# Towards a Hybrid Approach for Software Project Management using Ontology Alignment

Abdelghany Salah Abdelghany
Department of Information Systems and Technology
Institute of Statistical Studies and Research
Cairo University, Egypt

Nagy Ramadan Darwish
Department of Information Systems and Technology
Institute of Statistical Studies and Research
Cairo University, Egypt

Hesham Ahmed Hefny
Department of Computer and Information Science
Institute of Statistical Studies and Research
Cairo University, Egypt

## ABSTRACT

Many software projects are still behind schedule, over budget estimates and fail to meet user requirements. One of the important reasons for those failures is a lack of project management. With the growing trend of adopting the agile methodologies, there are two opposite sides: agile and traditional project management approach. It is advisable to use both approaches or to combine them within a software company. This may be useful for software companies that run different projects using different project management methodologies. So, the question is how to integrate both approaches in a single project management framework. In this paper, the researchers propose a novel approach that combines different project management methodologies based on the alignment of their ontological models. Two project management methodologies are considered: Scrum and PMBOK representing agile and traditional approach respectively. This approach also helps introduce new tools or integrate the existing tools for supporting this integration.

## General Terms

Software Engineering, Project Management, Ontology, Software projects

## Keywords

Software Development, Agile Methodologies, PMBOK, Scrum, Ontology Alignment Process, Ontology Matching Techniques

## 1. INTRODUCTION

Many software projects are still behind schedule, over budget estimates and fail to meet user requirements. One of the important reasons for those failures is a lack of project management. A survey conducted by Standish Group showed that 39% of all software projects are successful (delivered on time, on budget, and with desired features, 43% are challenged (late, over budget, and/or with fewer than the desired features) and 18% failed (either cancelled prior to completion or delivered and never used). Thus, it is clear that project management plays an important role in the success of software projects [1].

With the emergence of agile methodologies, it is clear that two opposite sides exist: agile and traditional project management approach. Traditional project management is based on linear development cycles like waterfall model and disallows change [2]. Agile project management is based on iterative and incremental development and allows responding to change at any time during development [3]. The PMBOK and Scrum are the two most popular project management methodologies representing traditional and agile approach respectively.

PMBOK is a recognized standard for project management. It is designed and developed by Project Management Institute (PMI). According to PMI, project management can be defined as “the application of knowledge, skills, tools and techniques to meet project requirements”. This application of knowledge needs the efficient management of suitable processes. These processes are structured into two axes: five process groups and ten knowledge areas. Each process has inputs, outputs and tools and techniques [4].

Agile development has been appeared to overcome the flexibility issues of traditional models [5]. The term "agile" was created when a group of software experts met together in February 2001. The idea was to discuss lightweight development methods and offer an alternative to traditional development processes [6]. The result of this meeting was the “agile manifest” that defines a set of values and principles describing the agile philosophy [7]. Agile development is not a methodology in itself. It is an umbrella term that includes several agile methodologies.

Agile methodologies include Extreme Programming (XP), Scrum, Dynamic System Development (DSDM), Feature Driven Development (FDD), Adaptive Software Development (ASD), Crystal methodologies, etc. Most of the agile methodologies promote development iterations, working software, customer collaboration, and process adaptability [8]. According to the Forrester state of the agile development report, the most practiced agile methodology by teams were Scrum (85%) [9].

Both agile and traditional methodologies have their strengths and weaknesses if compared to different project characteristics. Traditional methodologies are more suitable for large-scale projects with a clear goal and stable requirements. On the other hand, small-scale projects with unclear goal and changing requirements are managed more with the agile methodologies. Consequently, the use of pure traditional project management or pure agile project management is not effective. A mixture of both methodologies is needed.

However, the literature has shown that there is a lack of hybrid approaches that show how to combine two or more methodologies and issues found during this integration. So, the question is how to create a project management framework based on different methodologies for effectively managing software projects. This framework should be adapted for managing a specific project. This may be useful for software companies that run projects in different heterogeneous environment.

In this paper, the researchers propose a novel hybrid approach that helps in integrating different project management methodologies based on the alignment of their ontological models. Two project management methodologies are considered: Scrum representing the agile approach and PMBOK representing the traditional approach. This approach also helps introduce new tools or integrate the existing tools for supporting this integration.

