

электронный журнал

МОЛОДЕЖНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК

Издатель: ФГБОУ ВПО «Московский государственный технический университет им. Н.Э. Баумана»

УДК 004.415.538

Моделе-ориентированный редактор справочных данных с поддержкой стандарта на основе платформы ISO15926 оделе-ориентированный редактор справочных данных с поддержкой стандарта на основе платформы ECLIPSE

Студент,

кафедра: «Информационные системы и телекоммуникации»: Шишкин Д.А.

Научный руководитель: А. М. Иванов

Ассистент кафедры «Информационный системы и телекоммуникации»

МГТУ им. Н.Э. Баумана

iu3@bmstu.ru

В настоящее время производство становится все более сложным. Растет как количество элементов, подсистем технически сложных объектов (ТСО) которые необходимо производить, так и разнообразие связей и типов связей (механические, электрические, информационные) между отдельными их частями. В итоге, сложность изделий такова, что невозможно построить их аналитическую модель для экспериментов и подбора параметров: в модели системы приходится учитывать изменения системы во времени, причинно-следственные связи, последствие, нелинейности, случайные переменные. Еще одним фактором является то, что любое изделие имеет свой жизненный цикл, а определенные параметры необходимо знать не только на этапе производства, но и на этапе эксплуатации и утилизации.

Поэтому, сейчас стремительно развиваются гетерогенные multi-physics системы имитационного моделирования, в которых можно произвести необходимые эксперименты на виртуальном объекте непосредственно перед производством. Результаты этих экспериментов позволяют найти оптимальные параметры для изучаемого объекта. После чего начинается производства этого изделия.

Анализ предметной области показал, что ТСО необходимо описывать:

семантически - описывать то, чем является сам объект, и его части, то есть связь модели ТСО с реальным объектом в реальном мире. К примеру, описывая модель автомобиля, необходимо указывать, какая часть является корпусом, какая колесом. А так же необходимо определить их параметры - мощность двигателя, радиусы колес, максимальная скорость самого автомобиля.

графически - описывать то, как выглядит объект и его части. То есть, необходимо описать из каких поверхностей состоит объект, параметры материалов и их физические

свойства.

Одна из основных задач при разработке системы имитационного моделирования состоит в разработке подсистемы визуализации. Данная система должна отображать и семантику изучаемого ТСО, и его графическое представление. Однако невозможно визуализировать ТСО, не описав его на каком либо языке. Таким образом, в данной статье рассматривается вопрос создания мета-модели языка описания ТСО и подсистемы древовидной визуализации высказываний на этом языке.. Требуется реализовать модуль, который позволит визуализировать, создавать, изменять и удалять модели технически сложных объектов.

Наиболее подходящим для целей разработки является стандарт [7] **ISO 15926-1:2004 Industrial automation systems and integration** — Integration of life-cycle data for process plants including oil and gas production facilities (Системы промышленной автоматизации и интеграция — Интеграция данных жизненного цикла установок непрерывного производства, включая нефтяное и газовое производственное оборудование). Стандарт описывает язык описания универсальных понятий, с помощью которых описываются данные жизненного цикла ТСО. По сути, этот стандарт описывает верхнюю онтологию. Такой подход является факт-ориентированным [10], что дает возможность более гибко и полно описывать существующие зависимости между объектами.

Платформой для разработки и работы редактора был выбран Eclipse, что объясняется его открытостью, а так же возможностями по расширению с помощью механизма плагинов. Выбор данной платформы позволит использовать стандартный интерфейс среды для обеспечения работы пользователя, что значительно сократит время разработки. Механизм плагинов Eclipse позволяет добавлять необходимые элементы пользовательского интерфейса к стандартным. Подробное описание архитектуры и функционала данной платформы дано в [5].

В качестве подхода к разработке редактора была выбрана концепция разработки программного обеспечения, управляемой моделями (Model Driven Development MDD). Идея, лежащая в основе данного подхода, заключается в предельной автоматизации процесса генерации кода, благодаря чему разработчики могут сосредоточиться на создании самого алгоритма. Так же стоит отметить, что программное обеспечение, основанное на мета-программировании, является гибким, так как оно написано не для конкретного случая данных, с конкретной структурой и с фиксированными зависимостями, а работает в общем случае. Это позволяет быстро и эффективно решать возникающие проблемы, а так же сделать данные семантически более насыщенными и явно выразить скрытую семантику структуры данных.

В качестве реализации концепции разработки программного обеспечения, управляемой моделями была выбрана среда Eclipse Modeling Framework (EMF). Данная среда соответствует стандарту [8], разработанному Object Management Group для концепции MDD. EMF, так же как и Eclipse, является открытой средой и в настоящее время активно развивается. Еще одним преимуществом подобного выбора является возможность генерации не только кода модели данных, но и редактора модели для среды разработки программного обеспечения Eclipse. Подробное описание архитектуры и функционала данной платформы дано в [2, 3]. Описание мета-моделей в EMF производится на языке Ecore [2].

В соответствии с подходом MDD, модель данных описывается с помощью мета-модели, по которой генерируется код.

Спроектированная мета-модель описана с помощью языка Ecore и полностью соответствует стандарту ISO15926, дополненная некоторыми классами, упрощающими работу с моделью. Некоторые части модели позаимствованы из языка OWL [9], переработанными под выбранный стандарт. Также модель дополнена некоторыми ограничениями на сущности, которых нет в OWL.

Для удобства работы, а так же возможности расширения количества моделей ТСО, с которыми можно проводить виртуальные эксперименты необходимо обеспечить импорт/экспорт моделей, описанных с помощью онтологий в формате OWL.

