

ПРИНЦИПЫ УПРАВЛЕНИЯ ОНТОЛОГИЯМИ, ИСПОЛЬЗУЕМЫЕ В СРЕДЕ ИНТЕГРАЦИИ ЗНАНИЙ

Т. В. Левашова

Санкт-Петербургский институт информатики и автоматизации РАН
199178, Санкт-Петербург, 14 линия, 39
oleg@mail.iias.spb.su

УДК 681.3

Т. В. Левашова. Принципы управления онтологиями, используемые в среде интеграции знаний // Труды СПИИРАН. Вып. 1, т. 2. — СПб.: СПИИРАН, 2002.

Аннотация. *Рассмотрены основные подходы к разработке онтологий, перечислены требования к онтологиям, операции работы с онтологиями и методы их реализации. — Библ. 38 назв.*

UDC 681.3

T. V. Levashova. Principles of ontology management used in the knowledge integration environment // SPIIRAS Proceedings. Issue 1, v. 2. — SPb.: SPIIRAS, 2002.

Abstract. *Main approaches to ontologies development are considered, requirements to ontologies, actions on ontologies and methods of their implementation are described. — Bibl. 38 items.*

Введение

Развитие информационных технологий и связанная с этим необходимость хранения больших объемов информации, привели к возникновению большого числа разрозненных баз данных и знаний, создаваемых по мере их необходимости для решения конкретных проблем. Это привело к появлению такой дисциплины в информатике, как управление знаниями, занимающейся организацией знаний. Попытки соединить накопленные знания с целью их повторного использования, то есть интегрировать знания, в том числе разнородные, привели к необходимости разработки методов, направленных на анализ содержимого знаний. Способ представления знаний, для которого применим интеллектуальный анализ, предлагают онтологии.

Онтологиями называются смысловые теории о разновидностях, свойствах объектов и связях между ними. Они предоставляют терминологию для описания предположительно существующих знаний.

Понятие онтологии в информатике возникло из необходимости компьютерного представления возможных видов знаний, попыток описать знания человека об окружающем его мире и представить его в структурированном виде. Знания человека закреплены в языке, который он использует для описания. В данном случае существует связь между структурой и языком. В то же время, одна и та же структура может быть описана различными языками, то есть она не зависит от языка. Универсальным механизмом для описания структуры становится онтология, включающая в себя словарь, средствами которого описываются знания, и саму описываемую структуру знаний. Посредством словаря онтология накладывает ограничения на используемый язык.

Выделяют три уровня онтологий [1–3]. Схема, показывающая связи между онтологиями, приведена на рис. 1.

Онтологии верхнего уровня описывают общие понятия, такие как пространство, время, материя, объект, событие, действие и т.п., которые не зависят от конкретной задачи в проблемной области.

