All Hands to the Pumps: The Product Owner Role in Small Companies

Julian M Bass[2], Sarah Beecham[1], John Noll[3], and Mohammad Abdur Razzak[1]

[1] Lero, the Irish Software Research Centre, University of Limerick, Ireland
{abdur.razzak,sarah.beecham}@lero.ie
[2] University of Salford, The Crescent, Salford, M5 4WT, UK
j.bass@salford.ac.uk
[3] University of East London, University Way, London, E16 2RD, UK
j.noll@uel.ac.uk

Abstract. The Product Owner role in scrum is critical for successful project outcomes. We want to understand how job activities within the Product Owner role might vary between smaller companies and large multi-national enterprises. To address this question we undertake a case study of the Product Owner role in an Irish SME.
Using data from observations and interviews, we compared Product Owner activities to those nine identified in an empirical study by Bass of Product Owners in large multinationals.
We observed six of the nine activities, and three new activities associated with the Product Owner. To validate these findings, members of the case study company independently assessed the frequency with which they perceived Product Owners perform the twelve tasks.
Our study corroborates the importance of product grooming, requirements prioritisation, communicating requirements and release management within the Product Owner role, regardless of company size.
Architecture, governance and risk assessment activities observed in large-scale enterprises were absent from our case study company. Whereas, Product Owners in the studied team also engage in Customer Relationship Management, serve as Gatekeepers for story acceptance, and occasionally perform quality assurance and testing.
We hypothesize that, while Product Owners in large companies can specialize and focus exclusively on Product Owner tasks, Product Owners in small companies may be required to perform multiple activities outside the conventional Product Owner role. The implication of this hypothesis is that small organizations who ask Product Owners to perform extraneous and conflicting activities, may ultimately compromise the overall product strategy.

Key words: agile, scrum, inter-team coordination, product owner role, SME

1 Introduction

Product Owners play a critical role in the success of agile projects [1, 2]. The Product Owner role has been well-studied in large companies, and in distributed software development contexts [4, 3].

We explore whether product owners in small companies perform the same activities as in larger companies. Product Owners in smaller companies might be called upon to perform a wider range of tasks than would normally comprise the Product Owner role, such as technical support or customer site preparation, simply because there are fewer staff available that can specialize in these activities. This phenomenon has been observed for other roles, such as when developers are also expected to handle technical support [5]. We call this lack of specialization in small companies the "all hands to the pumps" hypothesis.

On the other hand, larger organisations have more employees, as well as more tightly defined organisational structures; as a consequence, staff, including Product Owners, can specialise and focus on a narrower range of activities. Also, Product Owners in *larger* enterprises might be called upon to undertake a wider range of *product owner specific* tasks, because of the size and scale of development programmes, which need closer oversight and more careful governance [3] Coordination of cooperating agile teams imposes restrictions on architectural innovation within teams [6]. Thus, teams must be encouraged to abide by architectural guidelines and standards set outside the team. These governance and architectural compliance activities can be seen as part of a wider product owner *team* perspective on product ownership. We call this the "specialist Product Owner" hypothesis.

This study attempts to answer the following research question: *How do the activities performed by Product Owners in small- and medium-size enterprises (SMEs) compare to those performed by Product Owners in large multi-national companies?*

In order to explore this question, we have undertaken a case study of a small software development company. Our study draws on earlier studies of product ownership in large enterprises [3].

We discovered that the Product Owners performed six of nine activities identified by Bass [3] in his study of Product Owners in large multi-national companies. This would appear to be consistent with the "specialist product owner" hypothesis. Three additional activities were observed that did not appear in Bass's original list: "Gatekeeper," "Tester," and "Customer Relationship Manager" (see Section 4). Two of these additional tasks ("Tester" and "Customer Relationship Manager") would appear to be outside the scope of the conventional definition of the product owner role, supporting the "all hands to the pumps" hypothesis.

