Project Management Development – Practice and Perspectives
International Scientific Conference on Project Management in the Baltic Countries
Riga, University of Latvia
e-ISSN 2501-0263

Agile Development and Open Innovation: Challenges and Trade-offs in Virtually Enlarging Innovation Sources

Stallinger Fritz, University of Applied Sciences BFI Vienna, Vienna, Austria

Abstract

Ever faster changing business and technological environments and increased competition urge companies to look for new sources of information to increase and speed up their innovation process. Open Innovation addresses this need through integrating knowledge and ideas from a company's ecosystem. On the other side, companies applying Agile Development base their innovation strategy on close cooperation with customers and on-site collaboration but tend to leave the innovation potential of a broader range of outside-stakeholders unused. In this context, the paper contributes to the research on the application of Open Innovation in Agile Development. Also addressing the pressure enforced by the recent pandemic for virtualisation of almost any collaboration, it identifies which challenges arise and which trade-offs to decide on when agile companies utilize the Open Innovation approach to integrate the knowledge of external stakeholders particularly by means of virtual communities.

A literature analysis following the principles of systematic mapping studies is applied to the topics of *Open Innovation* and *Agile Development*, but also related areas that can provide examples of virtual collaboration and openness. The overall goal of the research is to contribute to adapting an innovation system customization framework to the context of Open Innovation in Agile Development.

The findings indicate several challenges counteracting a straightforward application of Open Innovation concepts in Agile Development and point out the need for companies to thoroughly evaluate adaptations of either agile principles or Open Innovation elements. Solution approaches indicate a trend towards re-shifting to rather classical approaches where stability, planning, and documentation play an important role. The insights from identifying these challenges and trade-offs are used to derive design decisions for adapting an innovation system customization approach to the given context.

Key words: *agile development, open innovation, virtual communities, stakeholder collaboration, innovation system customization.*
JEL code: O22, 036, L17, L22

Introduction

Emerging from the software industry, agile development in contrast to traditional development approaches focuses on streamlining procedures, direct communication, self-organization, and close collaboration with the customer (Beck et al., 2001). Particularly in information technology, agile approaches and methods proofed to raise productivity and quality as well as employee and customer satisfaction (Dybå & Dingsøyr, 2008). The move away from an 'introverted' development by integrating customers throughout the whole development endeavour is a major advantage of Agile Development and particularly small companies and teams benefit from the reduction in time-to-market and early customer feedback (Conboy & Morgan, 2011). On the other hand, limiting the cooperation only to customers or customer representatives could compromise the innovation potential of a company as new ideas and respective proposals for implementation often emerge from the end-users of a product or service or from external experts (Reichwald & Piller, 2009).

The Open Innovation approach in consequence, strives for exploiting the innovation potential of the various stakeholders of the whole ecosystem of a company to augment the company-internal innovation capabilities (Chesbrough, 2011). As innovations typically emerge

from the collaboration of several involved people rather than from the work of an individual, a special focus of the Open Innovation approach relates to virtual communities. Their facilitator role and importance for the Open Innovation approach is underpinned by the steady growth of the internet and continuously emerging new or improved means of virtual collaboration. Consequently, Open Source Software Development and applications in the context of Software Ecosystems can be analysed as examples of how to use virtual communities to encourage user innovation (Reichwald & Piller, 2009).

In general, innovations tend to disturb well-established routines in an organization and bear the inherent potential for conflicts (Hauschildt, Salomo, Schultz, & Kock, 2016). Therefore, an organization must determine rules on how to handle innovations. Typical elements of such innovation management systems are identified by Davila, Epstein, and Shelton (2013) and comprise amongst others *innovation strategy*, *organizational structure*, *innovation process*, and *innovation culture*. Due to different business characteristics, each organization must develop or at least customize its own innovation management system. For the context of software businesses pursuing a Software Product Line Engineering approach Stallinger, Neumann, and Schossleitner (2014) proposed a business characteristics-driven approach for systematically customizing an organization's innovation management system.

The research presented here addresses the challenges that arise when organizations applying Agile Development decide to apply the concept of Open Innovation by integrating potential external stakeholders via virtual communities. In a longer-term step, it is envisioned to provide an innovation management system customization framework for the context of Agile Development and Open Innovation, capturing the main issues and practices identified. The research questions underlying the work in this paper are thus threefold: Firstly, which challenges emerge when agile companies use the Open Innovation approach to integrate external stakeholders via virtual communities? Secondly, based on the results of Research Question 1, how could an existing framework for innovation management system customization be conceptually adapted to support innovation system customization at the intersection of Agile Development, Open Innovation, and virtual stakeholder integration? Thirdly, but out of the core scope of this paper, which practices for virtual stakeholder integration applicable in the context of Open Innovation and Agile Development can be identified for integration in the framework? Research question 1 comprises the focus of this paper and is addressed by literature search and analysis following the guidelines for systematic mapping studies as proposed by Petersen, Feldt, Mujtaba, and Mattsson (2008) and Kitchenham, Budgen, and Pearl Brereton (2011). Research question 2 addresses particularly step 3 of a six-step design science-based approach (cf. Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007) to develop the innovation management system customization approach comprising the steps of problem identification and motivation, objectives for a solution, design and development, demonstration, evaluation, and communication, with the first two steps and the design part of step 3 covered within this paper.

Several related works address Open Innovation in the context of Requirements Engineering, e.g. (Alspaugh & Scacchi, 2013), (Knauss, Damian, Knauss, & Borici, 2014), (Linåker, Rempel, Regnell, & Mäder, 2016), or (Wnuk, Pfahl, Callele, & Karlsson, 2012), but overall, these studies do not explicitly address the management of external stakeholders. According a research agenda for Requirements Engineering in Open Innovation by Linåker, Regnell, and Munir (2015) stakeholder management in open systems needs to be further investigated, particularly with respect to the mode of approaching external stakeholders, the way to use existing requirements artifacts, or the integration of processes and methods from the Open Source Software Development. In their research in the context of software development Conboy

and Morgan (2011) conclude that "there is a lack of understanding of what constitutes innovation in software development in general and to what extent agile methods actually facilitate this process". Accordning Munir, Wnuk, and Runeson (2016) the combination of Agile Development and Open Innovation "seems to create barriers in transferring the ideas outside the team's boundaries". In a bachelor thesis supervised by the author, Kordon (2017) distilled four problem areas hindering the integration of external stakeholders in agile systems: the need for transition to online artifacts, issues of knowledge sharing and long-term knowledge retention, increased planning insecurity for both the agile company and the external stakeholders, and reaching a network's critical mass given the small agile team size.

