Abstract—Developing quality software product is a challenging activity and yet necessary asset for most software companies especially small and medium size companies. For a Software process improvement (SPI) activity to be effective in such companies, the day to day product-focused activities needs to align with SPI activity.

Different process maturity models have been developed, among them the Capability Maturity Model Integration (CMMI) is the most widely used process maturity framework. However, it is considered difficult to implement for small and medium companies due to various reasons related with its difficulty and financial requirement. On the other hand, the recent software development approach is being directed towards application of agile software development methodologies. But some difficulties have also been reported regarding agile methods limitations of long term and continues guide to process improvement.

Researches findings and experience reports demonstrated that, combination of agile methods and CMMI has advantages where most of which can be considered as a solution to problems associated with implementing process maturity activities in immature companies.

In this research in-progress paper an integrated framework that combines agile methods with specific practices of the key process areas in CMMI framework is proposed. The framework can be an alternative guide to formal process improvement through agile methods.

Index Terms—Integrated framework, CMMI and agile methods, institutionalization.

I. INTRODUCTION

The design and development of software is still the most challenging and complex process; where most products suffer from un-attained requirements and fail to maintain the expected quality. Many problems are attributed to failure of software products. Regardless of how it is explained, the set of problems that are encountered in the development of computer software is not limited to software that "doesn't function properly". The affliction encompasses problems associated with how we develop software, how we support a growing volume of existing software, and how we can expect to keep pace with a growing demand for more software [1].

The quality of a software product is considered to be the output of the software process used in the production process. Different software organizations follow varieties of process maturity models to improve the software process to maintain and upgrade the quality of software developed.

Continuous improvement of the software process is considered as a means to guide software developing companies towards standard practices to have a software process with better capability. A standard capability model is created, as a result significant improvement on the quality of the software products, through the introduction of new techniques and new tools, is achieved [2].

A number of software process maturity models have been developed to determine process capability and used as a reference for improvement. Capability Maturity Model (CMM) is developed in Carnegie Mellon University and later extended to CMMI by incorporating more other capability models. The model developed by the International Organization for Standardization (ISO) brought ISO15504 maturity assessment model which is compatible with CMMI. In addition, different models have been developed with the intention of contextualizing the standard models to a certain context. This approach has brought different models such as separate models for small companies, development of national model, and so on.

According to practical observations reported on the approaches, both CMMI and agile methodologies have shortcomings on some settings which can lead to failure in meeting the desired result. On the other hand some studies have shown the chance of getting the synergy in using the agile and CMMI to compensate each other and most of the shortcomings of the two approaches are found to be complementary [3]. According to [4], CMMI can help companies in institutionalizing agile methods consistently in a structured manner to address processes by defining what to address. On the other hand, agile methods have a significant advantage in guiding implementation of CMMI requirements by answering the how question which is not successfully addressed by CMMI framework.

Research findings related with combining some agile methodologies with the Key Process Areas (KPA) in the CMMI framework have demonstrated the two approaches can work together and can also be even better if they are implemented thoughtfully than they are individually. In line with this the attempts made and the results obtained have shown tangible evidence of achievements in software development and SPI through combining CMMI and agile methods.

II. BACKGROUND

The implementation of SPI programs differs according to the priority of parameters considered for a particular context.
Studying the initiatives and their success stories is a valuable input to study and evaluate alternative approach in developing cost effective, simple and context aware SPI framework. In the subsequent subtopics the major SPI implementation strategies with their underlined approach is reviewed.

**CMMI:** Capability Maturity Model for software (CMM-SW) was developed in Carnegie Mellon University in 1991 [5]. This model contains five maturity models with a number of KPAs to be followed in the process of passing to the next higher level. Later on, it is extended to CMMI by incorporating more other capability models. Even though CMMI has been dominantly used as a process maturity framework for a number of years, a number of difficulties reported regarding its successful implementation specially for small and medium companies [6]-[8].