The implication of this hypothesis is that small organizations must be careful about which activities they ask Product Owners to perform. For example, Product Owners who also do technical support were observed to give stories associated with customer issues higher priority, sometimes in conflict with the higher-level product strategy. As a result, the team may fail to meet release

targets due to missing or incomplete features. Also, companies must be careful when appointing Product Owners from other parts of the organization: a Product Owner must have sufficient experience and authority to make a convincing case for senior technical team members when prioritizing the product and sprint backlogs.

The rest of the paper is structured as follows: a discussion of previous research addressing product ownership in agile projects is presented next. This is followed by a description of the research methods we have adopted. In Section 4, we describe the findings of our study, which in turn is followed by a discussion of these findings in Section 5. Finally, we provide conclusions and possible future directions in Section 6.

2 Previous Research in Agile Product Ownership

In extreme programming an on-site customer is advocated to represent client perspectives and be available to the team on a full-time basis [7]. On-site customers have worked with development team members on user experience design, such as providing detailed evidence to support the development of personas [8]. On-site customers have been reported to work in pairs to help each other make tough decisions, bounce ideas off each other and sanity check interpretations of meetings [9]. On-site customers also typically conduct preparation in advance of the first iteration to set the direction for the team and create an overall vision for the product [10]. The on-site customer role has also been seen as a source of political advice to help navigate organisational politics [11]. Finally, on-site customers also help ease communication between technical and non-technical specialisms in organisations.

The Product Owner role is also one of the roles explicitly identified in Scrum [1, **?**]. Product Owners play an important role in the overall software development process [12], and are responsible for communication between the customer and development teams [13]. Product Owners develop and maintain the product backlog, a list of user stories defining requirements for the project. However, Product Owners are not always knowledgeable regarding best practice in requirements engineering [14].

In larger scale agile projects, Product Owners organise themselves into teams [3]. Product Owner teams are required where project sponsors need additional human resources to properly support a larger number of cooperating teams. In some Product Owner teams, area Product Owners are each given responsibility for a subset of product features and report to an overall Product Owner [15]. In contrast, some proxy Product Owners operate a shared responsibility model. Both approaches offer both strengths and weaknesses [15]. While a broader conception of Product Owner teams, sees technical and governance issues added to the Product Owner team responsibilities in order to coordinate cooperating agile teams [3].

In the Scaled Agile Framework (SAFe) the Product Owner is a member of a development team responsible for team user stories and prioritising the team

backlog [16]. A product manager is responsible for the overall product backlog in SAFe. The Product Owner is co-located with the team and acts as a customer proxy.

2.1 Skills Framework for the Information Age

The most widely used and sophisticated taxonomy of IT skills is the Skills Framework for the Information Age (SFIA), formally established in 2000 and now in Version 6, and adopted in nearly 200 countries [17]. SFIA comprises 97 skills along with detailed descriptions of up to seven levels at which the skill might be exercised. Job descriptions, job titles and person specifications can be created by combining skills, at an appropriate level of seniority, from SFIA.

There is no Product Owner role in SFIA. However, the SFIA skill categories most closely associated with product ownership are programme managers, product managers and project managers. Programme managers, at level 7, the highest level in SFIA, are responsible for aligning programme objectives with business objectives and authorising the selection and planning of all related projects and activities. Product management is commonly viewed of as a market facing activity. Product managers in SFIA at level 4 are focused on marketing and customer support at level 5 manage the lifecycle of one or more products acting as an owner or advocate for that product and at level 6, the most senior level for that skill initiate the creation of new products [17].

Project manager skills in SFIA range from Level 7 to Level 4, at level 7, project managers are responsible for planning, scheduling, controlling and reporting activities for strategic, high impact, high risk projects. While project managers at Level 4 are responsible for projects with smaller scope, for example they: typically last less than six months, have limited budget, have limited interdependency with other projects, and no significant strategic impact.

