Full PRINCE2 Project

In this approach, a full PRINCE2 framework is stood up to facilitate the delivery of the project. This will involve setting up and running reporting lines, budgets, stakeholders etc. In reality many elements of his will need to be done in any eventuality, however, if PRINCE2 is adopted, certain PRINCE2 practices and timings come into play (see Figure 1).

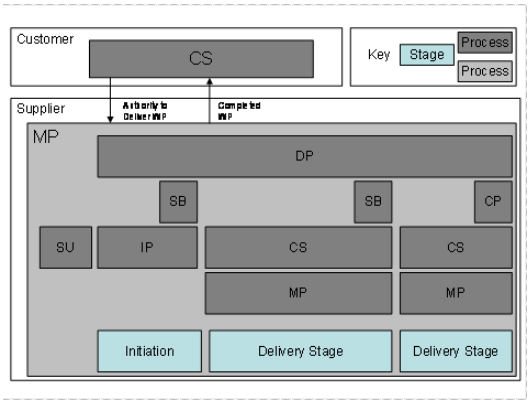

Figure 1: Delivering under the umbrella of PRINCE2

PRINCE2 at Work Package

On the assumption that the client has a PRINCE2 project, by defining a supplier's project as a Work Package it is possible to side step the majority of the PRINCE2. This may be considered a good or bad position depending on the level of benefit you perceive PRINCE2 brings.

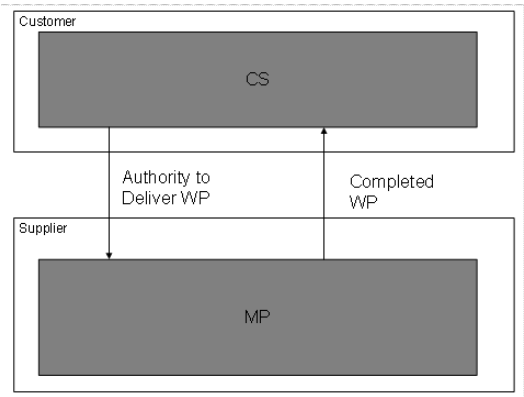

Figure 2: Delivering under the umbrella of a Work Package

As the technical method of delivery within a Managing Product Delivery (MP) is not defined and should be a single contiguous period of time, the mechanics of how work is undertaken within this time period is wide open.

The remainder of this document will focus on delivery using a full PRINCE2 project controlled and run by the supplier. Note in Figure 1, that two PRINCE2 projects are being run, one customer side, the other supplier. This is fully compatible with PRINCE2 as both the customer and supplier will be working to two different business cases, and the supplier's contribution may only be part of the customer's wider project.

Plans and Stages

As indicated by figure 1, the breakup of stages needs to be agreed; note however these are "management" stages and do not necessarily translate into "technical" stages. The division of technical stages into management stages is a critical decision, as failure to create the correct stage boundaries could negate PRINCE2's manage by exception principal, or empower the Project Manager with too much authority. Under PRINCE2, management stages cannot overlap, as they form the basis upon which projects invest resources (i.e. you cannot start another stage unless authorised by the board).

The division of stage boundaries and their alignment to technical frameworks (agile or otherwise) will vary from project to project. The following are some examples for outline project plans in a variety of frameworks on the Agile spectrum (from Unified Process to Scrum).

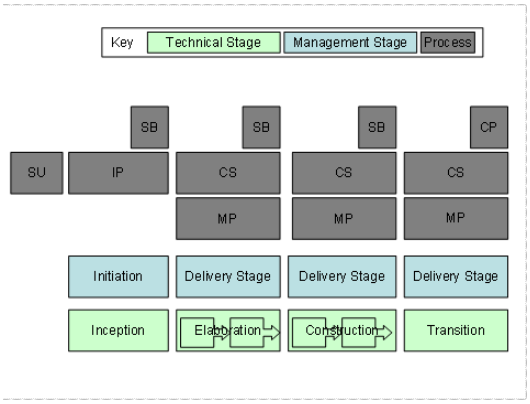

Figure 3: Example Stage boundaries using Unified Process (UP)

