PIONEERING THE COMBINED USE OF AGILE AND STAGE-GATE MODELS IN NEW PRODUCT DEVELOPMENT – CASES FROM THE MANUFACTURING INDUSTRY

Saeema Ahmed-Kristensen
Technical University of Denmark,
Produktionstorvet, Building 426, DK-2800
Kgs. Lyngby, Denmark
sakr@dtu.dk

Jaap Daalhuizen
Technical University of Denmark,
Produktionstorvet, Building 426, DK-2800
Kgs. Lyngby, Denmark
jaada@dtu.dk

ABSTRACT

Stage gate models have long been the norm in manufacturing industries. Whereas agile models, such as scrum, are standard amongst software industries. These two models have been traditionally been pitted against one another, each with its own advantages and limitations. A new trend is being observed, where gated models are becoming more agile, and agile models more gated. This trend indicates a need to combine the models, recognizing the limitations of both. We explore a number of cases where the models have been combined, very much based on trial and error, with adaptions being made as needed. The findings are relevant to product- and engineering design theory and education as stage gate models are commonly seen as the basis for modeling and teaching design. Certain qualities of agile are expected to be integrated into stage gate models, from which new process models might emerge. Such processes are expected to integrate more interdisciplinary collaboration along different stages of product development, facilitate more flexibility in setting requirements and allowing change. This article offers managers the possibility to understand how to combine stage gate and agile models and why, based upon cross company learning, hence fast tracking this process.

THEORETICAL BACKGROUND

Stage gate models

Stage-gated models have long been part of product development in the manufacturing industry. A stage gate model can be characterized as a “conceptual and an operational model for moving a new product from idea to launch” (Cooper, 1990, p.1). It does so by prescribing a number of stages or phases in the product development process each guarded by a gate where certain pre-determined results need to be delivered before moving to the next stage. Stage gate models are typically introduced in companies once they grow over a certain size, in order to make the innovation process more manageable, to facilitate coordination of information and tasks. Such processes have been part of the engineering design field for many decades, with the recognition of the need for methods to become more predictive and evaluative (Gregory, 1966). Early contributions to product development focused on systematizing and rationalizing the product development process (see e.g. Alexander, 1664; Archer, 1965; Jones, 1972; Hubka, 1982). Over time, stage gate models evolved in various generations of models (Rittel, 1984; Rothwell, 1992). Innovation models evolved from being simple, linear technology push to more and more integrated models that balanced technology push and market pull, interaction between stages and stakeholders (Tidd and Bessant, 2009). With their evolution, models have been labelled in many ways, for example plan-driven models, waterfall models, etc. In the context of this paper, we will use these terms interchangeably. That is, stage gate models are an example of planned product development models. They typically begin with a planning phase, moving to development through to testing and full-scale production. There are variations of the model, for example for development of complex systems (see e.g. Ulrich and Eppinger, 1995).

Agile product development models

Agile product development models emerged in the 1980's out of the need for organizations to become faster and more flexible in their innovation efforts (Takeuchi and Nonaka, 1986). Agile models were developed as a response to the sequential stage-gated models and have a more holistic nature in which multidisciplinary teams work in parallel along the product development process. In the words of Takeuchi and Nonaka, teams work "*as in rugby, the ball gets passed within a team as it moves as a unit up the field*" (1986, p. 1). These authors characterised agile product development in terms of six qualities:

- First, an agile process has built-in instability. This means that projects are kicked off by top management with broad goals and extremely ambitious targets. In this way, tension is built into the project which, with the right team, will drive creativity.
- Second, agile development requires self-organizing teams. This means that teams are given goals that are so ambitious that they are driven to a state of "zero information" and start acting like a start-up organization.
- Third, development phases overlap. This means that instead of moving from gate to gate, fulfilling all requirements per gate and handing over results, a team moves along development phases together. This typically happens while diverging and converging to deliver certain results along the process.
- Fourth, agile processes induce "multi-learning". Due to the tight interactions between team members, learning happens along individual, group and corporate level as well as across functions.
- Fifth, management exercises "subtle control". Management leaves teams largely on their own, yet builds in sufficient checks to avoid chaos, emphasizing self-control, control through peer pressure and control by passion.
- Lastly, team members also transfer their learning to others outside of the group across the organization.

