

# ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ОНТОЛОГИИ ПРЕДМЕТНОЙ ОБЛАСТИ ДЛЯ ОПРЕДЕЛЕНИЯ СЕМАНТИЧЕСКОЙ МОДЕЛИ DSL

**А.А. Курсанова, А.Е. Беляков**

*Южно-Уральский государственный университет, г. Челябинск*

При интенсивном информационном развитии остро становится проблема быстрого и качественного создания информационных систем, а также поддержки уже созданных систем. Затраты на модернизацию и расширение системы зачастую оказываются больше чем при первичном создании этой системы. Существующие паттерны проектирования информационных систем сводят процесс проектирования и модернизации к оперированию внутренними составляющими системы, а не всей системой в целом. В данной статье рассматриваемые подходы применяются в рамках платформы, адаптирующей информационную систему под конкретную предметную область путем преобразования входных данных в виде формального описания предметной области в объектную модель и дальнейшей работы с ней. Рассматривается возможность использования онтологии как инструментария для формального описания предметных областей и рассмотрен собственно процесс перехода от формального описания к объектной модели. Полученная объектная модель затем используется языковым модулем платформы уже для написания бизнес-логики полученной системы.

*Ключевые слова: информационная система, предметная области, языковой модуль, предметно-ориентированный язык, онтология.*

## **Введение**

Все методики проектирования объектно-ориентированных информационных систем сводятся к построению системы из «кирпичиков», их правильному взаимодействию друг с другом и использования шаблонов проектирования (так называемых «паттернов»). Каждый раз при проектировании берутся основные базовые составляющие каждой системы, адаптируются под предметную область заказчика и вписываются в систему. При каждой новой итерации проектирования данный процесс повторяется для каждого отдельного модуля системы или для целых подсистем, из-за чего создание и модификации системы становится нетривиальным процессом.

Если предположить, что будет существовать платформа, для которой можно разработать набор процедур для адаптации информационной системы под конкретную предметную область, то такая платформа сможет подстраиваться под не только новую, но и под постоянно изменяющуюся предметную область.

Принцип работы такой платформы может быть основан на допущении, что информационные системы имеют схожие архитектуры [1, с. 46–48], а также на синергии (суммирующий эффект взаимодействия двух или более факторов, характеризующийся тем, что их действие существенно превосходит эффект каждого отдельного компонента в виде их простой суммы) отдельных компонентов системы, что позволило бы манипулировать не внутри каждого отдельного модуля, а всей системой в целом. Это способствует формированию обширных методик и паттернов, которые позволили бы малыми силами модифицировать систему.

## **1. DSL как инструмент языкового модуля адаптивной платформы создания информационных систем**

В первую очередь такая платформа позволит повысить производительность труда разработчиков системы. Разработчики будут делиться на два типа:

- программисты предметной области, которые создают алгоритмы работы системы в терминах предметной области этой самой системы;
- проектировщики системы, которые бы настраивали систему на работу с предметной областью на базе существующей платформы.

Большинство информационных систем имеют многозвенную архитектуру (рис. 1) за счет наилучшей гибкости и масштабируемости [2, с. 67–81].

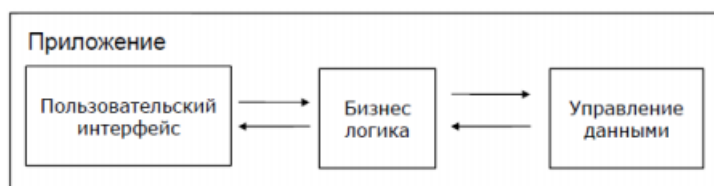


Рис. 1. Трехзвенная архитектура информационной системы

При объектно-ориентированном подходе к проектированию информационных систем также можно использовать и объектно-ориентированную БД (например, MongoDB). Далее при упоминании БД для хранения данных клиентской системы будет подразумеваться объектно-ориентированное хранилище, созданное специально для исследуемой платформы. Такое хранилище помогает полностью отстраниться от работы с конкретной реляционной БД и проектированием соответствующих таблиц, позволяя просто «кидать» и извлекать объекты в хранилище полностью в соответствии с объектно-ориентированным подходом.