2.2 The Product Owner Role

In an earlier empirical study of Product Owner teams, one of the authors of this paper (Bass), identified nine job activities in the context of large-scale offshore software development programmes: Groom, Prioritiser, Release Master, Technical Architect, Governor, Communicator, Traveller, Intermediary and Risk Assessor [3]. That study argued that Product Owner teams require members with different specialist skills. The skills are combined, in ways that vary in different projects and organisation, to form job descriptions and job titles, such as "Product Owner," "architect," and "proxy product owner." We can now briefly describe each of these job activities that form part of the Product Owner role.

The *Groom* gathers requirements from business clients, running workshops with users and domain experts producing a list of requirements in the form of a product backlog. In scrum requirements are conventionally recorded in the form of user stories written by the groom. The groom is also require to establish and maintain awareness of market trends that may impact the product portfolio.

Awareness of competitor product features and market opportunities and gaps that can be converted to business value by developing new products or features.

The *Prioritiser* ensures that requirements bring value to the business, resolving conflicting demands between project stakeholders. Items in the backlog are initially prioritised and then re-prioritised in advance of planning for each sprint. The prioritiser must have sufficient authority to support, and if necessary override, the demands of key clients, vocal user groups or senior executives in the interests of realising a wider business goal.

The *Release Master* manages and approves release plans. Release masters assess the quality of working and tested code delivered by development teams and ensure quality targets are met prior to customer release. The range of independent features implemented must at least form some minimum viable product in order to fulfil some overall business need. The release master decides at which point the first release is made and when subsequent upgrade or update releases are made.

The *Technical Architect* provides architectural coordination to large scale offshore enterprise software development programmes. the technical architect produces a reference architecture, a working software skeleton with diagrams and documents, that defines the overall architecture style, or software structure for the development programme and constituent products [6]. The technical architect seeks compliance from cooperating teams with the reference architecture and corporate architectural standards. The technical architect also seeks to influence corporate architectural standards by identifying new and emerging technologies, methods and techniques and promoting their value as business enablers where appropriate.

The *Governor* ensures project compliance with corporate guidelines and policies. The governor identifies and applies quality assurance processes ensuring cooperating teams follow guidelines and standards. For example, reviews should be properly conducted with appropriate participants on designated development artefacts at defined points in the development process. Governors also ensure that testing is conducted according to defined plans and processes.

The *Communicator* connects onshore and offshore geographical distribution. This often involves attending video or audio conference calls outside normal office hours, to manage communication between different time zones. The communicator uses formal and information channels to ensure adequate information flows. This may include encouraging and monitoring the use of a range of technologies, such as instant messaging, wikis and other collaboration tools.

The *Traveller*, from an offshore perspective, spends time onshore at client sites, gathering first-hand knowledge of a clients needs. Such visits are sufficiently long, often weeks or months rather than days, to build a network of key informants. Information is collated from sources and disseminated to interested parties involved in the development programme. Travellers also act in a responsive mode, chasing down specific requests for knowledge and disseminating the information gathered.

The *Intermediary* acts as an interface with senior executives, driving large-scale offshore enterprise software development programmes, and disseminating domain knowledge to teams. The intermediary gathers an understanding of, typically board level, vision and business strategy. This vision and strategy is then disseminated to development teams and used to inform requirements prioritisation.

The *Risk Assessor* evaluates technical complexity. Risk categories are identified and reviewed, typically each sprint. Mitigation plans are prepared for each risk and triggered when necessary.

These job activities can be used to construct job descriptions for specific members of the Product Owner team. It is most unlikely that a single individual could successfully complete tasks involved in all the job activities. However, the Product Owner role requires successful completion of all these tasks.

2.3 Software development in large vs small companies

Some previous studies have tended to conflate company size as a variable (such as [18] and [19]). However, organisational context plays an important role in software processes and in software process improvement [20]. The importance of context has also been recognised in other branches of engineering project management [21]. Further, software process tailoring, in practice, is driven by the organisational context [22]. Understanding organisational culture also plays an important part in avoiding communication breakdowns in global virtual teams [23].