Although Takeuchi and Nanoka based their research on a number of cases from the manufacturing industry, however agile product development has mainly been taken up in the software industry, with many organizations using agile models for their new product development efforts (Kettunen, 2009). Agile methodologies such as Scrum, eXtreme Programming (XP), and Feature-Driven Development (FDD) have long been in use to deal with issues like increasing global competition, emergence of new technologies and the diversification of customer demands. The trend of using agile models in software was sparked by the formulation of the concept of Agile Manufacturing (AM) (Goldman et al., 1995; Preiss, 2005). The publication of the Agile Manifesto further popularized the movement by formulating a set of principles to drive agile development in organizations (Fowler and Highsmith, 2001). Both AM and many agile methodologies can be viewed as being in line with each other, with AM covering many of the larger-scale business and organizational areas related to new product development more extensively than agile methodologies do (Kettunen, 2009). More specifically, going beyond the operational AM covers areas such as the interface between NPD and business, supply chain and workforce factors (Sanchez and Nagi, 2001; Yusuf and Adeleye, 2002; Vazquez-Bustelo and Avella, 2006). The key benefit of agile development is the increased ability of project teams and organisations to respond to change (Kidd, 1997). Agile models are focused on the company-market interface (Katayama and Bennett, 1999). Being able to respond to change enables an organization to be flexible, responsive and pro-active (Kettunen, 2009; Zhang and Sharifi, 2007). According to Kettunen (2009) flexibility and responsiveness are related to the ability to respond to unpredictable changes in demand in a timely and cost-effective manner. Furthermore, Kettunnen (2009) describes the ability to be pro-active as being related to the ability of an organization to create new organizational capabilities ahead of any changes that have been observed.

The use of stage gate & agile models

As already described above, stage gate models are predominantly employed in manufacturing industries. A key characteristic of physical product development is that capital investments are typically committed at an early stage. That is, decisions that are made during concept design commit a large proportion of the total cost of a project. This is due to the dispositional effect of early decisions on manufacturing processes – e.g. tooling, production, assembly, and distribution – later on in the development process (Ullman, 1997; Andreasen and Olesen, 1990). To deal with this, product development processes have long been organized as stage-gate systems in which requirements and specifications are formulated early and fixed in subsequent stages. Stages are guarded by a gate at which point reviews are organised where go/no go decisions are typically made. Stage-gate models are used to manage innovation efforts and reduce cycle time of product development projects and improve chance of success (Cooper, 1990).

Agile models have long been dominant in the software industry. In software development, the production of code is typically linked to human resources. That is, the definition of specification and the production of code are both driven by human capital, often at the same time. Thus development cycles can be relatively fast (order of hours, days) and requirements can be changed along the development process. As capital investments are linked to human resources along the whole development process, there are usually no physical rework constraints and cost of change is thus low. In addition, due to the flexibility of code production, even radical redesigns are possible a late stage in the process (Kettunen, 2009).

Combining stage gate & agile models

In current literature, very few studies address the combining of planned models with agile models. A possible explanation for this, which is confirmed later in the results section of the paper, is the relative newness of the approach in manufacturing companies. A couple of papers that do look at this area are reviewed here. In the context of software development, Boehm and Turner (2003) have pointed to the combination of these models. They suggest their combination is based on the type of risk the projects is likely to be exposed to and suggest that a project is divided into parts that are agile and those that are planned, depending upon whether the risks are primarily agile or planned. If the risks are primarily agile then that part of the process will be planned, and if the risks are primarily related to planned process then these parts of the process are agile. Boehm and turner (2003) identify three types of risk: (1) risks that are a result of the environment of a project, (2) risk that are related to the use of agile processes, and (3) risks that are related to the use of plan-driven processes. Determining a project's risks is done by assessing to what extend a project is close to, or defers from, the so called "home ground" of either of the processes. Home ground is defined as the set of conditions under which a model is most likely to contribute to success. The home grounds for agile and plan-driven processes are characterized along four dimensions: application, management, technical and personnel (see table 1).

Table 1. Agile and Plan-driven home grounds. Adapted from Boehm and Turner (2003) with some minor revisions.

