

Piotr Szwed, Jan Werewka, Grzegorz Rogus

AGH University of Science and Technology, Institute of Automatics

ONTOLOGY BASED ALIGNEMENT OF CLASSIC AND AGILE PROJECT MANAGMENT FOR AN IT ENTERPRISE

Abstract

For IT enterprises running projects in different heterogeneous environments building hybrid models integrating classical and agile project management methodologies is considered as the very important issue. The paper discusses the problem of alignment of two project management methodologies: classical represented by PMBOK and agile represented by Scrum. The presented approach consists in building ontological models of both methodologies and performing ontology matching to find or justify mappings between concepts. We focus on modeling and matching of time related behavior in project for both methodologies.

1. INTRODUCTION

In IT enterprises different projects may run using different project management methodologies. Such situation may be caused by customer requirements, different team experiences and competencies, enterprise environmental factors or organizational process assets. From the viewpoint of the enterprise executive managers it is important to have a certain level of alignment of different methodologies. This may be helpful while analyzing and comparing states of currently running projects, enable easy switching between different methodologies on team and management level; clarify and provide common understanding of project management terms, roles and activities for different methodologies.

In this paper we propose to align classic and agile project management methodologies based on matching (alignment) of ontologies [1] describing their concepts and relations between them. The proposed approach consists in two steps:

- building ontological models of selected methodologies,
- performing ontology matching that can be done manually or with the support of ontology matching techniques.

In our research, two representative project management methodologies are considered: PMBOK [2], a classical methodology based on a project plan and Scrum [3,4] an agile methodology based on a product value. We build their ontological models and propose an initial alignment for concepts defining their time related behavior (e.g. project phases, iterations). The matching of concepts related to the time planning and decomposition of

project activities along the time axis is considered as one of the most important issues while building heterogeneous approaches to project management integrating practices of classical and agile methodologies.

The ontologies of PMBOK and Scrum methodologies provide a good starting point for elaborating hybrid models for project management. Such hybrid models can be quite general and in fact represent well founded proposals of assignment of elements of classical or agile methodologies to the project lifecycle. On the other side, hybrid models can be created within an organization to support different projects running according to classical or agile methodologies. Despite the project variety, each organization needs a common framework for project planning, monitoring, reporting and project portfolio management.

In the presented approach we assumed that any decision about substitution of entities in a hybrid model of project management should be well founded by finding corresponding matching of ontological concepts. It should be emphasized, however, that ontology matching should be treated as one of techniques helping to assess and validate elaborated hybrid models. The other approaches should consider a validation by community members, e.g. groups within PMI [5] and Scrum Alliance [6] organizations and also take into account an empirical experience.

This chapter is organized as follows: in Section 2 we discuss the problems of ontology alignment. Section 3 describes the architectures of PMBOK and Scrum ontologies. The chapter focuses on alignment of time related behavior in PMBOK and Scrum project management that is modeled in ontologies by events, therefore in Section 4 and Section 5 we describe the in detail models of events in PMBOK and Scrum. Finally in Section 6 we propose an initial alignment of events of the two methodologies.

2. ONTOLOGY ALIGNMENT

Ontology alignment is the process of determining correspondences between concepts belonging to two ontologies O and O' . The term ontology alignment or ontology matching also refers to specification of such correspondences.

The formal definition of ontology alignment [7] defines it as a set of tuples: (e, e', R, n) , where

- e and e' are entities (classes, properties, individuals) belonging to ontologies O and O' .
- R is the relation between e and e' , e.g.: equivalence ($=$), subsumption (\geq), disjointness (\perp) and overlapping (\bullet).
- n is the confidence factor, typically in range $[0,1]$

The book [7] discusses several techniques that can be used to determine alignment of ontologies. They include:

- Name-based that establish an alignment by comparing strings: entity names (or URIs) but also labels and comments. They can use several metrics to calculate string distances between entity names. For distances above a certain threshold an appropriate tuple with the confidence factor equal to the calculated distance is added to the alignment.
- Structure based methods determine the correspondence of entities by analyzing their internal structure (relations and data types used to express properties) as well as their position in the ontological hierarchy.
- Extensional techniques can be applied in a certain situation: if individuals are available. This gives an opportunity to match ontological classes that are treated as

intentional specification of individuals based on relations between sets of individuals (equality, inclusion, overlapping and disjointness).