Agile development in large vs small companies The most important concerns in using agile methods for large-scale development programmes are scaling, portfolio management, inter-team coordination and architecture [24]. Scaling has been explored in [**?**]. An important emphasis is on more attention to up-front design prior to starting iterations. Inter-team coordination by using a governance framework has been proposed [**?**]. The issue of architecture, in large scale agile, has been addressed by Eckstein [25]. In this context architecture is shared across teams and changes impact on teams in lock step.

Research on agile method use within smaller companies has tended to predominate (for example [**?**]). This research has followed the trend for companies adopting XP towards adoption of Scrum. More recently, teams multi-tasking on multiple project in an SME context has been investigated [**?**].

In summary, company size is likely to impact on how and what practices and activities an agile team is expected to conduct. Since a study on large scale companies has been undertaken on the specific actions performed by the PO, we now consider how well the expectations of the PO role translates to a team operating in an SME context. This leads us to our research question, *How do the activities performed by Product Owners in small- and medium-size enterprises (SMEs) compare to those performed by Product Owners in large multi-national companies?*.

3 Methods

The study employed a participant-observer approach to study a development team in a small software company. Using observations and semi-structured interviews, we compared activities performed by the team's Product Owners to a list of large-company Product Owner activities identified in earlier research by Bass [3]. The result was a list of activities performed by the team under study. These activities were then validated using a survey (see Section 3.4 of the company's project and software development management).

3.1 The Case

The company we studied, which we shall call PracMed, is a medium-sized Irish-based software company that develops practice management software and lab management software for the optical industry. PracMed employs approximately fifty staff members in its software development organization, including support and management staff. PracMed's annual sales approach €50 million, from customers across the British Isles, continental Europe, Scandinavia, and North America. The company has also established a presence in China.

Our study focused on TeamA, whose responsibility is to tailor the company's product for a large customer in North America. The members of TeamA are distributed over four countries in two continents, with up to eight hours difference in timezones between locations. Table 1 shows the distribution of team members; of these, two team members play the Product Owner role, five are developers, one is the QA/Test lead, and one is Project Manager. In TeamA, the Project Manager also plays role of Scrum Master. Also, the Product Owners report to the Product Manager, who is based in Spain and is responsible for the strategic direction of the product.

Table 1. Team Distribution

Country	***Agile Roles***	***No of Team Members***
Ireland	Product Owner	1
	Software Developer	3
	Quality Assurance	1
Canada	Scrum Master	1
	Product Owner	1
	Software Developer	1
USA	Technical Lead (Software Developer)	1
Spain	Product Manager	1

Table 2. Project Management interviewees.

Country	*Roles*	*No of Staff*
Ireland	Project Manager	3
	QA Manager	1
	Senior Database Administrator	1
	Director of Development	1
Canada	Project Manager	1
Poland	Project Manager	1
Spain	Product Manager	1

3.2 Data Collection

We observed TeamA from January, 2016 to December, 2016. Specifically, one of the authors observed TeamA's Scrum ceremonies, including daily standups, sprint planning, backlog grooming, and sprint retrospectives. Due to the fact that the team members are distributed across Europe and North America, the observations were made by joining the video conference session for each ceremony. The same author also conducted semi-structured interviews of each member of TeamA, which were recorded and transcribed. the interview protocol is available from [**?**]. The observer also made contemporaneous hand-written notes during both the ceremony observations and interviews. Finally, the interviewer summarized the interviews using a mind-map, and presented the result to five interviewees in an online workshop to validate the insights gained from the interviews.

Subsequently, three of us interviewed members of PracMed's Project Management staff, including five Project Managers, the QA Manager, Director of Development, Product Manager, and Senior Database Administrator (see Table 2). These interviews followed the same general structure and as the TeamA interviews, but also had two of us observing and taking contemporaneous handwritten notes, while one of us conducted the actual interview.

3.3 Data Analysis

We followed a deductive data analysis approach. First, we examined interview transcripts and observation notes for documentation of activities performed by TeamA's Product Owners. Then, we compared the resulting list of activities to nine Product Owner activities identified by Bass in a study of Product Owners in large software companies. Finally, we created three groups of activities: those from Bass's study that were also performed by TeamA's Product Owners, new activities performed by TeamA's Product Owners that do not appear on Bass's list, and activities on Bass's list that were not performed by TeamA's Product Owners.

