R&D MANAGEMENT

Agile Stage-Gate Management (ASGM) for physical products

John J. Salvato[1,*] and Andre O. Laplume[2]

[1]Mechanical Engineering - Engineering Mechanics, Michigan Technological University, 1400 Townsend Drive, Houghton, MI 49931, USA. jjsalvat@mtu.edu
[2]School of Management, Entrepreneurship & Strategy, Ryerson University, 55 Dundas Street West, Toronto, M5G 2C3, Canada. alaplume@ryerson.ca
*Permanent Address: 3946 Fieldstone CircleGalesburg, MI 49053, USA

We present a qualitative study of Agile Stage-Gate Management (ASGM),: a hybrid new product development methodology that combines Agile and Stage-Gate Management (SGM) approaches for the coordination of new product development. When applied to software projects, Agile is expected to deliver reduced development times, improved resource utilization, and greater financial success. We examine whether ASGM practitioners realize similar outcomes in a sample of global firms developing complex electro-mechanical products (e.g., automobile components, railway propulsion systems, and medical devices). Our grounded theory approach articulates an understanding of ASGM through extensive interviews of experienced professionals. Our thematic analysis supports many expected benefits (i.e., speed to market, innovation enabling), but also does not encourage others, and reveals new pitfalls that deserve recognition (i.e., resource inefficiency). ASGM is not a panacea for all product development. Overall, physical product firms adopting this method can expect reduced development times and higher levels of innovation but will expend more resources to complete development projects, but a dichotomy exists. Physical product developers using ASGM experience a negative impact on project resource efficiency due to the need for dedicated resources, frequent product demonstrations, and duplicative management structures.

1. Introduction

[ASGM] has the potential to be the most significant change to our thinking about how new-product development should be done since the introduction of today's popular gating systems thirty years ago! (Cooper and Sommer, 2016a, p. 167)

We are witnessing an explosion of interest in Agile methods for New Product Development (NPD) resulting from a widespread belief that Agile generates significant benefits for firms. Agile is becoming ubiquitous in the software world, where it started, and is now migrating to physical product firms, where it is integrated into an existing Stage-Gate Management (SGM) structure for managing NPD. This hybrid approach is known as Agile Stage-Gate Management (ASGM) (Karlstrom and Runeson, 2005; Sommer et al., 2015; Cooper and Sommer, 2016b). ASGM shows great initial promise for physical product firms (Cooper and Sommer, 2016b); (Rigby et al., 2015), but it maybe idiosyncratic (Dikert et al., 2016; Bianchi et al., 2018).

We know from prior research on management fads and fashions that managerial technologies are adopted in contexts where they may not be appropriate (Abrahamson, 1991; Abrahamson, 1996; Rogers, 2010). Examples include job enrichment programs (Hackman, 1975), T-groups and matrix

structures (Byrne, 1986), quality circles (Lawler and Mohrman, 1985), knowledge management practices (Lev, 2000), employee–management techniques (Abrahamson and Eisenman, 2008), as well as talent management (Iles et al., 2010).

Physical product firms that implement ASGM have no standard metrics to assess its value concretely. Yet, early insights suggest that ASGM delivers improved development efficiency, decreased work effort per project, improved team communication, and is more responsive to changing customer needs (Sommer et al., 2015; Cooper and Sommer, 2016a, 2016b). Given the substantive differences between companies developing physical products and the typical software firm, we might expect some of the benefits of Agile not to materialize or new advantages to emerge. Quoting Cooper and Sommer (2018b, p. 513, 514):

…research into very recent industry experience suggests that this hybrid model, which incorporates Agile development methods, has significant potential benefits for manufacturers of physical products… Sadly…there has been little or no academic research in this new area of hybrid [ASGM] methods for physical new products.

There is a clear gap in our knowledge about the implementation of ASGM for physical product firms and their benefits. To answer the call from these authors and to expound current research, we seek clarification for the following research question. To what extent does the set of Agile principles, as applied to software, deliver similar results for physical products?

To answer this question, we use a content analysis method (Krippendorf, 2004), along with a grounded theory approach (Glaser, 1999; Graebner and Eisenhardt, 2004; Eisenhardt and Graebner, 2007), which comprises theme identification, linking, and elaboration. We asked 29 product development professionals from five physical product businesses of four multinational firms about their ASGM practices. We transcribed, coded, and analyzed the content generated from our respondents when asked about their ASGM implementations and the outcomes they experienced, as well as why they thought they experienced specific results. Our thematic analysis reveals that these firms experience many of the expected benefits, including development speed and greater alignment with intended customers. Conversely, they experienced higher resource costs, which were not expected. Our results regarding innovation and speed advantages reinforce elements of the extant literature and should be encouraging for ASGM proponents; with this in mind, we believe our study goes further by elaborating resource inefficiencies, including staffing strain due to the need for dedicated resources, frequent product demonstrations, and duplicative management structures.

2. Background

Much of the extant literature on Stage-Gate Management (SGM) focuses on physical products, where SGM is the quintessential process framework used for managing NPD (Christiansen and Varnes, 2009; Jespersen, 2012). SGM is a risk reduction scheme through the application of a gate review structure (Cooper and Kleinschmidt, 1993; Mohan et al., 2007), where NPD activities are organized into phases, a Go/No-Go review is conducted to assess the work completed at the end of the stage along with a decision to move onward (Cooper and Kleinschmidt, 1991; Kalluri and Kodali, 2014). SGM teams toil to define as much of the product design as early as possible (Munthe et al., 2014). The earlier product specifications are confirmed, the more risk that is 'retired', the sooner follow-on phases may commence (Iansiti, 1995; Biazzo, 2009). Developing new products supports business growth if appropriately executed, how firms manage NPD is essential (Cooper and Kleinschmidt, 1991). Common SGM criticisms levied against the method are aimed at an inability to accommodate all project types, insufficient fluidity for late specification freeze which drives organizations toward incremental projects, and a negative impact on speed due to method rigidity (Cohen et al., 1998; Hutchins and Muller, 2012; Bers et al., 2014; Sommer et al., 2015; Bianchi et al., 2018).

As an antidote for the challenges of SGM, Agile is a well-developed system designed to overcome innovation barriers to deliver improved success rates for software products (Juricek, 2014; Rigby et al., 2015). Characteristics of Agile include linear planning with inputs and outputs, established time-bounded Sprints[1] supported with explicit knowledge, and defined closure activities (Schwaber, 2004). Agile, as implemented, has roots from other methodologies, such as Extreme Programming (XP), Scrum, Adaptive Software Development, Crystal, Feature-Driven Development, and Pragmatic Programming (Alliance, 2001). Scrum was first described as an 'all-at-once' product development approach, scalable, and team-based, with an emphasis on overlapping phases, modeled after the Rugby scrum (Takeuchi and Nonaka, 1986). Innovation involves variability and uncertainty, Scrum, embraces helpful variation by iterative development, adaptation/transparency, and simultaneously reduces risk (Rubin, 2013). For our aspirations, we link Agile and Scrum together

since Scrum is the most utilized Agile method for hardware development (Cooper and Sommer, 2016b). From the Agile Manifesto, the method harbors a deep desire for lightweight processes with four main elements (Alliance, 2001):

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

Agile teams commence projects with limited product definition and tend to be comfortable learning throughout development (Pich et al., 2002; Munthe et al., 2014). Agile seems to be favored for projects that are smaller in size with dynamic and unpredictable requirements, whereas SGM fits larger projects with stable conditions (Boehm and Turner, 2004). Large scale Agile transformations within the software world are not without challenges. These transformations at larger software development organizations moving from SGM (i.e., waterfall method) to Agile must be mindful of bureaucracy concentrations, structural duplication, well-defined Agile roles, and goals; in other words, equalizing flexibility with rigidity (Dikert et al., 2016). Agile has been organized into nine elements, three each, roles, artifacts, and tools (Cooper and Sommer, 2016a):

