

7. Мартыненко Г.Я. Ритмико-смысловая динамика русского классического сонета. СПб., 2004.
8. Мартыненко Г.Я. Футбольная таблица как ценологическое распределение. Общая и прикладная ценология. № 7, М., 2007.
9. Марусенко М.А. Методы атрибуции анонимных и псевдонимных текстов. Л., 1990.
10. Пропт В.Я. Морфология волшебной сказки. Исторические корни волшебной сказки. М., 1998.
11. Шерстинова Т.Ю. Построение интонационных моделей для языков с малоизученной фонетической системой // Материалы XXXVI межд. филолог. конф. СПб.: СПбГУ, филологический факультет. Секция «Полевая лингвистика. Интегральное моделирование звуковой формы естественных языков». 2007 (в печати)
12. Шрейдер Ю.А., Шаров А.А. Системы и модели. М.: Наука, 1972.
13. Marbe K. Über den Rhythmus der Prosa. Giessen, 1904.

КОМПОНЕНТЫ ОНТОЛОГИЧЕСКИХ СИСТЕМ И ИХ РЕАЛИЗАЦИЯ В СОВРЕМЕННЫХ ПРОЕКТАХ

Л. М. Пивоварова, В. Ш. Рубашкин

Санкт-Петербургский государственный университет

Санкт-Петербург

Данная работа является обзором основных точек зрения на представление знаний и управление знаниями в онтологических системах. В настоящее время уже существует несколько обзоров онтологических систем (см., например [1][12][6][4]). Помимо того, что подобные обзоры быстро устаревают, все они различаются как по своим исходным позициям, так и по выбору характеристик, которые авторы считают существенными при сопоставлении различных систем. Мы считаем наиболее важными следующие аспекты:

- Базовый язык представления знаний и аксиоматика онтологии.
- Наполнение онтологии знаниями и интеграция накопленных ранее знаний.
- Ограниченный логический вывод.
- Связь онтологии с естественным языком.
- Инструментарий для работы с онтологиями (онторедатор).

Хотя все перечисленное важно для создания полнофункциональной онтологической системы, реальные проекты нередко сосредотачивают внимание только на некоторых из этих пунктов. Далее рассматриваются четыре крупных и, по-видимому, наиболее авторитетных онтологических проекта: Protégé [11], SUMO (Suggested Upper Merged Ontology) [13], KAON2 (The KArllsruhe ONtology and Semantic Web tool suite 2) [5] и CYC [2]. Эти системы по-разному функционально ориентированы, и их трудно сравнивать по единым критериям. Если говорить об основной функциональной ориентации, то Protégé — онторедатор; SUMO — универсальная онтология; KAON (далее рассматривается последняя версия KAON2) — система логического вывода и ответов на запросы к концептуальной системе, CYC — технология, направленная на поддержку бизнес-процессов и создание интеллектуальных агентов. В то же время, разнонаправленность этих систем позволяет осветить различные аспекты инженерии онтологий.

Protégé — это онторедатор и инструмент для построения баз знаний, созданный на медицинском факультете стенфордского университета группой медицинской информатики (Stanford Medical Informatics) под руководством профессора Марка Мьюсена [10].

Protégé не имеет никакой встроенной онтологии и рассчитан на создание прикладных онтологий с нуля. Более того, разработчики Protégé считают методически правильным начинать разработку онтологии, «подвешивая» понятия прикладного характера (*Пицца, Страна, Вино*) непосредственно, как подклассы понятия *Нечто* [4][9], и тем самым фактически игнорируют проблему общей организации концептуальной модели. Связь с естественным языком также не предусмотрена, хотя Protégé позиционируется как встраиваемый компонент для работы в системах извлечения знаний [6].

Несмотря на эти недостатки, Protégé активно используется в различных прикладных проектах, т. к. позволяет быстро и относительно просто сконструировать небольшую предметную онтологию и имеет расширяемую архитектуру, которая позволяет легко встраивать его в прикладные программы.

Языком описания онтологии в Protégé является OWL (либо еще более упрощенная фреймовая структура). Соответственно, могут быть реализованы основные выразительные возможности OWL — различение индивидов, классов и свойств; иерархия классов и свойств; указание области применения и области значений для объектных свойств и др. Индивид может принадлежать нескольким классам одновременно или не принадлежать ни одному классу. Также существует возможность указать, что пара или группа классов являются взаимоисключающими. Классы могут не только описываться, но и определяться через указание его необходимых и достаточных свойств.

