

A Model-driven Ontology Approach for Developing Service System Applications

Anatoly Petrenko

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine, Email: tolja.petrenko@gmail.com

In recent years different levels of ontologies (Domain, Application, and Task ontologies) are used in service systems development. OMG (Object Management Group) proposed a hierarchical system of service systems semantic description for their modeling, which consists of an M0 level (objects), the M1 level (models), the M2 level (meta-model or language level), and the M3 (meta-meta-model or language description level) which is often called Meta Object Facility (MOF).

In general, meta-models are language specifications, not only of modelling, but also of arbitrary languages. If two Software Engineering tools agree on the same meta-model, they impose the same structure on their models, so that they can easily exchange them. Software processes, being specific work flows, can be meta-modelled and used to construct software environments.

The process of designing system services can be modelled as a process of transformation models from a meta-model as input, and using a set of conversion rules. The transformation itself is also a model. Transformation models can have a variety of applications:

- Creating low-level models, and as a result the initial code, moving from models of a higher level.
- Display and synchronization between models at the same level or at different levels of abstraction.
- Creating a presentation about the system on the basis of requests.
- The evolution of model problems, such as refactoring pattern, i.e. change internal system structure in order to make it easier to be understood and further changeable, while not altering the existing functionality.
- Reverse design from the lower level models (or even code) to models of a higher level.

Models conversion is a key element in services system design, which provides ontological processing means for generating conceptual models. Languages of models conversion models support different types of transformations, such as model-model (MM) or a model-code (MC). A feature of this approach is that most of the time a developer works not with the code, but with the models.

MDA (the Model - driven approach) uses the MOF-based models for the creation and manipulation of accurate, detailed, computer-readable description of the application structures independent of programming languages, operating systems and databases, which can be used to implement them. MDA, based on a three-layer approach, provides:

- Computational independent model (CIM), which describes a system with independent computing point of view, highlighting the structural aspects of the system, changing the emphasis from a domain modeling to architecture modeling. A CIM is sometimes called a domain model.
- Platform independent model (PIM), which can be considered as the definition of the system from point of view of a neutral virtual machine or computer abstraction. The platform independent viewpoint focuses on the operation of a system while hiding the details necessary for a particular platform. So it is suitable for use with a number of different platforms of similar type.
- Platform Specific Model (PSM), which usually covers the technical concepts and services comprising the implementation platform. That is, this model is aimed at a specific implementation services system technology. A PSM combines the specifications in the PIM with the details that specify how that system uses a particular type of platform.

