

Issues in Successfully Implementing PRINCE2

This short note discusses some issues which I have seen causing difficulties with PRINCE2 implementations. I hope that this discussion will remove some of these allowing organisations to move forward.

Where to Start?

First, we should discuss the PRINCE2 qualifications. There are two, a relatively straightforward multiple-choice examination lasting an hour, the Foundation examination, and a three hour exam where PRINCE2 principles are to be applied to realistic scenarios, the Practitioner examination.

To obtain these qualifications PRINCE2 course delegates are naturally concerned to apply the principles of PRINCE2 in a mechanistic manner. They are concerned not to step out of whatever the boundaries of PRINCE2 are perceived to be.

It seems to me that this approach unfortunately leads to dogmatic application of PRINCE2 principles. In fact, these principles often seem to be overstated in many projects to imply that they have significant management overhead (I am also thinking that in some ways this is a hidden form of contingency planning, whereby when the project gets into trouble the vast allocation of effort to pure-PRINCE2 work can be moved to more directly productive areas).

Recently I came across an application of PRINCE2 to a smallish project. A technician was asked by his boss to move some data files from one PC to another. They began by writing their Job Descriptions, with the help of HR! Now, I know that this is incorrect application of PRINCE2 principles, but it is of concern when any right-minded person can think that this is an appropriate method.

What Should We Have?

PRINCE2 is the result of many years of development work, starting with PROMPT. It suffers from this growth by accretion in that there does not appear to have been a stepping back to look at PRINCE2 from the outside in.

In particular, the very valuable and essential, principles of PRINCE2 are not to be found gathered together. They are scattered about, and in fact, I doubt if anyone could be sure that they have collected all of them together given the varying terminology in the PRINCE2 manual. So, I am arguing for a clear statement of PRINCE2 principles. This would provide a basis for commonsense application of the PRINCE2 techniques and sensible execution of the processes. As it is, we have to try to glean the principles from reading, and re-reading, the PRINCE2 manual ... not the most exciting book in the world.

So, a priority would be a PRINCE2 Principles abstract, to be used as a guide for the real-world application of PRINCE2.

I am hoping to act as a collator of such a document over the next few months.

Agile / Lightweight Methodologies and PRINCE2

Having a technical background I have been familiar with the technologies of SDLCs for many years. The funny thing is that when the methodology comes from the technical staff, the focus is on heavyweight technical aspects and management aspects are very light. When the methodology

comes from the management staff, as in PRINCE2, it is precisely the other way around. In fact, PRINCE2 goes out of its way to put the technical work way down into MP! This is probably what you should expect, but it does make it very interesting when trying to merge the technical and management-based methodologies.

Lately, Agile or lightweight, software development processes have appeared and they do genuinely offer efficient development environments (with the usual provisos that all of the Agile requirements are followed and not just cherry-picked). Agile techniques recognise some truths about software development: We often do not know what will be precisely produced; The users do not know precisely what they want; We do not know how long it will take; These challenges to software production would seem to directly contradict PRINCE2 principles and hence rule PRINCE out as a framework for managing Agile projects. One of the results of this is that methodologies such as DSDM are starting to appear (in its latest incarnation) as a solution to the PRINCE2 / Agile conflict. Yet I maintain there is no conflict, and we should not deviate from the application of PRINCE2 principles and terminology without very good reason.

Agile projects run in Stages. So does PRINCE2. The PRINCE2 SU still makes sense to apply to Agile processes as it is surely not too much to ask if the project is worth doing. IP (which can be combined with SU when appropriate) can be used to identify the major features of the project, the quality controls, techniques and so on. What it cannot do is be prescriptive about precisely what products will be produced. It should be able to identify and prioritise the business benefits, which, as ever, provide the background for the project, agile or otherwise. CS / MP is the bread and butter of Agile stages. What is produced in these stages is decided as the project run, at the weekly meetings, from a list of "tasks" (sometimes called the Project Backlog) which have been prioritised by the embedded User. Each Agile cycle typically runs 2 to 4 weeks. This fits perfectly with PRINCE2, where commonsense dictates that we do not (pretend to) plan to levels of detail which are impossible and which uses stages to provide frequent sanity-checks on project progress.

Agile projects frequently deliver product using demonstrations and tests as the reporting vehicle. The Project Board may have to get used to visiting the team for such demonstrations, However with the correct level of QA, I expect most Boards would prefer this to the reading of another pile of reports, assured or otherwise.

Summary

The production of a distilled set of PRINCE2 principles is a matter of some urgency if PRINCE2 is to be successfully applied to modern Agile development techniques. We should resist the introduction of other management techniques to such projects unless there is a very good reason, when in fact, it would be preferable to adapt PRINCE2 to the new requirements.

If anyone is interested in helping to maintain a PRINCE2 set of Principles please contact the author at george@oaklodgeconsulting.co.uk