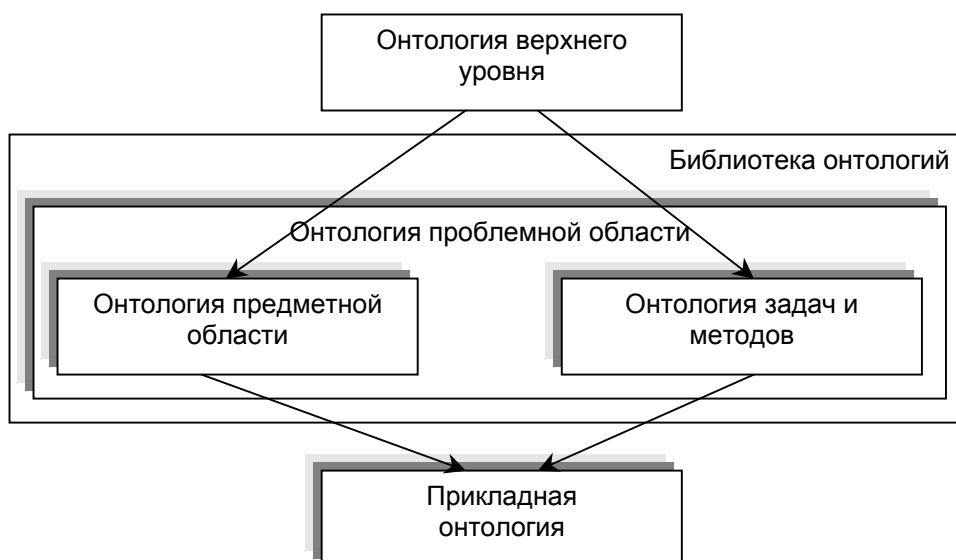


Рис. 1. Виды онтологий и отношения между ними

Онтологии предметной области и онтологии задач и методов описывают, соответственно, словарь, привязанный к родовому домену (например, медицина или автомобили) или родовую задачу или деятельность (например, диагностика или продажи) средствами словаря, представленного в онтологии верхнего уровня.

Прикладные онтологии описывают понятия, зависящие, как от конкретной проблемной области, так и от задачи. Эти понятия часто соответствуют ролям, играемым сущностями проблемной области при выполнении определенной деятельности.

В [4] описан общий подход к моделированию знаний, основанный на использовании онтологий. Предложена следующая схема: (1) неформальное описание, (2) разработка модели решения задачи (метода), (3) построение модели предметной области, (4) построение прикладной онтологии.

Далее будут рассмотрены и обобщены подходы, используемые при построении перечисленных онтологий.

1. Обзор систем по работе с онтологиями

1.1. Поддержка онтологий верхнего уровня

Интеграция информации является основной областью приложения онтологий. Для двух систем, пользующихся одним и тем же словарем, не существует гарантии, что они смогут договориться о чем-либо, пока не примут одну и ту же концептуализацию. Под концептуализацией понимается множество концептуальных связей, существующих в данной проблемной области. Две модели могут пользоваться пересекающимися онтологиями, в то время как сами не будут пересекаться. Это означает, что восходящий метод к интеграции систем, основанный на интеграции многих частных онтологий, может не работать, особенно, если частные онтологии нацелены только на концептуальные связи, значимые для конкретного контекста, и, следовательно, они являются слабым

приближением моделей. Следовательно, более удобно опираться на одну онтологию верхнего уровня вместо того, чтобы полагаться на договоренности, основанные на пересечении различных онтологий [1]. Онтология верхнего уровня является расширением для онтологии предметной области и онтологии задач и методов.

Назначение онтологии верхнего уровня — предоставить терминологию для описания общих понятий. Даже в случаях, когда некоторая задача может казаться характерной только для конкретной проблемной области, представление знаний может потребовать онтологии, описывающей более общие знания. Такую общую терминологию предоставляет онтология верхнего уровня [5]. В [1, 2] предполагается, что универсальная онтология верхнего уровня является результатом интеграции частных онтологий до понятий, независимых от конкретных проблемных областей.

Некоторые подходы, используемые при определении понятий онтологии верхнего уровня, приведены в табл. 1. На верхнем уровне онтологии находится абстрактное понятие “thing” или сходная с ним абстракция.