Детальная структура модуля импорта/экспорта показана на рис.2:

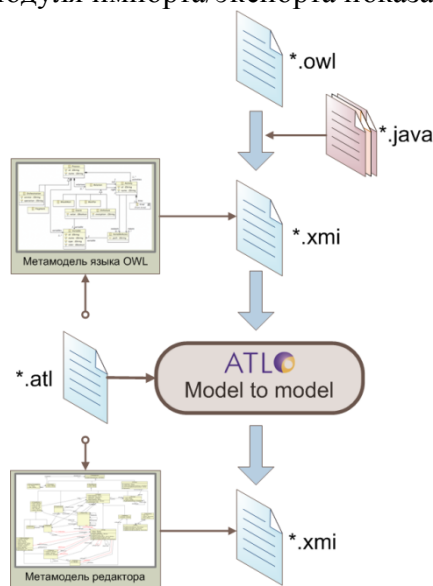


Рис 2 Импорт онтологий в формате OWL

Так как был выбран модель-ориентированный подход, то мы можем воспользоваться его преимуществами и трансформировать модель OWL в модель разрабатываемого языка. Для этого необходимо написать несколько правил трансформации на языке ATL [1]. Однако, формат OWL необходимо сконвертировать в формат XMI. Это решено с помощью отдельного Java модуля. Таким образом, загрузка данных в систему может быть реализована несколькими способами:

- созданы непосредственно в редакторе;
- путем импорта OWL-онтологий.

Во втором случае происходит ATL трансформация из модели OWL в модель разработанного языка.

- Таким образом, в ходе разработки редактора получены следующие результаты:
- разработана мета-модель языка описания структурно-имитационной модели ТСО.
- разработана система визуализации описания семантики ТСО.
- определены возможные пути дальнейшего развития системы.
- получен опыт применения методологии моделиориентированной разработки программного обеспечения.

На рис.3 приведена демонстрация работы приложения.

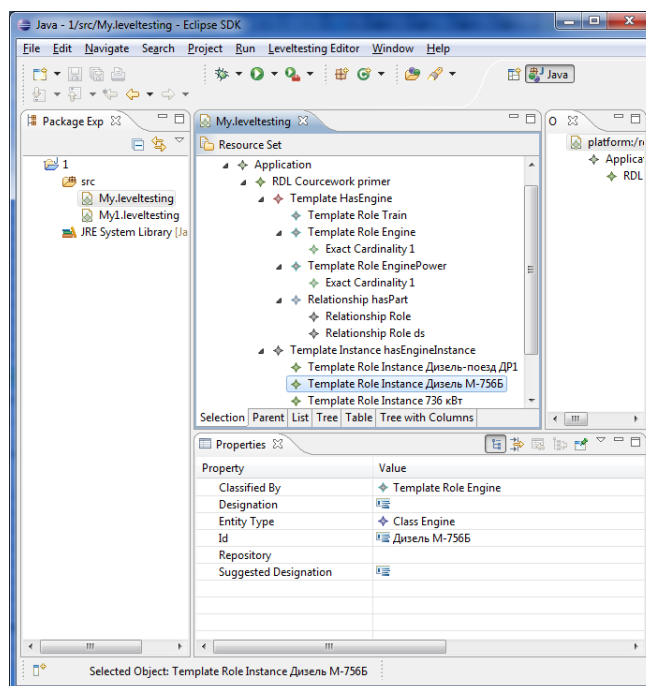


Рис 3. Демонстрация работы приложения

Визуализация описаний ТСО выполнена в виде древовидного редактора-плагина для Eclipse.

В будущем система может быть улучшена различными способами. Система может быть интегрирована с каким-либо свободным языком моделирования, например Modelica. Это позволит описывать функционирование отдельных подсистем не в программном коде, а с помощью специально предназначенного для этого языка.

Возможно применение в системе методов автоматического извлечения формальных моделей программных каркасов и использования этих моделей в составе методов искусственного интеллекта для упрощения разработки процесса преобразования моделей ТСО. Например, некий планировщик может анализировать описания моделей ТСО, формальную модель используемых программных каркасов и предопределенные шаблоны Java-кода для автоматического построения программного модуля, реализующего модель ТСО и привязки состояния трехмерной сцены.

Система может быть дополнена средствами описания не только привязок состояния объектов трехмерной сцены, но и средствами описания привязок к подсистеме обработки ввода, к подсистеме физической симуляции.

Описанные улучшения позволят сделать систему универсальным средством автоматизации разработки систем, использующих трехмерную визуализацию.

Литература

1. ATL - a model transformation technology – Режим доступа: <http://www.eclipse.org/atl/>, свободный
2. Eclipse Modeling Framework Documentation [Электронный ресурс] / The Eclipse Foundations – Режим доступа: <http://www.eclipse.org/modeling/emf/docs/>, свободный.
3. Frank Budinsky, Dave Steinberg, Ed Merks, Ray Ellersick, Timothy J. Grose. Eclipse Modeling Framework. Second Edition, 2009.
4. J. Mukerji, J. Miller, eds., "Model Driven Architecture", 2001.
5. Jon Siegel, Developing in OMG's Model Driven Architecture, 2006.
6. Eclipse Documentation [Электронный ресурс] – Режим доступа: <http://www.eclipse.org/documentation/> свободный.

7. ISO 15926-2. Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities. Part 2: Data model [Текст]. – 2003. – 241 с.
8. MOF Core specification – Режим доступа: <http://www.omg.org/spec/MOF/2.0/>, свободный
9. The World Wide Web Consortium - Режим доступа: <http://www.w3.org/>, свободный.
10. Левенчук А. Факт-ориентированное моделирование [Текст]. – Стенограмма доклада. - Режим доступа: <http://ailev.livejournal.com/699549.html>, свободный.