**Agile Methods:** Agile methods are a set of development methods recently introduced to the software development community. The major priority of agile development process is customer satisfaction through adaptive planning, iterative development and integration. The agile manifesto developed by a group of practitioners in 2001 has brought a set of principles with a main principle stated as “We are uncovering better ways of developing software by doing it and helping others do it” [9]. The most widely used agile methods are (Extreme Programming) XP and scrum. The main limitation of agile methods related with difficulties related with long term process improvement.

**Agile and CMMI:** Despite the advantages and opportunities of combining agile methods and standard practices, this research agenda has been overlooked due to the fact that the standard models and agile practices were focusing on being dominant over another one rather than collaborate with each other for better achievement. Recently, the search for a better approach of combining the CMMI and agile methods have been researched by amalgamating an agile practice with some of the KPAs in CMMI. As a result, some encouraging achievements have been found [10], [11]. This can be a very good opportunity for companies which fail to implement standard SPI practices due to reasons of model difficulty, economy and required skill.

The approach taken to assess agile methods using CMMI has shown the practices can work together in selected process areas from CMMI and agile methods [11]. The research has demonstrated how agile practices from agile methods can be introduced in an organization following plan driven development. The results have shown that, there is a relationship between the practices but fully addressing the KPAs of all CMMI maturity levels need further investigation and analysis.

The research done so far have demonstrated, possibilities and advantages associated with the collaboration of CMMI and agile methods. In line with this the research direction focuses on two main areas, adding agility to matured organizations implementing CMMI and introducing CMMI to companies with agile development environment. The research targeting in the two areas only considered companies following one of the two approaches of software development methods and ways to incorporate in either side advantages to the existing practice.

### III. RELATED WORK

Agile and CMMI have been considered incompatible set of practices by practitioners in both sides and referred as not suitable to be implemented together. The research towards close examination on similarities and defining possible complementary relationship is becoming a research topic after some success stories of experience reports on integration of agile and CMMI. Recently the research trend is to get the benefit of both approaches and find a better and thoughtful approach of implementing both Agile and CMMI to collaborate each other and possible approach to develop a unified set of model. The following summary of related works demonstrate the approaches considered, methodologies used, and findings reported by the research with the limitation of each to show the research gap.

- **K. Miler and J. Lukasiewicz** developed a reference model called a C-S model used for mapping specific practices of the second and the third level of maturity in the CMMI Staged representation of the CMMI (V 1.2) model with the activities described by the Scrum methodology [12]. The study considered Scrum to cover some practices of level 2 and 3 staged representation and not demonstrated comprehensive coverage of all the specific practices in the selected levels.

- **M. Pikkarainen and A. Mäntyniemi** presented application of CMMI framework for assessment of agile software development[11]. The research defined some basic relationship between the practices though not a complete model, it only focused on assessment aspect and no process improvement is presented. It is more of fragmented approach of application of CMMI for assessing companies implementing agile methods.

- **Application of scrum practices to the CMMI practices environment to introduce agility to a plan driven environment is an approach demonstrated by Ana Sofia, C. Marçal et al.** The approach was focused on aligning practices from scrum to CMMI project management process areas with assessment of their relationships and differences[13]. But the main contribution of this research is on the project management process areas on CMMI and scrum practices. The process areas other than project management were not considered.

- **Muhammad Ishfaq Khan et al.** propose small companies to adopt Agile practices and transform themselves to a CMM based company with a recommended approach that, expertise in agile will help software developers to adopt CMM [14]. Here the relationship between agile and CMMI practices and clear guidelines on how to implement is not indicated.