Project characteristics	Agile home ground	Plan-driven home ground
Application		
Primary goals	Rapid value, responding to change	Predictability, stability, high assurance

Size	Smaller teams and projects	Larger teams and projects
Environment	Turbulent, high change, project focused	Stable, low change, project and organization focused
Management		
Customer relations	Dedicated onsite customers, focused on prioritized increments	As-needed customer interaction, focused on contract provisions
Planning and control	Internalized plans, qualitative control	Documented plans, quantitative control
Communications	Tacit interpersonal knowledge	Explicit documented knowledge
Technical		
Requirements	Prioritized informal stories and test cases, undergoing unforeseeable change	Formalized project, capability, interface, quality, foreseeable evolution requirements
Development	Simple design, short increments, refactoring assumed inexpensive	Extensive design. longer increments, refactoring assumed expensive
Test	Executable test cases define requirements, testing	Documented test plans and procedures
Personnel		
Customers	Dedicated, collocated performers who are collaborative, representative, authorized, committed, and knowledgeable.	Performers who are collaborative, representative, authorized, committed, and knowledgeable., not always collocated
Developers	At least 30% full-time highly mature method users. No novice or uncommitted method users	50% highly mature experts early in process, 10% throughout process. 30% novice method users is workable. No uncommitted method users.
Culture	Comfort and empowerment via many degrees of freedom (thriving on chaos)	Comfort and empowerment via framework of policies and procedures (thriving on order)

Cooper (2001) argued that (agile) software development happens in relation to for example hardware development, marketing efforts, production planning which are typically governed using stage-gate models. He argues that both models can be used together, and suggests that agile processes are used between gates for software aspects of the products, indicating that the splitting is based in software and hardware. He describes this as almost a micro- process, for software-related activities between gates. In relation to this, Karlstrom and Runeson (2006) discuss the combined use of both models, arguing that stage-gated models are needed both for communication purposes within a project as well as for decision makers that sponsor projects or that pay for the outcome of a project. Indeed, Boehm (2002) argues that a combined use of the models can provide software developers with a rich set of methods and strategies, each with their own benefits. However, Boehm also points to the importance of a responsible use of both models. This leads us to the question: *how are manufacturing companies combining agile and stage gate models and what are the necessary conditions for successful integration?*

Aim of the research

The motivation for this research came from two aspects: 1) an observation that manufacturing firms are beginning to combine agile processes with more formalized planned processes such as stage gate models, and; 2) an absence of a body of literature that formalizes the combinations of the models explaining the way that they are combined, and the results that should be expected from these processes. The aim of the research was to investigate the combined use of agile methods with plan-based product development methods, such as the stage gate model in manufacturing companies. The following research question guided our research:

- Why are the models combined, and how?
- What are the necessary conditions for the combined use of models?

Prior to conducting the interviews and based upon the understanding from theory of agile and planned product development models, the following hypotheses were formed.

1) *Agile processes are more likely to be used for innovative products, where uncertainty and/or complexity is high.*
2) *The models are more likely to be combined in situations of high uncertainty, leading to the use of agile to break down tasks, and then returning to stage gate (planned process).*
3) *The combining of agile and planned processes requires that requirements are kept more abstract in order to allow change when agile process are integrated as part of stage gate models.*

RESEARCH METHODS

An empirical study was conducted to investigate the research questions, the methods for data collection, and analysis are described here.

Data collection

Interviews were conducted in order to investigate how stage gate and agile models are combined in the manufacturing industry. The criteria to select the case companies for the interviews were the following. The companies needed to be: using both agile and planned (stage gate) product development models and; manufacturing products with some part of the product that was hardware (software and electronics could also be included). This put a limitation on the type (and number) of companies that could be possible candidates, as these are first movers in product development processes, with implementers who are heavily engaged in keeping up with the latest research and models. The interviews were conducted with project managers or vice presidents of research and development who were involved in the design and implementation of the product development process. Semi-structured interviews were conducted as these allowed for consistency across the interviews by using the same questions, but also allowed the participants to expand upon their answers more freely. The interviews lasted one hour and were audio-recorded with permission of the participants. Both researchers were present at each interview, with one taking the lead in the questioning and the other primarily taking notes. An interview guide was prepared to ensure consistency of questions across the interviews, and the questions focused on a number of themes, including:

- Understanding how the two models are perceived individually with respect to dealing with change, uncertainty, level of innovativeness resulting from the process, the complexity of products, competences and size of teams.
- Understanding how, when and how the models are combined.
- The challenges faced to the project and team and how these are addressed through adaptions made to agile and the planned process.

The interviews were conducted in four companies, all acting on an international level. Three interviews were conducted in Denmark, and one in the Netherlands. The case companies, experience level in combined use of agile and planned process in hardware, and the interview participants are summarised in Table 2.

Table 2. Overview of the company cases involved in the study.