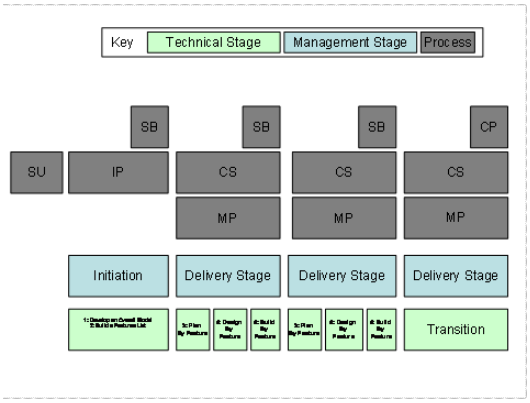

Figure 4: Example Stage boundaries using Feature Driven Development (FDD)

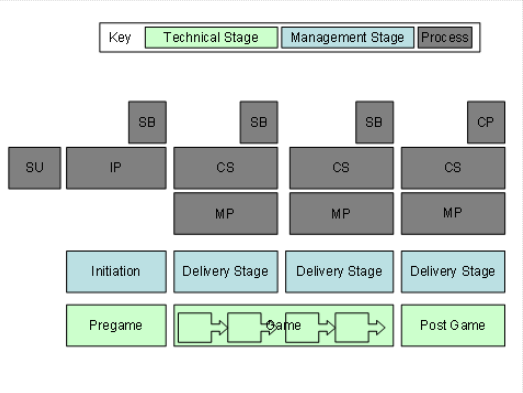

Figure 5: Example Stage boundaries using Scrum

Common to all approaches is scoping and derisking of the project. Is should be undertaken during the Initiation Stage (IP) so that the Project Initiation Document can reflect the various options and risks associated with the project. Iterative development, should (dependant on project timescales) contain one or more management stage boundaries during the development so that progress can be formally assessed and the project board has the opportunity to address any matters arising. Finally, a management stage will be required to handover created products (i.e. to finalise testing, packaging, delivery etc.) prior to the project formally being closed down.

Note that a solution to the issues inherent to BUFD (e.g. Waterfall) is to have overlapping phases (e.g. writing executable code/prototypes whilst designing, developing code whilst testing etc). Ironically, this solution presents more of an issue to PRINCE2 than Agile, as management stages must not overlap.

Outputs

The least likely bedfellow to Agile is the Work Package and associated Product Descriptions – the output of the Work Package. It is likely that the products being defined will be at a reasonably high level when compared to the individual "products"/components each developer will be response before – i.e. the products will describe the end result of the development and any associated documentation.

The PRINCE2 Project Product Descriptions and Product Description are high-level summaries of what the project is producing and their associated success criteria. Work Packages details the high level mechanics and protocols for developing the Products. Of particular importance is to achieve flexibility in the following criteria:

Product Descriptions

- Quality Responsibilities – Reviewers must include user representation who are able to make a business priority call in accordance to Agile principles
- Quality Criteria and Tolerance are pragmatic and support a fit for purpose delivery

Work Packages

- Tolerances – Identifying where the "give" in the task is. For agile, this should most significantly be in scope so that every possible chance of preserving Time and Cost can be made.

By agreeing up front, Work Packages and Product Descriptions are a useful instrument to document specifically areas where scope and quality can be compromised in order to maintain the deliver schedule.

Although it would be unlikely that formal PRINCE2 product description are handed down to the developer at the component level, the exact same information is likely to be provided – i.e.

scope and purpose, timescales, quality criteria etc. This is another example where PRINCE2 introduces what is generally perceived as "common sense".

Roles and Responsibilities

Generally speaking the role and responsibilities definitions within the Agile frameworks do not interfere with PRINCE2. This is as a result of the Agile frameworks naturally focusing on the technical aspect of delivery, whilst PRINCE2 focuses on the more general governance.