The rest of the paper is organized as follows: Section 2 provides a background overview of the two most popular project management methodologies. Section 3 presents the necessity for the integration between them. Section 3 presents the most important works related to this research field. Section 4 describes the proposed approach and its main components. Section 5 discusses the challenges and future directions. The last section concludes the paper with final remarks.

## 2. BACKGROUND OVERVIEW

This section consists of two parts. The first part presents the most popular agile methodology: Scrum. The second part provides an overview of PMBOK Guide along with its knowledge areas.

### 2.1 Scrum Methodology

Scrum is a framework based on the agile principles for managing complex product development. It was initiated by Ken Schwaber in 1995.The Scrum framework is composed of Scrum teams and their related roles, meetings, artifacts and rules. The team model in Scrum includes three primary roles: product owner, development team and Scrum master. All management responsibilities are divided among these roles [10].

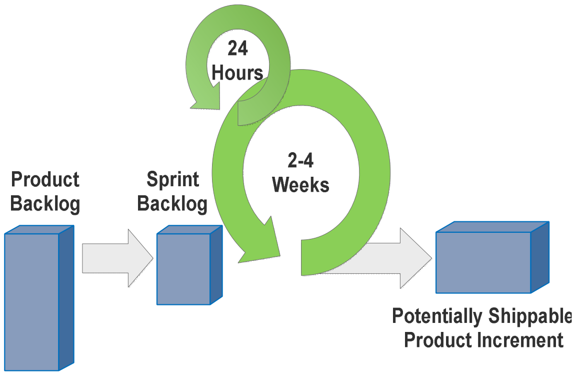

**Fig 1: The Scrum Process Flow [11]**

As shown in Fig. 1, Scrum structures product development in short iterations called Sprints, which are typically between 1–4 weeks in length. Sprints take place one after the other. Each Sprint begins with a Sprint planning meeting in which a cross-functional team select items from a prioritized list of user stories (requirements) called the Product Backlog. The Product Backlog is managed by the Product Owner who orders it according business value. Then, the team commits to complete the selected items (user stories) by the end of the Sprint [11].

During the Sprint, the team holds a meeting called a Daily Scrum. This is a 15-minutes meeting that occurs every workday. At this meeting, the team inspects progress toward the Sprint goal. At the end of the Sprint, a Sprint review meeting is held, where the team presents each user story that it completed in the Sprint to the product owner and other stakeholders. Following the Sprint Review, the team holds a Sprint retrospective. In this meeting, the team inspects its process and adapts it to improve its effectiveness and productivity [12].

### 2.2 Overview of PMBOK Guide

The first edition of PMBOK Guide was released in 1996. The second edition was released in 2000. The third edition was released in 2004. The fourth edition was released in 2008. The fifth and current edition was released in 2013 [13]. As mentioned before, PMBOK can be organized into process groups and knowledge areas. Fig. 2 shows how the process groups interact with each other. The process groups are the following [14]:

- Initiating
- Planning
- Executing
- Monitoring controlling
- Closing

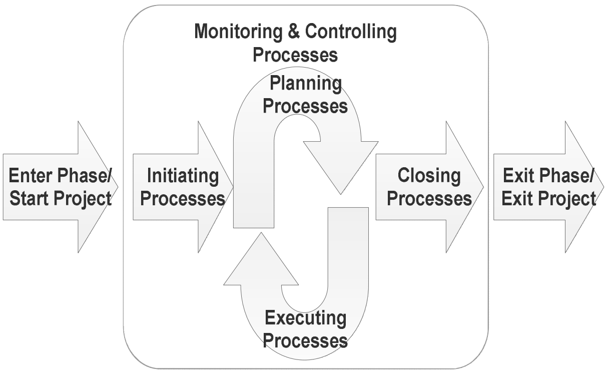

**Fig 2: PMBOK Process Groups [14]**

In this paper, the focus in on knowledge areas because they offer a more precise view of what is project management and provide the overall picture. Each knowledge area consists of a set of processes. These areas are as follows [15]:

1. *Project Integration Management* describes the processes that integrate and coordinate the different processes and activities of project management. It consists of the following processes:

   a. Develop project charter
   b. Develop project management plan
   c. Direct and Manage Project Work
   d. Monitor and Control Project Work
   e. Perform Integrated Change Control
   f. Close Project or Phase

2. *Project Scope Management* describes the processes needed to control the project scope. It consists of:

   a. Plan Scope Management
   b. Collect Requirements
   c. Define Scope
   d. Create WBS
   e. Validate Scope
   f. Control Scope

3. *Project Time Management* describes the processes needed to manage the timely completion of the project. It consists of:

    a. Plan Schedule Management
    b. Define Activities
    c. Sequence Activities
    d. Estimate Activity Resources
    e. Estimate Activity Durations
    f. Develop Schedule
    g. Control Schedule

4. *Project Cost Management* includes the processes concerning estimating, budgeting, funding, managing, and controlling the cost. It consists of:

    a. Plan Cost Management
    b. Estimate Costs
    c. Determine Budget
    d. Control Costs

5. *Project Quality Management* describes the processes involved in determining quality policies, objectives and responsibilities. It consists of:

    a. Plan Quality Management
    b. Perform Quality Assurance
    c. Quality Control

6. *Project Human Resource Management* describes all the necessary processes for organizing, managing, and leading the project team. It consists of:

    a. Plan Human Resource Management
    b. Acquire Project Team
    c. Develop Project Team
    d. Manage Project Team

7. *Project Communications Management* includes the processes concerning the timely planning, distribution and disposition of project information. It consists of:

    a. Plan Communications Management
    b. Manage Communications
    c. Control Communications

8. *Project Risk Management* includes the processes concerned with identifying, analyzing and controlling project risk. It consists of:

    a. Plan Risk Management
    b. Identify Risks
    c. Perform Qualitative Risk Analysis
    d. Perform Quantitative Risk Analysis
    e. Plan Risk Responses
    f. Monitor and Control Risks

9. *Project Procurement Management* includes the processes concerned with acquiring the products and services required to complete a project. It consists of:

    a. Plan Procurements
    b. Conduct Procurements
    c. Control Procurements
    d. Close Procurements

# 3. THE INTEGRATION BETWEEN SCRUM AND PMBOK

On one hand, agile methods have already proved its worth in the software development. But, they do not cover all aspects of project management. In an earlier work [16], a mapping has been conducted between Scrum practices and project management process areas of CMMI. The results showed that 37% of these processes are satisfied, 17% are partially satisfied and 46% are unsatisfied. This indicates that Scrum do not satisfy all the project management requirements of CMMI. More work is necessary to fill these gaps found in Scrum Methodology. These gaps could be filled by extending Scrum with some management elements described in other project management models or references such as PMBOK Guide.

On the other hand, PMBOK Guide focuses on project management generally and do not prescribe a specific methodology for software development. However, PMBOK tends to follow the traditional waterfall model and does not acknowledge iterative development. Therefore, connecting PMBOK with agile methods is a great challenge and beneficial for software project management. Therefore, the question has to be raised: how to realize the integration of these to seemingly different methodologies into a single framework? Furthermore, both approaches use different vocabularies. This leads to the problem of terminology conflicts. This problem needs to be solved before conducting the integration.

# 4. RELATED WORK

There are many works reported the need for integrating different project management approaches and methodologies for effectively managing software projects. However, there are few works focus on how this integration can be done. In general, these works can be categorized into two main categories according to their contents and research focus. These categories will be discussed briefly in the following subsections.

## 4.1 Comparing Agile Methodologies and Traditional Project Management Processes

This category focuses on mapping or comparing a set of project management processes as described in different project management models (i.e., PMBOK, ISO or CMMI) and a number of agile methodologies. The goal is to identify the gaps, differences and discrepancies existing between them. These include the following:

- Fitsilis [17] presented a comparison between PMBOK processes and a number of agile project management processes: XP, Scrum and FDD. The goal is to identify gaps and discrepancies existing between them. The results showed that agile methods do not cover the following knowledge areas: risk management, cost management and procurement management.

- M. Spundak [18] presented how the agile project management approach differs from traditional project management approach. This study concluded that both traditional and agile approaches have different advantages and disadvantages if compared to different kinds of projects. It also recommended combining the traditional and agile approaches into a single project management methodology.