Company	Industry	Number of	Level of	Participant

		employees	maturity in using combined model	interviewed
Case A	Danish headquarters of a US company producing medical devices, such as blood measuring equipment.	> 2700 worldwide	1-2 years with hardware. Experience with one project.	VP Research and Development, Responsibilities include implementation of process
Case B	Danish company working with hearing measurement devices.	> 500	4 years for Hardware and mechanical parts (5 years for Software)	Product Development Manager Responsibilities include implementation of process
Case C	Dutch multinational company manufacturing healthcare, consumer products and lighting products.	> 100.000 worldwide	3-4 years with hardware (10-15 years with software)	User Experience director. Responsibilities include development and organization wide implementation of new processes for innovation.
Case D	Danish company manufacturing large industrial valves.	> 24.000 worldwide (5400 in Denmark)	Experience in software. Not yet implemented in hardware but planned next few years.	Product Development Process Expert of Refrigeration and Air conditioning Department.

RESULTS

The interviews were analysed qualitatively, and the findings are presented here. First we describe how and when the processes are combined. Then the adaptions are described that were made by the companies diverging from traditional theory of agile processes and product development processes. Last, we outline the challenges seen with combining stage gate and agile models.

How are when are the processes combined?

Prior to the interviews, two hypotheses were formed relating to the combined use of the processes:

1) *Agile processes are more likely to be used for innovative products, where uncertainty and/or complexity are high.*
2) *The models are more likely to be combined in situations of high uncertainty, leading to the use of agile to break down tasks, and then returning to stage gate (planned process).*

From the cases described, neither of these two hypotheses were found to be the case. The models were combined in different ways in the different organizations, and treating the project as whole, with iterations going across gates (see table 3).

All four companies were making a transition from stage gate models to a combined use of stage gate and agile models, and the use of these models was across projects. The cases studies, although limited in number, provided a rich picture that was significantly different to that described in literature. All cases combined the models treating a project as a whole, contrary to the view that a project is split into agile and planned parts. For example, Boehms (2002) suggested that a project is split based on type of risk. Cooper (2001) suggested that projects are split in software and hardware tasks with agile being used for software tasks and in iterations within gates, as a micro process.

The combination of the models was very much embedded into the planned process, i.e. agile thinking and processes were combined within an overall stage gate model. So, contrary to the original hypothesis, the processes were integrated, rather than parts of the project being planned in agile processes and others in stage gates. They were numerous reasons for doing so, including strict regulations within the particular domain, i.e. regulations for medical devices for documentation (Case A). In addition, all of the cases described the need for planned processes and this is exemplified with one participant from Case A stating "it is dangerous to allow too much flexibility; you need a good idea of what you are doing and to understand the direction". The planned models were seen as the primary model needed for regulations, with related quality system and deliverables, whereas the agile processes were being used at and across the gates to drive development of content. In Case A, the role of the regulations in dictating the documentation that is needed to be delivered at the gates was highlighted, and although also mentioned in the other cases, this was to a lesser extent. In all cases where the combination had been implemented (Cases A-C), the iterations of the agile went across the gates, i.e. where not simply meant to produce deliverables for the gates. In those cases, deliverables for the gates were dealt with by one or two people from each team working on the deliverable whilst others continued with the iteration. This practice was used to keep momentum.