The remainder of the paper is structured as follows: Section 2 summarizes relevant background on requirements and innovation in Agile Development, Open Innovation, stakeholder integration via virtual communities, and innovation system customization. Section 3 summarizes the challenges and trade-offs identified for the combination of Open Innovation and Agile Development. Section 4 consolidates the insights from section 3 into basic design decisions for an innovation system customization framework. Section 5 concludes the paper.

Background

Innovation and Requirements in Agile Development

Agile approaches typically address the issue of frequent changes and the ongoing demand for innovation by maintaining a constant exchange with the customer and thereby focus on close interpersonal collaboration (Beck et al., 2001; Dybå & Dingsøyr, 2008). It is consequently the involved people who represent the most significant bottleneck in the innovation process. Further, from the viewpoint of the influence of developers on the innovation process Hevner and Green (2000) observe in the context of software development that with more perceived control by and more involvement of developers particularly technical innovations get adapted more quickly and effectively. Moreover, a growth of a company might negatively affect the innovation potential of developers as scaling up typically implies moving away from interdisciplinary task fields to rather specialized ones with less responsibilities and space for creativity (Moe et al., 2012). Overall, since a close relationship to the customer constitutes a significant and human resources-intensive part of Agile Development, it seems to rather prevent a company from applying a broader and more open approach to innovation (Conboy & Morgan, 2011).

To better integrate real end-users into the innovation process, the combination of user-driven approaches with Agile Development is proposed by several authors (e.g. Chamberlain, Sharp, & Maiden, 2006; P. Näkki, K. Koskela, & M. Pikkarainen, 2011). According Chamberlain et al. (2006) – for the case of *User Centered Design* – five dimensions can be identified showing a direct correlation with Agile Development. These comprise direct user involvement into the development process; collaboration and culture stimulating active communication and collaboration with users; prototyping enabling users to provide feedback; a project lifecycle giving enough time to early identify user needs and requirements; a way of project management guiding the interplay of Agile Development and User Centered Design without too strict rules. Particularly, the fourth dimension here might contradict to the agile claim for short iteration cycles if additional time must be spent on elicitation and communication of user needs and requirements.

Agile methods generally view requirements as information that is subject to quick change and that cannot be elicited at once prior to development (Sillitti & Succi, 2005). With respect to agile requirements engineering practices, several principles can be identified (cf. Cao & Ramesh, 2008): requirements elicitation via on-site communication between developers and customers and documentation in only high-level descriptions; iterative negotiation of requirements with

increasing level of detail as development progresses; extensive and repeated prioritizing of requirements together with the customer; constant planning to maintain flexibility when dealing with changes; delivery of prototypes and mock-ups to the customer for early requirements validation; regular meetings to evaluate the requirements to keep track of project status and validate the match of the requirements with customer's need. – Overall, the outlined practices pose the need for high-qualified staff due to an inherent lack of documentation and rigor and the high amount of transfer tasks with the customer (Savolainen, Kuusela, & Vilavaara, 2010).

The artifacts used in requirements engineering in Agile Development correspond to the focus on on-site communication and emphasize simple methods that can easily be performed with pens and paper (Sillitti & Succi, 2005). Particularly, the physical character of such artifacts (like e.g., user stories) is regarded a key promotor of collaboration, communication, and self-organization. On the other hand, the trend to virtual teams and the use of information sharing systems causes issues with maintaining this promotor (Sharp & Robinson, 2008). Similarly other artifacts like e.g. a product backlog lack detailed specifications, underlie high dynamics and reprioritization and do not provide a sound basis for differentiating between user requirements and system requirements (Savolainen et al., 2010).

Innovations and Open Innovation

From a terminological perspective, 'innovation' is not restricted to a new idea or invention, but also encompasses its exploitation and successful introduction in a market. Consequently, the term 'innovation' only applies if there is some novel combination of – in broad sense – "tools" and "purpose" and such innovations generally can be categorized according various criteria, like content (e.g. process innovation or product innovation), intensity/novelty (e.g. incremental or radical innovations), or subjectivity (different perception of novelty by different actors) (Hauschildt et al., 2016).

The innovation process is a central element of innovation management, and many process models are proposed in literature. Generally, an innovation passes through a series of phases, activities and decisions. While many of the models are based on rather sequential stage-and-review concepts, more recent models explicitly comprise parallel and iterative activities. Although the number and properties of the involved phases varies between sources, the generation and gathering of ideas and their commercial exploitation typically represent fixed starts and ends in such processes (Herzog, 2008). Figure 1 shows an ideal-typical model extending the mentioned two phases with a decision-oriented idea acceptance phase.

