Investigation on Australian Agile Software Development Organizations: An Exploratory Study of Adaptable SCM Process Implementation

Usman K. Durrani
RMIT University/School of BITL, Melbourne, Australia
Email: usman.durrani@rmit.edu.au

Joan Richardson and John Lenaricic
RMIT University/School of BITL, Melbourne, Australia
Email: {joan.richardson, john.lenarcic}@rmit.edu.au

Abstract—There have been many studies and examples in the literature separately demonstrating the benefits of Agile software development and Software Configuration Management process to software development organizations. There is, however, a general misconception that these two practice domains cannot coexist in a single environment without compromising the values of each other, and there is little reporting on such phenomenon and/or how practitioners can benefit from it.

The aim of this paper is to present the findings of phase 1 of our research and to identify Australian Agile Software Development Organizations having such coexistence. This study will use ‘organization size’ as a lens to analyze implementation variations of Agile and SCM practices and the perceived importance of SCM process in the agile environment. This research as a whole will support practitioners, and IT practitioners in general by providing better understanding of SCM Process in agile environments. It will also contribute knowledge towards theory of lean.

Index Terms— Software Configuration Management, Agile Software Development, Lean Thinking, Adaptable Environment, Value-based

I. INTRODUCTION

Current technological trends reveals the continuous improvement in software development methodologies [1], [2] but on the other end, it also shows that they have not yet reached the desired satisfaction level of the practitioners [3], [4]. Hence, the software engineering process will continue evolving and consensus on any one software development methodology is not possible at present.

Many researchers have already attempted to address whether Agile software development methods are just an ad hoc approach or are truly based on some sound principles [5]. They suggest that, although Agile methods have evolved out of best practice, they are in fact consistent principles as proposed by organizational theories [5].

Study of Software Configuration Management (SCM) practices [4], [6] and Agile software development practices [7], [8] reveal that these are a combination of various interrelated practices and managerial policies. According to [9], regardless of the software development method in use, it is important that it remains under control, and SCM is the method of bringing this control to the software development process.

To allow for the coexistence of such practices, an organization needs to develop an adaptable process. This adaptability comes with the application of Lean Thinking [10] that promotes the culture of implementing only those practices that can add value for the process, project, and organization. But such strategies often come with associated physical and policy constraints [9], [11] and organizations need to overcome these constraints to attain their goals.

The development of an adaptable software configuration management process that promotes the coexistence of Agile and SCM practices is the key area of this research. This study will help to describe and understand the phenomenon of alignment and implementation [12] of Software Configuration Management in the Agile software development environment. A sample population of sixty organizations having both Agile and SCM practices in place and considers both practice domains as important for their software development activities were surveyed.

This paper is structured as follows: In the next section, we present the literature related to this study. Afterwards, we introduce our proposed conceptual framework, followed by a research methodology. We then discuss our results, discussion and conclusion. Last two sections present research contribution and future research.

II. LITERATURE REVIEW
A. Agile Methods and Software Configuration Management Process

Despite continuous efforts to introduce new software development methodologies, it is still very costly due to poor quality (defects, uncontrolled changes, traceability etc.) misaligned process that often results in project failure.

Software Configuration Management is a method of bringing control to the software development process, and is recognized as an inseparable part of quality-oriented product development regardless of the development method. However, SCM is often considered a bureaucratic method; an approach that causes additional work and more documentation [13]. Nonetheless, the value of SCM should not be underestimated in the case of Agile software development methods [9]. The potential for having the engagement of Agile software development methods together with the SCM process exists more in organizations having an adaptable software development environment than in those still using traditional ways of development.

For the purpose of this study, the researcher has investigated agile values and principles [14] along with most widely used agile methods and SCM practices [15].

B. Lean Thinking and Adaptable Environment

Two pillars that supports Lean Thinking are, “continuous improvement”, and “Respect for People” [16]. Continuous improvement also called as “kaizen” which emphasize on “Challenging everything” by creating an atmosphere of continuous learning and embracing changes. Such an environment can only be created where there is a respect for people [16].

