LEVEL 0 SUMMARY

Instruction

This summary will be shared with L1, L2 and L3. Keep in mind that these levels do not have a full understanding of the subject. Try to write something easy to understand but not simplistic. Your summary should explain the main contribution of the paper with your own words. Furthermore, you can use simple examples, if necessary, to better explain the main ideas. Your grade will take into account the quality of your summary, the formal English language in which it has been written, and whether it helps the levels above in their own work.

Name of student:

BURLON Carla

Name of your Level 1:

GONCALVES MELIM Maria Lolita

Source (e.g. scholars.google.com):

Paper available on Github

Paper title:

Jesus Sanchez Cuadrado, Esther Guerra, and Juan de Lara, A Component Model for Model Transformations, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, JULY 2013

Keywords specific to the paper:

Model-Driven Engineering, Model Transformation, Reusability, Genericity, Component-Based Development.

Summary of the main contributions:

Model-driven Engineering is a methodology that aims to create and exploit conceptual models which are abstract representations of the data and interactions of a specific field. It is a part of software design that focuses on the design and efficiency of IT models.

This article proposes a component based approach of model transformations which aims to offer a reusable way of conceptualizing and implementing these transformations. The approach is supported by an implementation in the Atlas Transformation Language, a model to model transformation language.

Transformation templates are represented by components that are configurable and can be chained together within composite components. Although there is no notion of pruning or automated transformation this offer a framework to build model transformation pipelines more effectively. The transformation approach was obtained based on insights from generic programming and component based software development and does not rely on any AI model.

This paper starts by discussing the limitations of using meta-models approaches. These require extensive knowledge and must be adapted to each specific metal model. A transformation approach designed for one metamodel cannot be used for another. This entails manual and error prone work.

The transformation component in essence consists of an input and output of typed metal-models or concepts. The concept word means a set of structural requirements that a meta model should follow. The nature of the transformations is described as a black box at first.An example of component is “Flow diagram to Petri Net” which takes a flow diagram as input and outputs a petri net.

As mentioned before theses components can be used on their own or chained together to form composite models.The details of the model are presented in 4 dimensions : abstraction, specialization , selection and integration. The notion of concept is the abstraction, the reuse is the ability to use the same components for multiple meta model.. The specialization is the mapping between concept and meta model (compatibility module) and the selection is provided by tags which may be attached to both concepts and transformations.The paper also provides a composition language that allows the integration to build component chains.

Method :

The generation of the component transformations is based on the binding which connects generic transformation templates to the models. By using High Order Transformations the paper proposes a way to modify or generate the appropriate transformations based one the concept. This means that to reuse the model you just need the appropriate bindings which , as an added benefit although they are meta-model-specific are not component-specific.

Supported by a software application? (If yes, provide more details)

The approach is supported by an implementation in the Atlas Transformation Language, a model to model transformation language. This language is widely used in practice for model-to-model transformations. This choice was made to demonstrate the practical applicability and effectiveness of the component model for model transformations within the Model-Driven Engineering (MDE) framework.