Хранилище основано на фиксированном наборе таблиц, хранящих информацию о сохраненных объектах. Однако хранилище имеет ряд упрощений. Во-первых, оно может хранить только объекты. У каждого объекта всегда есть жестко заданный набор идентификаторов типа Guid, и вся ссылочная целостность всегда жестко задана, т. е. если ссылка на объект есть, то нужно сначала удалить объект, на который идет ссылка, а потом уже тот объект.

По сравнению с БД хранилище объектов выглядит гораздо более упрощенным: нет различных типов ссылочной целостности, нет отношений между таблицами и т. д. Однако и работа с хранилищем получается гораздо более упрощенной, что экономит трудозатраты при разработке системы.

Имея объектное хранилище, подсистемой управления базой данных можно пренебречь. Зная, что система состоит из уже известного конечного набора подсистем, становится ясно, что весь процесс разработки информационной системы сводится к ее адаптации к той или иной предметной области.

Любая предметная область – это множество данных реального или виртуального мира в рамках требуемого контекста, представленных с необходимым заказчику уровнем абстракции. Задача платформы состоит в преобразовании поступающих на вход неких описаний предметной области и преобразованию в набор модулей будущей информационной системы, чтобы функционировать в рамках предметной области.

Можно предположить, что функционирование системы будет обеспечиваться взаимодействием расширяемого ряда модулей, например, объектное хранилище, языковой модуль, который делает возможным написание алгоритмов системы на языке предметной области, виртуальная машина для выполнения созданных алгоритмов, интерфейсный модуль для отображения данных и т. д. Этот список в дальнейшем может быть расширяем. Но в данном исследовании рассматривается только языковой модуль предметно-ориентированного языка (ПОЯ или DSL), на котором планируется дальнейшее написание алгоритмов системы, которые затем выполняются виртуальной машиной платформы.

## 2. Семантическая модель DSL

Основная задача проектирования DSL – это отражение семантики предметной области в формальную систему, описываемую синтаксисом и правилами трансформации. Три составляющие любого языка: семантика, синтаксис и правила трансформации. Синтаксис для упрощения нужно брать жестко диктуемый платформой создания системы, семантика же полностью зависит от исходной предметной области.

Но тогда для проектирования системы платформой необходимо иметь на входе четкое формализованное описание предметной области.

Описание предметной области можно оформлять в виде сценариев описания бизнес-процессов (что достаточно универсально), диаграмм SADT (IDEF0, IDEF3, DFD) / ARIS(eEPC и т. д.) / UML (Business Use-case Diagram (BUC) + Activity Diagram + Sequence Diagram) или онтологии.

Основной вопрос формализации семантики до сих пор не решен [3]. Одним из направлений представления семантики является онтология, которая на мой взгляд позволит иметь на входе системы четко формализованное описание предметной области.

В данном исследовании в качестве формата для представления предметной области будет рассматриваться именно онтология – способ представления формализованных данных некой



Класс **OnthologyObject** является абстрактным базовым классом для всех остальных классов онтологии.

**OnthologyClass** хранит метаданные объектов предметной области: свойства и связи между объектами (поведение, прототипы методы).

**OnthologytDataProperty** – класс,с хранящий информацию о свойстве объектов. В нем задается тип свойства и название свойства.

**OnthologytDataPropertyValue** хранит значение свойства объекта.

**OnthologyInstance** – класс-объект онтологии. Объекты этого типа будут являться экземплярами сущностей онтологии.

**OnthologyObjectProperty** – класс, хранящий информацию о взаимосвязях между классами онтологии.

В самой онтологии не хранится информация о строгой типизации отношений между объектами. Характер связи становится понятен по свойствам этой связи (функциональная, транзитивная, инверсивная и т. д.) и по названию связи, суть которой по стандарту должна быть понятна из самого названия отношения, например, «hasParent», «isChildOf» и т. д.

Однако для использования объектной модели предметной области из онтологии в информационной системе все эти отношения между объекту нужно разрешить и дать им формальное описание.