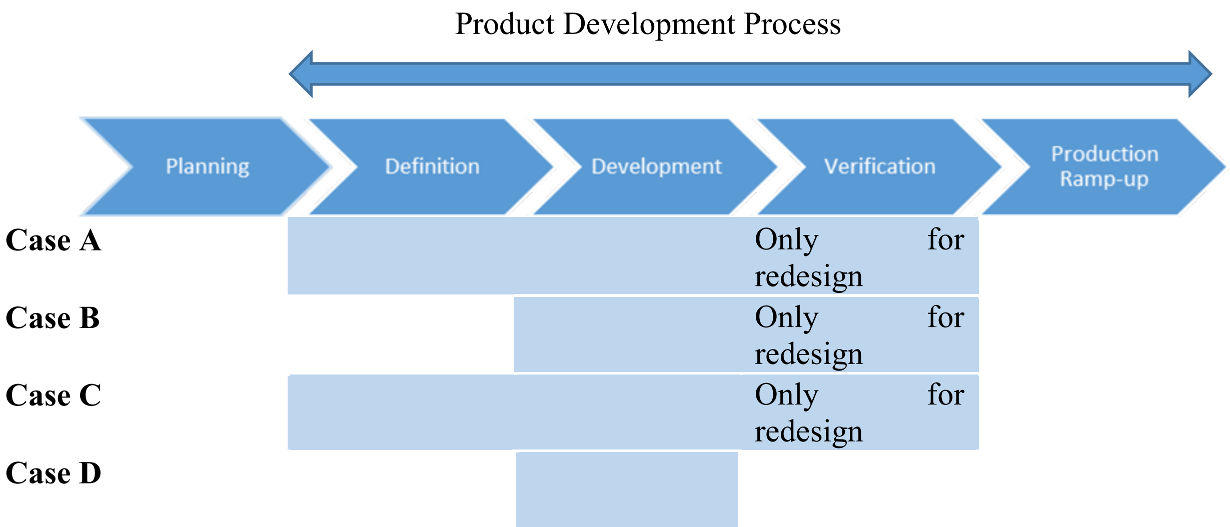


Figure 1. The use of agile process within planned Product Development Processes indicated with shaded segments. For the sake of simplicity, a generic Stage Gate Model is used, and the phases of the individual company's processes is mapped onto this.

Agile processes were combined with a stage gate model when agility was welcome. This is a somewhat obvious statement; however it can be interpreted to refer to a situation in which it is still possible to have changes in the design, and to still derive/change requirements. This in turns relates to the handling of requirements of projects and when these are considered complete. More specifically it relates to the moment when design is frozen, i.e. when no further changes are accepted for that project and the design is moving towards being fully defined for verification and production. An overview of the phases of the stage gate model, where agile processes are combined is illustrated in Figure 1. The primary phases where agile concepts were used is during development (concept and detailed design phase), and to some extent in verification, i.e. testing of products, systems and sub-systems if this led to new development tasks. A difference was observed in how early in the development process agile concepts were integrated, with two of the cases (A and C) utilised agile thinking to define the specification (this is discussed further later in the paper), and others preferring the specification to be completed before moving into agile.

Table 3. Overview of how stage gate and agile models were combined in case companies.

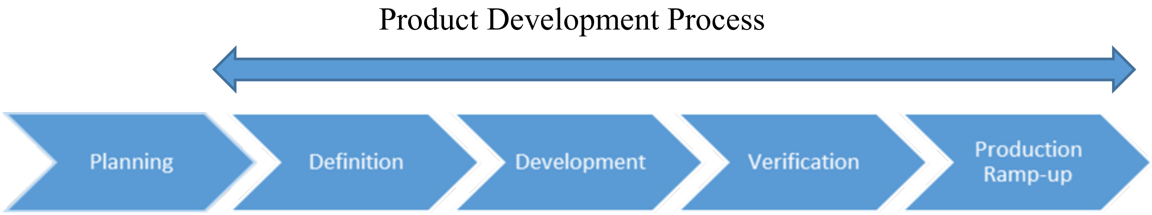
	Case A	**Case B**	**Case C**	**Case D**
Projects where processes are combined	Across all projects			Rolled into all projects in next few years.
Where in the planned processes are agile concepts combined	Definition and development, and to a limited extent in verification.	In development, and to a limited extent in verification.	Definition and development, and to a limited extent in verification.	In development.
Defining tasks: who and how	Stream owner and project team. Planned deliverables from gates used to define sprints within single disciplines. Stream owner for each team defines tasks. Team prioritises tasks together.	Product owner/product manager.	Project team. User stories are defined by product owner and business analysis. These are used to define tasks on team level.	Project manager and project team. Planned deliverables from gates used to define tasks for agile iterations. Tasks defined within single disciplines
Iterations/sprints and gates.	Iterations of agile process go across gates.			Iterations within gates
Multiple projects	Multiple projects avoided if possible. If this is not possible, guidance issued to priorities projects and tasks.	Scrum teams work on multiple projects, and scrum meetings address all these projects.	Individual members of the team can work on both scrum and stage gate models. This can cause a mismatch in the amount of time expected from a team member and the amount he/she can dedicate to the agile tasks.	Does not envisage a problem, don't expect 100% on some sprints, and culture of multiple projects.

Dealing with requirements and change of requirements.