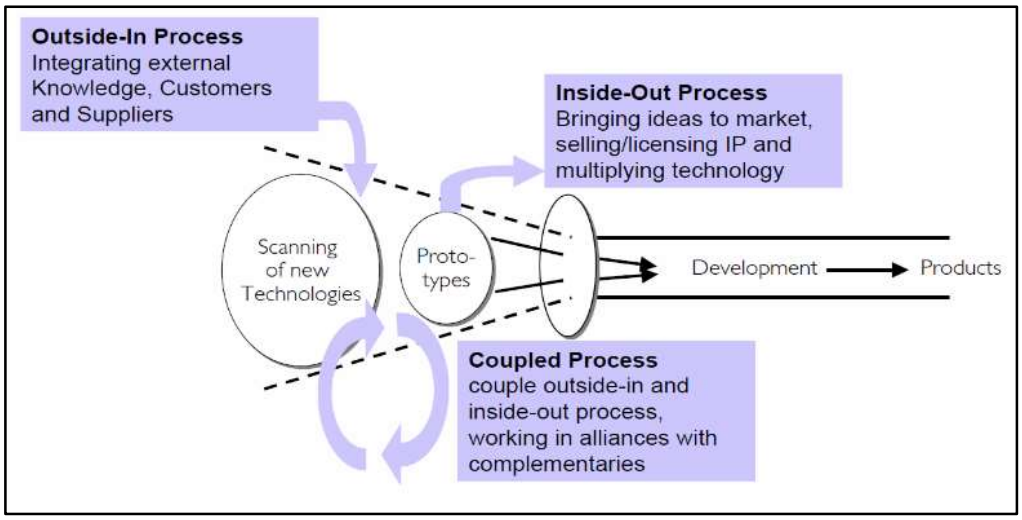
1. Idea generation	2. Idea acceptance	3. Idea realization
1.1 Determination of search area	2.1 Evaluation of the ideas	3.1 Concrete implementation of the novel idea
1.2 Idea finding	2.2 Preparation of realization schedules	3.2 Sales in desired market
1.3 Proposal of ideas	2.3 Decision on a plan to realize	3.3 Check on acceptance

Source: author's construction adapted from (Thom, 1980)

Fig. 1. **Three-phase innovation process**

In classical 'closed innovation', novel ideas are generated through knowledge within the company without being shared with others in the same application domain or market. Being further developed, they are typically regarded as intellectual property and competitive advantage, secured by patents and exploited through existing business models without allowing other potential contributors to consider further applications or uses (Chesbrough, 2011).

On the other side, to enlarge innovation capability, companies could leverage innovation potentials across their boundaries following the idea of 'collective invention' where free exchange of information allows companies to identify and follow the most relevant and promising technical developments (Allen, 1983). The concept of Open Innovation therefore tries to weaken the company's boundaries by involving external actors and utilizing the surroundings of a company through externalizing ideas which do not fit current internal business models, and by internalizing ideas and knowledge from the outside and using these within the company's innovation process (Chesbrough, 2011). Reichwald and Piller (2009) extend this concept by networks of experts and other value-creation partners and actively integrating users and their needs in all phases of the innovation process, particularly the earlier phases of idea generation and concept development. Such an Open Innovation approach can be conceptually structured into three cores 'archetypes' of innovation processes (cf. Figure 2).



Source: (Gassmann & Enkel, 2004)

Fig. 2. **Archetype processes of Open Innovation**

For the outside-in process, users and customers are of high importance as they are directly linked to the market of the company or the technology (Slaughter, 1993). A key factor for the success of this process is the ability of a company to develop competence in identifying the relevant external information carriers, to integrate them, and to assimilate the respective knowledge. Cohen and Levinthal (1990) denote this as 'absorptive capacity'. However, this capability might conflict with the 'Not Invited Here'-syndrome denoting the resistance of long-tenured internal groups against outside events and technological developments (Katz & Allen, 1982). For the case that a company's business model does not allow the integration of an idea emerging from the company's innovation process, the company can externalize this idea via the inside-out process and leverage technological multiplication and commercialization e.g. through licenses (Gassmann & Enkel, 2004). The coupled process, finally, combines both previous

processes into a collaborative ecosystem of companies, users, and experts and provides ways for finding the best purposes and markets for existing technologies and for developing novel tools and technologies to fulfil a given market's needs (Gassmann & Enkel, 2004).

1. Stakeholder Integration in Virtual Communities

Open Source Software Development is regarded as one of the most popular examples of collaborative networks driving innovation in terms of openness of process as well as of out-comes (Huizingh, 2011). Several factors like high numbers of ideas generated and programmers involved as well as the selection mechanisms and criteria for new features favour the transfer of principles to Open Innovation. Two basic approaches can be identified: 1) the user-based approach in form of a collaborative development of features, and 2) the vendor-based approach organized by a company managing feature and requirement selection and prioritization with involvement of users (Laurent & Cleland-Huang, 2009). Particularly this second approach highly overlaps with the context of Open Innovation and is further analysed hereafter.

Requirements in Open-Source Software Development ecosystems are quite different compared to requirements in closed systems. They are dispersed across and evolve within a plethora of different artifacts, online conversations, and repositories, and further include requirements-like artifacts like feature-requests. Most such artifacts describe a desired behaviour or feature rather in the solution than in the problem space (Alspaugh/Scacchi, 2013). This online and dispersed nature of requirements causes a series of challenges and problems, comprising among others: insufficient exploitation of stakeholder collaboration due to deficiencies of online tools in matching stakeholders with similar ideas; insufficient user-side mechanisms for prioritization resulting in users unsatisfied with the vendor-side handling of their effort and input; insufficient vendor-to-user communication, not satisfactorily allowing to keep track of user needs and including users into requirements elicitation; insufficient feedback and status updates on requests leading to contributors perceiving their input unnoticed or ignored (cf. Laurent & Cleland-Huang, 2009).

These challenges require a company to set up adequate and transparent decision making tools and processes for identifying innovation critical information in order not to get overwhelmed with a plethora of ideas and suggestions (Dahlander & Magnusson, 2008). Processing all ideas and requests with universally applicable processes could lead to discarding innovative ideas, as these are often immature and incalculable in their early stages. Wnuk et al. (2012) therefore recommend a segregation of requirements processes and refinement of prioritization methods as countermeasures as the plethora of requirements artifacts forms a comprehensive network of distributed knowledge and it is crucial for a company to know these artefacts and related communication structures.

To determine appropriate collaboration and communication strategies for virtual communities, the basic Open Innovation strategies to integrate external actors must be analysed. According Dahlander and Magnusson (2008) three basic strategies to integrate Open Source Software Development communities into business models can be identified:

- ***Accessing***: integration of new or existing communities to enlarge a company's innovation potential,
- ***Aligning***: alignment of deferring intentions of the community and the company regarding free availability or commercialization of the development or product, e.g., by means of licensing policies or incentives to influence the community's development,

- ***Assimilating***: after successful integration (i.e. accessing) and alignment, active pushing of the integration of the community's outputs, but also in return the provision of content to the community in order to leverage legitimacy.