3.4 Validation

In order to validate our observations, we conducted a survey of PracMed's Project Management staff, which comprises Product Owners, Scrum Masters, Project Managers, and the Director of Development. Each person was asked to rate each of the Product Owner activities identified by Bass [3] or our observations, on a three point scale according to how frequently they performed an activity ("frequently," "sometimes," or "rarely/never"). In the case where the respondents were not Product Owners, they were asked to asses how frequently they *perceived* their Product Owner colleagues performed an activity.

Respondents were also allowed to say an activity was not applicable, or that they didn't understand the activity. Finally, the survey asked if there were any additional activities that the respondent felt should be added to the list.

4 Findings

Table 3. PO activities

Activities	***Case-Study***	***Large-Scale***	***SAFe***
Groom[a]	✓	✓	✓
Prioritiser[a]	✓	✓	✓
Release Master[a]	✓	✓	✓
Communicator[a]	✓	✓	✓
Traveller[a]	✓	✓	✗
Intermediary[a]	✓	✓	✗
Gatekeeper (Story Acceptance criteria validator)	✓	✓	✗
Technical Architecture Coordinator[a]	✗	✓	✗
Governor[a]	✗	✓	✗
Risk Assessor[a]	✗	✓	✗
Tester	✓	✗	✗
Customer Relationship Manager	✓	✗	✗

[a] These activities were identified by Bass [3].

As discussed in Section 3, one of us observed TeamA performing Scrum ceremonies (daily standup, sprint planning, backlog grooming, and end of sprint demonstration and retrospective). Also, each member of TeamA was interviewed. These observations and interviews suggest that six of the nine Product Owner activities enumerated by Bass [3] were also performed by TeamA's Product Owners.

TeamA's Product Owners were also observed performing two additional activities that were not on Bass's original list: 1. Tester, and 2. Customer Relationship Manager. Our observations also indicated that Product Owners performed the important *Gatekeeper* activity of authorising completed and tested features for release.

4.1 Tester

In PracMed, we have observed that Product Owners also perform testing activities. Most of the tasks comes to the QA to end of the sprint and for that purpose, PO perform QA tasks sometimes. According to one of the TeamA's developers, it does make sense for TeamA's Product Owners to perform testing because they are the ones who know the specifications well.

4.2 Customer Relationship Manager

In PracMed, *Customer Relationship Managers* are responsible for a wide range of tasks, including implementation of new installations, data conversion review, second line support, online training and webinars, server installations, "go-live" pre-checks, trade shows, and general on-site training. PracMed's Product Owners also act as a customer relationship managers, which is not an activity of product ownership per se, but rather an additional responsibility. This can create difficulties; as one Product Owner noted, Customer Relationship Manager tasks are "more technical" so there is a shift in mindset required in addition to divided responsibilities.

4.3 Gatekeeper

Finally, our observations indicated that Product Owners had the final say on whether a feature or story was complete. This activity, which we call *Gatekeeper*, was part of the *Release Master* activity in Bass's catalog. However, in the context of TeamA, where releases are made according to a customer-driven schedule rather than by feature readiness, the determination of whether a feature is complete is a separate event that is mostly disconnected from the release process. As such, we have added the new *Gatekeeper* activity to capture the notion that the Product Owner decides on feature completeness.

4.4 Validation

In order to confirm these insights from observations and interviews, we distributed a survey of the twelve activities shown in Table 3 to ten members of PracMed's Project Management office, which comprises Product Owners, Scrum Masters, and Project Managers, and is headed by the Director of Development. Eight surveys were completed, one was returned incomplete, and one was not returned as of this writing due to the person being on leave.

Of the eight completed surveys, three were submitted by Product Owners, two were submitted by Scrum Masters, and two were submitted by Project Managers, and one by the Director of Development (see Table 4).