Таблица 1. Сопоставление используемых в онтологиях верхнего уровня абстракций

Автор, система	Категории, используемые для описания абстракции верхнего уровня	Используемый формализм
Guarino N. [6]	<ol style="list-style-type: none"> 1. <i>Частности</i>: существующие в мире объекты; 2. <i>Универсальные понятия</i>: понятия, необходимые для описания частных 	Мета-свойства (тождественность, строгость, целостность, зависимость, собственная тождественность) и ограничения, вытекающие из определений перечисленных свойств. Более подробно см. [7, 8]
Sowa J. [9, 10]	<ol style="list-style-type: none"> 1. <i>Понятие независимого</i>: понятия, существующие независимо от наличия каких-либо других объектов (человек, железо и т.п.); 2. <i>Понятие зависимого</i>: понятия, используемые для определения игровых ролей (жена, учитель и т.п.); 3. <i>Понятие посредствующего</i>: среда или контекст, где реализуется предыдущее понятие, (семья, школа); 4. <i>Понятие непрерывного</i>; 5. <i>Понятие дискретного</i> 	Примитивные свойства (реальность, форма, понимание, препозиция, звено, намерение) и категории верхнего уровня (объект, процесс, схема, сценарий, соединение, участие, описание, история, структура, ситуация, причина, цель)
СУС [11]	<ol style="list-style-type: none"> 1. <i>Понятие единичного</i>: константа, обозначающая единичный объект, в отличие от множества объектов 2. <i>Понятие неосязаемого</i>: набор нефизических объектов (не являющихся сделанными из чего-либо, не результат кодирования, нематериальны) 3. <i>Понятие временного</i>: набор понятий, имеющих протяженность (действия, осязаемые объекты, соглашения, абстрактные единицы времени) 	СусL — язык исчисления предикатов первого порядка (язык формальной логики) Термы языка: константы, неатомарные термы, переменные. Термы объединяются в значимые выражения, используемые для вывода утверждений в базу знаний. Каждому понятию из базы знаний соответствует некоторая константа.
GUM (Generalized Upper Model) [11]	<ol style="list-style-type: none"> 1. <i>Конфигурация</i>: конфигурация элементов, участвующих в некоторой деятельности 2. <i>Элемент</i>: единичный самостоятель- 	The LOOM Knowledge Representation Language Анализ лексико-грамматических систем естественных языков, осно-

Автор, система	Категории, используемые для описания абстракции верхнего уровня	Используемый формализм
	<p>ный объект или понятие</p> <p>3. <i>Последовательность</i>: сложная ситуация, когда различные виды деятельности или конфигурации связаны отношениями таким образом, что образуют последовательность</p>	<p>ванный на использовании метафункций (экспериментальная, логическая, текстовая, взаимодействия высказывающегося и слушателя). Метафункции предоставляют модели верхнего уровня набор ограничений относительно того, что модель должна содержать, чтобы описать этими средствами различные грамматические конструкции [13].</p>
WebKB [14]	<p>1. <i>Ситуация</i>: объект, «происходящий» во времени и пространстве</p> <p>2. <i>Сущность</i>: объект, который может быть «вовлечен» в ситуацию</p>	<p>Формат Resource Description Framework (RDF), включающий 3 базовых метакласса (Class — класс, Property — свойство и ConstraintResource — ограничения) и категорию «отношение» (relation), как экземпляр класса Property. Метаклассы составляют набор классов (Collection)</p>
WordNet [15] — лексическая база данных	<p>1. <i>Живой объект</i>: организмы (растения, животные, человек)</p> <p>2. <i>Неживой объект</i>: вещи (артефакты, вещества, еда)</p>	<p>Лексическая матрица, семантическая сеть. Рассматриваются семантические (синонимы, антонимы, гипонимы, меронимы) и морфологические отношения между словами</p>

1.2. Поддержка онтологий предметных областей и онтологий методов

Расширением для онтологий предметных областей и онтологий методов (онтологий задач и методов) является онтология проблемной области. Онтология предметной области состоит из объектов предметной области и отношений между ними, описываемых в терминологии конкретной проблемной области. Онтология методов — онтология, определяющая понятия и связи, описывающие механизм работы метода или алгоритм решения задачи [16] средствами словаря проблемной области или, для методов являющихся общими для нескольких проблемных областей, онтологии более высокого уровня.

Сравнительные характеристики информационных систем, занимающихся построением и управлением онтологиями рассматриваемого типа, приведены в табл. 2.

Таблица 2. Информационные системы, поддерживающие управление онтологиями проблемной области

Система	Назначение, область применения	Используемый формализм	Поддерживаемые функции
Chimaera [17]	Соединение фрагментов баз знаний	<p>OKBC (Open Knowledge Base Connectivity)</p> <p>Используемые понятия: класс, подкласс, слот (собственный и шаблонный)</p>	<p>Изменение структуры таксономии и преобразование структуры знаний в таксономическую, разрешение конфликтных имен в базе знаний, соединение баз знаний, просмотр онтологии, хранение онто-</p>