The combination of these strategies can lead to hybrid structures within a company, encompassing proprietary as well as open parts or to structures with varying degrees of community involvement. Overall, a company that takes into account to open its boundaries, but also wants to maintain its competitiveness has to identify the appropriate degree of openness, sometimes by giving priority to control and limiting openness (Dahlander & Magnusson, 2008).

Following the concept of vendor-based Open Source Software Development, so-called Software Ecosystems form a reference of how a collaboration-based approach can be integrated into a company's business model. Jansen, Brinkkemper, and Finkelstein (2013) define a Software Ecosystem as "a set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts".

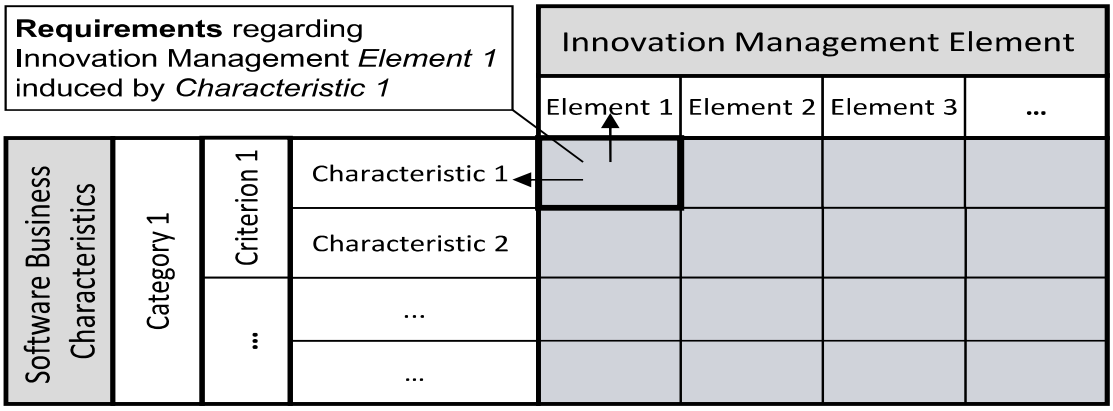
Exemplary studies of such Software Ecosystems allow drawing basic conclusions and illustrating behavioural patterns in the respective virtual communities. The study of Linåker et al. (2016) on an open source ecosystem showed a core team of stakeholders of platform users as well as platform providers with quite stable influence and collaborative behaviour to be the main drivers for ongoing development. In contrast, the study of Knauss et al. (2014) on a commercial ecosystem revealed a need for key players (platform providers, etc.) to ensure ecosystem stability. As these players have the power to change the direction of an ecosystem, niche players might get ignored or even forced to leave.

Similarly, to Open-Source Software Development systems, the representation of knowledge is crucial for the health of the ecosystem. Although consumers can rather easily provide feedback or request requirements, this information is typically only represented on a quite specific and narrow abstraction level, which requires experienced and skilled staff to derive appropriate interpretations using in turn their closest network, thus tending to lead to very close links and encapsulation of knowledge (Knauss et al., 2014).

Innovation System Customization

The management of innovations in a systematic way has been a subject to research for a long time and mainly driven by product and service businesses in consumer or business-to-business markets. Software Product Line Engineering as primarily an engineering approach emphasises a systematic and widely prescriptive management of product variability and proactive planning. Potential innovations to a Software Product Line may thus be hindered if they require changes to these pre-planned models and structures. (Stallinger & Neumann, 2013)

To help organizations that apply a Software Product Line Engineering approach to systematically and better exploit their innovation potential, Stallinger and Neumann (2013) present a conceptual framework for innovation system customization in the set context, which is further extended by Stallinger et al. (2014) with the proposal of an assessment-based innovation system evaluation and adaptation method. The framework captures and prescribes generic innovation management system requirements for the context of Software Product Line Engineering across two dimensions (cf. Figure 3): firstly, innovation management elements, like innovation strategy or innovation process, that must be considered in defining an innovation system; secondly, software business characteristics, that differentiate businesses or organizations. The requirements are then defined at the intersection of a specific software business characteristic with an innovation management element (Stallinger & Neumann, 2013).



Source: (Stallinger & Neumann, 2013)

Fig. 3. **Conceptual framework for innovation system customization**

According Davila et al. (2013) the following innovation system elements require consideration: *Innovation Strategy, Organizational Structure, Innovation Process, Innovation Culture, Innovation Measurement, Incentives and Rewards,* and *Learning.* Their adoption and interpretation for use in the framework is described in more depth in (Stallinger & Neumann, 2013). The criteria used to characterize and differentiate software businesses are organized into the following categories (cf. (Stallinger et al., 2014) for more details): *Customers and Market* (e.g. anonymity of customers, strength of customer-relationships), *Products and Services* (e.g. typical product life-span, degree of customization), *Engineering and Production* (e.g. repeatability of the process, number of involved engineering disciplines), and *Organization* (e.g. position in the value chain, structure of supplier networks). For each criterion in the software business characteristics dimension a set of typical characteristics or values the criterion could take for a specific organization is foreseen.

Open Innovation via Virtual Communities in Agile Development: Challenges & Trade-offs

This section tries to distil and structure the main results and insights from literature as summarized in subsections 1 to 3 under 'Background' above. The focus is on identifying major potential challenges and issues that arise when companies try to combine Agile Development with Open Innovation by integrating external stakeholders via virtual communities (cf. Research Question 1). The results are presented in Table 1 clustered into three categories according to the interplay of the involved approaches. Pointers to literature sources elaborating on the respective item or its underlying assumptions are provided where possible, as well as an indication which innovation system elements (cf. subsection 4 of 'Background' above) might be appropriate to address the respective item.