- Semantic-based techniques consist in determining the correspondence of entities of two ontologies by comparing their meaning with respect to an external formal specification that is usually represented by an upper ontology. Those techniques require an initial step where compared ontologies are anchored in an external ontology. This can be performed manually or automatically with use of described earlier techniques.

We have performed several experiments with ontology matching tools that shown that the most suitable techniques for the described in this chapter problem are structural and semantic-based techniques.

3. PROJECT MANAGEMENT ONTOLOGIES

The PMBOK and Scrum ontologies relate to the common domain of Project Management. In consequence, they have similar high level structure reflecting general domain concepts. Moreover, they use the common vocabulary of relations (properties). Basic groups of classes in both ontologies are Artifacts, Roles, Processes and Events.

Artifacts represent objects that are created, processed, stored and delivered during the project realization. In the PMBOK ontology this group contains all inputs and output of processes listed in the methodology description [2] with subsidiary classes used to define their internal structure. In the Scrum ontology this group contains well recognized artifacts as ProductBacklog, SprintBacklog, BurndownChart, but also several management artifacts and, similarly to PMBOK ontology, classes used to define details of internal structure.

In both ontologies appear Roles defining agents responsible for executing processes and managing artifacts. Main roles are: Stakeholder, Team, TeamMember, Project Manager and FunctionalManager (PMBOK only) and ProjectOwner and ScrumMaster (Scrum only). It should be remarked, that PMBOK and Scrum define differently the role of Stakeholder – in PMBOK, a Stakeholder is equivalent to a Role, whereas in Scrum, a Stakeholder represents an external role, not committed to the project execution.

Both ontologies define explicit (PMBOK) and inexplicit (Scrum) sets of processes that are described by their inputs, outputs and that are related to the project stages (Events) during which they are executed. In PMBOK ontology there are 42 processes classified in specification [2] by their adhesion to ProcessGroups and KnowledgeArea. In the Scrum ontology 13 processes were identified and described in the similar manner.

The last main group of concepts are Events. Events are time related, they may be defined in terms of occurrence time, duration, frequency, but also agents (participants) and objects [8]. DurableEvents representing intervals can be linked by several temporal properties *after*, *before*, *during*, *finishes*, *overlaps*, *starts*), defined according to [9].

4. MODELLING OF TIME RELATED BEHAVIOUR IN PMBOK PROJECT MANAGEMENT

Project's life cycle in PMBOK is composed of one or several phases. The PMBOK phase can be considered as a project (project consists of one project phase) or subproject (project contains more project phases). There are temporal relationships between phases (*after*, *before*, *during*, *finishes*, *starts*). In PMBOK three phase-to-phase relationships are distinguished (Fig. 1.a.):

1. sequential relationship (next phase can start *after* the previous one is finished)
2. overlapping relationship (next phase can start *during* another phase)
3. iterative relationship (planning for the next phase is performed *during* the current phase is executed)

Each phase contains 5 events: PhaseInitiating, PhasePlanning, PhaseExecuting, PhaseControlling&Monitoring, and PhaseClosing. Those events define time periods during which different interacting activities belonging to following process groups: initiating, planning, executing, monitoring & control and closing may be performed. PhaseInitiating is an event which always starts each project phase and is executed only once. During the PhaseInitiating event the project or subproject is authorized, and then next project activities can be started. During the PhasePlanning a project management plan which contains subsidiary plans (e.g. WBS, schedule, cost, quality plans) is defined. Activities performed during PhaseExecuting coordinate people and other resource to carry out the plan. PhaseControlling&Monitoring overlaps with PhasePlanning and PhaseExecuting and its activities consist in monitoring and measuring project progress and taking corrective action, when necessary. Finally, PhaseClosing formally finishes the phase of a project and is an end of the corresponding project or subproject.