Prior to the interviews it was hypothesized (Hypothesis 3) that : *The combining of agile and planned processes requires that requirements are kept more abstract in order to allow change when agile process are integrated as part of stage gate models.*

This was found to be true, with three different strategies observed, summarized in Figure 2. Case A defined the requirements as part of the combined process, and froze this as the end of the development phase. Case B defined the majority of requirements as part of the planned process, outside of the combined process. Within the combined process around 20% or so of the specification was defined. Case C, also evolved the specification as part of the agile process, they also adopted a strategy of postponing definition of requirements for hardware until after the majority of requirements for software were defined. Thus allowing the hardware functionality to be reduced to a minimum and embody as much functionality as possible in software. In line with this strategy, they kept requirements on a system level as long as possible, postponing decisions to embody certain requirements into hardware rather than software. The final case (D), expected the specification to be fully defined prior to combining agile thinking. One of the key differences between stage gate models, and agile processes as described earlier, is dealing with changes: change is welcome in agile process, whereas stage gate models prescribes definition of requirements to happen early in the process, with the specifications being fixed. However, numerous studies show that requirements are often changed, recognising the co-evolution of solution and requirements (see e.g. Sudin, et al., 2010; Dorst and Cross, 2001). A body of research in the field of engineering change classify the need to change a design as either for: improving the product or, error rectification (Vianello and Ahmed-Kristensen, 2012; Jarratt et al., 2011). As the costs of a development project accumulate with time in the stage gate process, the cost of change also increase, and this is seen as undesired, so the majority of changes that are seen late in the process are error rectification (whereas those related to product development are often saved for the next generation of products).

Across all the cases, an advantage of the combined agile process was stated as faster identification of the need to change a requirement in comparison to the case of using the stage gate model alone. In manufacturing companies that utilize stage gate models alone, previous findings show that the majority of changes (need to change a specification) are discovered during the manufacturing phase (Ahmed and Kanike, 2007). They analysed over 1500 change reports for the development of an aero-engine over an eight year period, including two years of service, and found that around 76% of all changes happen in the manufacturing phase. Early identification of the need to change a requirement is one of the strengths in agile process and relatively easy to implement in software domains. At the same time, late changes in product development are not desired where hardware is involved, due to amongst other reasons the cost of redesigning manufacturing tools. Hence the use of agile processes, where issues are likely to be identified faster is a significant improvement, with the main action being the identification of where extra resources are needed. An analysis of engineering change documentation, and a comparison of when changes arise in combined process models as compared to planned processes alone would be required to further substantiate this. From these findings regarding the use of combined models there is an implication of how requirements are defined, and when they are left open, in order to take full advantage of agile thinking. This challenges traditional descriptions of engineering design process, and stage gate models. Should the trend to combine agile and planned PDP continue within manufacturing firms, new models to deals with requirements would be expected.



Case A	Defined in agile	Design Frozen at	

	process	end	
Case B	80% of Specification defined	Around 20% of specification	
Case C	Defined in agile process	Hardware may be left open until software systems are defined	
Case D	Specification defined		

Figure 2 The defining of requirements during the Product Development Processes. The shaded segments show where agile processes are employed indicating whether requirements are evolved within or outside of these.

Adapting models for combined use

The cases interviewed all showed evidence of moving away from pure agile, or pure planned processes. It was evident that the companies were very much learning from their experience, and using this to adapt the processes. In addition to the way requirement are dealt with, which has already been described, adaptions were seen in three levels: 1) the roles of the team members; 2) communication or knowledge sharing within and outside of the team and; 3) the governance models. These are summarised in Table 4. Case A, has chosen to remove the role of the scrum masters as insufficient value was perceived from the role in relation to the time commitment requirement. Case C, split the product owner role into two to ensure that two perspectives were constantly being considered: the technical perspective as well as the business perspective through the role of a 'business analyst. Case C, also developed the specification within the agile perspective, and hence the splitting into similar roles ensures that the specification addresses both perspectives. All of the cases, as stated, adopted the agile processes within the structure of the stage gate model, and thereby adopted governance models from planned processes or traditional project management. One reason for this was related to knowledge sharing which is described further here.

Table 4. Adaptions made to the Agile and planned Product Development Models.