In addition to the above lean pillars, three associated principles [10] which facilitate the lean process are: the identification of value, the elimination of waste, and the generation of flow (of value to the customer). These principles, along with others [16], [17] when applied appropriately results in an adaptable process through which various software engineering practices and collaborative management systems can prevail to deliver value.

There were six observations [5] drawn to develop an approach to balance discipline and agility. According to [5], future trends are towards application developments that need both agility and discipline.

For the purpose of this study, researcher will incorporate lean thinking principles as part of conceptual research framework (Fig. 1).

III. CONCEPTUAL FRAMEWORK

For the coexistence of Agile and SCM practices, an organization needs to develop an adaptable process. This adaptability comes with the application of Lean Thinking [10] that promotes the culture of implementing only those practices that can add value for the process, project, and organization. But such implementation often comes with associated physical and policy constraints [9], and organizations need to overcome these constraints for the attainment of their goals (Fig. 1). The development of an adaptable process is the motivation of this study.

Hypotheses for this research are described as follow:

- H1: Medium and small Australian agile software development organizations use agile methods more frequently than the large agile software development organizations.
- H2: Large Australian agile software development organizations use SCM process more frequently than the medium and small Australian agile software development organizations
- H3: Some SCM practices are considered more important than others in Australian agile software development organizations

IV. METHODOLOGY APPROACH

In our research, identification of the respondents was a challenge. This is because Agile software development is still an emerging approach that has gained popularity only in the last seven to eight years. In organizations, often only a few teams are practicing Agile software development along with the Software Configuration Management process. Therefore, it was a challenge to identify the specific respondents who would qualify to respond to our questionnaire. After a considerable amount of research on this issue, the “linkedin” online community was identified for the collection of data.

Since this was an exploratory study, the researcher gathered data from software professionals who have
worked with both Agile methodology and the Software Configuration Management process. This type of sampling method is called judgment sampling, a type of purposive sampling. According to [18], “judgment sampling occurs when a researcher selects sample members to conform to some criterion”.

In order to increase validity of the survey, the researcher also conducted a pilot test survey with a small user group. Based on the feedback of the pilot group, the content of the survey was then updated and full-scale survey was conducted.

The research question (Is there any coexistence of Agile and SCM practices in Australian Agile Software Development Organizations?) requires an exploration of the phenomenon, therefore, descriptive statistics was used for the assessment purpose. Descriptive statistics are commonly used in survey studies [19].

V. RESULT

After the validation and verification of the online survey, over the period of two months, questionnaires were posted to 183 practitioners, mostly from different organizations, selected through personal references. From this mail out, 76 practitioners responded but 16 responses were excluded from further analysis because they did not represented organizations that used both agile and SCM practices; 107 practitioners did not respond.

For this research, “Organization size” variable was selected as a lens to understand its association with the adoption of Agile and SCM practices, and the importance given to both these set of practices. Table I shows the cross tabulation between organization size located in various Australian states.

<table>
<thead>
<tr>
<th>Organization size</th>
<th>NSW</th>
<th>QLD</th>
<th>SA</th>
<th>VIC</th>
<th>WA</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-20</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>21-200</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>6</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>Total</td>
<td>5</td>
<td>9</td>
<td>3</td>
<td>6</td>
<td>24</td>
<td>60</td>
</tr>
</tbody>
</table>

Based on the analysis we identified that regardless of the size of the organization and/or the size of the development team, all the participant organizations did considered tools and technology as either important or very important for their software development activities.

<table>
<thead>
<tr>
<th>Value Categories</th>
<th>Respondent’s Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value for the Project/Products</td>
<td>17</td>
</tr>
<tr>
<td>Value for the Process</td>
<td>16</td>
</tr>
<tr>
<td>Value for the Organization</td>
<td>20</td>
</tr>
<tr>
<td>No Value</td>
<td>7</td>
</tr>
</tbody>
</table>

From Table II, we can infer that SCM process/practices are adding value for the organization, project/products and other software development processes.