- Galvan et al. [19] addressed the compliance analysis of the project management processes of three main agile methods (SCRUM, XP, and UPEDU) with the ISO standard. The compliance analysis was conducted on three particular categories of items: roles, activities and artifacts. The analysis results indicate that UPEDU and SCRUM are more complaint with the ISO, while XP has a moderate level.

- Bougroun et al. [20] conducted a mapping between CMMI level three and the practices of three agile methods: Scrum, XP and Kanban. The mapping results showed that Scrum satisfies 44% of the third level of CMMI, XP covers 45% of this level and Kanban covers only 6%. Combining the practices of the three methods covers 58% of the CMMI model level three.

- Farid et al. [16] presented a new approach for mapping between CMMI project management process areas and Scrum method. The goal is to show how Scrum practices cover these process areas. They also designed a new score index that could measure the importance of specific Scrum practices in the CMMI practices coverage. The result showed that CMMI process areas are not fully covered with Scrum practices.

## 4.2 Blending Agile Best Practices with Other Project Management Practices

This category focuses on blending or mixing some of the best practices of agile methodologies into traditional project management processes or vice versa. The goal is to combine their strengths while suppressing their drawbacks. For example:

- Batra et al. [21] presented a case study of a large distributed strategic project which faced several issues. These issues include the following: changes in scope, changes in user requirements, increases in cost and lack of risk analysis, etc. Then, they discussed how both agile and structured approaches were used to deal with each of the key project issues. This study concluded that blending of agile and structured practices is not only workable, but is essential.

- Hayata and Han [22] proposed a hybrid model to IT project development and management by blending Scrum method into traditional software development. This hybrid model concentrates on three levels. First, “waterfall-upfront” is applied to specify requirements. Next, Agile methods can be applied during the design, implementation and unit-testing. Finally, “waterfall-at-end” can be applied to conduct an integration test.

- Lozo and Jovanović [23] defined a new flexible hybrid methodology that combines the traditional and the agile method for managing IT projects. This proposed methodology has four different phases. The first and the last are always done in the traditional way. The second and third are in the flexible style. Flexible elements can be executed either with the traditional or agile way depending on the type of the project.

- Binder et al. [24] proposed a novel model called cocktail that combines the Agile Principles (AP) with the structure of the ISO standard. They identified the correlation between the ISO processes and the AP and made the necessary adaptations for incorporating the strengths of agile into each process. One of the limitations of this model is that the ISO standard provides only a high level definition of processes.

- M. Rahmanian [2] compared two hybrid methods combining the traditional and the agile method for IT project management. The first hybrid method uses the agile approach and the PMBOK Guide. The other hybrid method applies the agile method Scrum into the traditional software development process. This study concluded that this combination is the most appropriate solution.

- Ahmad et al. [25] proposed a novel hybrid framework called XSR which integrates the strengths of the three agile methods: XP, RUP and Scrum. XP provides software engineering practices such as user stories. RUP helps in providing a formal and structured process throughout the project lifecycle. The focus of Scrum is to agile effective management practices.

## 4.3 Summary of Previous Work

Although all of the previous works provide a good way to connect different project management methodologies, they do not provide a full solution involving a greater level of integration among their concepts. Table 1 presents a summary of all the works that were reviewed. This table includes the authors, the publication year, the focus of the study and its main shortcomings. It can be concluded some gaps related to this field:

- There is a lack of case studies that have been conducted on hybrid project management models.
- There is no a hybrid approach that provides a general process for successfully integrating two or more methodologies for software project management.
- In particular, there is limited research on integrating specific methodologies that have proven its effectiveness such as Scrum and PMBOK.