Для этого в классе **OnthologyObjectProperty** введено поле RelationshipTypeId, которое является ссылкой на класс **RelationshipType**, которое отвечает за тип отношения между объектами в предметной области.

Логика работы отношений объектов предметной области базируется на двух шаблонах проектирования: «Стратегия (Strategy)» и «Посредник (Mediator)» (рис. 3).

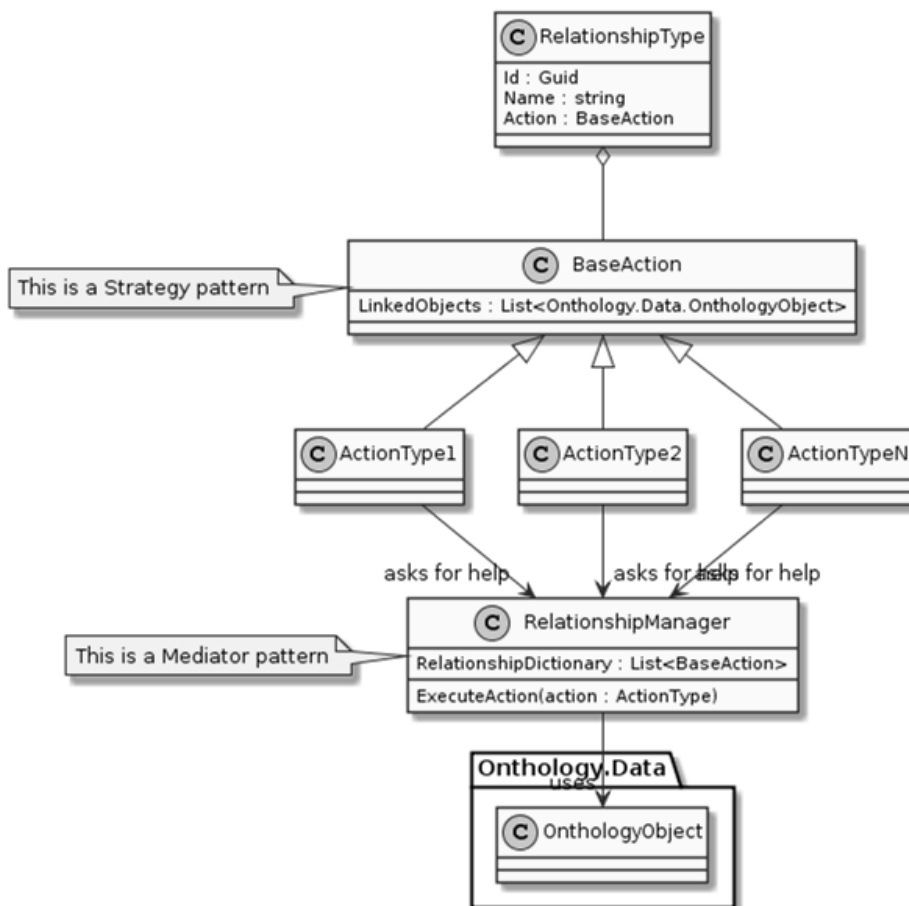


Рис. 3. Диаграмма классов модели для разрешения отношений между объектами предметной области

## Краткие сообщения

---

Шаблон «Стратегия» позволяет объектам не знать точно, какое поведение они будут реализовывать. Плюсом данного шаблона является, что объекты не зависят от исполняющих классов, и что поведение объектов можно менять «на лету» при необходимости. Суть заключается в том, что объект хранит интерфейс того или иного отношения, но не его реализацию. Сами объекты не знают, какое именно поведение они реализуют [4, с. 300–309].

Паттерн «Посредник» обеспечивает слабую связь между взаимодействующими друг с другом объектами. Так как заведомо мы не знаем какого рода отношение может быть между объектами, какой алгоритм поведения задаст настройщик системы, то жесткие связи между объектами могут явиться препятствием в создании информационной системы [4, с. 263–272].

Таким образом, имея множество различных реализаций, все из которых должны реализовывать тот или иной интерфейс, все разрешения отношений между объектами отправляются в класс **RelationshipManager**, только этот класс хранит информацию обо всех отношениях в системе, обо всех объектах, которые реализуют те или иные отношения, и отвечает за корректное взаимодействие объектов друг с другом.