SUMO (Suggested Upper Merged Ontology, Рекомендованная онтология верхнего уровня интеграции) — это бесплатная, свободно распространяемая онтология, принадлежащая IEEE (Institute of Electrical and Electronics Engineers). SUMO не предназначена для решения какой-либо конкретной задачи и преследует цель интеграции существующих онтологий в единую, всеобъемлющую, способную к расширению структуру, которая имела бы статус универсального стандарта и могла бы использоваться в различных прикладных и исследовательских проектах [8].

SUMO является онтологией верхнего уровня и содержит около 1000 понятий и 5000 аксиом, из них около 800 правил. Более конкретные понятия хранятся в отдельных отраслевых онтологиях и могут подключаться по мере необходимости. Для связи между SUMO и отраслевыми онтологиями разработана онтология среднего уровня MILO (Mid-Level Ontology). Весь доступный на сегодня комплекс онтологий содержит около 20000 понятий и 60000 аксиом [13].

Самым общим понятием SUMO является *Сущность (Entity)*. Сущности разделяются на *Физическое* и *Абстрактное*. К категории *Абстрактного* в онтологии SUMO относятся такие понятия как *Количество*, *Атрибут*, *НаборИлиКласс*, *Отношение*, *Суждение*, *Граф* и *ЭлементГрафа*. *Физическое* разделяется на *Объекты*, *Процессы* и *НосителиСодержания (ContentBearingPhysical)*. В онтологии SUMO допускается множественное наследование [8].

Изначально SUMO разрабатывалась на диалекте языка KIF, в настоящее время выполнен ее перевод на OWL. Разработана полная система соответствий понятий SUMO с лексическими единицами WordNET (для английского языка). SUMO является онтологией в чистом виде и не имеет ни достаточно развитого онторедатора, ни машины логического вывода; создатели SUMO предоставляют лишь информацию, которая может обрабатываться программно и включаться в качестве составной части в различные приложения и обрабатываться средствами этих приложений.

KAON2 (The Karlsruhe ONtology and Semantic Web tool suite 2) — это инфраструктура для управления онтологиями, созданная в университете Карлсруэ при сотрудничестве с Манчестерским университетом. KAON2 позволяет управлять онтологиями, имеет программный интерфейс для взаимодействия с другими приложениями, может использоваться в качестве онтологического сервера, обладает машиной логического вывода и механизмом извлечения знаний из реляционных баз данных [5]. Система KAON2 также не разрабатывалась под какую-либо конкретную задачу, однако интересы ее разработчиков лежат в области Semantic Web и бизнес-процессов.

Ни проект KAON2, ни другие проекты, осуществляемые в группе по управлению знаниями университета Карлсруэ, не ставят своей целью разработку онтологий, т. е. непосредственно концептуализацию. Создателей KAON'a интересует не столько содержание онтологии, сколько ее формальная структура. С точки зрения структуры, принципиальным является разделение онтологии на терминологическую часть (TBox) и часть данных (ABox, «А» означает «assertions», утверждения). Поскольку проект ориентирован, главным образом, на разработки в области Semantic Web, наиболее значимыми оказываются онтологии с компактной Т-частью и большой А-частью (большое количество документов аннотируется на основе относительно простой онтологии [7]).

С вычислительной точки зрения основной функцией KAON2 является логический вывод и ответы на запросы о составе и свойствах концептуальной системы, поэтому модель знаний KAON2 наиболее детально проработана и единственная из всех рассмотренных в данном обзоре имеет логико-теоретическое обоснование.

KAON2 обладает интерфейсами для импорта онтологий на языках OWL-DL, SWRL и F-Logic, однако для задач логического вывода используется собственный внутренний язык, основанный на клаузах Хорна. В процессе ответа на запросы Т-часть онтологии переводится в программу логического типа, которая затем исполняется, используя А-часть (аксиомы) в качестве данных [7].

CYC [3] — это многоцелевая база знаний и машина логического вывода, направленная на семантизацию программного обеспечения за счет базовых интуитивных знаний (здорового смысла). CYC является коммерческой технологией, ориентированной, в первую очередь, на использование в бизнес-процессах — для таких задач как интеграция разнородных баз данных, интеллектуальный поиск, распределенный искусственный интеллект и др.