**Table 1. Summary of the Previous Work**

| Author | Year | Focus | Shortcomings |
|---|---|---|---|
| Fitsilis [17] | 2008 | Present a comparison between PMBOK project management processes and a number of agile processes. | This study does not provide detailed mapping between PMBOK processes and agile methodologies. |
| Batra et al. [21] | 2010 | Present a hybrid approach that integrates agile and structured approaches to benefit from their strengths. | More case studies are needed to test this hybrid approach in different software development environments. |
| Hayata and Han [22] | 2011 | Propose a hybrid model for IT project management based on Scrum. | More research effort is needed to include different agile methods other than Scrum. |
| Lozo and Jovanović [23] | 2012 | Define a new flexible hybrid methodology for managing IT projects. | More case studies are needed to strengthen their findings and conclusion. |

**Table 1. Summary of the Previous Work**

| Author | Year | Focus | Shortcomings |
|---|---|---|---|
| Spundak [p2] | 2014 | Present how the agile project management approach differs from the traditional approach. | This study does not provide any guidance on how to define a hybrid methodology. |
| Galvan et al. [19] | 2014 | Address the compliance issues of three agile methods with the ISO standard. | A full compliance study is still missing. |
| Bougroun, Zeaaraoui and Bouchentouf[20] | 2014 | Conduct a mapping between CMMI level three and the practices of three agile methods. | This mapping is high level and needs to be extended by considering the process areas related to the higher levels of CMMI model. |
| Bindera, Aillaudb and Schillia [24] | 2014 | Propose a novel model called cocktail that combines the Agile Principles with the ISO standard. | The cocktail model needs to be applied in practice to assess its effectiveness and discover the issues on implementing this model. |
| Rahmanian [2] | 2014 | Investigate and compare two hybrid methods for IT project management. | These two hybrid methods need to be tested empirically through case studies. |
| Ahmad et al. [25] | 2014 | Propose a novel hybrid framework called XSR which integrates three agile methods. | This framework needs to be tested in a real practical environment to address its shortcomings. |
| Farid et al. [16] | 2016 | Present a new approach for mapping between CMMI process areas and Scrum method. | More work is needed to cover other process areas that have not been covered on this study. |

# 5. THE PROPOSED APPROACH

The objective of the proposed approach is to integrate different project management methodologies based on alignment of their ontological models. The results of this alignment will provide a formal description that includes mapping of elements belonging to both methodologies. This approach also helps introduce new tools for supporting this integration. As shown in Fig. 3, the proposed approach consists of three steps which are discussed briefly in the following subsections:

1. Developing the domain ontologies of the selected methodologies.
2. Performing ontology alignment (Matching).
3. Creating a conceptual model that helps generate new tools.

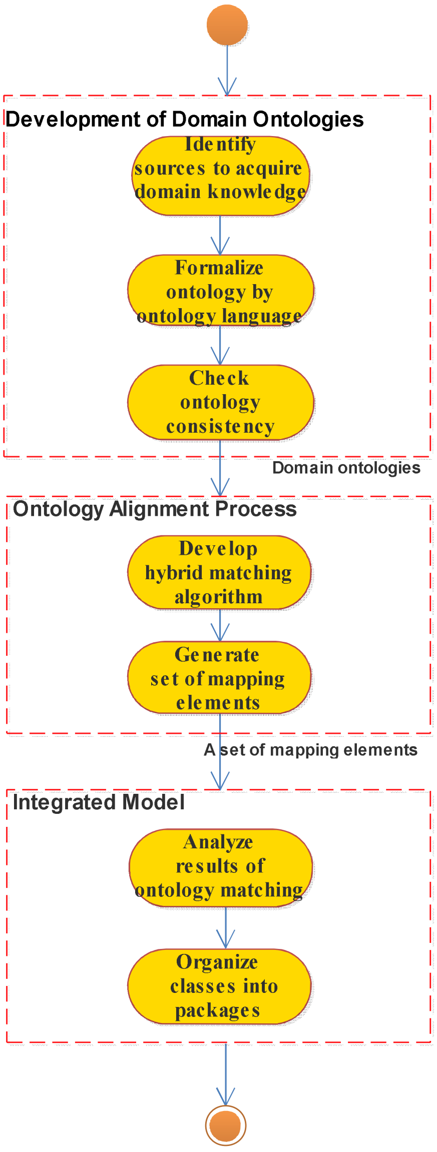
Development of Domain Ontologies
Identify sources to acquire domain knowledge
Formalize ontology by ontology language
Check ontology consistency
Domain ontologies
Ontology Alignment Process
Develop hybrid matching algorithm
Generate set of mapping elements
A set of mapping elements
Integrated Model
Analyze results of ontology matching
Organize classes into packages