- **Z. Lina and S. Dan** presented the relationship between agile methods and CMMI and the benefits of using both approaches to improve efficiency of small and medium size companies. The approach focused on the relationship between scrum and CMMI with the intention of defining a project management framework. The research has presented how software project management can be enhanced through the implementation of scrum under the umbrella of CMMI KPAs [15]. The limitation of the research is, it addresses only the project management aspect of software process without considering other
KPAs in level 2 or 3 in CMMI framework which cannot be addressed by scrum.
• Carsten Ruseng Jakobsen and Jeff Sutherland presented an experience report on implementation of scrum in CMMI level 5 companies and claimed improved efficiency in software development than using the traditional development models, waterfall is used in this context [16]. Agile and CMMI combination is expressed in terms of using CMMI to help an organization institutionalize agile methods. The focus in this practice report however is mainly introducing agile methods specifically scrum practices to the matured organizations with the objective of maximizing software development efficiency. Application of agile methods to implement the early stages of process improvement is not addressed.
• J. Diaz et al. Provide a more comprehensive model complemented with a detailed analysis of relationship between CMMI practices and scrum practices with documented case study [17]. The research only addresses CMMI level 2 practices and the model presents very little additional practices which may help to address all of the CMMI level 2 goals equally well.
• T. Kähkönen and P. Abrahamsson presented concepts about the relationship between XP and CMMI and presented that CMMI level 2 KPAs can be covered by using XP practices [18]. The focus of this research is on application of CMMI for assessment of XP practices and does not present a guideline on how companies can mature their process capability using a unified and integrated set of activities.

According to the review made so far it has been demonstrated that the two paradigms not only have significant similarities but also complement each other. Mainly two approaches have been considered in researches and experience reports regarding combining agile methods and CMMI. The first one is introducing agile methods to organizations having matured process or already achieved some levels in CMMI capability levels. The other approach towards combining these two approaches is application of CMMI for assessing software development process in companies implementing agile methods.

Since the idea is introduced recently, and is coming out of the idea that “agile and CMMI are completely incompatible set of practices”, it needs a thorough investigation in all dimensions to enrich the preliminary yet promising results achieved so far. So far large percentage of the approaches taken is fragmented approach by picking practices from both approaches and considering their complementation to benefit from.

One of the areas which can be worth considering is integration of the two paradigms as a unified framework to be used as an alternative path for companies with low maturity and limited financial capacity. Integrating the practices in a unified model can provide an excellent opportunity especially software companies with low maturity level to benefit from their process improvement at the same time keep the pace on the business by responding to customers’ demand. Such a framework can satisfy the need for product focused approach and can help to define process improvement activity to upgrade process capability, which is considered “best-fit” approach of SPI in companies with skill and financial limitation.

### TABLE I: SAMPLE RELATIONSHIP MATRIX OF CMMI KPAS SPECIFIC PRACTICES AND PRACTICES OF AGILE METHODS

<table>
<thead>
<tr>
<th>CMMI KPAs</th>
<th>Specific Goals</th>
<th>CMMI specific practices</th>
<th>Definition of Specific practices</th>
<th>XP practices</th>
<th>Scrum practices</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPA 3, Project Monitoring and Control (PMC)</td>
<td>SG 1, Monitor the Project Against the Plan</td>
<td>SP 1.1 Monitor Project Planning Parameters</td>
<td>Monitor actual values of project planning parameters against the project plan</td>
<td>Onsite customer</td>
<td>Daily sprint meeting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.2 Monitor Commitments</td>
<td>Monitor commitments against those identified in the project plan.</td>
<td>Pair programming</td>
<td>Daily sprint meeting, Sprint review meeting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.3 Monitor Project Risks</td>
<td>Monitor risks against those identified in the project plan.</td>
<td>Continuous integration, small release</td>
<td>Sprint planning meeting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.4 Monitor Data Management</td>
<td>Monitor the management of project data against the project plan.</td>
<td>Daily sprint meeting, Sprint review meeting</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.5 Monitor Stakeholder Involvement</td>
<td>Monitor stakeholder involvement against the project plan.</td>
<td>Onsite customer</td>
<td>Sprint planning meeting</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.6 Conduct Progress Reviews</td>
<td>Periodically review the project’s progress, performance, and issues.</td>
<td>Daily sprint meeting, Sprint review meeting</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SP 1.7 Conduct Milestone Reviews</td>
<td>Review the project’s accomplishments and results at selected project milestones.</td>
<td>Small release</td>
<td>Sprint review meeting</td>
</tr>
</tbody>
</table>

### IV. INTEGRATED FRAMEWORK OF AGILE AND CMMI

SPI implementation has been tried in various types of approach through contextualization of the standard processes to a particular setting. In most of the researches, “context” is defined in terms of the day-to-day product development activity, product focused than process focused culture, and organization size and financial capability. The application of Integrated framework of agile and CMMI can satisfy the need for product focused approach and increase flexibility of SPI to be contextualize to any setting and define process improvement activity to upgrade process capability. The approach will focus on how to retain knowledge and experience by targeting software development activities in a particular project using agile methods. Integrating unified framework to be used as an alternative path of software process improvement for companies with low maturity and limited financial capacity.
A. Defining the Framework