Table 5 shows that the members of PracMed's Project Management Office have a similar view of the Product Owner role to our observations: the Project Management Office viewed six of nine original activities (as identified

Participant role	# participants
Product Owner	3
Scrum Master	2
Project Manager	2
Director of Development	1

Table 4. Activity validation survey respondents from PracMed's Project Management Office.

by Bass [3]). This is important as it indicates our observations of TeamA are indicative of the Product Owner role across PracMed's agile teams.

Table 5 also compares responses from Product Owners to the other Project Management Office members. The responses show substantial similarity, indicating that non-Product Owners see the Product Owner performing the same activities as Product Owners feel they are performing.

	frequently			sometimes			rarely			not applicable			#
Activity	PO	Other	Tot	PO	Other	Tot	PO	Other	Tot	PO	Other	Tot	Responses
Groom	1	3	4	2	2	4	0	0	0	0	0	0	8
Prioritizer	2	3	5	1	3	4	0	0	0	0	0	0	8
Release Master	2	0	2	1	4	5	0	0	0	0	1	1	8
Communicator	2	3	5	1	1	2	0	0	0	0	1	1	8
Traveller	0	2	2	3	3	6	0	0	0	0	0	0	8
Intermediary	0	2	2	0	3	3	3	0	3	0	0	0	8
Gatekeeper	3	1	4	0	3	3	0	1	1	0	0	0	8
Tech. Arch. Coord.	0	1	1	0	1	1	3	1	4	0	3	3	8
Governor	1	0	1	0	3	3	2	1	3	0	1	1	8
Risk Assessor	0	1	1	3	3	6	0	1	1	0	0	0	8
Tester	2	1	3	1	4	5	0	0	0	0	0	0	8
Cust. Rel. Mgr.	2	4	6	0	0	0	0	0	0	1	1	2	8

Table 5. Validation survey results.

5 Discussion

In this research we have explored how Product Owner activities in small- and medium-size enterprises (SMEs) compare to those performed by Product Owners in large multi-national companies. Our thinking evolved around two apparently conflicting hypotheses: the "all hands to the pumps" hypothesis and the "specialist Product Owner" hypothesis. All hands to the pumps refers to the lack of job role specialization in small companies. While large enterprises can draw on a bigger pool of staff members, with more focused job descriptions, in the specialist Product Owner hypothesis.

In PracMed, we observed TeamA's Product Owners were performing three additional activities that were not on Bass's [3] original list: 1. Tester, 2. Customer Relationship Manager and 3. Gatekeeper. Testing is not conventionally an activity within the Product Owner role. However, in our study we found Product Owners contributing to test execution. Customer Relationship Management is an important activity observed in our study. Larger enterprises generally have dedicated, specialist customer account managers. However, in this study we found Product Owners taking a hands-on approach to managing relationships with major clients. These findings support the all hands to the pumps hypothesis, which suggests staff in smaller companies are required to turn their hands to a wide range of activities as and when required, due to the smaller overall pool of employees available in the organisation. In PracMed, we also observed TeamA's Product Owners authorising completed features for release. This activity was identified in previous research as part of the Release Master activity in which release plans are defined. However, in PracMed, we have recognised the importance of Product Owners seeing demonstrations of working code and deciding when code is ready to be released, independent of the release schedule. As such, we think it is helpful to separate the future planning of releases from the approval of working code that may be eventually incorporated into a release.

Whereas Product Owners in Paasivaara et al[15] are co-located and sharing experiences from supporting separate project teams and independent features, the Product Owners in PracMed are geographically distributed and collaborating to support the same virtual teams.

Our extensive observations produced no evidence of the Governor, Technical Architect, and Risk Assessor job activities, identified in Bass [3]. The Governor activity is focused on compliance with corporate standards which is a more important issue for larger-scale development programmes. For example, the Governor ensures compliance with corporate process and quality standards in organisations with CMMI Level 5 accreditation. There is a tendency to push architecture away from a client-facing business concern and delegate the issue to technical development teams. We have observed this tendency from this study. However, drawing on evidence from Bass [3] that architecture should be more carefully aligned with the delivery of business needs, we are inclined to view technical architecture is a concern for Product Owner teams as well. The Risk Assessor conducts assessments during each sprint and each release, but was not viewed as a core activity in PracMed.