We asked the respondents to identify if they had made any customization to the SCM process for alignment with their current software development process in the organization, 39 respondents indicated “Yes”. The remaining 21 respondents selected “No”.

V. RESULT

The resulting means (shown in Table III) as generated through Cronbach’s alpha test reveals that participants had a high mean rating (more than 2.0) on 5 SCM practices out of a total of 8 and for Agile practices this was 15 out of a total of 22. For the research purpose, this high mean rating corresponded to a response of “Often” or “Always”.

The ANOVA test was used to compare the means of different size organizations (small-, medium-, and large-scale) in terms of their implementation of Agile practices. Rather than comparing the means for each of the 22 Agile practices (Table III), the researcher used the technique of combining the weights as given by each respondent (in various organization size categories) against all 22 Agile practices. These combined Agile practice weights were then used to perform the ANOVA test, and the respondents were grouped according to their organization sizes.

Since we have three groups of organizations by size (small, medium and large), the possible combinations for the mean comparison are “Small and Medium”, “Small and Large”, and “Medium and Large”. Levene’s test
(Table IV) for equality of variance was performed prior to ANOVA testing.

| TABLE IV: LEVENE’S TEST FOR EQUALITY OF VARIANCE |
|-----------------|------|------|------|
| Levene Statistic | df1 | df2 | Sig. |
| .602            | 2   | 55  | .551 |

There were no significant results from the ANOVA test between the three groups of organizations, as shown in Table V.

| TABLE V: ANOVA TEST RESULT FOR SIGNIFICANCE |
|-----------------|------|------|------|
| Sum of Squares  | df  | Mean Square | F    | Sig. |
| Between Groups  | 120.899 | 2 | 60.450 | .576 | .565 |
| Within Groups   | 5770.222 | 55 | 104.91 | .602 | .551 |

Therefore, based on the test performed above, the hypothesis H1 is not supported, that “Medium and small Australian agile software development organizations use agile methods more frequently than the large agile software development organizations”. In other words, software development organizations, regardless of their size, consider Agile software development practices important and use these frequently for their software development operations.

Levene’s test for equality of variance was also performed prior to ANOVA testing for H2 (as shown in Table VI).

| TABLE VI: LEVENE’S TEST FOR EQUALITY OF VARIANCE |
|-----------------|------|------|------|
| Levene Statistic | df1 | df2 | Sig. |
| 2.873           | 2   | 55  | .065 |

The ANOVA test between the three organizational groups proved significant, as shown in Table VII.

| TABLE VII: ANOVA TEST RESULT FOR SIGNIFICANCE |
|-----------------|------|------|------|
| Sum of Squares  | df  | Mean Square | F    | Sig. |
| Between Groups  | 128.011 | 2 | 64.006 | 3.271 | .045 |
| Within Groups   | 1076.265 | 55 | 19.568 | .602 | .551 |

Based on the tests performed above, the H2 hypothesis is accepted, that “Large Australian agile software development organizations use SCM process more frequently than the medium and small Australian agile software development organizations”. In other words, large software development organizations consider Software Configuration Management practices comparatively more important than medium and small software development organizations. However, these medium and small software development organizations are gradually realizing the benefits of the SCM process and are adopting selective SCM practices for their operations using value-based mindset.

For the purpose of H3, we had already calculated the means for each of the Software Configuration Management practices and each is ranked according to means score and standard deviation (Table III). However, this does not provide us with any information about whether the means of two successively occurring ranked list of practices are significantly different from each other. To test this, we performed a pair-wise t test, 20 out of 28 pairs met the significance level of less than 0.05.

