Integrating Network Technique into Distributed Agent-Oriented Software Development Projects

Christian Röder and Lawrence Cabac

University of Hamburg, Faculty of Mathematics, Informatics and Natural Sciences,
Department of Informatics, http://www.informatik.uni-hamburg.de/TGI/

Keywords: Project Management, Network Technique, Software Engineering, Petri Nets, Modeling

Extended Abstract

The management of local software projects is challenging, due to its complexity. In case of distributed development projects, the complexity in project management increases even more [1]. In this publication we introduce and adapt the well-proven *network technique*[1] into Paose, a distributed agent-oriented software development approach, by directly integrating a modeling tool for network technique into the development environment of Paose. We support the project participants in modeling their interdependent project activities and reason about them more easily, using an illustrative, graphical syntax. By providing the mapping of network technique diagrams onto Petri nets, we utilize formal semantics and improve integration into our development approach, which is based on Petri nets.

The participants of distributed software development projects have to tackle several challenges: Project members, which are organized in sub-teams, perform activities at spacial and temporal distance from one another. These distances have an impact on communication, coordination and control [1]. If, in order to counter these challenges, concepts from agile project management that value self-responsibility are applied, the sub-teams are required to self-organize and self-manage their own activities. Therefore, the sub-teams require support.

Naturally, a sub-team has to plan and perform a large number of activities. These activities may depend logically and temporally. Furthermore, not only may dependencies exist between the activities of one single sub-team, but also between activities of multiple sub-teams. Quantities and interdependencies of activities complicate the sub-teams capabilities of planning and scheduling: Statements regarding the duration of the overall project or about efficiently scheduling activities are not made easily by participants of the sub-teams. In order to ease

[1] The terms *network technique* or *network scheduling* (German: *Netzplantechnik*) subsume - amongst other concepts - the more known methods *CPM* (critical path method), *MPM* (metra potential method), *PERT* (program evaluation and review technique) or *PDM* (precedence diagramming method). For a short overview about network technique, see [6, pp. 101 - 108].

the sub-teams' project management, participants of sub-teams require support in modeling the complex system of linked activities.

Distributed software development projects can be executed by following the **P**etri-net and **a**gent-**o**riented **s**oftware **e**ngineering (Paose) approach [2,4]. In prior works, an issue tracking system and continuous integration support were integrated into Paose in order to support the participants in performing project management activities [3]. To allow scheduling of activities and reasoning about time properties further work is required. The current research of Röder [5] is concerned with the integration of time planning techniques into Paose.

The discipline of project management provides - among other things - the well-proven *network technique*. Using network technique, project managers are able to graphically model projects. Thereby, project activities and relationships are explicated as *net schedules* and elements of project flows. In addition, project managers can make use of the *critical path method* to obtain prioritization guidance by identifying activities that are critical in regard to the project duration.

In this poster we present a prototypical tool for applying concepts of a network technique to model sets of project activities as net schedules, enabling the identification of sequences of time-critical activities. The modeler can map net schedules to Petri nets, whereby the semantics can be altered in principle. This mapping has three benefits. Petri net analysis can be applied indirectly on net schedules. Project participants obtain explicit semantics about the net schedules they created. Furthermore, net schedules can indirectly be incorporated as executable Petri nets into the Petri net-based implementations of Paose.

In the future, the modeling tool can be enhanced in different ways: More concepts of the network technique can be implemented into the tool, to enrich the usable syntax. More complex semantics for the mapping of net schedules to Petri nets can be provided, to generate Petri net-based implementations for various purposes.

References

1. Pär Ågerfalk et al. A framework for considering opportunities and threats in distributed software development. In *Proceedings of the International Workshop on Distributed Software Development. Austrian Computer Society*, pages 47–61, August 2005.
2. Lawrence Cabac. *Modeling Petri Net-Based Multi-Agent Applications*, volume 5 of *Agent Technology – Theory and Applications*. Logos Verlag, Berlin, 2010.
3. Matthias Güttler. Integration einer agilen Projektmanagementumgebung in ein verteiltes Team. Diploma thesis, University of Hamburg, Department of Informatics, Vogt-Kölln Str. 30, D-22527 Hamburg, November 2013.
4. Paose web page. http://www.paose.net/. Last accessed on 31st May, 2015.
5. Christian Röder. Prototypische Integration von Netzplantechnik in einen agentenorientierten und verteilten Softwareentwicklungsprozess. Diploma thesis, University of Hamburg, Department of Informatics, Vogt-Kölln Str. 30, D-22527 Hamburg, planned for September 2015.
6. Stefano Tonchia. *Industrial Project Management: Planning, Design, and Construction*. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2008.