The Product Owners in PracMed do spend time on the Communicator activity. However, they spent comparatively little time travelling to other sites within the virtual team, possibly because the sites are all captive development centres within the organisation, with a shared organisational culture. The Traveller job activity is perhaps rather more important for outsourcing vendors where travelling to onshore clients is an important aspect of relationship building.

Similarly, the Product Owners in the current study do not seem to spend much time on the Intermediary activity. The presence of two Product Owners

reporting to a product sponsor risks diffusion of decision making, especially regarding story priority.

The Prioritiser, Groom and Sprint Planner are conventional Product Owner job activities identified in the practitioner literature[1]. However, our research revealed some areas of overlap in the assignment of activities to members of the Product Owner team.

6 Conclusions

In this research we investigated Product Owner activities in small- and medium-size enterprises (SMEs) in order to compare with the activities performed by Product Owners in large multi-national companies. We performed a participatory observational case study between January and December, 2016, and conducted semi-structured interviews.

The contribution of this paper is a comparison of job activities within the Product Owner role observed in our case study company, PracMed, with those obtained from a previous study of large scale multi-national enterprises [3]. We assess areas of commonality, and explore in more detail those areas of difference between the job activity taxonomy of Bass [3] and the Product Owner activities actually performed in our case study organization.

We corroborate previous research highlighting the importance of product grooming, requirements prioritisation, communicating requirements and release management job activities within the Product Owner role. There is consensus around the need for sprint planning, participation in demonstrations of working code and retrospectives in all projects. We discovered that PracMed did not perform technical architect, governance and risk assessment activities previously observed in large-scale enterprise development programmes [3].

We discovered that Product Owners in our case study company also engage in Customer Relationship Management, occasionally performing test execution while also serving as Gatekeepers for feature acceptance into working code releases.

Using the evidence from these empirical findings, we hypothesise that Product Owners in small companies must perform multiple activities including some outside the conventional Product Owner role. We call this the “all hands to the pumps” hypothesis. Since, smaller companies have few staff in general, there are fewer specialist staff to draw on, to accomplish tasks. Hence available staff have to undertake a wider range of job activities, sometimes crossing conventional role boundaries. compared with large enterprises. In contrast, large enterprises are able to attract specialists that focus on a narrower set of job activities.

From our “all hands to the pumps” hypothesis we can infer that smaller companies need to pay careful attention to the job activities they expect Product Owners to perform so that the selected activities contribute to the overall business strategy.

7 Acknowledgments

We thank the members of TeamA and members of the Project Management Team for their generous and thoughtful collaboration on this study, and PracMed, for allowing us to study their software development efforts. This work was supported, in part, by Science Foundation Ireland grants 10/CE/I1855 and 13/RC/2094 to Lero - the Irish Software Research Centre (`www.lero.ie`).

