7 Conclusions

Language of software model to model transformation is presented in this paper. The motivation of performing such a research is to obtain extensible model to model transformation language supporting several concrete syntaxes, allowing representation of software models and transformation rules both in compact and detail view. Graph representation of software models allows providing a bridge between UML (or other modeling language with graphical concrete syntax) and analytical tools for software models processing and analyzing (1)-(3). Proposed representation of transformation rules is also compatible with graph representation of software models (10)-(13). Such representation allows considering complex expression for performing transformations, including several preconditions, or software model elements that are not linked each other directly.

Concrete syntax of the M2MTM, proposed in this paper, allows considering transformation process both on metalevel and model level [6]. General transformation ideas and software models notations can be analyzed on metalevel.

Considering of sub-graphs and software models at level of elements permits analyzing transformations in details. Doing this existing transformation rules can be refined and new transformation rules also can be designed.

8 FURTHER WORK

Propose an approach of resulting software model designing grounded on problem domain ontology analysis. Visualized resulting software model should consider possibilities of human cognitive abilities for perception (Chebanyuk and Markov, 2015).

Define operations that are used for analysis of software model before and after transformation (for example refinement or merging). Extend M2MTL abstract and concrete syntaxes for preforming these operations and propose corresponded analytical tools.

Develop a software tool for extracting information from initial software model designed in different modeling environments.

References

[1]Brambilla M., Cabot J., Wimmer M., 2012. *The book* Model-Driven Software Engineering in Practice. Synthesis Lectures on Software Engineering. Morgan & ClayPool publishers. Pages 1-182

[2] Chebanyuk E., Markov K., 2015. Software model cognitive value. *International Journal "Information Theories and Applications", Vol. 22, Number 4, ITHEA 2015 http://www.foibg.com/ijita/vol22/ijita22-04-p04.pdf*

[3] Chebanyuk E., Markov K., 2016. Model of problem domain "Model-driven architecture formal methods and approaches" International Journal "Information Content and Processing", Vol. 22, Number 4, ITHEA 2016, p.202-222. http://www.foibg.com/ijicp/vol03/ijicp03-03-p01.pdf

[4] Chomsky, N. 1957. *The book*. Syntactic Structures. Mouton publishers, Eilenberg: Mac Lane The, Hague, *1945 - 1957. ISBN 90 279 3385 5. p.107.*

[5] ATL 2016 . ATL Transformation Language, 2016 . http://www.eclipse.org/atl/

[6] IBM, 2016. *http://researcher.ibm.com/researcher/files/zurich-jku/mdse-07.pdf*

[7] OCL, 2014. Object Constraint Language Version 2.4 OMG standard. *http://www.omg.org/spec/OCL/2.4/PDF*

[8] OMG, 2016. Meta Object Facility™ (MOF™) Core http://www.omg.org/spec/MOF/

[9] QVT, 2016. Meta Object Facility Query/View/Transformation, v1.3 http://www.omg.org/cgi-bin/doc?formal/2016-06-03

[10] Rumpe B., Weisemöller I., 2011. A Domain Specific Transformation Language. *In: ME 2011 - Models and Evolution, Wellington, New Zealand. Ed: B. Schätz, D. Deridder, A. Pierantonio, J. Sprinkle, D. Tamzalit, Wellington, New Zealand, Okt. 2011. https://arxiv.org/ftp/arxiv/papers/1409/1409.2309.pdf*

[11] Santo I.S., MC Andrade R., Santos P.A., 2015. Templates for textual use cases of software product lines: results from a systematic mapping study and a controlled experiment. *Journal of Software Engineering Research and Development (2015) 3:5 DOI 10.1186/s40411-015-0020-3 https://jserd.springeropen.com/articles/10.1186/s40411-015-0020-3*

[12] Sendall, S., Kozaczynski. W., 2003. "Model transformation: The heart and soul of model-driven software development." *IEEE Software 20.5 (2003): 42-45. http://ieeexplore.ieee.org/document/1231150/*

[13] XMI, 2015. XML Metadata Interchange. *Access mode http://www.omg.org/spec/XMI/*