With respect to the effects between Agile Development and Open Innovation, basic agile principles and practices like small team sizes, minimization of documentation and reliance on the knowledge of individuals limit the potential of a company to fully exploit the three basic Open Innovation processes of out-side in, inside-out, and coupled. Minimization of documentation and

the resource-intensive communication with stakeholders results in humans turning out as the bottleneck for open knowledge sharing within a broader network.

Table 1

Challenges and issues when opening up Agile Development via virtual communities

Challenge/Issue	ISE [1)]
Agile Development vs. Open Innovation	
Minimization of documentation and focus on individuals as knowledge carriers limit inside-out/coupled processes (cf. Cao & Ramesh, 2008; Gassmann & Enkel, 2004; Munir et al., 2016)	S, P
Small team sizes limit absorptive capacity and outside-in process (cf. Cao & Ramesh, 2008; Cohen & Levinthal, 1990; Gassmann & Enkel, 2004; Munir et al., 2016)	S, P, O
Minimization of documentation causes human bottlenecks in knowledge sharing as main principle of Open Innovation (cf. Conboy & Morgan, 2011; Savolainen et al., 2010)	S, P
Increasing instability through extensive requirements changes and reprioritizations caused by early integration of end-users in outside-in-process (cf. Cohen & Levinthal, 1990; Daneva & Pastor, 2016; Reichwald & Piller, 2009)	P
NIH-Syndrome counteracts absorptive capacity and outside-in process (cf. Cohen & Levinthal, 1990; Katz & Allen, 1982)	C
Virtual Communities vs. Agile Development	
Move to online artifacts counteracts focus on physical artifacts as major agile practice (cf. Sillitti & Succi, 2005)	P
Move to online, time-/location independent communication counteracts on-site and local communication as major agile practices (cf. Sharp & Robinson, 2008; Sillitti & Succi, 2005)	P
Growing, distributed knowledge base challenges agile principle of knowledge sharing in small teams (cf. Knauss et al., 2014; Munir et al., 2016)	P
Solely/increasing use of online artifacts for requirements limits exploitation of stakeholder collaboration (cf. Laurent & Cleland-Huang, 2009)	P
Solely/increasing use of online artifacts for requirements limits support for user-side feature prioritization (cf. Laurent & Cleland-Huang, 2009)	P
Small agile team size creates tendency to join existing instead of creating new communities (cf. Dahlander & Magnusson, 2008)	S
Community growth reduces importance of customer as main agile principle (cf. Conboy & Morgan, 2011)	S, P, C
Community growth diminishes developer innovation as an effect of scaling-up (cf. Moe et al., 2012)	S, O
Virtual Communities vs. Open Innovation	
Measures to maintain community stability compromise the network's ability to rapidly adapt to environmental changes (cf. Knauss et al., 2014; Linåker et al., 2016)	S, O
Attracting/maintaining community requires appropriate management of intellectual property (cf. Dahlander & Magnusson, 2008)	S
Solely online artifacts for requirements limit vendor-user communication and user inclusion in requirements elicitation (cf. Laurent & Cleland-Huang, 2009)	P, O
Solely online artifacts for requirements limit feedback and status updates to contributors (cf. Laurent & Cleland-Huang, 2009)	P,O
Low abstraction level of information representation leads to knowledge encapsulation in experts' closer networks (cf. Knauss et al., 2014)	P, O

[1)] Innovation System Elements: S(trategy), P(rocess), O(rganization), C(ulture), L(earning), M(easures), I(ncentives)

Source: Own compilation

With the Open Innovation approach to gain momentum, the huge number of requirements emerging from consumers, end-users and other network players could lead to increasing instability for all players, because of increased change and re-prioritization efforts. Further,

independently from the specifics of Agile Development, psychological factors like the 'Not-Invented-here'-syndrome might hinder the company to fully exploit input from the outside.

Regarding the relationships and effects between virtual communities and Agile Development, the dominance of online artifacts and online communication limit the advantages of fundamental agile principles and practices like physical artifacts and onsite, personal communication. Further, the growing, spread and only virtually represented knowledge might overwhelm a small agile team. The move to online artifacts also limits collaboration with and among stakeholders and lacks provision of support for consolidation and joint prioritization of feature and requirements requests among the external stakeholders. Weighting the envisioned efforts for reaching and maintaining a virtual community's critical mass against the possibilities of small agile teams often results in a decision to join an existing community instead of creating a new one specific to the company's business objectives. Assuming the opening-up to be successful, the increasing number of new stakeholders and stakeholder types to deal with might decrease the customer focus inherent to agile principles and, if the company reacts with scaling-up, developer innovation as one of the drivers of agile innovation might decrease.

Concerning potential effects between the use of virtual communities and Open Innovation, the analysis of virtual networks shows that there is a certain need for key players to provide direction and stability. However, such players have the potential to compromise the ability of the network to adapt quickly to changes in the environment. In addition, the appropriate management of intellectual property with respect to the community plays an important role. The move to online artifacts, limits from a vendor perspective the communication with users and their inclusion in requirements elicitation, while on the other side contributors within the network might not get the expected feedback, updates, or recognition. Finally, the quality and abstraction level of information and its dispersed online representation might result in the need of experts to interpret and differentiate such information, which creates a tendency of accumulation and encapsulation of the respective knowledge by these experts and their closest network contacts, compromising the core ideas underlying truly Open Innovation.

Analysing the above-identified relationships and effects, one can derive that applying the concept of Open Innovation to Agile Development and using virtual communities as a basic means for realization is not a self-enforcing combination, but requires decisions with respect to several trade-offs with the following appearing to be most important ones:

(1) Defining the appropriate degree of openness to increase a company's innovation potential vs. losing competitive advantages.
(2) Balancing the degree of agility vs. the innovation potential resulting from Open Innovation.
(3) Managing the appropriate mix of sources for requirements and ideas by balancing the influence from key players and the contributions from a broader stakeholder network.
(4) Identifying the appropriate level of detail and abstraction of documentation and overall knowledge representation to balance the effectiveness of internal agile processes with efficient and effective community communication and collaboration.