To exploit the best advantages and opportunities of combining agile methods and standard practices integrating different models as an approach is expected to define a smooth unified model with detailed analysis of relationship between the practices of agile and CMMI. In this research an integrated framework which combines agile methods with key process areas of CMMI framework is developed to guide formal process improvement through agile practices to upgrade process maturity for a better success in software development. The framework uses continuous representation of CMMI V1.3 framework [19] with XP and Scrum development approaches from agile methods. In the continuous representation, the capability improvement is implemented on each KPA regardless of the maturity level it belongs. The framework considers specific practices of each KPA to be addressed by practices of the selected agile methods.

B. Mapping CMMI KPAs to Agile Practices

The relationship between CMMI KPAs and practices from agile methods has been presented through definition of the concepts represented by the practices, case studies and experiments from different researches. Table I shows sample relationship matrix where each specific practice is mapped with the corresponding agile practice which can address it. The definition of each specific practice of the KPAs is directly taken from CMMI v1.3 to keep the context of the specific practices as presented by the CMMI framework. In addition to the specific mapping between the practices the coverage of CMMI practices through implementation of agile methods in general is not indicated. The synergy which can be obtained from this effect could be vague to quantify and present with explicit mapping to specific practices.

V. THE STUDY FRAMEWORK

Using agile methods to develop a software product has advantages in the context of that particular project. Structuring this experience in a predefined set of framework helps an organization to benefit from previous projects through scientifically experimented and validated approaches. Hence, mapping each practice carried out in that particular project with activities predefined based on CMMI process improvement framework will help the organization process improvement to evolve based on the organization context and future plan.

In this research an integrated framework of agile and CMMI for software process improvement is presented through the institutionalization of practices of agile methods based on CMMI framework of software process improvement. Regarding CMMI, the CMMI v1.3 of continuous representation of process improvement is considered due to its flexibility in selecting any KPA at any instance of software development process. The approach can make the software process improvement more product focused and the process improvement can be executed with minimal resource allocation and without losing focus on the daily software development business activity of the organization. Capability improvement of selected KPAs follows institutionalization process defined in the institutional theory. The generic goals and practices of CMMI recommend structures to be institutionalized in the process of advancing through software process improvement.

Software process improvement is institutionalization of practices in a predefined structure through identification of KPAs and associated specific practices. It needs to be implemented based on the organization context or improvement plan or any approach considered relevant by that particular organization. Hence, the approach used in processes of institutionalization can be used in advancing through maturity levels in software process improvement.

Component processes of institutionalization as discussed by [20] consists of Innovation, Habitualization, Objectification and Sedimentation as shown in Fig. 1. Those component processes define specific properties of a particular practice according to the level of institutionalization in a certain context. In addition, the component processes demonstrate ways of institutionalizing a particular structure practice or process.

![Fig. 1. Component processes of institutionalization (Tolbert & Zucker, 1996).](image)

**Innovation** refers to creation of new structure and **Habitualization** deals with the creation of new structural arrangements in response to a specific organizational problem or set of problems, and the formalization arrangements of structures in the routine procedures of that organization. The third process component called **Objectification** imply the diffusion of structure. It involves the development of some level of recognition by decision makers and adoption of a structure, and the increasing adoption by organizations. The final process is **Sedimentation** which implies full institutionalization of a process. It refers to a continuation of a structure in a certain setting as a norm and it is also characterized by its level of fully spread across actors considered as adopters of the structure.

In continuous representation of software process improvement, the capability level for a particular process area passes through 4 levels (incomplete, performed, managed, and defined), in CMMI V1.3. The levels define requirements to be attained by a process area at each level.