### Заключение

Проведен анализ требований к платформе, нацеленной на адаптацию информационных систем к конкретной предметной области. Для данной платформы сформулированы входные данные, необходимые для проведения процедур по адаптации информационной системы к конкретной предметной области. Выбран инструментарий для формального описания предметных областей и рассмотрен собственно процесс перехода от формального описания предметной области в объектную модель платформы, которой оперирует языковой модуль платформы для написания алгоритмов бизнес-логики информационной системы.

### Литература

1. Фаулер, М. Шаблоны корпоративных приложений / М. Фаулер. – М.: Вильямс, 2012.
2. Таненбаум, Э. Распределенные системы. Принципы и парадигмы / Э. Таненбаум. – СПб.: Питер, 2007.
3. Попов, М.Ю. Реализация семантико-синтаксического анализатора естественного языка в системе классификации полнотекстовых текстов / М.Ю. Попов, С.А. Фоменков, А.В. Заболева-Зотова // Успехи современного естествознания. – 2004. – № 5.
4. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влссидес. – СПб.: Питер, 2007.

**Кирсанова Александра Александровна**, программист Вычислительного центра, преподаватель кафедры электронных вычислительных машин, Южно-Уральский государственный университет, г. Челябинск; alexander.a.kirsanov@gmail.com.

**Беляков Александр Евгеньевич**, программист Вычислительного центра, преподаватель кафедры электронных вычислительных машин, Южно-Уральский государственный университет, г. Челябинск; alex.beliakov@susu.ru.

*Поступила в редакцию 18 марта 2016 г.*

## POSSIBLE USE OF ONTOLOGY AS A TOOL OF DESCRIBING DSL SEMANTIC MODEL

Kirsanova A.A., alexander.a.kirsanov@gmail.com,

Belyakov A.E., alex.beliakov@susu.ru

South Ural State University, Chelyabinsk, Russian Federation

With intensive information evolution of our society the problem of fast and qualitative information system development and support becomes more and more topical. Expenses to update and scale existing systems turn out to be even more than that to create one from scratch. In this article approaches under review are described within the adaptive platform that adapts information system to specific domain by the means of transforming incoming domain formal description into the object model and then manipulating it. The possibility of using an ontology as a tool for domain formal description and the process of transforming ontology into the object model is described. The resulting object model is then used by language module of the platform to describe system business logic.

*Keywords: information system, domain, language module, domain-specific language, ontology.*

### References

1. Fowler M. *Shablony korporativnykh prilozheniy* [Patterns of Enterprise Application Architecture]. Moscow, Williams, 2012.

2. Tanenbaum E. *Raspredelemnnye sistemy. Printsipy i paradigmy* [Distributed Systems. Principles and Paradigms]. St. Petersburg, Piter, 2007.

3. Popov M.Y., Fomenkov S.A., Zaboleeva-Zotova A.V. [Realization of Semantic Syntaxes Analyzer of Natural Language in Classification System of Full Text Copies]. *Uspekhi sovremennogo estestvoznaniya* [Achievements of Modern Natural Science], 2004, no. 5. (in Russ.)

4. Gamma E., Helm R., Johnson R., Vlissides J. *Priemy ob"ektno-orientirovannogo proektirovaniya. Patterny proektirovaniya* [Elements of Reusable Object-Oriented Software]. St. Petersburg, Piter, 2007.

*Received 18 March 2016*

### ОБРАЗЕЦ ЦИТИРОВАНИЯ

Кирсанова, А.А. Возможности использования онтологии предметной области для определения семантической модели DSL / А.А. Кирсанова, А.Е. Беляков // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». – 2016. – Т. 16, № 2. – С. 154–159. DOI: 10.14529/ctcr160218

### FOR CITATION

Kirsanova A.A., Belyakov A.E. Possible Use of Ontology as a Tool of Describing DSL Semantic Model. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*, 2016, vol. 16, no. 2, pp. 154–159. (in Russ.) DOI: 10.14529/ctcr160218