5. MODELLING OF TIME RELATED BEHAVIOUR IN SCRUM PROJECT MANAGEMENT

Classification of events in Scrum methodology is based on the Mike's Cohn description, who presented the agile development on the *onion diagram* [10]. This diagram was then modified by the company VersionOne [11] in order to better adhere to general practices of agile development and this version gained acceptance of Mike Cohn (Fig. 1.b).

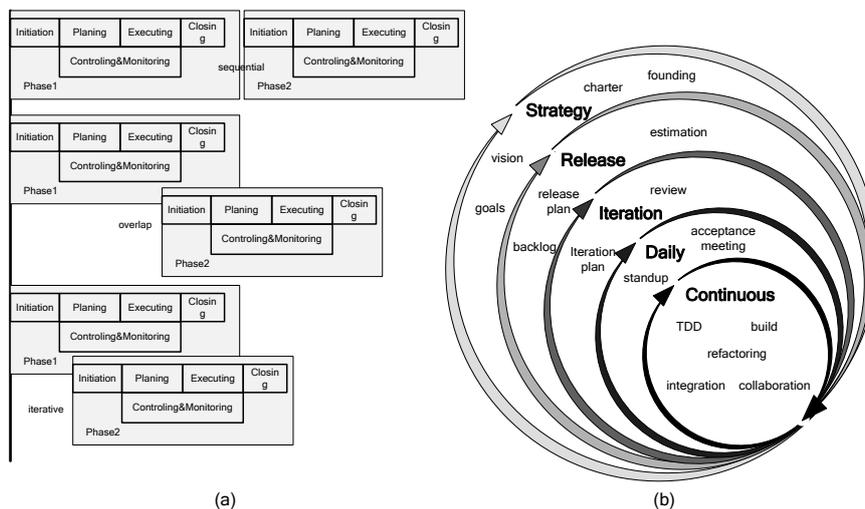


Fig. 1. Comparison of phase relationships in (a) PMBOK to (b) agile planning horizons

The onion diagram shows a containment hierarchy between layers corresponding to several planning horizons: Strategy (product development), Release (release planning), Iteration (single iteration), Daily (every day activities) and Continuous. Those planning

horizon were modeled as corresponding concepts StrategyPlanningHorizon, ReleasePlanningHorizon, Sprint, Daily and Continuous. The Event category contains also four events describing meetings: SprintPlanningMeeting (planning the iteration scope), SprintReviewMeeting (review of iteration results), SprintRetrospectiveMeeting (lessons and experience from the previous iteration, recommendations for the future) and DailyScrumMeeting (analysis of progress, impediments and new tasks).

The containment hierarchy of planning horizons is modeled with temporal relations using the property: *during*. Meetings are aligned on a time axis with reference to the Sprint (SprintPlanningMeeting *overlaps* Sprint, SprintReviewMeeting *ends* Sprint, etc.). Those relations define precisely temporal dependences between various Sprint events.

6. PROPOSED ALIGNMENT FOR TIME RELATED BEHAVIOUR

Analysis of events and their relations in PMBOK and Scrum ontologies indicates that there are two candidate mappings:

- In the first mapping a phase leads to an external release, one-phase project delivers only one release, whereas multiphase project can deliver multiple releases. Consecutive overlapping PhasePlanning and PhaseExecuting events of PMBOK correspond to a Sprint and Daily events.
- The second mapping shown in Fig. 2. assumes an equivalence between a Project and ReleasePlanningHorizon. The goal of a project is an external release. In this case the StrategyPlanningHorizon is not mapped directly, it is related to a product development achieved by several projects. A project phase is mapped to a Sprint, and events occurring during the phase are linked with overlapping relation to Scrum daily activities.