База знаний Cys описывается на языке CysL, который является гибридным языком, сочетающим в себе свойства фреймов и логики предикатов [4]. CysL различает индивиды, классы, предикаты и функции. Предикаты могут иметь любое количество аргументов, при этом основное содержание онтологии — это предложения, сконструированные из предикатов и представляющие некоторые сведения о реальном мире. Предпочтение отдается не столько фактам (предполагается, что их интеллектуальный агент получает из баз данных, Интернета и других источников), сколько знаниям, необходимым для их понимания [10]. Различие между коллекциями и индивидами определяется лишь применяемыми к ним предикатами: коллекции выстраиваются в иерархию с помощью предиката *IS_A*, а индивиды связываются с коллекциями через предикат *быть экземпляром* [2].

На глобальном уровне к онтологии не предъявляются требования непротиворечивости — поскольку «наивные» человеческие представления, которые она призвана отражать, также противоречивы. Вместо этого онтология разбивается на отдельные микротеоии, которые выстраиваются в иерархию, при этом число уровней иерархии достигает 50. Все утверждения в микротеоии опираются на некоторые общие допущения и не должны противоречить друг другу.

Верхний уровень онтологии содержит микротеоии, описывающие индивиды, коллекции, логические отношения; далее — пространство, время, движение, действие и т. п. Средний уровень — астрономия, физические объекты, химия, география, погода, экология, растения и животные, социальное поведение, организации, политика, бизнес, военное дело и др. Еще ниже расположены доменные минитеоии. В настоящее время онтология СУС содержит около 400 тысяч концептов и более 3,5 миллионов утверждений [2].

Важнейшим компонентом базы знаний СУС является **лексикон**, также построенный на основе WordNET. СУС не является онтологией значений слов: не все концепты имеют ссылку на обозначающее их слово и, наоборот, не все значения слов лексикона отображены в онтологии. Концепты (значения) добавляются в онтологию только в том случае, если это необходимо для поддержки логического вывода на основе здравого смысла. Хотя содержательно лексикон сильно отличается от онтологии, с формальной точки зрения он является никак не выделяемой частью базы знаний и описывается на том же языке CycL, что и основная онтология.

Таким образом, можно констатировать, что крупные онтологические проекты с точки зрения их функциональности не столько конкурируют, сколько дополняют возможности друг друга. Одновременно можно, видимо, констатировать, что пока не существует проекта, который мог бы использоваться как готовый и наполненный знаниями полнофункциональный ресурс, готовый как для непосредственного использования, так и для дальнейшего развития в широком спектре интеллектуальных информационных технологий.

Работа выполнена при финансовой поддержке РФФИ (проект № 06-06-80434).

ЛИТЕРАТУРА

1. Овдей О. М., Проскудина Г. Ю. Обзор инструментов инженерии онтологий // Труды Шестой Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции», 2004
2. Cyc 101 Tutorial [Электронный ресурс]. — Режим доступа: <http://www.opencyc.org/doc/tut/>
3. Cycorp, Inc. [Электронный ресурс]. — Режим доступа: <http://www.cyc.com>
4. Gomez-Perez A., Fernando-Lopez M., Corcho O. Ontological Engineering with examples from the areas of Knowledge management, e-Commerce and the Semantic Web – 2nd. Ed. – London, Springer-Verlag – 2004
5. KAON2 — Ontology Management for the Semantic Web [Электронный ресурс]. — Режим доступа: <http://kaon2.semanticweb.org/>.
6. Mizoguchi R. Ontology Engineering Environments // Handbook on Ontologies; ред. Staab S., Studer R. – Berlin, Springer-Verlag — 2003
7. Motik B., Sattler U. A Comparison of Reasoning Techniques for Querying Large Description Logic ABoxes // Proc. of the 13th International Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR 2006) — Phnom Penh (Cambodia) — 2006
8. Niles, I., and Pease, A. 2001. Towards a Standard Upper Ontology. // Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001); ред. Welty C., Smith B — Ogunquit (USA) — 2001.
9. Noy, N. F., McGuinness, D. L. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report - 2001.
10. Panton K. et. al Common Sense Reasoning — From Cyc to Intelligent Assistant // Ambient Intelligence in Everyday Life, pp. 1-31, LNAI 3864, Springer, 2006.
11. The Protégé Ontology Editor and Knowledge Acquisition System [Электронный ресурс]. — Режим доступа: <http://protege.stanford.edu/>.
12. L. Stojanovic, B. Motik Ontology Evolution within Ontology Editors // Proceedings of EON2002 Evaluation of Ontology-based Tools, 2002.
13. Suggested Upper Merged Ontology (SUMO) [Электронный ресурс]. — Режим доступа: <http://www.ontologyportal.org/index.html>.