We observed that the respondents opined that the establishment of version management and baseline for the project and releases is the most important practice of agile software development organizations. This was followed by “coordination of software releases through well established procedures”, and then the “identification of project artifacts at the start of project”, and “execution of well-defined software build procedure for a particular version of software”. Hence, based on the tests performed above, the H3 hypothesis is accepted, that “Some SCM practices are considered more important than others in Australian agile software development organizations”.

VI.DISCUSSION AND CONCLUSION

This quantitative exploratory study investigated the existence of Adaptable Software Configuration Management process in Australian software development organizations. The comparative analysis was performed between various organizations based on their sizes (small, medium, and large) and the practices they performed, along with the frequency and importance. It also explored the different tools and technologies used and their importance for these organizations in different software development areas, such as, programming language, software development IDEs, SCM system, software build management system, software defect/issues system, and project management system etc. In addition to that, various SCM dynamic capabilities were also studied along with the importance of each.

This study has found that Agile methods and the associated practices are considered important and used frequently by all-sized organizations (small, medium, and large). The outcome of this research is in sync with the result of the workshop conducted by [20] with large organizations, in which it was unanimously agreed that Agile methods helped them to become more flexible and adaptive to change [21]. In the researcher’s opinion, more discipline and tool support is required for Agile practices, and they are generally far more prescriptive than traditional development practices [6]. We found that all participants in this research are using various tools and technologies extensively for software build management, software defect and issue management, and release management activities.

Another critical finding of this research is that, although large organizations are still the highest consumers of traditional processes such as SCM process, an increasing number of medium and small organizations are adopting a customized version of SCM process for their specific software development needs. SCM process is one of the key supporting software development processes that provides discipline to Agile methods and helps in the development of capabilities to integrate, build, and reconfigure the competencies for rapidly changing software requirements.

VII.RESEARCH CONTRIBUTION
While the Agile software development methods are gaining more popularity compared with classical software development methods, the evidence of their success has been mostly subjective, and it is very difficult to obtain hard numbers [1], [2]. Similarly, very scant empirical studies exist of Software Configuration Management in Agile methods [9], [23]. By using the quantitative method approach, this study aims to generate empirical evidence to contribute to the body of knowledge in the relevant areas.

On the practical side, this research can also provide support to IT businesses in general, and Software Development Organizations in particular, with the streamlining of the internal operational environment for the facilitation of an adaptable software development environment and the resulting coexistence of value-added Agile and SCM practices.

VIII. FUTURE RESEARCH

There is a need to further investigate how different strategies are used by these organizations in order to develop adaptable software development environments through the application of Lean Thinking. It is also necessary to understand different constraints faced (and resolutions performed) by such organizations during the implementation of Software Configuration Management practices in accordance with Agile values and principles [14][23].

REFERENCES


Mr. Usman K. Durrani is currently doing his PhD from School of Business Information Technology and Logistics RMIT University in Melbourne, Australia. He is a seasoned management consultant, working in the area of software configuration management for more than twelve years. His research interests include agile software development methodologies software configuration management, lean thinking concepts and principles, and management and control aspects of cloud computing.

Associate Professor Joan Richardson has been an academic for the last 20 years. She completed a PhD on ‘An episode of change in a contemporary higher education setting’ and has published widely on IT education issues. She has completed a number of learning and teaching projects funded by Learning and Teaching Infrastructure Funds and was awarded with an Australian Learning and Teaching Council (ALTC) citation in 2010. Dr. Joan also supervises PhD students and is the deputy Head of School responsible for all Learning and Teaching matters.

Dr. John Lenarcic is a physicist and applied mathematician by training, an IT academic by fortunate accident and an armchair philosopher by conscious choice. He currently teaches usability analysis and computer ethics in the School of Business IT and Logistics at RMIT University in Melbourne, Australia. Dr. Lenarcic’s research interest includes social media, linguistic issues in systems development, design thinking in software engineering and the social and ethical implications of information systems.

© 2013 ACADEMY PUBLISHER