РАЗРАБОТКА СРЕДСТВ ПОСТРОЕНИЯ И ИСПОЛЬЗОВАНИЯ ОНТОЛОГИЙ ДЛЯ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Д.В.Абрамов[2], В.В.Андреев[1], Е.В.Симонова[2], П.О.Скобелев[2]

[1]Институт проблем управления сложными системами РАН
443020, Самара, ул. Садовая, 61, Россия
тел: +7 (8462) 32-39-27, факс: +7 (8462) 33-27-70

[2]ООО НПК «Маджента Девелопмент»
443110, Самара, ул. Осипенко 1а, 5-й этаж
abramov@magenta-technology.ru
тел: +7 (8462) 70-66-84; 70-66-85

тел: +7 (846-2)706-684, +7 (846-2)706-685

Ключевые слова: *онтология, бизнес знания, декларативные знания, процедурные знания, редактор онтологий, мультиагентные системы*

Abstract
The article describes tools for development and usage ontologies for decision making support. Main parts of the ontology constructor are considered. Main functions to work with the ontology are given.

Введение

Открытый характер современного информационного общества и глобальной рыночной экономики обусловливает необходимость разрабатывать новые методы и средства организации и управления, направленные на решение задач управления, связанных с динамическим перераспределением ресурсов при постоянно изменяющемся рынке.

При этом отметим, что традиционные системы поддержки принятия решений, основанные на точных математических моделях, оказываются малоэффективными, постольку являются почти не адаптируемыми к изменениям окружающего мира. Для адекватного отображения динамической картины мира становятся востребованы системы управления знаниями, позволяющие системам принятия решений оперативно реагировать на изменение ситуации.

Все это приводит к необходимости появления качественно новых систем принятия решений, функционирование которых будет основано на изменяющихся знаниях о мире. В такой постановке особую значимость приобретают системы управления знаниями, позволяющие строить, изменять и оперировать с моделями реального мира.

В данной статье мы рассмотрим механизм представления знаний – онтологию, и систему конструирования онтологий, ориентированную на поддержку систем принятия решений.

1 Использование онтологии в процессе принятия решений

Онтология – это формализованные концептуальные знания о предметной области. Концептуальность знаний онтологии означает, что эти знания формулируются в терминах основных концептов (наиболее общих понятий и отношений), описывающих фрагменты окружающего мира.

При проектировании онтологии деятельности предприятия все необходимые нам знания мы можем разделить на:

- знания предметной области;
- знания, относящиеся к методу решения задачи или принятия решений.

Таки образом, онтология деятельности предприятий представляет набор двух миров – мира предметной области, в какой то мере внешнего по отношению к системе и определяющего объекты реального мира, которыми мы оперируем при принятии решения, и мира, в котором отражены знания о том, каким именно образом мы принимаем решения. При этом одним из достоинств такого построения является то, что для одних и тех же знаний предметной области мы можем применять различные знания о методе решения, изменяя тем самым поведение системы.

Ключевыми понятиями онтологии принятия решений являются: «цели», «критерии принятия решений», «предпочтения», «объекты» и «субъекты» деятельности, «сценарии действий», «стратегии», «результаты действий». В онтологии предприятия мы можем определять конкретные цели, стоящие перед нами, декларировать возможные пути их достижения, критерии, которыми мы руководствуется при принятии решения, взаимосвязи между различными понятиями. При этом определенные критерии будут связаны с целью, выражая тем самым то, что успешность достижения цели определяется данными критериями и т.п. Отметим, что в онтологии не задается четкий алгоритм решения поставленных задач, а определяются возможные пути получения решения с указанием свойственных каждой альтернативе компромиссов.

Система поддержки принятия проектируется способной понимать и рассуждать в рамках описанной модели, другими словами, она, опираясь на знания, описанные в онтологии, осуществляет поиск решений в соответствии с нашими целями и предпочтениями, находя определенный баланс между различными компромиссами, такими, как например, стоимость, удовлетворенность клиентов, риск и т.п.

2 Онтологический базис

Система представления знаний «модель мира» предлагает для пользователя такие базовые концепты как «объект», «сценарий действий», «отношение» и «атрибут». При этом объекты определяются как набор сценариев действий и набор атрибутов, каждый сценарий задается своим набором свойств и телом, состоящим из набора действий, отношения трактуются как определенные виды связей между всеми другими концептами, а атрибуты задаются набором типичных элементарных компонент типа «символьная строка», «целое число» и т.д. Эта модель получила условное название «модель Аристотеля», который первым определил «объекты – суть свойства». Данный онтологический базис является наиболее общим и позволяется описывать любые знания, не ориентируясь на их специфику.