- Roles: product owner, scrum master, the development team
- Artifacts: time-boxed sprints, daily scrums, retrospective reviews
- Tools: product/sprint backlogs, scrum board, burndown charts

Early ASGM desires were based upon adaptability, flexibility, agility, and speed, enabled by 'spiral' development, simply, a repeating set of build, test, feedback, and revise loops (Cooper, 2014). The differences can be described as:

SGM has been described as a macro-level model designed to help project selection, identification of stages and best practices, along with roles and responsibilities; with Agile thought of as a project management method focused on agility, adaptability, and speed to development through the usage of micro-planning tools. (Cooper and Sommer, 2016b)

In other words, ASGM attempts to balance Agile and SGM, by establishing a healthy tension between fixed planning and iterative problem solving, between process control and productive disorder (Sommer et al., 2015). Agile, as a method, seeks rapid value and responsiveness to change, preferring to construct prototypes to determine which features will add value, whereas SGM, has goals of predictability, stability, and assurance (Boehm and Turner, 2004).

With limited research, ASGM scholars have suggested that the technique delivers several benefits (Rubin, 2013; Sommer et al., 2015; Cooper and Sommer, 2016a, 2016b), including improved efficiency and productivity, reduced development effort with less rework, rapid response to changing needs, enhanced team communication, greater customer alignment, faster releases, improved coping with uncertainty, and fewer resource issues. Early positive indications aside, a broader dataset does not yet exist. The small number of businesses studied are either country- or industry-specific and may not be generalizable. Clearly, there is an opportunity to build on this emerging knowledge.

3. Methodology

Since our goal is the development of theory, not to quantitatively test theory, we follow a grounded theory approach, where researchers inductively develop theory based upon the systematic collection and analysis of data (Glaser, 1999; Laplume and Dass, 2014). We conduct comparisons without a hypothesis since our goal was not to test variables but to search for similarities, differences, successes, failures, and patterns (Yin, 1999, Eisenhardt and Graebner, 2007). Theoretical insights are extracted from the respondent descriptions of their NPD implementations of ASGM. We use theoretical sampling because the selected cases fell within the study intent and were suitable to highlight the practices employed. As we were looking deep into the real-world application of ASGM, we chose the case study method because it allows for several respondents to provide their perspectives on the same phenomenon and is flexible in dealing with uncertainty between the focal phenomenon and associated context (Yin, 1999). We leveraged three levels of coding to build a robust thematic set: Open – identify concepts and properties; Axial – analyze iteratively and organize based on wording; Selective – integration and refinement of theory (Strauss and Corbin, 1998).

Interviews with knowledgeable respondents are an efficient way to gather rich data (Homburg et al., 2014). We focused on manufacturing multinationals as they contrast the most common implementation context for Agile (e.g., small software companies). We studied five distinct Business Units (BU) from four global firms with development sites across the globe. We selected companies that showed signs of ASGM and whose employees told us so, developing complex electro-mechanical products such

as automobile components, railway technologies, and medical devices. The cases are industry leaders within the Automotive Components, Railway Technology, Process Monitoring, Perimeter Access, and Medical Device markets. We leverage multiple cases to build a more cogent argument for theory creation (Yin, 1999). We want to ensure the development of grounded, accurate, and generalizable theory, which enabled comparisons to determine whether an emergent finding was singular or familiar (Goggin, 1986). We marked cases not based upon attribute sampling, but for balance, and the opportunity to learn (Stake, 2005). We treated each BU as an individual case; our firms had thousands of employees who were geographically dispersed and competed in differing markets. One company-provided two examples ('Railway' and 'Monitoring'), that being said, based on corporation size (i.e., revenue, employees), BU geographic distance, bereft NPD corporate governance, dramatically different products, and no intercommunication between development teams, we treated these cases as unique, see Table 1.

We required study respondents to be current NPD professionals with greater than five years of experience leading, managing, or supporting teams that utilized ASGM. Our respondents had different skill sets, backgrounds, and industry experiences. Participant education was mostly technical (e.g., mechanical, electrical, or software engineering), although several respondents went on to obtain advanced degrees (e.g., MBA) or professional certificates (e.g., project management). We provide descriptive information about our sample, including geographies, roles, and industries in Table 2. With 29 interviews, we believe the number of cases is appropriate for theory development (Eisenhardt and Graebner, 2007). We sought to diversify responses by leveraging respondents from varying functions (e.g., engineering, marketing, manufacturing), as well as, different organizational levels (e.g., project team, project management, leadership) to minimize retrospective alignment (Eisenhardt and Graebner, 2007).

We collected field data through live discussions using video calls, lasting 30–75 min. We transcribed each interview, then organized to ensure traceability. We conducted the interviews using a semi-structured approach with open-ended starter questions, which were pointed enough to probe (Roulston et al., 2003; Laplume and Dass, 2014). We asked about ASGM implementation, the outcomes experienced, and their causes. Transcribed interviews are categorized into themes using an iterative process of identifying subthemes relating to performance outcomes and then searching for repeating textual patterns supporting cross-unit presence and generalizability. This 'cycling' through case data leads to objective (i.e., replicable) theory induction based on data (Eisenhardt and Graebner, 2007). Transcripts are mined for relevant quotes, which are presented in tables in the results section (Tables 3–5). We made sure that respondent quotations were fastidiously edited to ensure each quote represented the intent and context (Sandelowski, 1994).

3.1. Case validation

Before we can start to answer our research questions, we validate that the BUs were, indeed, using ASGM. This was a crucial step to filter out firms that are transitioning to, or just testing out ASGM, or not using ASGM at all. Guided by ASGM knowledge (Sommer et al., 2015; Cooper and Sommer, 2016a, 2016b) and inspired by the Agile Manifesto (Alliance, 2001), we organize Agile into eight tenets, then evaluated each case against these tenets to ensure the BUs implemented ASGM. The principles are: (1) Team Interface, (2) Product Demonstration, (3) Customer Involvement, (4) Specification Flexibility, (5) Team Structure, (6) Time-Bound, (7) Feature Prioritization, and (8) Communication (see Appendix A).

We review the transcripts line by line and search for the Agile keywords to use as evidence of the tenet being implemented in each BU. If a respondent quote, including context, is found within the transcript that supported the Agile doctrine, including keywords, we considered it as evidence of Agile. Transitioning from individual respondents to aggregate responses for each case, we made use of a simple majority of respondents. We found three of eight Agile techniques in all cases, with five techniques observed in four of five cases. Overall, we found a high occurrence of Agile tenets across all respondents, from a low of 69.0% to a high of 96.9%, establishing confidence that our firms were practicing ASGM and not still in the process of transitioning. Despite reliable indicators of ASGM implementation, we did find friction between the project teams and BUs when interfacing with senior leaders who did not understand or appreciate the Agile doctrine. We did not evaluate ASGM implementation at the executive organizational levels, which was most often leadership personnel responsible for approving the projects but resided externally to each case. Notwithstanding, this may be a limitation of our study.