![Fig. 2. Capability levels of continuous representation (CMMI V1.3).](image)

Adopting the institutionalization process from the institutional theory to the capability levels of continuous representation of process improvement in CMMI V1.3 is considered in this research as best fit approach as “Integration Bridge”. The institutionalization process components are presented to define the activities to be carried out and
requirements to be achieved in the process of upgrading a particular KPA through the capability levels. The capability improvement of a particular KPA can be achieved through the improvement of capability level of its associated specific practices. Improving capability of specific practices can be implemented through institutionalization of the agile practices used during development of a software project based on their relationship and their level of institutionalization according to the requirements of the specific practices.

The process component of institutionalization used is considered as an approach to institutionalize practices to the capability levels of agile methods through the related practices in KPAs of CMMI. The representation of the modified model as shown in Fig. 3 incorporates the three components of institutionalization excluding Innovation which is the first level. The basic assumption in this regard is innovation deals with creating new approach or structure, but in software process improvement the practices are identified and the framework is also defined hence no new structure is going to be created.

Institutionalizing practices fulfilling CMMI goals at different levels within different types of KPAs can be tracked through measuring how a particular practice from agile development method can address specific practices in a particular KPA. Implementation of specific practices can be a means to achieve the corresponding goals of CMMI KPAs capability levels. Hence, the level of institutionalization of agile practices in relation with the specific practices can be taken as an indicator for the achievement of the specific goals.

Tracking the level of capability of a particular KPA is represented by the capability level of its specific practices. The micro-level KPA capability tracking approach at specific practice level is implemented to meaningfully trace the capability of KPAs which in turn represent the process improvement in a software development company.

The KPA capability improvement tracking model adapted from OWPL micro evaluation framework shown below display hypothetical capability levels of specific practices of a particular KPA and how the capability level of the specific practices can be used to assess the capability level of the parent KPA. As it is discussed earlier, in this framework the capability improvement of a selected KPA is implemented through the improvement of its specific practices capability. As Fig. 4 represents the capability of a particular KPA is achieved through the institutionalization of all specific practices and the level of institutionalization will be determined by the capability levels defined. SP 1 for example is a specific practice in a certain key process area KPAx; this specific practice passes through the capability levels from the first level which is level 0 to level 3. Capability of KPAx can be evaluated by the sum of its specific practices capability levels as indicated in the graph.

VI. RESEARCH METHOD

Case study method is selected for the study for a number of reasons related to the research context. According to Yin, case study approach is a good approach to answer questions related with how and why which are related with operational links to be traced over time [21]. The study will follow case study steps iteratively based on the guides in case study research described by Benbasat as preparing data collection, collecting evidence, analyzing case study evidence and presenting case studies [22].

The research targets SPI through integration of agile practices with CMMI. Once the framework is developed based on the analyzed concepts pertinent to the models selected; specific implication will be expected to be drawn through in depth and rich insight in to the phenomena.

The first phase is a detail analysis of the practices in the models. Taxonomy comparative analysis will be used to
perform detailed analysis of the models considered to take part in the framework.

The main part of the study follows a case study. The approach will be used to evaluate the integrated framework to be developed based on the result of the analysis on the first phase.

Data collection will follow multiple evidence based on the approach suggested by [21]. Data collection instruments including interviews, group discussion, and software development documents will be used to collect required data for the study. The study will be situated in selected ICT incubation centers at Hawassa University and Addis Ababa University, Ethiopia.

VII. CONCLUSION AND FUTURE WORK

Among the success factor for SPI implementation aligning the SPI activity with the daily software development tasks is considered essential. One of the approaches in this regard is combining agile methods and CMMI which has been presented in different approaches. The major consideration in this research focus on relating specific practices of CMMI with practices from agile methods to unearth the close relationship of software process improvement with day to day software development activities. The future steps in this research is evaluating the framework to demonstrate the implications based on practical application of the approach. Once the framework is evaluated detailed analysis of components will be performed to get rich insight of associated implications.

REFERENCES


Tatek Engdashed is a PhD student in the IT program of Addis Ababa University, Ethiopia specializing in software engineering. Tatek has been a lecturer in Institute of Technology Hawassa University, Ethiopia in the Department of Informatics. His research interests include software process improvement, agile methods, software measurement, and process improvement and knowledge management.