Innovation System Customization Approach

Above insights into potential challenges and trade-offs when applying the Open Innovation approach to Agile Development by virtually integrating stakeholders allow drawing basic decisions along the first three of six steps of a design science-based approach for the development of an innovation system customization approach (cf. Research question 2).

Regarding the first step of *problem identification and motivation*, the analysis of challenges and trade-offs provides essential insights into the problem to be solved. While the main motivation

is to support companies applying Agile Development with a structured approach to customize their innovation management system when opening, the underlying problem turns out as quite challenging and complex. While Open Innovation is generally expected to foster innovation in Agile Development, the detailed view on Table 1 shows several particularly negative cause-effect relationships in both directions between the elements of Open Innovation, Agile Development, and virtual communities. Customizing an innovation system in this context is thus much more about identifying and evaluating respective trade-offs than a straightforward decision-making exercise and involves strategic decisions by the company.

With respect to the second step of defining the *objectives for a solution* the overall objective can be set as supporting companies applying Agile Development with a structured approach to customize Open Innovation based on virtual stakeholder integration according to their business objectives and business environment. More detailed objectives comprise efficient and effective applicability of the framework and respective methods, ease of understanding and applicability by non-innovation management experts, and coverage of a wide range of business types and business sizes in the domain of product and/or service development.

Concerning the third step of *design and development* the insights from the analysis carried out in the previous section suggest several key design decisions:

(1) Simply enhancing the existing innovation system customization framework for Software Product Line Engineering by e.g., including Agile Development characteristics appears no longer a meaningful option. The perceived complexity added by virtual collaboration and the substantial differences between the concepts of classical innovation and Open Innovation suggest the provision of a dedicated framework.

(2) The evaluation of several trade-offs and respective decision-making appears to clearly exceed the scope and responsibilities of core innovation management and affect the overall business model and strategy of a company. Therefore, the enhancement of the method for the application of the customization framework as laid out by Stallinger et al. (2014) by an up-front method for trade-off determination and strategic decision-making is suggested.

(3) An extension of the innovation management elements dimension (cf. Figure 3) comprising e.g., Innovation Strategy, Organizational Structure, and Innovation Process by a further element to capture requirements on the collaboration and communication infrastructure and particularly respective knowledge representation appears necessary.

(4) With respect to the criteria used to characterize and differentiate businesses no extensions at the level of categories appear necessary, but extension of the characteristics within certain categories, e.g., an extension in the Customers and Market category to cover ecosystem and respective stakeholder characteristics, or an extension in the Engineering and Production category to cover properties of the agile process.

(5) A subset of these business characteristics or respectively the values they can take has to serve to systematically link the results of the method for trade-off determination and strategic decision-making (cf. (2)) to the core customization framework.

The proper step of developing the framework (cf. Research question 3) exceeds the scope of the present paper and comprises systematically gathering 'good practices' at the intersection of Open Innovation, Agile Development, and virtual communities for abstraction and inclusion as innovation system requirements in the framework. Part of these practices can serve as input for the method for trade-off determination and strategic decision-making, e.g.:

- analysis of the basic Open Innovation strategies (i.e. accessing, aligning assimilating) in order to determine an adequate combination of these strategies and appropriate collaboration and communication strategies (cf. Dahlander & Magnusson, 2008), or
- opening-up only partly to lead users to limit the number of stakeholders an agile team has to collaborate with at the cost of a reduction of innovation potential (cf. P. Näkki et al., 2011).

Other practices can serve as requirements on innovation system elements for the core customization framework, e.g.:

- set up adequate and transparent decision making tools and processes for identifying innovation critical information in order not to get overwhelmed with ideas and suggestions (cf. Dahlander & Magnusson, 2008), or
- segregate requirements processes and refine prioritization methods in order not to get overwhelmed with ideas and to understand the multiple types of requirements artefacts and communication structures in the network of distributed knowledge (cf. Wnuk et al. (2012).

Particularly the innovation system requirements part of the framework is envisioned as a living artefact, to reflect best practices increasingly and continuously.

Conclusions

The research presented in this paper addresses the challenges that organizations applying Agile Development face when applying the concept of Open Innovation to lift limitations on their innovation potential resulting from a mere customer focus. The Open Innovation approach strives for exploiting the innovation potential of the whole ecosystem of a company and puts a special focus to virtual communities as a facilitator for Open Innovation.

Based on an analysis of requirements practices and innovation in Agile Development, the concept of Open Innovation, and of stakeholder integration via virtual communities in the fields of Open-Source Software Development and of Software Ecosystems, major challenges for opening-up Agile Development via virtual communities are identified. A deeper analysis of these challenges shows that beyond 'simple' challenges it is particularly the interplay of these challenges in the triangle of Open Innovation principles, Agile Development principles and practices, and virtual community and virtual collaboration peculiarities that creates a series of trade-offs. These trade-offs must be carefully evaluated, and respective strategic decisions made. Sample trade-offs relate e.g., to defining the appropriate degree of openness without losing competitive advantages, to balancing the degree of agility vs. the potential resulting from Open Innovation, or to balancing the influence of key players and community leaders against that of a broader user and stakeholder network while maintaining network health.

The results obtained are used to derive basic design decisions for the adaptation of an innovation system customization framework for Software Engineering Product Line Businesses to the context of Open Innovation via virtual communities in Agile Development, particularly for the context of product and/or service development. A basic decision is suggested to provide a separate and dedicated framework for the combination of Open Innovation with Agile development via virtual communities. Beyond that, major conceptual adaptations suggested referring to the need for provision of an up-front method for trade-off determination and strategic decision-making and to an extension of the innovation management elements dimension by a further element to capture requirements on the collaboration and communication infrastructure and knowledge representation. Further, several extensions of the business characteristics within certain characteristic categories appear necessary, e.g., extensions to cover ecosystem and respective stakeholder characteristics, or to cover properties of the applied agile process.