4. Performance study results

To answer our research question, we identified the themes that emerged from our thematic analysis

Table 1. Case characteristics

Case name	Auto	Railway	Monitoring	Perimeter	Medical
Case number	1	2	3	4	5
Company number	1	2	2	3	4
Schedule flexibility	High	Medium	Low	Low	Low
NPD portion	Up front	Complete	Complete	Complete	Complete
Path to market	Intermediary	Platform	Direct	Direct	Platform
Market or business turmoil	High	Medium	Low	Medium	High
Respondent experience (avg years)	29.7	18.6	18.3	24.5	13.7
Agile exposure	Low	High	Medium	High	Low
Team dispersion	Low	Low	Low	Medium	High
Agile tool	No	Yes	Yes	Yes	Yes
Firm Revenue	~ $20B	~$36B	~$36B	~$3B	~$15B
Firm employees (#)	~100,000	~140,000	~140,000	~10,000	~40,000
BU project team size	10–100	25–75	10–30	20–50	25–100
BU business areas	Automotive structural components for global OEM's	Railway propulsion and electrical control systems	Process quality equipment for high-tech manufacturing	Residential and commercial security control products	Automated surgical equipment and implants for humans

Table 2. Interview characteristics

Characteristic		Case number	Company number	Number of interviews
Industry	Home and Office Products	4	3	11
	Transportation and Logistics	2	2	5
	Hardware	3	2	4
	Health Care Equipment	5	4	6
	Automotive	1	1	3
	Sub-Total			29
Role	Leadership			2
	Individual Contributor			12
	Resource Manager			6
	Program Management			9
	Sub-Total			29
Geography	NA – United States			14
	NA – Canada			4
	EU – Germany			6
	EU-Switzerland			5
	Sub-Total			29
Discipline	Technical			21
	Business			8
	Sub-Total			29

Table 3. Evidence for speed to market (STM)

Second-order themes	Representative respondent quotations
Lightweight process	…we are basically taking an accelerator model, and modified it to a concept model…we try to make that be very light in terms of what the requirements are… – **Leadership, Auto**
	…focus on doing minimal amount of paperwork…make sure we are not bogged down by the process. The process itself is not the end game…What allows us to be faster…minimizing the process – **Program Manager, Auto**
Team communication	…I had clear visibility…[engineers] try to fix things in the background and then you get hit with it months later…[but] able to get things done quicker by having an open communication and being collocated and having an open dialogue. – **Individual Contributor, Perimeter**
	…it is useful, and it saves time ultimately. Though it is 15 min out of your day, it saves time because we can meet with those people on a daily cadence, getting roadblocks out of the way…It's a lot more efficient than an email…- **Individual Contributor, Medical**
Process flexibility	…if it is a completely new product…and you're at the cutting edge of technology…depending upon what product and how much hardware/software…it can be more rapid…in case of…hardware…I would leave it flexible…- **Resource Manager, Railway**
	If you don't accept change…during development, you will have a huge problem…I think a market changes even during development…we have to have the ability to react on change. – **Project Manager, Medical**
Project control	Everybody…at this meeting…has something to say…So everyone has a word to say about this, but at the end…there's the gate with all the direction…final decision is made by those people at the end at the gate… – **Individual Contributor, Monitoring**
	…sprints would be the execution towards the milestones in the daily management…we weren't using the backlog to drive dates…we had dates and we're using the tool to figure out what work had to be done…then track towards it – **Resource Manager, Perimeter**
Project communication	The product owner…puts down the backlog…has a very good overview on what we need to achieve…what features we need to achieve…puts this down into a textual description…from an upper level, [the VP] looks down into what needs to be done, and he has the big picture- **Individual Contributor, Medical**
	…we do monthly NPD review[s]…teams have to come in…give a brief update, showing…that everything is on track – or if they're not on track, then they have to give them more detail review…they have their formalized toll gates with their deliverables list. They complete all their deliverables, and show they're completed to move to the next phase- **Leadership, Monitoring**

Table 4. Evidence for innovation enabling

Second-order themes	Representative respondent quotations
Entrepreneurial mindsets	[we are] being cutting edge, we have to push the boundary, and this team is not apprehensive about trying the new and unusual. I find that invigorating**- Individual Contributor, Medical**
	It's having a lot of input where no one even worries about what the answers are because something new will come out of that conversation. What allows us to be faster upfront…but I think it's also the mindset of the person. Do they have a business mindset, are they thinking it like a consumer? – **Program Management, Auto**
Relationships	It's critical and paramount, if we're actually working with mutual benefits in the mind. It's really got to be both of you working in partnerships that delivers some value. – **Leadership, Auto**
	…an engagement with a startup company…project that is enabled by that relationship…we're doing investment committee portfolio reviews about the overall opportunity…We're connecting the two so that we're not doing investment from things where we don't have a …relationship. – **Program Management, Auto**
Managing requirements	…having good fundamental core requirements nailed down is probably the most important thing that we can do. And we often cut it short just to get moving, seeing how we're going to figure it out when we go, or we think that it's not going to change, and it sure as heck does. – **Resource Manager, Perimeter**
	…you've got to manage the scope and the scope creep…Early on, there was a lot of scope creep. We finally…shut the door on part of it…goes back to…rigorous definition…discovery processes…making sure that you understand what you're getting into…get as much of that alignment early on…that's what Agile helps you do… – **Program Management, Perimeter**
Customer value	For now, one thing would be to be able to have in mind that we can propose something to the customer that is a subset of what they want in the ideal world, I would say…the MVP…are always using MVP, but the MVP for them is not the same as for us.- **Resource Manager, Monitoring**
	If we see that we are running in the right direction and confirm in such a manner…you have to show something to them…they tell you what they like and don't like and then…it's like the options and directions into the product. That's very useful**- Program Manager, Railway**

Table 5. Evidence for increased resource utilization

Second-order themes	Representative respondent quotations
Staffing strain	In terms of resource usage…my experience is that the overhead is increasing, because you need people to be trained, you need…Scrum masters, you need a lot of people working on preparing the backlog items and prioritizing…– **Individual Contributor, Perimeter**
	…[in] our Scrums, everybody was accountable…everybody has to stand up in front of the [class]…they would very quickly be self-managed…because they were the only one that was there trying to speak and had nothing to show for it…accountability was huge…– **Leadership, Perimeter**
	Maybe one of the downsides is…what we've seen in general is that…the whole development team is dedicated to the work…you know what the cost is going to be because, essentially, the team is working…R&D projects generally get more expensive. – **Individual Contributor, Railway**
	…the cost is that you have a room full of people for 20 min. But 25 people, that's a lot. So, if we ever try to balance the overhead component, because you could really burn a lot of time if you're not careful. – **Program Manager, Perimeter**
Demonstration woes	…verifying…'Are we on the right track?' We would do the Sprint demos…at the end of every Sprint… 'We want to physically demonstrate to you where we're at, and does everyone feel like we're on the right track?'…there was a lot of collaboration there…– **Leadership, Perimeter**
	…[another] way that is very useful and that is very sometimes complex to do on a regular basis, is to put prototypes in the end of the customer…The actual customer. Then have the engineering team…whether it's the service guys…[or] the engineers themselves, in front of the customer and discuss the product. – **Program Manager, Monitoring**
Structural duplication	…we live it through… a SGM model with…very specific semantics of what a project has to deliver at a certain gate and integrate it with the SGM model. We have a marriage between pure SGM business decision model in addition with a product development model…[a] list of what documents you have to deliver… **Resource Manager, Railway**
	This is what they are used to. [they] think in projects…and that is what they are comfortable with. We somehow have to map our approach to the stuff that they know…trying to move them into an Agile way of thinking… That is something that we see that is a problem and that we actually work with and work with again at this point in time. – **Program Manager, Railway**