Для описания предметно-ориентированных знаний предлагается использовать расширения онтологического базиса – механизм метаконцептов, т.е. синтетических понятий сконструированных на базисе уже имеющихся. Оперируя понятиями «объект», «сценарий действий», «отношение» и «атрибут», мы можем строить на основе них более сложные такие так: «цели», «критерии принятия решений», «ограничения», «объект» и «субъект» деятельности, формируя тем самым метаонтологию мира принятия решений.

Расширяя данные понятия (создавая наследников) пользователь может вводить конкретные цели, критерии их достижения и т.д., тем самым конструируя релевантную для него модель принятия решений.

Подчеркнем, что создание «миров знаний» кардинально отличается от построения традиционных моделей знаний предметной области (например, принятых в экспертных системах), поскольку использует совершенно иную систему представления знаний, базирующуюся на встроенных понятиях (онтологический базис и расширяющие его метапонятия) о пространстве и времени, сценах и сценариях действий, а также другую логику рассуждений.

3 Программные средства для управления онтологиями (Ontology Management Toolset)

Ontology Management Toolset (OMT) – это библиотека инструментальных программных средств Java для поддержки создания, редактирования и навигации по онтологии и для создания и редактирования сцен, основанных на онтологиях.

OMT позволяет:

- проектировать онтологии в виде семантических сетей. Используя OMT, разработчик и пользователь может создавать и редактировать онтологии, специфицируя свои концепты (классы) и устанавливая связи между ними, и также, формируя сценарии действий;
- создавать сцены и работать с ними. Сцена – это набор экземпляров классов, описывающих некоторую ситуацию в мире. Каждый из экземпляров набора связан с некоторым концептом онтологии отношением вид-род;
- работать с экземплярами классов и экземплярами отношений;
- осуществлять навигацию по семантическим сетям онтологий и сцен (получение связанных сущностей, проверка наличия пути и т.д.). При этом функции по навигации доступны как разработчикам с помощью программного интерфейса приложения (API), так и в методах концептов.
- хранить онтологии и сцены в реляционной базе данных, либо в XML хранилище;
- импорт/экспорт онтологий в XML.

В архитектуре OMT можно выделить следующие компоненты (рисунок 1):

- хранилища онтологий и онтологических сцен, объединяющие в себе все знания компании;
- средства отображения и навигирования по онтологиям и сценам, позволяющие пользователям просматривать корпоративные знания (поддерживается отображение онтологии в виде словаря понятий, либо семантической сети);
- средства редактирования онтологий и сцен;
- ядро системы, которое обеспечивает всю работу с онтологиями и сценами, при этом с использованием определенного API конструктор онтологий может интегрироваться в другие системы и использоваться в них.

4 Представление процедурных знаний

Для представления «сценариев действия» в OMT используются понятие метода концепта. Метод концепта является понятием аналогичным методу в объектно-ориентированном программировании. Для каждого метода определяется имя, список параметров и возвращаемое значение. Тело метода кодируется на языке Java, что, на наш взгляд, выгодно отличает OMT от других конструкторов онтологий, имеющих собственный язык для представления процедурных знаний, таких как OntoEdit, Protégé и т.п., поскольку:

- Java является широко распространенным языком, поэтому подготовка специалистов займет меньше усилий и затрат, нежели в случае разработки и использования собственного процедурного языка;
- использование Java позволяет в OMT применять подход, при котором по онтологии генерируются java классы, которые потом используются в приложении. Это значительно увеличивает скорость исполнения процедурных частей по сравнению с интерпретируемыми языками других конструкторов онтологий.

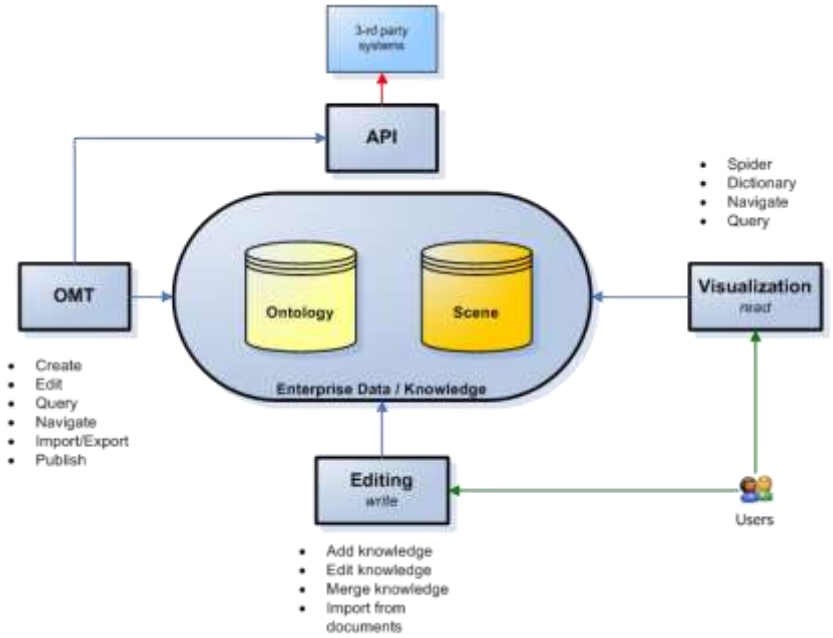

Рисунок 1 – Архитектура OMT

Также для представления процедурных знаний в OMT в качестве расширения онтологического базиса, предлагается для формализации процедурных знаний использовать модель активности, аналогичную определяемой в стандарте UML 2.0 Activity Diagram. С этой целью в онтологию в качестве метапонятий вводятся: действие (action) и переход между действиями (transition).