**Fig 3: The Proposed Approach**

## 5.1 Developing the Domain Ontologies of the Selected Methodologies

As specified in the proposed approach and illustrated in Fig. 1, the first step is to build the ontologies of the selected methodologies. Ontology has many definitions. The most widely cited one was given by Gruber which defined the ontology as: "*An ontology is an explicit specification of a conceptualization*" [26]. A conceptualization refers to the objects, concepts and entities existing in a domain of interest and the relationships existing among them. Explicit means that the concepts identified and the restrictions applied to them should be defined clearly. Formal means that an ontology is machine-readable [27]. An ontology can be formulated as follows [28]:

$$O = (C, P, I) \qquad (1)$$

Where:

- $C$ is the set of concepts that exists in the domain of interest.
- $P$ is the set of relations holding among the concepts of the domain.
- $I$ is the set of individuals representing the instances of a concept.

The ontologies of PMBOK and Scrum are a good starting point for creating hybrid models for software project management. Developing these ontologies follows a methodology based on iterative process which allows refining the ontologies to build more accurate models of the selected domain. As shown in Fig. 4, this methodology consists of the following steps:

1. Determine the domain and the scope of the ontology.
2. Identify the sources used to acquire the domain knowledge (ex., experts, documents, existing ontologies, etc.).
3. Develop the glossary that contains the key concepts in the domain.
4. Classify the concepts in a hierarchy (taxonomy) and their relations.
5. Formalize the ontology by an ontology language (ex., RDF, OWL, etc.).
6. Evaluate the ontology completeness and consistency.

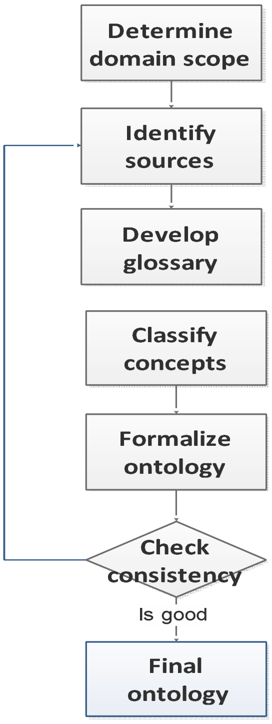

**Fig 4: Methodology of Creating Ontology**

## 5.2 The Ontology Alignment Process

Ontology alignment (matching) is the process of determining relationships or correspondences between entities of a pair of ontologies. Fig. 5 shows the general matching process [29] which can be seen as a function which receives as input two ontologies O1 and O2 and outputs an alignment A between these ontologies. An alignment contains a set of mapping elements (correspondences) between entities belonging to the matched ontologies. A correspondence can be expressed by several cardinalities: 1:1 (one-to-one), 1:m (one-to-many), n:1 (many-to-one) or n:m (many-to-many). Given two ontologies, a mapping element is defined as a 5-tuple:

$$A = (id, e_1, e_2, r, n) \qquad (2)$$

Where:

- *id* is an identifier for each mapping element.
- $e_1$ is the entity of the first ontology.
- $e_2$ is the entity of the second ontology.
- *r* is the relationship existing between $e_1$and $e_2$.
- *n* is the confidence level, typically in rang [0,1].

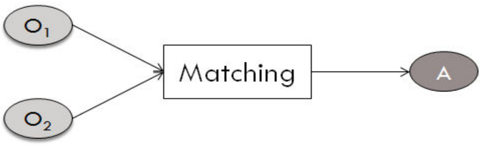

**Fig 5: The Ontology Alignment Process [30]**

The mapping element can be generated by using one or more matching algorithms. Ontology matching algorithms [30] can be classified into three main categories: Element, Extensional and semantic-based algorithms. Element Level algorithms focus on ontology entities and instances in isolation with their relations. Extensional algorithms can be applied in case the matched ontologies share the same set of individuals. Semantic-based algorithms focus on using external resources to find a common ground for the matched ontologies.

Each matching algorithm provides a numerical value that evaluates the similarity between two entities of the matched ontologies. To achieve high matching accuracy, a single matching algorithm is unlikely to be successful. So, the researchers suggest using a hybrid matcher to aggregate the similarity values generated by multiple matching algorithms to find correspondences between two ontologies O1 and O2 as shown in Fig 6.