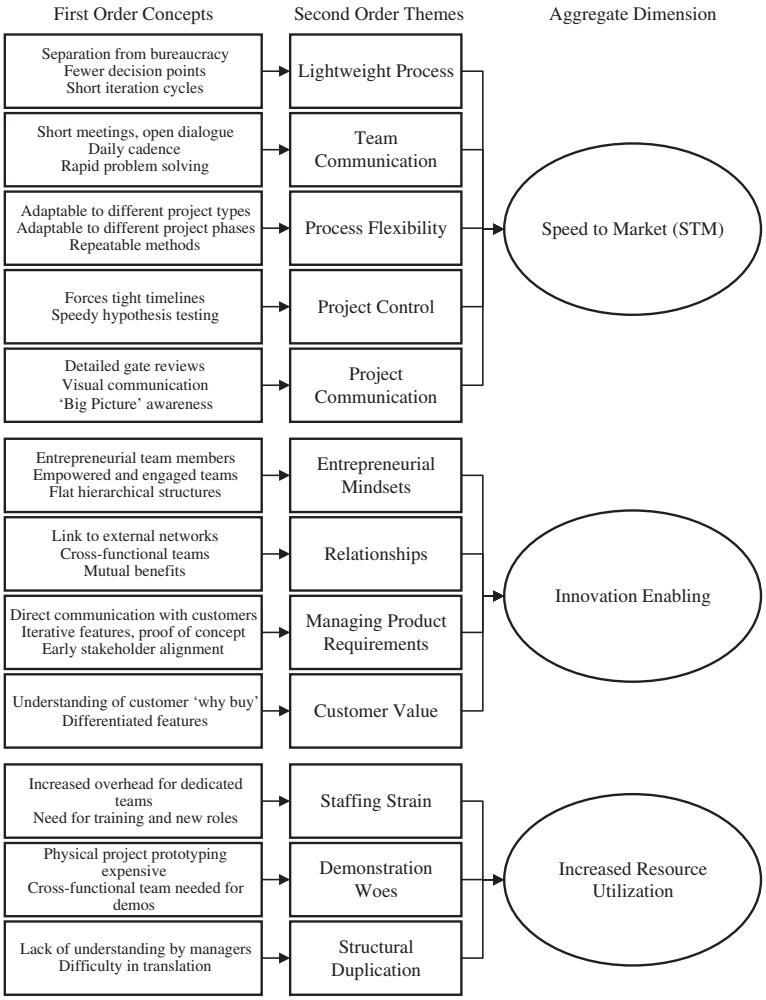

Figure 1. Overview of inductive theme development.

of the interview transcripts. We identified several first-order concepts supported by direct quotations. We linked the first-order concepts to second-order themes that explain a mechanism contributing to ASGM performance, then organized those themes under aggregate dimensions (see Figure 1).

4.1. Theme 1: speed to market (STM)

STM was regularly reported by our respondents, and their descriptions can be categorized into the subthemes of lightweight processes, team communication, project flexibility, and project control. See Table 3 for supporting participant quotations.

4.1.1. Lightweight process

The respondents expressed that ASGM delivered consistent results without being burdensome; moreover, excessive-decision points were not instituted. This indicates a reduction in process, allowing the teams to spend their time on product designs, and not process steps. We understood that a simplified, lean approach allowed project teams the ability to traverse the pathway of development vigorously.

4.1.2. Team communication

Our project teams enjoyed the increased visibility enabled by 'Dailys', which were quick (e.g., 15–30 min) frequent, cross-functional meetings. Respondents highlighted improved transparency, which allowed critical issues to be raised, leading to fewer surprises and enhanced alignment, coupled with a reduction in development time. This connection also led to happy, engaged, well-communicating teams, which delivered significant accomplishments. These strong personal relationships, trust between peers, positive team rapport, closeness among members, were all indicators of elevated engagement.

4.1.3. Process flexibility

The ability of ASGM to manage development activities, as well as the adeptness for modifications to

accommodate project styles, such as technology exploration, research, platforms, and feature additions, was crucial. We discovered that the process must support change over time to suit new needs or challenges. For 'Greenfield Projects', where requirements were vague, our cases regarded SGM as a poor fit due to a desire to define requirements early, reducing risk thoroughly. For projects with a higher degree of uncertainty, the flexibility of ASGM fit well.

4.1.4. Project control

We found teams used several supporting techniques to manage projects, including the Systems Engineering V-Model and APQP (Advanced Product Quality Planning) tools. The ASGM framework guided and controlled activities to ensure a repeatable and reliable outcome, most notably thorough gate reviews. Clear milestones and detailed review checklists were essential for communicating expectations, defining Sprint work packages, and aligning project timing, ASGM, indeed, suggests a balance between the rigidity of SGM and the flexibility of Agile.

4.1.5. Project communication

Communication between the project team and stakeholders, including gate reviews, project status reports, burndowns, financial reviews, and generic status updates, was instituted to assess progress, along with project health. This was not merely data and reports, but notably, the methodology or language used to communicate with leadership. Agile encourages frequent communication, the daily ritual of quick discussions to share information had pulled teams closer, but also assisted individual members see a holistic understanding of the entire project, not merely their area, the 'big picture' was unmistakable.

4.2. Theme 2: innovation enabling

Innovation Enabling was the second most prevalent theme, to achieve higher levels of innovation, our cases demonstrated an entrepreneurial mindset, built external relationships, gathered requirements early, and understood customer value. We summarize the quotation evidence for the second-order theme of Innovation Enabling in Table 4. Much like STM, Innovation Enabling was expected and widely observed. Since innovation is a popular topic of instruction and theorizing, it is informative to know that an NPD process can deliver positive results.

4.2.1. Entrepreneurial mindsets

We understood that the old ways of business had to be discarded, the 'same old' thinking, inclusive of tools, processes, and hiring practices had to be revisited. Teams were encouraged to question orthodoxy and were intrinsically connected to customer needs and desires. Our cases acted like small businesses, during a time of Facebook, Apple, and Google, old-line, staid manufactures had to change. Thinking creatively and unboundedly, like a struggling startup, seemed to foster an environment for bolder solutions.

4.2.2. Relationships

Our cases looked to extend relationships beyond internal resources for development; teams were scouting technologies globally, opening new avenues outside of traditional supply chains. Market disruptors were pursuing products such as self-driving cars; in some cases, they were not conventional automobile manufacturers, but technology companies focused on disruption, not plants, tooling, and production. Teams looked outside of their organizations for ground-breaking technologies, realizing that not every element of a product had to be designed internally.

4.2.3. Managing product requirements

Detailed and testable product requirements were crucial. Physical products often have hundreds of requirements; engineering teams live and die by nuanced definitions of product specifications. Our cases pursued precise product requirements, even early in development, allocating time, and resources to accomplish this endeavor. We found our cases recursively cycled through potential specifications, searching for a prioritized Minimal Viable Product (MVP) definition. A top feature could plummet to the bottom based on feedback, managing this information supported teams with their innovation quest. 'Scope Creep', the rigorous management of project intent, was also crucial. Teams, in the past, allowed marketing, or the broader businesses, to add content (e.g., features, markets) throughout development, effectively 'moving the goalposts', with ASGM prioritization, this was more difficult.

4.2.4. Customer value

A desire of the business to align with the requirements their customers wanted, needed, and, most importantly, amenable to pay for, was vital. These features were often the differentiators one firm had over another; in other words, the 'Why Buys?' that would be articulated in future sales brochures. A tremendous amount of work went into finding these features. This usually was not a long list to fulfill a tender or contract but was a shortlist of what made one product remarkable. If the end customer was enamored with a unique feature, but would not pay for it, the feature was arguably worthless. The real

winners were products or features that were unique, wanted, with compelling stories, and ones that brought considerable value to customers.

4.3. Theme 3: increased resource utilization

Theme 3 emerged because of concerns regarding strained staffing, elaborate demonstrations, and organizational structure duplication. We reviewed Agile claims from the extant literature and concluded that we should expect a resource utilization improvement; yet, we found the opposite. Our cases harangued that resource costs increased. Unlike software products, which experience resource reductions, units of manufacturing firms had to invest generously to make ASGM work. For example, without sufficient investment in staff hours, structure change, translation capabilities, and training, the other advantages of Agile are not likely to materialize, see Table 5 for respondent quotations.