Действие представляет собой онтологический класс, у которого определен абстрактный метод execute(). Переход между действиями представляет собой отношение, связывающее два действия, и условие применимости перехода в виде метода isApplicable().

В дальнейшем, онтолог может создавать наследников от перечисленных выше метаконцептов, тем самым специфицируя конкретные действия, и соединять их друг с другом (с помощью отношений перехода). Полученная семантическая сеть и определяет модель активности.

Главными преимуществами такого подхода к спецификации процедурных знаний являются следующие:

- знания формализуются в виде сети (диаграммы), что значительно упрощает понимание пользователем описанной логики работы и обеспечивает простое редактирование частей алгоритма (отдельных действий);
- возможно создание предметно-ориентированной библиотеки действий, из которой, как из кирпичиков, может собираться конкретная модель деятельности.

5 Особенности реализации

Одним из основных упоров при создании конструктора онтологий являлось создание среды управления знаниям масштаба предприятия. При этом необходимыми условиями успешного функционирования такого рода систем являются:

- Масштабирование;
- Надежность;

- Безопасность;
- Кроссплатформенность;
- Интеграция и совместимость;
- Распределенность.

С этой целью для хранения онтологий и онтологических сцен в ОМТ используется реляционная база данных. Использование базы данных обеспечивает такие свойства как масштабируемость (хранение и оперирование большими объемами данных, что характерно для ряда предметных областей и применений [4]), надежность и распределенность. При этом стоит отметить, что только ограниченное число известных нам конструкторов онтологий, такие как OntoEdit, Protégé, WebODE, обеспечивают хранение онтологий и сцен в базе данных.

Для обеспечения работы с большими объемами данным к ОМТ предусмотрены средства частичной загрузки сцен, выгрузки неиспользуемых частей из памяти, управление политикой использования памяти (соблюдение баланса между производительностью и потреблением ресурсов).