Scrum

One notable exception however is Scrum which is arguably more of a management methodology than of development (a topic for another paper). Scrum's mantra is the high performance self organising team which sets itself at odds with more traditional hierarchical management structures adopted in most other frameworks – PRINCE2 included. This in itself is not an issue for PRINCE2, however the responsibility of Scrum's "Scrum Master" role and the Project Manager role in PRINCE2 needs care to ensure the essence of Scrum (i.e. the self organising team) is not jeopardised.

PRINCE2 sets out a set of responsibilities for the Project Manager, of specific issue being "The Project Manager is … responsible for the project producing a result capable of achieving the benefits defined in the Business Case". This is at odds with Scrum as this responsibility is taken on by the team, with the Scrum Master role helping to facilitate success and manage the team's processes. The danger of combining the PRINCE2 Project Manager and Scrum master roles is that this distinction will be lost, the team will look to the Project Manager for leadership, or the Project Manager will impose leadership. In this scenario, the defining attribute of Scrum will be lost – i.e. the high performance self organising team. It is therefore strongly recommend that, if at all possible, the Scrum Master and Project Manager roles are not combined. If they are not, the Project Manager must be fully versant and aware of the sensitivities around the dual role. A suitable organisation structure is defined below.

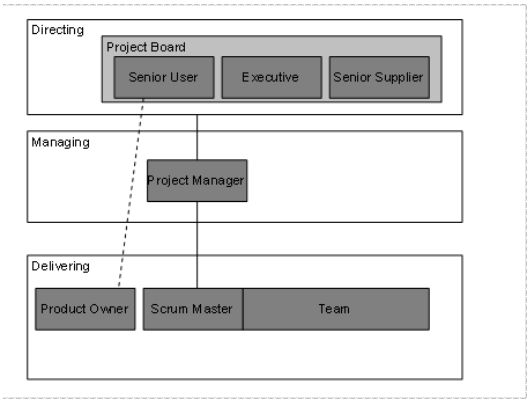

Figure 6: PRINCE2 Organisation structure in Scrum

A number of key points should be noted:

- The Project Manager should be considered a scrum "stake holder" and therefore should be invited to the Sprint Review Meeting. The Project Manager should take a passive "chicken" role if they attend the daily stand-up.
- The Scrum Product Owner and the Senior User share responsibility for the benefit to be derived from the project. It is advisable that a direct reporting line between them is setup so that user perspective on the benefits derived is shared. In this sense the Product Owner will be undertaking a degree of Assurance.

- Scrum Master will be taking on the Team Manger role. As per all Agile frameworks, Work Packages will need to be crafted to fix the time and effort as opposed to scope (although the estimated should have been assessed to ensure it has a reasonable chance of delivering the requested output).

The wider structure for the project in the scenario whereby the supplier was running their own PRINCE2 project would be:

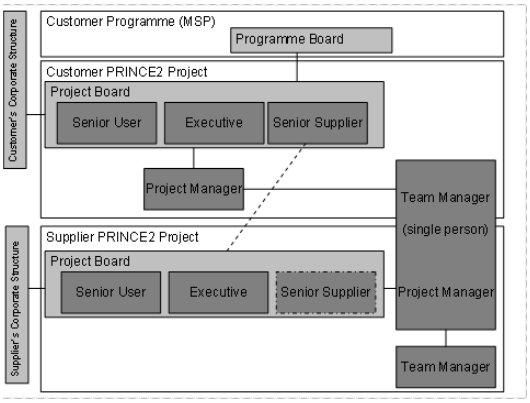

Figure 7: Wider Governance Structures

In this scenario, the "Project Manager" has a dual role; firstly the Team Manager reporting to the Customer's Project Manager within the customer's project (accepting Work packages/delivering products), and secondly, the Project Manager for the supplier, reporting into the supplier's project board. Note that although this looks on paper bureaucratic and cumbersome, with or without PRINCE2 you would end up with something very similar (the structure is a natural one).