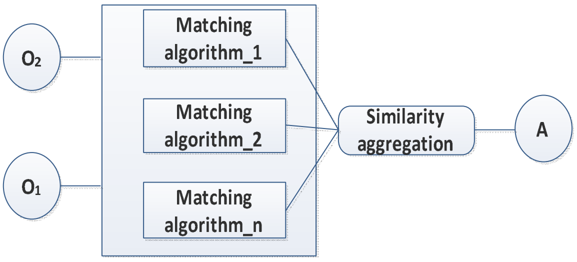

**Fig 6: Hybrid Matcher for Ontology Matching**

## 5.3 Creating the Conceptual Model

The alignment of concepts described in the previous step provides a basis to create a hybrid model for software project management. The goal of the model proposed in this paper is to realize the integration between PMBOK and Scrum concepts. This model helps generate new tools for supporting this integration. It consists of three main packages as shown in Fig. 7:

a) One for the concepts of PMBOK.
b) One for the concepts of Scrum.
c) A common package that joins the concepts that exists in both approaches.

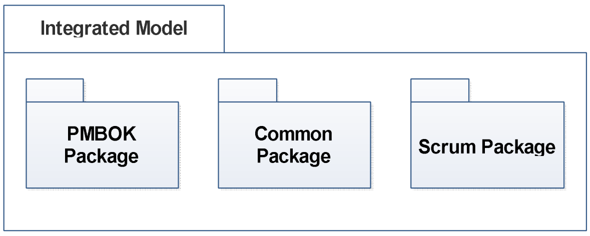

**Fig 7: Integrated Model for Scrum and PMBOK**

## 6. CHALLENGES AND FUTURE WORK

Although some works have been done towards tackling the problem of combining two or more methodologies for optimizing software project management, there are still open issues that impose new challenges and emphasize new directions for the future work. This paper briefly comments on such challenges:

- To solve the problem of semantic conflict between the selected methodologies.
- To improve the process of ontology alignment (matching).
- To propose new software project management framework that combines different project management methodologies.
- To test and validate the proposed framework using tangible evidence (e.g., case studies).
- To help introduce new tools or integrate the existing tools for supporting software project management.

## 7. CONCLUSION

This paper presented a novel approach to address the problem of integrating different project management methodologies for effectively managing software projects. This approach consists of three steps. The first step is to build the ontologies of the selected methodologies. In this paper, two project management methodologies are included: Scrum representing the agile approach and PMBOK representing the traditional approach. The second step is to perform ontology alignment using a hybrid matching algorithm. The final step is to create a hybrid model based on the alignment to realize the integration between the concepts of both methodologies. This model could assist in generating new tools for supporting this integration. This approach also provides a solution for the problem of terminology conflicts that arises when creating hybrid models for software project management. This may be useful for providing a common understanding of software project management concepts of different methodologies and easily enable the project team to switch between them.

## 8. REFERENCES

[1] Standish Group, (2014, May). CHAOS MANIFESTO. [Online].Available:https://people.eecs.ku.edu/~saiedian/811/Papers/standish-2013-report.pdf

[2] M. Rahmanian, "A Comparative Study on Hybrid IT Project Management," *International Journal of Computer and Information Technology*, vol. 3, issue 5, 2014.

[3] R. Hoda, J. Noble and S. Marshall, "Agile Project Management," in *Conf. New Zealand Computer Science Research Students (NZCSRSC)*, vol. 6, 2008 pp. 218-221.

[4] Project Management Institute, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 5th ed. Project Management Institute, 2008.

[5] A. I. Khan, R. J. Qurashi and U. A. Khan. "A Comprehensive Study of Commonly Practiced Heavy and Light Weight Software Methodologies," *International Journal of Computer Science Issues*, vol. 8, issue. 4, no. 2, 2011.

[6] K. M. Calo et al., "A Quantitative Framework for the evaluation of Agile Methodologies," *JCS&T*, vol. 10, no. 2, 2010.

[7] "Manifesto for Agile Software Development." Internet: http://agilemanifesto.org, [Feb. 5, 2017].