None of the cases experienced a positive resource impact from ASGM implementation; our respondents accentuated that Agile worsened resource loading, driving up staffing costs. Our cases were focused on team alignment, communication, product features, and schedule attainment, as opposed to achieving resource reductions. Three subthemes potentially explain why our BUs were not more efficient during NPD as expected from Agile-managed software products.

4.3.1. Staffing strain

'Dailys' were intended to communicate priorities and concerns in a quick and organized method. Each person spoke in front of their peers, which created an element of accountability. We found that teams were located together and dedicated, at least at the 'core' team level (e.g., design, project management, test). Our cases minimized distractions from other projects to be fast and innovative. NPD projects often have an uneven activity level over the duration, where a dedicated team approach may negatively impact costs. With SGM, resources could support multiple projects, usually the main project with secondary ones. We understood that this 'off-peak' time was financially allocated elsewhere; in contrast, with a dedicated team, the costs remain with the main project. To counteract this dichotomy, Agile mandates flexibility, meaning team members operate across functional boundaries to deal with critical and immediate activities. If this 'cross-over' fails to materialize, then the 'off-peak' hours remain aligned to the main project creating an ASGM penalty. Constant sprinting was also seen as negative by some respondents. The unrelenting focus on project tasks induced fatigue, where the 'off-peak' time was not only a financial concern but a human one. As ASGM introduced new roles, such as Agile Coach and Scrum Master, we found these tended to be incremental to traditional staffing.

4.3.2. Demonstration woes

According to the Agile manifesto, progress is more important than comprehensive documentation. Our cases spent significantly building prototypes such as full products or systems for review with customers to secure alignment on critical details. This meshed well with the Agile desire of early and regular customer interactions where learning was encouraged to ensure the team was 'heading down the right track'. Meaningful prototypes for software products can be quick to produce (e.g., hours) using minimal resources, whereas demonstrations for physical products can require cross-functional personnel weeks to create. We believe that physical firms, adopting the frequent demonstration mindset, may have placed too much into elaborate prototypes, potentially driving up development costs.

4.3.3. Structural duplication

Our respondents, particularly managers and leadership, expected formal, detailed checkpoints to confirm project control, this included milestones, technical reviews, and included a comprehensive project schedule. Respondents felt a holistic understanding of the project, with critical dates defined, was crucial. We found the ASGM implementations to be Agile PLUS SGM. In other words, teams were BOTH Agile and SGM, not a hybrid between the two. Agile promotes a lightweight process, focused on necessities, and layering a structured SGM system on top of Agile may run counter to its principles and add bureaucracy. Our cases leveraged the classical elements of Agile such as Sprints, Epics,[2] and Backlogs.[3] Unfortunately, management did not fully understand Agile terminology, nor were they interested in learning. SGM simply provided a better perception of control. Management wanted to conduct business in an SGM fashion even though development teams were operating under Agile principles, requiring additional resources. Several respondents discussed the need to 'translate' between the two approaches, particularly during gate reviews or project updates.

4.4. ASGM implementation levels

Bianchi et al. (2018) demonstrated for Italian software developers, Agile practices varied by levels, that is to say, controls were included such as the

developers' age, freelance status, and leadership role, coupled with the project team and organization size, along with stratification of Agile tenets, particularly, specification delay, Sprint utilization, and customer feedback. Within the framework of our qualitative study, all three of these Agile levels were strongly represented; that being said, our respondent's behaviors were complicated.

4.4.1. Specification delay

Our cases recognized this as a benefit of Agile. BUs sought out the ASGM approach as a means to embrace delayed specification freeze as a goal, driven by, in many cases, competitive and market pressures. As much as our cases wanted to learn from customer engagements, which they did, there was a strong desire to define the product early on.

4.4.2. Sprint utilization

For our conglomeration of physical product producers the Sprint was most evident in thought and in practice. Teams fastidious with its implementation had experienced primary benefits of speed and customer alignment, enabled from improved team focus and communication. We specifically inquired about Sprints and their associated details (e.g., duration, staffing, planning) during our interviews.

4.4.3. Customer feedback

Our respondents invested in customer feedback sessions leveraging prototypes as complete products, sub-systems, or even components, aligning vigorously with this Agile tenet. In fact, this was another perceived driver toward ASGM. Our cases benefitted from an increased understanding of customer needs and wants, often updating product specifications. Conversely, the teams may have relied too heavily on physical prototypes leading to a resource penalty chasing Agile.

5. Discussion

The nascent body of ASGM research that exists for physical products has been presented by notable NPD process management experts. See Table 6 for a summary of current ASGM research, along with a comparison of our study results. We agree with Sommer et al. (2015) about the virtues of increased development process flexibility, improved team communication, coordination, morale, motivation, and productivity, as well as task prioritization, and alignment between process and methods. Similarly, we found a lacking Agile culture in the organization as well as an increase in bureaucratic documentation, but we did not find evidence of delays from resource distribution, reward systems misalignment, or inadequate knowledge management. Further research may reveal why delays occur in some cases but not in others. Perhaps reward systems can be designed to reduce misalignment? It is not clear why knowledge management is problematic in one context but not others.

Our results are also consistent with Karlstrom and Runeson (2005), where better internal team communication and morale, progress metrics for management, efficient planning, customer feedback, and adaptability to market needs are achieved by adopting ASGM. We also corroborate the downside of frustration from a lack of frozen product specifications. However, our study results do not support their findings of team isolation, stifled long-range planning capability, role conflicts, or stakeholder preference of partial project approvals. Perhaps moderating factors could be identified to help explain why some teams may experience isolation but not others, or why role conflicts occur in some organizations but not others.

Our results are compatible with Bianchi et al. (2018), who conducted a study of 181 Italian software developers and found that Agile, specifically Sprints, had a positive impact on project speed, cost, and quality, as well as structural duplication issues. Bianchi et al. (2018) also found a marginally negative association between Agile specification practices (i.e., delayed definition) and project speed, which we did not. Papers that address ASGM performance are few and future researchers may have the opportunity to parse results at different levels to better estimate the performance of ASGM implementations.

Cooper and Sommer (2016a, 2016b) raise several research questions that our study addresses; these are summarized in Table 7. We have reservations about concluding that SGM and Agile being completely symbiotic. Our cases frequently demonstrated with, perhaps overly, a fervent reliance on physical prototypes which negatively affected cost in a way that software firms do not experience. ASGM seemed most appropriate for innovation projects with high uncertainty, then again, we did see the methodology used in all development phases, including early research, technical, or late-stage commercialization ramp-up. Dedicated resources are essential for the doctrine; significantly, our cases were able to loosen this requirement for some cross-functional members. The implementation of ASGM did not eliminate gates, nor differentiated between hardware and software activities. Stakeholders (e.g., executive leadership) approval was granted with teams presenting project