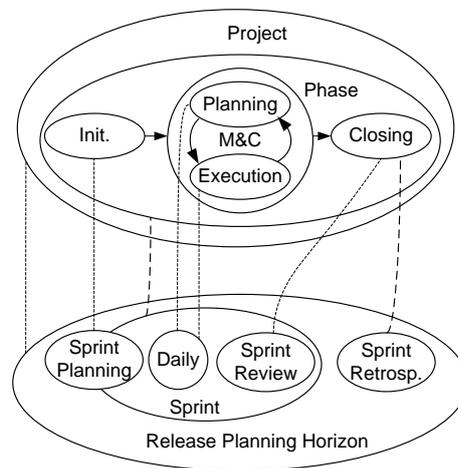


Fig. 2. Mapping of PMBOK phase to SCRUM sprint planning horizon

This mapping should be treated as an initial mapping for planned application of structural-based matching method that is expected to calculate the confidence factors and find matching between other concepts, in particular processes executes during events.

7. CONCLUSIONS

Construction of ontology involves an analysis and a systematization of knowledge of a certain domain. This builds good foundations for communicating between specialists, team members and other persons involved in the project. We expect that the research described in this chapter may be especially interesting for companies working in heterogeneous project environment. The obtained solutions will help to obtain suitable level of alignment of different methodologies in an enterprise.

The presented work is a part of a broader research and practical experiments in real IT companies environment, which should lead at to developing instructions for leading projects in heterogeneous environment and developing common solutions for agile and classic project management in an IT enterprise.

BIBLIOGRAPHY

- [1] Gruber T.: A Translation Approach to Portable Ontology Specifications. Academic Press Ltd., 1993
- [2] A guide to the project management body of knowledge (PMBOK® Guide) - Fourth Edition: Project Management Institute, 2008.
- [3] Schwaber K., Beedle M: Agile software development with SCRUM. Upper Saddle River, N.J., Prentice Hall, 2002.
- [4] Mike Cohn: Succeeding with Agile: Software Development Using Scrum, Addison Weseley, 2009
- [5] Project Management Institute, <http://www.pmi.org>
- [6] ScrumAliance, <http://www.scrumalliance.org/>
- [7] Euzenat J., Shvaiko P., Ontology matching, Springer Verlag 2007
- [8] Kaneiwa K., Iwazume M., Fukuda K.: *An Upper Ontology for Event Classifications and Relations*. Australian Conference on Artificial Intelligence 2007: 394-403
- [9] Allen J. F., Ferguson G.: Actions and events in interval temporal logic. Technical Report TR521, Computer Science Department, University of Rochester, Rochester, New York 14627, July 1994.
- [10] Cohn M.: Agile Estimating and Planning, Prentice Hall/PTR, 2006, s.330
- [11] Agile Development Poster, <http://pm.versionone.com/AgilePoster.html>

ZASTOSOWANIE ONTOLOGI DO INTEGRACJI KLASYCZNYCH I ZWINNYCH METODYK ZARZĄDZANIA PROJEKTAMI W PRZEDSIĘBIORSTWIE INFORMATYCZNYM

Streszczenie

Współczesne przedsiębiorstwa branży IT poszukują rozwiązań z dziedziny zarządzania projektami polegających na integracji metodyk klasycznych i zwinnych. W artykule analizowany jest problem integracji dwóch metodyk: klasycznej reprezentowanej przez PMBOK ora zwinnej reprezentowanej przez Scrum. Zaproponowane podejście polega na budowie ontologicznych modeli obu metodyk oraz przeprowadzeniu uzgadniania ontologii, aby znaleźć lub potwierdzić zaproponowane odwzorowania pojęć. W artykule skupiono się na zagadnieniach modelowania i uzgadniania pojęć definiujących relacje czasowe w obu metodykach.