[8] A. Garg, "Agile Software Development," in *DRDO Science Spectrum*, 2009, pp. 55-59.

[9] D. L. Giudice, "The 2015 State Of Agile Development," Forrester Research, August 3, 2015.

[10] K. Schwaber, J. Sutherland. (2011, Oct.). The Scrum Guide. Scrum.org.[Online].Available:http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-DE.pdf

[11] J. Sutherland. (2010, Jul.). Scrum Handbook. Scrum Training Institute.[Online].Available:http://www.knowledgehut.com/images/scrumhandbook.pdf

[12] Microsoft Team, (2010, May). MSF for Agile Software Development v5.0. Microsoft. [Online]. Available: http://msdn.microsoft.com/en-us/library/dd380647(v=vs.100).aspx

[13] R. Xue, "Analysis and Comparison of Project Management Standards and Guides," in *Int. Conf. Mechanics, Materials, Mechanical Engineering and Chemical Engineering*, 2015.

[14] J. Heagney, *Fundamentals of Project Management*. 4th ed. American Management Association, 2012.

[15] C. S. Snyder, A User's Manual to the PMBOK Guide. 5th ed. Wiley, 2010.

[16] A. B. Farid, A. S. Abdelhgany, Y. M. Helmy, "Implementing Project Management Category Process Areas of CMMI Version 1.3 Using Scrum Practices, and Assets," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 7, no. 2, 2016.

[17] P. Fitsilis, "Comparing PMBOK and Agile Project Management Software Development Processes," *Advances in Computer and Information Sciences and Engineering*, pp. 378–383, 2008.

[18] M. Špundak, "Mixed agile/traditional Project Management Methodology – reality or Illusion?," in *Proc. Social and Behavioral Sciences*, vol. 119, 2014, pp. 939 – 948.

[19] S. Galvana et al., "The Project Management Cocktail Model: An Approach for Balancing Agile and ISO 21500," in *Proc. Computer science*, vol. 64, 2014, pp. 188 – 195.

[20] Z. Bougroun, A. Zeaaraoui and T. Bouchentouf, "The Projection of the Specific Practices of the Third Level of CMMI Model in Agile Methods: Scrum, XP and Kanban," in *Int. Conf. Information Science and Technology (CIST)*, Tetouan, 2014, pp. 174-179.

[21] D. Batra et al., "Balancing Agile and Structured Development Approaches to Successfully Manage Large Distributed Software Projects: A case Study from the Cruise Line Industry", *Communications of the Association for Information Systems*, vol. 27, 2010.

[22] T. Hayata and J. Han, "A hybrid model for IT Project with Scrum," in *Int. Conf. Service Operations, Logistics, and Informatics (SOLI)*, 2011, pp. 285-290.

[23] G. Lozo and S. Jovanović, "A Flexible Hybrid Method for IT Project Management," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 3, no. 7, 2012.

[24] J. Bindera, L. Aillaudb and L. Schillia, "The Project Management Cocktail Model: An Approach for Balancing Agile and ISO 21500," in *Proc. Social and Behavioral Sciences*, vol. 119, 2014, pp. 182 – 191.

[25] G. Ahmad et al., "XSR: Novel Hybrid Software Development
Model (Integrating XP, Scrum & RUP)," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, issue. 3, 2014.

[26] T.R. Gruber, "Toward Principles for the Design of Ontologies Used for Knowledge Sharing," *International Journal of Human-Computer Studies*, vol. 43, pp. 907-928, 1995.

[27] F. Ruiz and J.R. Hilera, "Using Ontologies in Software Engineering and Technology," in *Ontologies for Software Engineering and Software Technology*, Berlin: Springer, 2006, pp. 49-102,

[28] X. Xue, Y, Wang and A. Ren, "Optimizing ontology alignment through Memetic Algorithm based on Partial Reference Alignment," *Expert Systems with Applications*, vol. 41, pp. 3213–3222, 2014.

[29] P. Shvaiko and J. Euzenat, "Ontology Matching: State of the Art and Future Challenges," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 1, pp. 158-176, 2013.

[30] M.Gawich et al., "Alternative Approaches for Ontology Matching," *International Journal of Computer Applications*, vol. 49, no. 18, pp. 0975 – 8887, 2012.