Table 6. ASGM literature comparison

Author(s), journal, & year	Extant literature findings	Study findings
Sommer et al. RTM 2015	Increased flexibility	**Supported** with Process Flexibility and Lightweight Process themes – Critical for the NPD process to incorporate flexibility in terms of duration, learning, and gates
	Improved communication, morale, & productivity	**Supported** with Team Communication theme – ASGM led to more engaged, motivated, productive, and transparent teams, where teams felt more connected to the project and more informed
	Improved task prioritization	**Supported** with Team Communication theme – Cases spent a significant amount of time prototyping designs with customer to rank, then re-rank features
	Better alignment between process and methods	**Supported** with Process Flexibility, Lightweight Process, Entrepreneurial Mindsets, and Project Control themes – Our cases were focused on lightweight processes reducing the bureaucratic burden to focus team energy on the product
	Resource delays	**Not Supported** – Did not observe this issue, our cases utilized a 'core' vs 'extended' co-located resource model; projects did not languish for resources
	Reward systems misaligned	**Not Supported** – Did not observe this issue; teams were focused on delivering products and accomplishing development tasks
	Lack of Agile culture	**Supported** with Structural Duplication theme – A lack of Agile understanding forced the project teams to use two 'languages' when communicating
	Poor knowledge management	**Not Supported** – Cases did not divulge knowledge issues, seem to have access to the right resources when needed. 'Agile' personnel central to Team Talent theme
	Project documentation bureaucracy	**Supported** with Structural Duplication and Lightweight Process themes – Because our cases operated in pluralistic fashion due to management concerns, gate documentation and communication became redundant
	Difficulty ending or handing over projects	**Not Supported** – Our cases were operating with an overarching schedule with precise dates and milestones, they constantly re-ranked priorities, and were sensitive to 'scope creep' especially in the later stages of development
Karlstrom and Runeson IEEE Software 2005	Better communication	**Supported** with Team Communication theme as mentioned above
	Better progress metrics	**Supported** with Project Communication theme – Cases used software (management) tools to help track activities and produce charts/metrics for management
	More efficient planning	**Supported** with Customer Value and Managing Product Requirements themes – Cases worked to extract what the customer valued, through backlog prioritization, critically watched for 'scope creep', resisting additions.
	Improved customer feedback	**Supported** with Customer Value and Managing Product Requirements themes – Cases spent significantly to find the 'right' feature, then turned those features and wants into actionable engineering specifications
	Dedicated resources lead to isolation	**Not Supported** – Did not find this concern, we found other negatives with dedicated resources, not this. Respondents felt well aligned to the project being developed with their co-located, cross-functional team
	Short v Long term planning clash	**Not Supported** – Our cases did not impinge their long-range planning, they routinely met cross-functionally to develop multi-Sprint/phase plans to ensure the project tracked to an overall schedule
	Conflicts with new Agile roles	**Not Supported** – Did not observe issues with reticent behavior driven by ASGM implementation, functioning Scrum Master and Product Owner roles were common
Cooper and Sommer IMM 2016	Adaptive to market	**Supported** with Customer Value and Managing Product Requirements as mentioned
	Partial approval preference	**Not Supported** – Cases used gates, management expected a full, but potentially rough, picture before approval
	Lack of frozen specifications	**Supported** through our Managing Product Requirements theme – Many of our respondents described robust requirements as critical to project success

Table 7. ASGM implementation insights

Author(s), journal, & year	ASGM questions	Study findings
Cooper and Sommer IMM 2016	*Are SGM and Agile symbiotic?*	**Yes**, but w/cost. Cases integrated SGM & Agile from an operational perspective. Integration resembled an excessive level of duplicity, management was more comfortable w/SGM, expecting rigid gates, detailed checklists, thorough schedules
	What styles of projects?	**Innovative** ones with higher levels of uncertainty. Cases felt ASGM was applicable where learning was a premium
	Which stages of a project?	**All**. Cases used ASGM throughout development, although one could sense a 'degree' of agility in-play. Early in development teams were very Agile and open to learning, as commercialization came closer, they were less 'Agile'
	Are resources dedicated?	**Yes**, our cases employed a 'core' and 'extended,' co-located team approach, where 'core' team was always present and 'extended' team (e.g., purchasing or regulatory) was allowed to flex depending on Sprint focus or phase
	Are gates needed?	**Yes**, the utilization of gates to be mostly unchanged. Gates were essential to success, preferring a high-level plan
	Different gates for SW?	**No.** Sprint durations could change depending on focus. Separation may lead to isolation negating ASGM benefits
	Stakeholder approval with a partial plan?	**No**, management approved projects in ASGM because it was presented mainly in the traditional language of SGM. It was unlikely that management would recommend an incremental approach, wanted a complete assessment at the start
	Early VoC still needed?	**Yes**, emphatically. Our cases sought more early-stage VoC, as detailed specifications were an enabler of project success
	Agile roles & terminology used?	**Yes**, we found this, in fact, the observance of Agile team roles such as Scrum Master, Product Owner, and Business Owner, were keywords used to assess the degree of ASGM implementation, which was very pronounced
	Stakeholder awareness and multiple projects?	**Yes**, but our cases presented to senior management in a more traditional SGM manner, so the receiving message here was mostly unchanged with tried and true phase gates, gate checklists, and high-level project schedule
Cooper and Sommer JPIM 2016	*Clear Definition of Done?*	**Yes**, this was a challenge to our cases, they developed a broader sense of a completed Sprint, not just in terms of product features, but with project tasks included breaking more complex tasks down to digestible Sprint durations.
	Dedicated resources?	**Yes**, but to a certain degree, as answered above
	Model and Sprint planning integrated?	**Yes**, teams met to establish macro-level phase, then Sprint cadences, but also to cull features, creating follow-on projects, allowing BU an opportunity to stay focused on the project at hand but create future model plans
	Higher success, better productivity, and improved speed?	**No**, but it did benefit product success and speed. Our cases felt strongly that the product would better align with customer needs, leading to improved market performance, as well as a positive impact on development speed, but came with a negative impact on development costs
	If so, why is ASGM better?	**Provides balance**. ASGM balances flexibility of learning (e.g., find features) and control, additionally, the methodology offers focus and resources, which energizes teams and improves performance
	Challenges and weaknesses?	**Resource costs**. Cases felt positive about ASGM, particularly the communication, morale, and accountability aspects, along with focus and speed, only the pursuit of 'getting the product right' according to customers
	Additional adjustments for physical products?	**Several**, project teams must utilize virtual prototyping techniques, provide additional Agile training, allow team member personal growth time, and optimize the intersection of Agile and SGM procedurally
	Do teams still spiral during development?	**Yes**, more prevalent in earlier phases. Cases spent a lot on demonstrations, placing a premium on customer feedback, however, learning was more encouraged in design phases, less so as manufacturing and commercialization encroached
	What is in a backlog?	**Design Activities** and design features. Tasks needed to complete a project, phase by phase, along with features, then decomposed into manageable Sprints, w/clear acceptance criteria to support 'Definition of Done'

plans using the traditional language of SGM, not as Agile developed partial plans even-though the cases assumed Agile roles and terminology. Stakeholders were aware of and oversaw multiple ASGM projects. Finally, the teams incorporated Voice of the Customer (VoC) activities early on in a comprehensive manner.

Teams defined completed Sprints with a few functional adaptions by including not only product-related tasks (e.g., test circuit board for output) but also design process related work (e.g., release drawings). Teams met to establish macro-level phase plans that aligned to Sprint cadences, but also to cull features, pushing for follow-on projects when necessary, allowing the business an opportunity to stay focused on the project at hand, simultaneously creating future model plans. ASGM honestly seems to balance the flexibility of learning, to build a better product, with a requisite level of project control, to achieve commercialization. Additionally, the methodology provides focus and resources, which energizes teams, with some downsides, and improves performance.