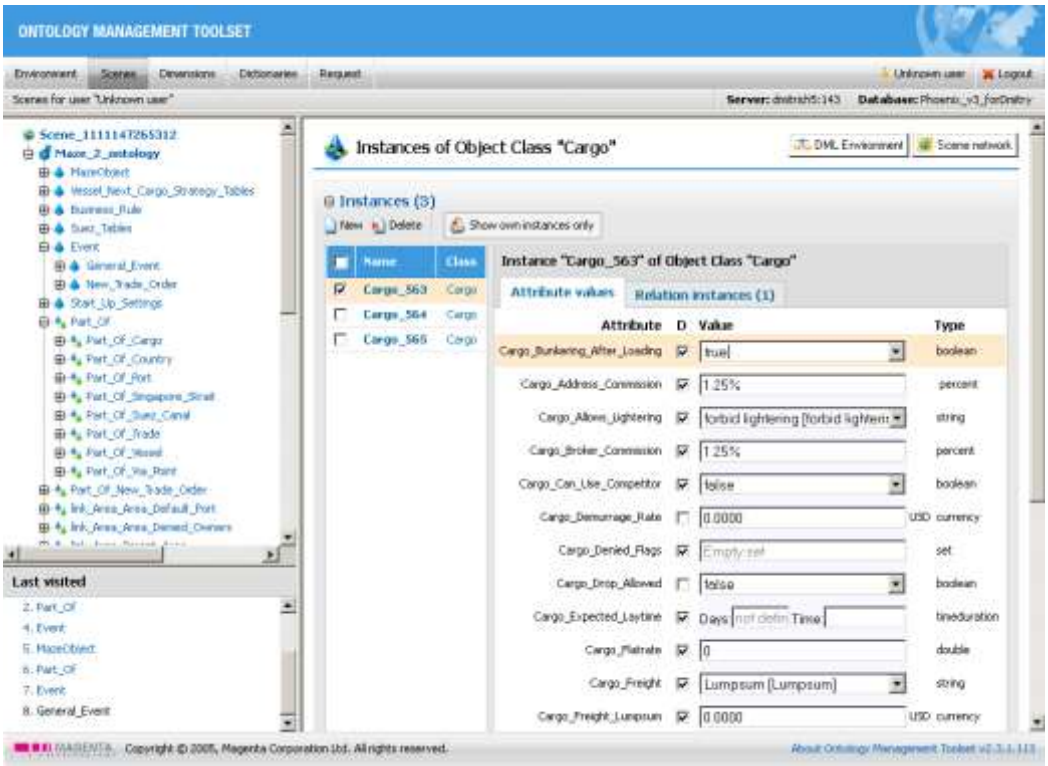

Рисунок 2 – Редактор онтологий и сцен

7 Среда для разработки онтологий

Интерфейс ОМТ предоставляет функции по визуальному конструированию онтологий и сцен (рисунок 2). Поддерживаются представления в виде дерева иерархии концептов и в виде семантической сети. Конструктор онтологий является web приложением и для своей работы требует только наличия Интернет браузера. Данная особенность конструктора онтологий

обеспечивает простой и легкий доступ к корпоративным знаниям, возможность работы с корпоративной базой знаний находясь в любой точке мира.

8 Перспективы развития и применение

Возможности конструктора онтологий используются в целом ряде систем, таких как система поддержки принятия решений по планированию грузоперевозок в морской логистике, система поддержки управления распределением ресурсов в проектах предприятий [5], система поддержки принятия решений по планированию сухопутных грузоперевозок [3].

Онтология в перечисленных приложениях служит для описания предметной области задачи, ограничений существующих в реальном мире, бизнес логики принятия решений.

В рамках развития ОМТ ведутся следующие работы:

- расширение функциональности языка запросов к сценам и онтологиям, поддержка логического вывода в запросах. Поддержка языка запросов RDQL;
- создание средств работы с распределенными онтологиями (слияние, выравнивание онтологий, более развитая поддержка совместной разработки онтологий);
- введение поддержки стандартов представления онтологий, таких как RDF, OWL;
- интеграция конструктора онтологий с Java IDE;
- разработка средств автоматического построения онтологий посредством анализа различных информационных источников (текстовых документов, баз данных предприятия и т.п.).

Заключение

Для работы в условиях высокой динамики и глобальной рыночной экономики открытые системы должны быть способны приобретать новые бизнес знания, чтобы гибко реагировать на изменяющиеся условия среды и адаптироваться к изменениям этой среды в процессе своего функционирования.

Возможности построения описания миров, добавления новых знаний в процессе работы, проверка и контроль вводимых знаний, анализ сценариев с точки зрения логики построения системы, интеграция знаний разных пользователей, а также способность работы с большим объемом данных - все эти особенности делают конструктор онтологий уникальной системой, помогающей при решении комплексных задач, в условиях динамичного окружения с множеством объектов, тем самым давая необходимый базис для систем поддержки принятия решений.

Список литературы

[1] В.Андреев, К.Ивкушкин, И.Минаков, Г.Ржевский, П.Скобелев Конструктор онтологий мультиагентных систем. // Труды III Международной конференции по проблемам управления и моделирования сложных систем, Самара, 4-9 сентября 2001. – Самара: СНЦ РАН, 2001, с. 480-488

[2] В. Андреев, К. Ивкушкин, И. Минаков, Г. Ржевский, А. Сафронов, П. Скобелев. Основные компоненты внутреннего устройства мультиагентной системы. // Труды 5-ой Международной конференции по проблемам управления и моделирования сложных систем, Самара, 17-21 июня 2003. – Самара: СНЦ РАН, 2003, стр. 304 - 316.

[3] Д. Абрамов, В.Андреев, С. Батищев, Т. Искварина, Д. Карягин, А. Сафронов Построение онтологий для решения задач транспортной логистики// Труды 6-ой Международной конференции по проблемам управления и моделирования сложных систем, Самара, 14-17 июня 2004. - Самара: СНЦ РАН, 2004, стр. 246 - 253.

[4] В.Андреев, С. Батищев, К. Ивкушкин, Т. Искварина, П. Скобелев Инструментальные средства для разработки мультиагентных систем промышленного масштаба// Труды 6-ой Международной конфе-

ренции по проблемам управления и моделирования сложных систем, Самара, 14-17 июня 2004. - Самара: СНЦ РАН, 2004, стр. 233 - 240.
[5] Иващенко А.В., Ивкушкин К.В., Никифоров В.А., Скобелев П.О Мультиагентная система поддержки управления распределением ресурсов в проектах предприятий. // Настоящий сборник
[6] Виттих В.А. Онтологии сложных систем – организаций. // Препринт ИПУСС РАН, № 3, Самара, 1998 - 15 с.

Published: Абрамов Д. В., Андреев В. В., Симонова Е. В., Скобелев П. О. Разработка средств построения и использования онтологий для поддержки принятия решений. Труды VII Международной конференции “Проблемы управления и моделирования в сложных системах”. Самара. 2005. – С. 435-440.