With respect to limitations, the paper mainly follows the claim by Conboy and Morgan (2011) that in order to understand the relationship between Agile Development and Open Innovation particularly the requirements practices of both worlds have to be analysed and understood. In turn, this might imply that the findings presented here apply more to the concept of incremental than of radical innovations. Further, the implicit assumption underlying the paper

with respect to Agile Development is 'agile in the small'. Further research would be necessary to elaborate the interplay of Open Innovation with Scaled Agile settings.

As an outlook to the proper step of developing the framework by systematically capturing good practices, examples of practices and strategic decision making are provided as identified in course of the literature work. These examples indicate a tendency towards rather traditional and classical approaches where stability, planning, and documentation play an important role.

Acknowledgment
The work on this paper was carried out within the project 'Multi-Project Management and Integrated Management' at the University of Applied Sciences BFI Vienna, funded by the City of Vienna/Austria.

Funded by City of Vienna Economic Affairs, Labour and Statistics

References

Allen, R. C. (1983). Collective invention. *Journal of Economic Behavior & Organization*, *4*(1), 1–24. https://doi.org/10.1016/0167-2681(83)90023-9

Alspaugh, T. A., & Scacchi, W. (2013). Ongoing software development without classical requirements. In *2013 21st IEEE International Requirements Engineering Conference (RE)* (pp. 165–174). Piscataway, NJ: IEEE. https://doi.org/10.1109/RE.2013.6636716

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., . . . Thomas, D. (2001). Manifesto for Agile Software Development. Retrieved from https://agilemanifesto.org/

Cao, L., & Ramesh, B. (2008). Agile Requirements Engineering Practices: An Empirical Study. *IEEE Software*, *25*(1), 60–67. https://doi.org/10.1109/MS.2008.1

Chamberlain, S., Sharp, H., & Maiden, N. (2006). Towards a Framework for Integrating Agile Development and User-Centred Design. In P. Abrahamsson, M. Marchesi, & G. Succi (Eds.), *Lecture notes in computer science, 0302-9743: Vol. 4044. Extreme programming and agile processes in software engineering: 7th international conference, XP 2006, Oulu, Finland, June 17-22, 2006, proceedings* (Vol. 4044, pp. 143–153). Berlin: Springer. https://doi.org/10.1007/11774129_15

Chesbrough, H. W. (2011). *Open innovation: The new imperative for creating and profiting from technology* ([Repr.]). Boston, Mass.: Harvard Business School Press.

Cohen, W. M., & Levinthal, D. A. (1990). Absorptive Capacity: A New Perspective on Learning and Innovation. *Administrative Science Quarterly*, *35*(1), 128. https://doi.org/10.2307/2393553

Conboy, K., & Morgan, L. (2011). Beyond the customer: Opening the agile systems development process. *Information and Software Technology*, *53*(5), 535–542. https://doi.org/10.1016/j.infsof.2010.10.007

Dahlander, L., & Magnusson, M. (2008). How do Firms Make Use of Open Source Communities? *Long Range Planning*, *41*(6), 629–649. https://doi.org/10.1016/j.lrp.2008.09.003

Daneva, M., & Pastor, Ó. (Eds.) (2016). *LNCS sublibrary. SL 2, Programming and software engineering: Vol. 9619. Requirements engineering: Foundation for software quality : 22nd International Working Conference, REFSQ 2016, Gothenburg, Sweden, March 14-17, 2016, Proceedings*. Switzerland: Springer. https://doi.org/10.1007/978-3-319-30282-9

Davila, T., Epstein, M. J., & Shelton, R. D. (2013). *Making innovation work: How to manage it, measure it, and profit from it* (Updated ed.). Upper Saddle River, N.J.: FT Press.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

Gassmann, O., & Enkel, E. (2004). Towards a Theory of Open Innovation: Three Core Process Archetypes. In *R&D Management Conference (RADMA) 2004,* Lisbon. Retrieved from https://www.alexandria.unisg.ch/publications/274

Hauschildt, J., Salomo, S., Schultz, C., & Kock, A. (2016). *Innovationsmanagement* (6., vollständig aktualisierte und überarbeitete Auflage). *Vahlens Handbücher*. München: Franz Vahlen.

Herzog, P. (2008). *Open and closed innovation: Different cultures for different strategies* (1st ed.). *Betriebswirtschaftliche Studien in forschungsintensiven Industrien*. Wiesbaden: Gabler Edition Wissenschaft.

Hevner, A. R., & Green, G. C. (2000). The successful diffusion of innovations: guidance for software development organizations. *IEEE Software*, *17*(6), 96–103. https://doi.org/10.1109/52.895175

Huizingh, E. K.R.E. (2011). Open innovation: State of the art and future perspectives. *Technovation*, *31*(1), 2–9. https://doi.org/10.1016/j.technovation.2010.10.002

Jansen, S., Brinkkemper, S., & Finkelstein, A. (2013). Business network management as a survival strategy. In S. Jansen, M. A. Cusumano, & S. Brinkkemper (Eds.), *Software ecosystems: Analyzing and managing business networks in the software industry*. Cheltenham: Edward Elgar. Retrieved from https://www.elgaronline.com/view/edcoll/9781781955628/9781781955628.00009.xml

Katz, R., & Allen, T. J. (1982). Investigating the Not Invented Here (NIH) syndrome: A look at the performance, tenure, and communication patterns of 50 R & D Project Groups. *R&D Management*, *12*(1), 7–20. https://doi.org/10.1111/j.1467-9310.1982.tb00478.x

Kitchenham, B. A., Budgen, D., & Pearl Brereton, O. (2011). Using mapping studies as the basis for further research – A participant-observer case study. *Information and Software Technology*, *53*(6), 638–651. https://doi.org/10.1016/j.infsof.2010.12.011

Knauss, E., Damian, D., Knauss, A., & Borici, A. (2014). Openness and requirements: Opportunities and tradeoffs in software ecosystems. In *2014 IEEE 22nd International Requirements Engineering Conference (RE)* (pp. 213–222). Piscataway, NJ: IEEE. https://doi.org/10.1109/RE.2014.6912263

Kordon, F. (2017). *Open Innovation in Agile Systems: Four problem dimensions impeding the integration of external stakeholders* (Bachelor Thesis). University of Applied Sciences BFI Vienna, Vienna.