Overall, we are left with five recommendations for future research that stand out to us. First, dedicated resources aided project speed, but the cost burden was solely allocated to the project in focus. Under SGM, resources could support other projects defraying costs. Our cases did not realize a resource benefit; we believe the Agile concept of team flexibility, where different roles cover for each other, may not have been adequately leveraged. How do the best teams manage dedicated resources without the cost? The resource penalty we observed seems simple: dedicated teams, considering the uneven nature of activities across a long project, should be assigned via an optimized staffing model. Our cases sensed that dedicated resources, when compared to a shared portfolio model, created an ASGM penalty; regardless of the staffing model, a more profound issue could be the cause. Agile team resources must be experienced, small business-minded, and confident, but also flexible in a cross-functional sense, when crucial Sprint tasks require completion. For example, a mechanical design engineer with strong writing skills could update a regulatory plan to achieve a Sprint timeline, according to Agile, this is the ideal team-mate. ASGM practitioners, to possibly defray the dedicated resource cost impact, should encourage project team members by highlighting, celebrating, and cross-training to create an environment where team members truly pitch-in regardless of functional workstream. Flexibility is not only for the management process of ASGM but also for team personnel.

Second, executive management in many of our cases was not entirely comfortable with Agile, necessitating a more complex and thorough control approach. They simply did not understand Agile doctrine; as such, they gravitated toward SGM, Agile planning was seen as incomplete. Perhaps training directed at senior leaders would help them adapt? Or will this problem eventually work itself generationally? We are confident that our cases implemented ASGM thoroughly; despite this, BU executives operating at the Agile–SGM interface were accustomed to the language of SGM.

Third, our cases prototyped throughout development; they embraced the feedback model of Agile; notably, we observed that demonstrations tended to be physical; for these products, this work comes with high costs. It is possible our cases could have leveraged virtual prototypes to a higher degree to mitigate the cost impact of frequent customer interactions. Practitioners of ASGM should exercise caution with demonstration frequency using tangible prototypes, such as machined or welded components and intensive control system layouts. Perhaps, blindly following Agile for physical products leads to a cost penalty unless current demonstration methods are employed effectively, for instance, digital modeling, 3-D printing, and virtual prototyping.

Fourth, managers should be wary of the human toll that relentless ‘sprinting’ has on staff. ASGM energizes teams; moreover, excessive stress on team members seems counterproductive, especially for longer duration physical product initiatives. There may be a need for research examining the wellbeing of employees subjected to continual ‘sprinting’.

Fifth, future research can examine ways of reducing structural redundancy. There is a need to understand how to merge Agile and SGM methodologies more efficiently to minimize redundancies. A lightweight, yet functional level of control, must be instituted within local policies and procedures, teams need to define their level of ‘balance’ that is the essence of ASGM to limit bureaucracy. Leadership that demanded an Agile to SGM ‘translation’ for phase reviews must adopt Agile language and, particularly, trust in its methods for overall project control. Software firms have demonstrated that learning, flexibility, and lightweight processes can still offer proper project control. Notably, deciphering between these two methodologies consumed additional resources. Extensive Agile training should be incorporated, not only for the project teams but also for the management ‘eco-system’ within which the units operate. We imagine that any process or methodology change requires sound, thorough, and repeated practice to establish a sincere attempt toward implementation. Also, new Agile inspired roles, such as Scrum Master and Product Owner, cannot merely be additive to an

organization, something else must give. For all of the positives, we wonder if manufacturing firms are truly managing NPD with a different ASGM model, not in today's Industrial Scrum approach (Sommer et al., 2015), or intra-phase spiraling (Cooper and Sommer, 2016a), but in a manner that acknowledges the heavy burden of manufacturing a complex physical product to be produced at scale.

6. Conclusion

Our study lends support to the idea that Agile methods can be combined with SGM with in physical product firms and elaborates on the mechanisms in play using rigorous tools from the social sciences to develop this theory, particularly for a complex process that is NPD. We make several contributions to the literature on NPD, new product management, and management in general. First, Agile can be organized into eight central tenets; these tenets can be used to assess the state of ASGM as practiced by development teams. Second, it is essential to understand how contextual differences matter, in this case, the physical product contexts of global firms rather than software companies. The prominence of resource investment inefficiency was an important discovery that was unexpected from a direct translation of Agile benefits in the software context; this suggests that ASGM is riskier if there is resistance to paying the additional resource costs. Thus, there are essential differences between implementing Agile methods for physical products, supporting the idea that administrative technologies may have different consequences for adopting firms when they are transferred to new contexts. Third, from the theme extraction, there are several critical behaviors that development teams should focus on to improve speed, increase innovation, and unlock resource efficiencies. Our study highlighted the speed and innovation improvements of ASGM for physical products, but also exposed the negative resourcing impacts of this same methodology. For future practitioners of ASGM in the physical product realm, we postulate that duplicative project management structures, dedicated project staffing, and excessive or complex product demonstrations can fulfill some of the benefits of Agile but are potentially the root causes of increased project costs.

References

Abrahamson, E. (1991) Managerial fads and fashions: the diffusion and rejection of innovations. *Academy of Management Review*, **16**, 586–612.

Abrahamson, E. (1996) Management fashion. *Academy of Management*, **21**, 254–285.

Abrahamson, E. and Eisenman, M. (2008) Employee-management techniques: Transient fads or trending fashion? *Administrative Science Quarterly*, **53**, 719–744.

Alliance, T.A. (2001) Agile Manifesto [Online]. Available: http://agilemanifesto.org/

Bers, J.A., Dismukes, J.P., Mehserle, D., and Rowe, C. (2014) Extending the stage-gate model to radical innovation – the accelerated radical innovation model. *Journal of Knowledeg Economy*, **5**, 706–734.

Bianchi, M., Marzib, G., and Guerinic, M. (2018). Agile, stage-gate and their combination: exploring how they relate to performance in software development. *Journal of Business Research*, **110**, 538–553.

Biazzo, S. (2009) Flexibility, structuration, and simultaneity in new product development. *Journal of Product Innovation Management*, **26**, 336–353.

Boehm, B. and Turner, R. (2004) *Balancing Agility and Discipline: A Guide for the Perplexed*. Boston, MA: Addison-Wesley.

Byrne, J.A. (1986) Business fads: what's in – and out. *Business Week*, **2929**, 52–56.

Christiansen, J.K. and Varnes, C.J. (2009) Formal rules in product development: sensemaking of structured approaches. *Journal of Product Innovation Management*, **26**, 502–519.

Cohen, L.Y., Kamienski, P.W., and Espino, R.L. (1998) Gate system focuses industrial basic. *Research Technology Management*, **41**, 4, 34–37.

Cooper, R.G. (2014, January–February) What's next? After stage gate. *Research Technology Management*, **2014**, 20–31.

Cooper, R.G. and Kleinschmidt, E.J. (1991) New product processes at leading industrial firms. *Industrial Marketing Management*, **20**, 137–147.

Cooper, R.G. and Kleinschmidt, E.J. (1993) Screening new products for potential winners. *Long Range Planning*, **26**, 74–81.

Cooper, R.G. and Sommer, A.F. (2016a) Agile-stage-gate: new idea-to-launch method for manufactured new products is faster, more responsive. *Industrial Marketing Management*, **59**, 167–180.

Cooper, R.G. and Sommer, A.F. (2016b) From experience: the agile–stage-gate hybrid model: a promising new approach and a new research opportunity. *Journal of Product Innovation Management*, **33**, 513–526.

Dikert, K., Paasivaara, M., and Lassenius, C. (2016) Challenges and success factors for large-scale agile transformations: a systematic literature review. *The Journal of Systems and Software*, **119**, 87–108.

Eisenhardt, K.M. and Graebner, M.E. (2007) Theory building from cases: opportunities and challenges. *The Academy of Management Journal*, **50**, 25–32.

Glaser, B.G. (1999) The future of grounded theory. *Qualitative Health Research*, **9**, 836–845.

Goggin, M.L. (1986) The "too few cases/too few variables" problem in implementation research. *Western Political Quarterly*, **39**, 328–347.