References

1. Schwaber, K., Beedle, M.: Agile Software Development with Scrum. Prentice Hall, Upper Saddle River, NJ, USA (2002)
2. Leffingwell, D.: Scaling software agility: Best practices for large enterprises. Addison Wesley, Boston, MA, USA (2007)
3. Bass, J.M.: How product owner teams scale agile methods to large distributed enterprises. Empirical Software Engineering **20**(6) (2015) 1525 – 1557
4. Bass, J.M.: Agile method tailoring in distributed enterprises: Product owner teams. In: 2013 IEEE 8th International Conference on Global Software Engineering, IEEE (2013) 154–163
5. Raninen, A., Merikoski, H., Ahonen, J.J., Beecham, S.: Applying software process modeling to improve customer support processes. Journal of Software: Evolution and Process **27** (2015) 274–293
6. Bass, J.M.: Artefacts and agile method tailoring in large-scale offshore software development programmes. Information and Software Technology **75** (July 2016) 1–16
7. Beck, K., Andres, C.: Extreme Programming Explained. 2nd edn. Addison Wesley, Boston, USA (November 2004)
8. Broschinsky, D., Baker, L.: Using persona with XP at LANDesk software, an avocent company. In: Agile 2008 Conference. 543–548
9. Martin, A., Biddle, R., Noble, J.: Xp customer practices: A grounded theory. In: Agile Conference, 2009. AGILE '09. (Aug 2009) 33–40
10. Martin, A.: The Role of the Customer in Agile Projects. PhD thesis, Victoria University of Wellington, New Zealand (2009)
11. Martin, A., Biddle, R., Noble, J.: The xp customer team: A grounded theory. In: Agile Conference, 2009. AGILE '09. (Aug 2009) 57–64
12. Raithatha, D.: Making the whole product agile: A product owners perspective. In: Proceedings of the 8th international conference on Agile processes in software engineering and extreme programming. XP'07, Berlin, Heidelberg, Springer-Verlag (2007) 184–187
13. Hoda, R., Noble, J., Marshall, S.: The impact of inadequate customer involvement on self-organizing agile teams. Information and Software Technology **53**(5) (May 2011) 521–534
14. Ktata, O., Lévesque, G.: Agile development: Issues and avenues requiring a substantial enhancement of the business perspective in large projects. In: Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering. C3S2E '09, New York, NY, USA, ACM (2009) 59–66

15. Paasivaara, M., Heikkilä, V.T., Lassenius, C.: Experiences in scaling the product owner role in large-scale globally distributed scrum. In: 2012 IEEE 7th International Conference on Global Software Engineering, Los Alamitos, CA, USA, IEEE Computer Society (2012) 174–178
16. Scaled Agile Inc: Product owner scaled agile framework. `http://www.scaledagileframework.com/product-owner/`
17. The SFIA Foundation: Version 6 of the skills framework for the information age SFIA
18. Estler, H.C., Nordio, M., Furia, C.A., Meyer, B., Schneider, J.: Agile vs. structured distributed software development: A case study. Empirical Software Engineering **19**(5) (October 2014) 1197–1224
19. de Cesare, S., Lycett, M., Macredie, R.D., Patel, C., Paul, R.: Examining perceptions of agility in software development practice. Commun. ACM **53**(6) (June 2010) 126–130
20. Dyba, T.: An empirical investigation of the key factors for success in software process improvement. IEEE Transactions on Software Engineering **31**(5) (May 2005) 410–424
21. Bosch-Rekveldt, M., Jongkind, Y., Mooi, H., Bakker, H., Verbraeck, A.: Grasping project complexity in large engineering projects: The TOE (Technical, Organizational and Environmental) framework. International Journal of Project Management **29**(6) (2011) 728–739
22. Campanelli, A.S., Parreiras, F.S.: Agile methods tailoring a systematic literature review. Journal of Systems and Software **110** (2015) 85 – 100
23. Daim, T., Ha, A., Reutiman, S., Hughes, B., Pathak, U., Bynum, W., Bhatla, A.: Exploring the communication breakdown in global virtual teams. International Journal of Project Management **30**(2) (2012) 199–212
24. Dingsøyr, T., Moe, N.: Towards principles of large-scale agile development. In Dingsøyr, T., Moe, N., Tonelli, R., Counsell, S., Gencel, C., Petersen, K., eds.: Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation. Volume 199 of Lecture Notes in Business Information Processing. Springer International Publishing (2014) 1–8
25. Eckstein, J.: Architecture in large scale agile development. In Dingsøyr, T., Moe, N., Tonelli, R., Counsell, S., Gencel, C., Petersen, K., eds.: Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation. Volume 199 of Lecture Notes in Business Information Processing. Springer International Publishing (2014) 21–29
26. Scaled Agile Inc: Scaled agile framework SAFe for lean software and system engineering. `http://www.scaledagileframework.com/`