Adaption of:	**Case A**	**Case B**	**Case C**	**Case D**
Roles of teams	Scrum master, role was removed after experiencing it as time consuming and not adding value.		Product owner role split into two to include both a business analysis perspective and a product owner (more technical perspective).	
Communication (within and outside team)	Visualization tools. Defining of streams and swim lanes defining interaction between disciplines in tasks.	Tools from project management, for example: interaction points. i.e. meetings with key stakeholders.		Utilise deliverables from gates of planned models to define tasks.
Governance	Governance models from stage gate (milestones, gates) review			

The use of agile teams resulted in a high level of informal knowledge sharing taking place. The companies interviewed all pointed to an increase in knowledge sharing within the team. However knowledge sharing outside of the agile teams became more difficult. To address this issue the companies had developed different strategies, and adopted approaches from stage gate processes and traditional project management tools, for example the use of interaction points with meetings where other relevant stakeholders within the company would be invited to enable updating and knowledge sharing, initially these meetings had been open to all, but now relevant stakeholders were invited only. All of the companies interviewed still used gates with review processes in place that also served as an approach to communicate outside the team. One of the companies also utilized visualization tools that could be printed out and placed in the office to ensure a visual status of the project is constantly present. The three companies that had adopted the combined process (Cases A-C) had also adopted the approach when using distributed teams, i.e. in global product development, however they all ensured at least one physical meeting was possible. The final case D, expected to use this only with co-located teams.

Knowledge sharing and creating shared understanding.

The cases described used both multidisciplinary teams and single discipline teams. Case B, have a software team and a hardware team, the latter being multidisciplinary in nature. The hardware team was seen as more challenging in relation to combining agile and stage gate. This was explained as due to being familiar with a stage gate model mindset, and having to adopt the new agile mindset and processes. It was also explained as being due to agile thinking being more common in software development than in hardware development as well as the multidisciplinary nature of the hardware teams. In Case C, the use of so called 'user stories' were used as a boundary objects in the multidisciplinary teams to facilitate shared understanding. User stories acted as intermediates that, all disciplines could relate to and could use to define tasks and drive requirement definition. Case A, adopted the use of so called 'swim lanes' to highlight how the interfaces between the individual disciplinary teams interacted. That is, tasks related to each disciplines were mapped and interactions between disciplines were indicated as 'swim lanes'. Case A also utilized visualization tools that would be physically present to communicate knowledge outside the agile teams, together with traditional project management tools (as was the case with the other cases). Although the use of combined processes led to an increase in knowledge sharing within the team, review processes and interaction points with key stakeholders was felt necessary to: 1) enable the team to see beyond small iterations and 2) facilitate knowledge sharing beyond the agile teams.

The use of the combined models, as shown in Table 2, also requires different mindsets that need to be combined. Agile models emphasize the same level of dedication from team members, which can be a challenge when dealing project members have multiple projects. As can be seen in Table 2, each company had its own approach ranging from avoidance (clearly prioritizing projects and their tasks), to ensuring the all team members in a team dealt with the same projects in the same combined processes, to allowing projects with combined processes and projects with pure planned processes to co-exist, paying attention that all team members work on any one project at the same time. All three case that had implemented this, recognized the challenge in dealing with multiple projects, and where still in a learning phase.

DISCUSSION AND CONCLUSION

This paper has described the use of combining planned product development processes together with agile processes within manufacturing companies. Four cases, at various stages of implementation have been described, and illustrate that companies are still at a learning stage with understanding the implication of this. Contrary to expectations, the complexity of the product, or level of uncertainty or innovativeness was not used within the companies to distinguish when to apply this approach, instead all cases adopted (or planned) to adopt this across all projects- of course the cases cover a narrow spectrum and no attempt has been made to distinguish them based on innovation level. Also contrary to literature, the use of the combined models was found to be integrated, and treating the project as a whole, with iterations going across the traditional gates of planned product development models, rather than the agile processes being sub-processes within the phases of the stage gate model.

From these findings and the use of combined models there is an implication of how requirements are defined, and when they are left open in order to take full advantage of agile thinking. This challenges traditional descriptions of engineering design processes and stage gate models where requirements are expected to be defined early in the process. Should the trend to combine agile and planned PDP continue within manufacturing firms, new models to deals with requirements are needed to support these processes.

From a managerial perspective, the paper draws experiences and lessons learnt across a number of cases, thereby increasing any individual company's knowledge. From the cases, a number of challenges were observed when combining the models, including the disparity of knowledge sharing within agile teams, and other stakeholders, understanding how to remain agile and welcome changes in design requirements whilst adhering to strict regulations. These challenges resulted in the companies adopting and combining elements of planned product development models and agile thinking to address these challenges.

The individual cases highlight that the adaptions made by the companies were both common (e.g. use of governance model) and individual strategies. The latter may be due to the individual context of the company, for example the role that regulations play, which may dictate how requirements are handled, and also indicates that the learning from the companies is not yet complete. Hence, there is a need for further research investigating additional cases and characterizing their context.

This paper informs on the situation when it is of benefit to use agile models together with the linear stage gate model and highlights the issues of integration that can appear and the adjustments required to address these issues.