Graebner, M.E. and Eisenhardt, K.M. (2004) The seller's side of the story: acquisition as courtship and governance as syndicate in entrepreneurial firms. *Administrative Science Quarterly*, **49**, 366–403.

Hackman, J.R. (1975) Is job enrichment just a fad? *Harvard Business Review*, **53**, 129.

Homburg, C., Wilczek, H., and Hahn, A. (2014) Looking beyond the horizon: how to approach the customers' customers in business-to-business markets. *Journal of Marketing*, **78**, 58–77.

Hutchins, N. and Muller, A. (2012) Beyond stage-gate: restoring learning and adaptability to commercialization. *Strategy and Leadership*, **40**, 30–35.

Iansiti, M. (1995) Technology integration: managing technological evolution in a complex environment. *Research Policy*, **24**, 521–542.

Iles, P., Preece, D., and Chuai, X. (2010) Talent management as a management fashion in HRD: towards a research agenda. *Human Resource Development International*, **13**, 125–145.

Jespersen, K.R. (2012) Stage-to-stage information dependency in the NPD process: effective learning or a potential entrapment of NPD gates? *Journal of Product Innovation Management*, **29**, 257–274.

Juricek, J. (2014) Agile project management principles. *Lecture Notes on Software Engineering*, **2**, 172–175.

Kalluri, V. and Kodali, R. (2014) Analysis of new product development research: 1998–2009. *Benchmarking: An International Journal*, **21**, 527–618.

Karlstrom, D. and Runeson, P. (2005, May–June) *Combining agile methods with stage-gate project management*. Lund, Sweden: IEEE.

Krippendorf, K. (2004) Reliability in content analysis: some common misconceptions and recommendations. *Human Communication Research*, **30**, 411–433.

Laplume, A.O. and Dass, P. (2014) Outstreaming for ambidexterity: evolving a firm's core business from components to systems by serving internal and external customers. *Long Range Planning*, **48**, 3, 135–150.

Lawler, E.E. and Mohrman, S.A. (1985) Quality circles after the fad. *Harvard Business Review*, **63**, 65–71.

Lev, B. (2000) Knowledge management: fad or need? *Research Technology Management*, **43**, 9–10.

Mohan, K., Jain, R., and Ramesh, B. (2007) Knowledge networking to support medical new product development. *Decision Support Systems*, **43**, 1255–1273.

Munthe, C.I., Uppvall, L., Engwall, M., and Dahlén, L. (2014) Dealing with the devil of deviation: managing uncertainty during product development execution. *R&D Management*, **44**, 203–216.

Pich, M., Loch, C., and De Meyer, A. (2002, August). On uncertainty, ambiguity, and complexity in project management. *Management Science*, **48**, 1008–1023.

Rigby, D.K., Berez, S., Caimi, G., and Noble, A. (2015) *Agile Innovation*. Boston, MA: Bain & Company.

Rogers, E.M. (2010) *Diffusion of Innovations*, 5th edn. New York, NY: Free Press by Simon & Schuster.

Roulston, K., Demarrias, K. and, Lewis, J.B. (2003) Learning to interview in the social sciences. *Qualitative Inquiry*, **9**, 643–668.

Rubin, K.S. (2013) *Essential Scrum – A Practical Guide to the Most Popular Agile Process*. Upper Saddle River, NJ: Pearson Education.

Sandelowski, M. (1994) Focus on qualitative methods: the use of quotes in qualitative research. *Research in Nursing & Health*, **17**, 479–482.

Schwaber, K. (2004). *RE: SCRUM Development Process*. Burlington, MA: Advanced Development Methods.

Sommer, A.F., Hedegaard, C., Dukovska-Popovska, I., and Steger-Jensen, K. (2015, January-February) Improved product development performance through agile/stage-gate hybrids: the next-generation stage-gate process? *Research-Technology Management*, 34–44.

Stake, R.E. (2005) Qualitative case studies. In Norman, K. Denzin & Yvonna, S. Lincoln (Eds.), *The Sage Handbook of Qualitative Research*. Thousand Oaks, CA: Sage Publications Inc. pp. 443–466.

Strauss, A. and Corbin, J. (1998) *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, 2nd edn. Thousand Oaks, CA: SAGE Publication Ltd.

Takeuchi, H. and Nonaka, I. (1986, January-February) The new product development game. *Harvard Business Review*, **64**, 137–146.

Yin, R.K. (1999) Enhancing the quality of case studies in health services research. *Health Services Research*, **34**, 1209–1224.

Notes

[1]Sprints are time-bounded groupings of development activity that produce tangible product designs.

[2]Epic is a macro set of product features, bundled together to deliver a significant portion of the product which align to critical customer needs.

[3]Backlog is a prioritized set of project tasks required to complete product development.

John J. Salvato is the President of Design Quality Services LLC based in Galesburg, Michigan, USA, a product design, and quality management consulting firm. As a 27-year industry veteran within the medical device and automotive markets that has held influential positions within quality assurance, manufacturing, and product development, John has championed New Product Development (NPD) activities around the world. His research has an emphasis on engineering management practices, specifically the integration between Agile and gate driven methods for firms that design and develop complex physical products.

Andre O. Laplume is an Associate Professor of Strategy and Entrepreneurship at Ryerson University in Toronto, Canada. His research lies at the intersection of strategy, innovation, and entrepreneurship and has been published in several leading academic journals, including the Journal of Management, Journal of International Business Studies, and Research Policy.

APPENDIX A.
Textual cues for Agile techniques

Agile element and occurrence	Element description	Element cue Example	Auto	Railway	Monitoring	Perimeter	Medical
Team interface (TI) – 96.9%	Individuals and interactions over process and tools	Teams focused on tasks at hand, finding solutions to product features, less interested in the path taken – Keywords: Autonomous, Flexibility	Y	Y	Y	Y	Y
Product demonstrations (PD) – 86.2%	Working software/ product over comprehensive documentation	Frequent product, feature, sub-system, component demonstrations as a means of illustrating progress – Keywords: Demonstration, Test, Customer, Integration, Evaluation	Y	Y	Y	Y	Y
Customer involvement (CI) – 75.9%	Customer collaboration over contract negotiations	Pulling in customers, internal or external, to ensure features are valued, less worried on formalized engagement – Keywords: Evaluation, Demonstration, Test, Feedback, Review	Y	Y	Y	Y	N
Specification flexibility (SF) 79.3%	Responding to change over following a plan	Planning often, accepting of specification change or learning from customer interactions – Keywords: Change, Flexibility, Update, Priority, Feedback, Learning	Y	Y	Y	Y	Y
Team structure (TS) – 86.2%	Product Owner/ Scrum Master/ Self-Organizing	Organizational structure that employs key Agile roles, such as an empowered Product Owner, and behaviors that illustrate empowerment – Keywords: Empowered, Autonomous, Engaged, Customer Focused, Accountable	N	Y	Y	Y	Y
Time-bound (TB) – 82.8%	Time bounded sprint activity with planning	Sprints that are concretely time bound with specific activities planned – Keywords: Week, Month, Quarter, Defined, Time	N	Y	Y	Y	Y
Feature prioritization (FP) – 69.0%	Establish product feature priorities, creating Epics/ Stories to support the importance	Clear establishment of feature/task priorities, culling of less desired features, reduction of scope, implementation of user stories to establish importance – Keywords: Minimum Viable Product (MVP), Priority, Learning, Feedback	N	Y	Y	Y	Y
Communication (Comm) – 86.2%	Scrum team meeting, team location, communication tools	Team communication patterns, frequent, quick meetings, interactions of members, tool usage (burndown charts or backlogs) – Keywords: Transparency, Communication, Alignment, Burndown, Engagement	N	Y	Y	Y	Y