A transformation of YAWL to AToMPM Modeling Language

Srinivasan Balakrishnan
Master of Computer Science
University of Antwerp

Abstract:

Modeling languages have been used for software development and industrial applications. YAWL(Yet Another Workflow Language) is one of the leading design pattern. Many business processes using this simulation environment to manage, analyze, flowchart their needs. The interesting thing is to design one modeling language in another. AtoMPM(A Tool for Multi-Paradigm Modeling) is the another simulation environment which works similarly like petri nets, Casual Block Diagram. In this study we are implementing YAWL in AToMPM together with its operational semantics. We first go through introducing the project and related work.

YAWL:

YAWL system is one of the most mature open-source workflow management system available at present(understanding user references). "Wikipedia saids that YAWL is an extended form of petri nets and improvised design environment". YAWL support for the business process management and organizational utilities which can deal with tons of data and applications for the industries. It has been part of the medical sectors which helps to analyzing, testing the data level abstractions. For computing software in a simple way and easy-to-use approaches the YAWL system is used. It support for the high level graphics, application designs, deployment, execution and monitoring for the software systems. Enabling the business development for the organizations and research oriented sectors as well. The step by step design used to analyze the data, deep overview of the software components, functionalities of entities used. It exhibits the clear view of the related information from the user point of view. [1]

It has an comprehensive support for the control-flow systems. YAWL contributes a large impact in modeling world with its operational and application level design patterns. It is a service oriented architecture, so any language can adapt with this system. It supports passing files as data and organize those data by this internal engine. The data's are sequentially scheduled by the YAWL engine. To adopt the YAWL in other language is a lot of work, so that it split in to three different sub parts. They are Control panel, Editor and Engine.[2]

AToMPM:

AToMPM is a web based modeling technique, which uses UML as a subset for the definition of modeling language. It is one of the visual modeling environment was developed by "Eugine

Syriani from MCGILL University, Canada". It works under local host over on-line through its client and server. It is a tool for multi paradigm modeling for building metamodels, transformations and for executing Formalism Transformation Graph(FTG). In AToMPM the languages like petri nets, state charts and CBD are metamodeled using the internal formalisms. The models are synthesized in AToMPM as abstract and concrete models. Every model works with AToMPM are metamodeled and modeled as abstract syntax and concrete syntax. AToMPM uses the UML as the main formalism to develop the other languages in to it. It uses class diagram, entity diagram, relational diagram to design. It is an improvised version of AToM3 which can be called as a successor of AToM3. [3]

AToMPM runs on a web browser and provides the coordination for distributed collaboration in real-time. So the data and design can be accessed only over the web. It support for Domain Specific Language(DSL) to collaborate with the different domain languages. The language is totally depends on the formalisms and simulation work flows. So, the internal operations requires the appropriate terminology to move on to further steps. Every level works under some rules, for example it starts by design class diagrams and transitions between them, transformation to graph model, scheduling the design and generating code. The following figure shows the general view of AToMPM window and some tool-bar.[3][4]

Related Work:

The work has developed similar to is Petri nets are modeled and transformed by AToMPM simulation language. Petri nets is one of the know designing language, which is more base for other modeling environment. Every step in petri nets is sequentially designed building process which makes the transitions work well. Many simulation languages are more similar to petri net model, so it gives an impression to the users that main design languages were developed with the help of petri nets. The beauty thing in formalisms are we build one in another with the help of their tools, to let the languages work on their same style. AToMPM is the simulation formalism has the ability to incorporate with some fewer modeling languages. [3]

Why is this study?

We are having two different modeling languages in hand to work on. Always there is challenging factor is to combine or collaborate each modeling language. Also by working with this we are gaining the hands-on experience with simulation environments. So that we are doing this analysis to let the YAWL language to fit into AToMPM modeling language. This process is very interesting to work with modeling languages. YAWL has a very latest business modeling technique. By developing that sensational business process model in to another visual paradigm. The goal is to develop YAWL operational semantics with the AToMPM floor. The paper will explain more clear and the procedures to change the languages.

References:

[1] Recker,J.(Ed.)., & La Rosa,M.(2012). Understanding User differences in open-source work flow management system usage intentions. Australia, Brisbane:Queensland University

[2] Wynn,M.T., De Weerdt,J., Ter Hofstede,A., van der Aalst,W., Reijers,H.A., Adams,M., Ouyang,C., Rosemann,M., & Low,W.Z.(2013). Cost-Aware Business Process Management: A Research Agenda. Australia, Brisbane:Queensland University

[3] Mustafiz,s., Denil,J., L'cio,L., & Vangheluwe,H.(2012). The FTG+PM framework for Multi-Paradigm Modeling: An Automotive Case Study. Belgium, Antwerp: University of Antwerp & Karel de Grote., Canada, Montreal: McGill University

[4] Lajmi,A., Martinez,J., & Ziadi,T.(2014). DSLFORGE: Textual Modeling on the Web. France, Paris: Universite Pierre et Marie Curie