Our cases indicate that models are being combined in different ways, with iterations going across gates. This leads to challenges, e.g., effective knowledge sharing, dealing with uncertainty and change, for example in defining requirements of products.

REFERENCES

Alexander, C. (1964). Notes on the Synthesis of Form (Vol. 5). Harvard University Press.

Ahmed, S. and Kanike Y. (2007). Engineering change during a product's lifecycle. International Conference on Engineering Design, Paris, France.

Andreasen, M. M., & Olesen, J. (1990). The concept of dispositions. Journal of Engineering Design, 1(1), 17-36.

Archer, B. (1965). Systematic method for designers. London: Council of Industrial Design.

Bessant, J., & Tidd, J. (2007). Innovation and entrepreneurship. John Wiley & Sons.

Boehm, B. (2002). Get ready for agile methods, with care. Computer, 35(1), 64-69.

Boehm, B., & Turner, R. (2003). Balancing agility and discipline: A guide for the perplexed. Addison-Wesley Professional.

Cooper, R. G. (1990). Stage-gate systems: a new tool for managing new products. Business horizons, *33*(3), 44-54.
Cooper, R. G. (2008). Perspective: The Stage-Gate® Idea-to-Launch Process—Update, What's New, and NexGen Systems. Journal of Product Innovation Management, 25(3), 213-232.
Dorst, K., & Cross, N. (2001). Creativity in the design process: co-evolution of problem–solution. Design studies, 22(5), 425-437.
Fowler, M., & Highsmith, J. (2001). The agile manifesto. Software Development, 9(8), 28-35.
Goldman, S.L., Nagel, R.N. and Preiss, K. (1995). Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer. Van Nostrand Reinhold: New York, NY, USA.
Gregory, S. A. (Ed.). (1966). The design method. Butterworths.
Jarratt, T. A. W., Eckert, C.M., Caldwell, N. H. M. and Clarkson, P. J. (2011). Engineering change: an overview and perspective on the literature. Research in engineering design 22 (2), 103-124.
Jones, J. C. (1992). Design methods. John Wiley & Sons.
Karlström, D., & Runeson, P. (2006). Integrating agile software development into stage-gate managed product development. Empirical Software Engineering, 11(2), 203-225.
Katayama, H. and Bennett, D. (1999). Agility, adaptability and leanness: a comparison of concepts and a study of practice. International Journal of Production Economics, 60–61, 43–51.
Kettunen, P. (2009). Adopting key lessons from agile manufacturing to agile software product development—A comparative study. Technovation, 29(6), 408-422.
Kidd, P.T. (1997). Agile enterprise strategy: a next generation manufacturing concepts. In: Proceedings of the IEE Colloquium on Agile Manufacturing, pp. 2/1–2/6.
Preiss, K. (2005). Agility—the origins, the vision and the reality. In: Proceedings of the International Conference on Agility (ICAM), pp. 13–21.
Sanchez, L.M. and Nagi,R. (2001. A review of agile manufacturing systems. International Journal of Production Research, 39 (16), 3561–3600.
Sudin, M. N. B., Ahmed-Kristensen, S. and Andreasen, M. M. (2010). The Role of a Specification During the Design Process. 11th International Design Conference, Dubrovnik, Croatia.
Takeuchi, H., & Nonaka, I. (1986). The new new product development game. Harvard business review, 64(1), 137-146.
Ullman, D. G. (2004). The mechanical design process. 1997. McGraw-Hill, New York, NY, USA.
Eppinger, S. D., & Ulrich, K. T. (1995). Product design and development. 1995.
Darlington, MJ and Cully SJ A model of factors influencing the design requirement. Design Studies, 25(4), 329-350.
Vazquez-Bustelo, D. and Avella, L. (2006). Agile manufacturing: industrial case studies in Spain. Technovation, 26, 1147–1161.
Vianello G and Ahmen-Kristensen, S. (2012) A comparative study of changes across the lifecycle of complex products in a variant and a customised industry. Journal of Engineering Design, 21 (2).
Yusuf, Y.Y. and Adeleye, E.O. (2002). A comparative study of lean and agile manufacturing with a related survey of current practices in the UK. International Journal of Production Research, 40 (17), 4545–4562.
Zhang, Z. and Sharifi, H. (2007). Towards theory building in agile manufacturing strategy — a taxonomical approach. IEEE Transactions on Engineering Management, 54 (2), 351–370.