Laurent, P., & Cleland-Huang, J. (2009). Lessons Learned from Open-Source Projects for Facilitating Online Requirements Processes. In M. Glinz & P. Heymans (Eds.), *Lecture notes in computer science, 0302-9743: Vol. 5512. Requirements engineering: Foundation for software quality: 15th International Working Conference, REFSQ 2009, Amsterdam the Netherlands June 2009, proceedings* (Vol. 5512, pp. 240–255). Berlin: Springer. https://doi.org/10.1007/978-3-642-02050-6_21

Linåker, J., Regnell, B., & Munir, H. (2015). Requirements engineering in open innovation: a research agenda. In D. Pfahl, R. Bendraou, R. Turner, M. Kuhrmann, R. Hebig, & F. Maggi (Eds.), *ICPS: ACM international conference proceeding series, 2015 International Conference on Software and Systems Process (ICSSP)* (pp. 208–212). New York, NY: Association for Computing Machinery, Inc. https://doi.org/10.1145/2785592.2795370

Linåker, J., Rempel, P., Regnell, B., & Mäder, P. (2016). How Firms Adapt and Interact in Open-Source Ecosystems: Analyzing Stakeholder Influence and Collaboration Patterns. In M. Daneva & Ó. Pastor (Eds.), *LNCS sublibrary. SL 2, Programming, and software engineering: Vol. 9619. Requirements engineering: Foundation for software quality : 22nd International Working Conference, REFSQ 2016, Gothenburg, Sweden, March 14-17, 2016, Proceedings* (Vol. 9619, pp. 63–81). Switzerland: Springer. https://doi.org/10.1007/978-3-319-30282-9_5

Moe, N. B., Barney, S., Aurum, A., Khurum, M., Wohlin, C., Barney, H. T., . . . Winata, M. (2012). Fostering and Sustaining Innovation in a Fast Growing Agile Company. In O. Dieste, A. Jedlitschka, & N. Juristo (Eds.), *LNCS sublibrary. SL 2, Programming and software engineering: Vol. 7343. Product-focused software process improvement: 13th International Conference, PROFES 2012, Madrid, Spain, June 13-15 2012 : proceedings* (Vol. 7343, pp. 160–174). Heidelberg: Springer. https://doi.org/10.1007/978-3-642-31063-8_13

Munir, H., Wnuk, K., & Runeson, P. (2016). Open innovation in software engineering: a systematic mapping study. *Empirical Software Engineering*, *21*(2), 684–723. https://doi.org/10.1007/s10664-015-9380-x

P. Näkki, K. Koskela, & M. Pikkarainen (2011). Practical model for user-driven innovation in agile software development. In K.-D. Thoben (Ed.), *17th International Conference on Concurrent Enterprising (ICE), 2011: 20-22 June 2011, Aachen, Germany* (pp. 1–8). Piscataway, NJ: IEEE.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. In *Electronic Workshops in Computing, 12th International Conference on Evaluation and*

Assessment in Software Engineering (EASE). BCS Learning & Development. https://doi.org/10.14236/ewic/EASE2008.8

Reichwald, R., & Piller, F. T. (2009). *Interaktive Wertschöpfung: Open Innovation, Individualisierung und neue Formen der Arbeitsteilung* (2., vollständig überarbeitete und erw. Aufl.). *Gabler Lehrbuch*. Wiesbaden: Gabler Verlag / GWV Fachverlage, Wiesbaden.

Savolainen, J., Kuusela, J., & Vilavaara, A. (2010). Transition to Agile Development - Rediscovery of Important Requirements Engineering Practices. In *Proceedings of the 2010 18th IEEE International Requirements Engineering Conference* (pp. 289–294). Los Alamitos, Calif.: IEEE Computer Society. https://doi.org/10.1109/RE.2010.41

Sharp, H., & Robinson, H. (2008). Collaboration and co-ordination in mature eXtreme programming teams. *International Journal of Human-Computer Studies*, *66*(7), 506–518. https://doi.org/10.1016/j.ijhcs.2007.10.004

Sillitti, A., & Succi, G. (2005). Requirements Engineering for Agile Methods. In A. Aurum & C. Wohlin (Eds.), *Engineering and managing software requirements* (pp. 309–326). Berlin, London: Springer. https://doi.org/10.1007/3-540-28244-0_14

Slaughter, S. (1993). Innovation and learning during implementation: a comparison of user and manufacturer innovations. *Research Policy*, *22*(1), 81–95. https://doi.org/10.1016/0048-7333(93)90034-F

Stallinger, F., & Neumann, R. (2013). A Framework for Innovation System Customization for Product Line-based Software Businesses. In O. Demirors & O. Turetken (Eds.), *Proceedings 39th Euromicro Conference Series on Software Engineering and Advanced Applications: SEAA 2013 : 4-6 September 2013, Santander, Spain* (pp. 94–97). Piscataway, NJ: IEEE Computer Society, Conference Publishing Services.

Stallinger, F., Neumann, R., & Schossleitner, R. (2014). Assessment-Based Innovation System Customization for Software Product Line Organizations. In A. Mitasiunas, T. Rout, R. V. O'Connor, & A. Dorling (Eds.), *Software Process Improvement and Capability Determination* (pp. 131–143). Cham: Springer International Publishing.

Thom, N. (1980). *Grundlagen des betrieblichen Innovationsmanagements* (2., völlig neu bearb. Aufl.). Königstein/Ts.: Hanstein.

Wnuk, K., Pfahl, D., Callele, D., & Karlsson, E.-A. (2012). How can open source software development help requirements management gain the potential of open innovation. In P. Runeson, M. Höst, E. Mendes, A. Andrews, & R. Harrison (Eds.), *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement: (ESEM 2012) Lund, Sweden, 20-21 September 2012* (pp. 271–279). Red Hook, NY: Curran Associates. https://doi.org/10.1145/2372251.2372301