Of particular importance to Agile is the Senior User roles. As requirements have not been fully articulated at the beginning of the project (i.e. the exact scope has not been defined), it is critical that the development teams have access to empower user representatives. As a Senior User on the supplier's project should be read into the supplier's business case which the supplier may not want to share externally, a decision is required:

- the supplier's business case may need to have private sections that the senior user is not party to, or
- a proxied senior user (i.e. requirements manager/lead business analyst) may be more suitable if the customer's senior user cannot commit.

(Note that similar sensitivities arise over the Senior Supplier role in the customer's project). As the project board is ultimately responsible for authorising changes to scope, adequate user representation is critical to ensure the impact of decisions are fully understood.

User Roles

The role of senior user may trigger alarm bells for any Agile practioner. This is because understanding the real user requirement by interacting with them is fundamental to delivering fit for purpose solutions. Although interaction with users continues throughout all agile lifecycles, the level and purpose of the interaction between the Initiation and technical delivery stages will vary depending on the agile approach. For example, more upfront design effort will be required with some frameworks (e.g. UP, FDD) whilst others will reduce upfront design to a minimum but rely heavily on it during development (potentially SCRUM dependant on the objectives, definitely XP).

A "senior" user as opposed to an actual user, especially one with a view on the overall business case as defined in PRINCE2, is likely to have a different take on requirements – this is an issue which would need to be solved on a case

by case basis. However, this is not a PRINCE2 issue, more a universal one which is more obvious when management responsibilities are documented (for example, it is common place to meet conflicting user requirements on a project). PRINCE2 does not discourage end user involvement in the technical delivery stages. Therefore the Senior user is a positive aspect of PRINCE2 as it actually adds some formal dispute resolution structures via an escalation path should the all to common occurrence of issues around end user requirements occur.

Responsibility

A common criticism of PRINCE2 is that it moves responsibility for decision making from the Project Manager upwards. This certainly is the case, with more emphasis placed on the Project Board – however for good reason. Not all aspects of the projects successful running is down to the Project Manager. This is of particular importance with agile methods which are likely to be used on projects with shorter more aggressive timescales. Issues will crop up such as:

- Contention of resources
- Performance of the delivery
- Issues around requirements.

It is therefore critical that structures are in place such that the Project Manager can escalate and quickly resolve issues quickly and agree change to ensure time is not wasted. This is central to the ethos of Agile, and explicit in Scrum. By appropriately defining Management Stage boundaries and appropriately delegated risk and change budgets, the Project Manager retains healthy levels of responsibility, however, in real life the "buck" seldom stops with the Project Manager. Senior executives are seldom quiet onlookers; therefore, it is healthy and correct to formally include directing level decision making. As indecision is Agile's number one enemy, PRINCE2 should be seen in a positive light.

Conclusion

It may be overstating the position to say that PRINCE2 and Agile naturally go hand in hand, although it would be wrong to suggest they are incompatible. With a sympathetic Project Board and knowledgeable Project Manager (hopefully this whitepaper helps) the two can quite happily co-exist on a project and, more importantly, it can make a lot sense to do so. To go one step further, the process and discipline introduced via PRINCE2 to project management, is akin to the process and discipline introduced by Agile for the developer. Therefore the adoption by a supplier of PRINCE2 within the organisation, rather than relying on the rather open mandate provided by the Manage Work Package process has .

The apparent additional overhead of running a PRINCE2 organisational structure is misleading as the structures required would be needed for any reasonable project governance. The artefacts can be tailored to meet the level of formality required (i.e. reports could be verbal, email or official documents). It is likely therefore that the only difference in cost is a small additional Project Management/PSO cost to maintain good order and discipline (e.g. registers and logs etc), a cost that this likely to be rewarded with reduced risk of project failure as a result of poor management.