			логов
КАКТУС [18]	Моделирование знаний о сложных технических системах с целью их многократного использования	<p>CML (Conceptual Modelling Language), EXPRESS (язык, используемый для обмена информацией о продуктах), Ontolingua (формальный язык, используемый для структурирования знаний о различных проблемных областях).</p> <p>Используемые понятия: категория (включает <i>concept</i> (CML), <i>entity</i> (EXPRESS), <i>class</i> (Ontolingua)); определение (самостоятельная единица знания, включает <i>concept</i>, <i>relation</i>, <i>inference</i> и т.п. (CML), <i>entity</i>, <i>type</i> (EXPRESS)); теория (набор определений, включает модель экспертных знаний (<i>expertise model</i>), знания логического вывода (<i>inference knowledge</i>), онтологию (<i>ontology</i>) и т.п. (CML), схему (EXPRESS), онтологическую теорию (<i>ontological theory</i>) (Ontolingua)).</p>	<p>Создание, поддержка и использование библиотек онтологий. Библиотеки онтологий представлены как база данных теорий и определений внутри теорий.</p> <p>Поиск по библиотекам онтологий; импорт библиотек, поддерживающих формализмы, используемые системой.</p> <p>Поддержка библиотеки онтологий Ontolingua (импорт/экспорт, графическое отображение, редактирование средствами системы).</p> <p>Функции перевода из одного формализма в другой и частично в C/C++</p>
OntoBroker [19, 20].	Основанный на использовании онтологий доступ к распределенной, разнородной и плохо структурированной информации, содержащейся в Web-документах. Информация аннотируется поставщиком знаний с использованием словарей онтологий	<p>Модифицированная фреймовая логика. Используемые <i>понятия</i> — класс, подкласс, экземпляр класса, атрибут, значение атрибута, связь “part-of”, <i>отношения</i>: выражения предикатов, используемые как обычные логические формализмы, термы в которых не могут быть аргументами и объектными выражениями.</p> <p>Онтология состоит из трех частей: (1) иерархии понятий, определяющей связи подклассов между различными классами; (2) атрибутов классов; (3) набора правил, определяющего связи между различными понятиями и атрибутами [20].</p> <p>Поддерживается RDF-формат</p>	Извлечение, вывод и генерация метаданных в WWW
Ontolingua [21]	Сервер онтологий	<p>KIF (Knowledge Interchange Format) [2] — декларативный язык для описания знаний, используется фреймовое представление знаний</p> <p>Основные понятия: класс, подкласс, слот (собственный и шаблонный), фасет (собственный и шаблонный), экземпляр класса, отношение “is-a” (возможна инверсия)</p>	Создание онтологий, поддержка распределенных, совместно редактируемых и просматриваемых онтологий
Protégé [22]	Создание онтологий предметных областей	<p>CLIPS, PAL (для ограничений), фреймовое представление данных, логика исчисления предикатов первого порядка</p> <p>Используемые понятия: класс, подкласс, экземпляр класса, слот, зна-</p>	Поиск структурных элементов по имени и по содержанию при наложении ограничений на условия поиска; простое объединение онтологий; экспорт он-

		чения слота, ограничения на значения слота, инверсный слот, фазет, отношения “is-a” (возможны инверсные отношения).	тологии в форматы JDBC, XML, RDF, html
--	--	---	--

1.3. Библиотеки онтологий, прикладная онтология

Роль библиотеки онтологий — хранить и систематизировать существующие онтологии проблемных областей для упрощения построения прикладной онтологии. Библиотека онтологий дает следующие преимущества: (1) не требуется опыта по построению прикладной онтологии, достаточно опыта эксперта проблемной области, чтобы приспособить прикладную онтологию к конкретной проблеме; (2) возможно многократное использование прикладной онтологии в зависимости от обстоятельств [23]. Можно выделить следующие серверы, поддерживающие библиотеки онтологий: Ontolingua, DAML Ontology Library.

Назначение прикладной онтологии: (1) устранить пробел, существующий между онтологиями предметных областей и онтологиями методов, соединив эти онтологии в прикладную онтологию; (2) позволить эксперту проблемной области использовать тот же самый язык, который принят в приложении, с которым он работает, и который может отличаться от языка, используемого в библиотеке онтологий [4].

Словарь прикладной онтологии ориентирован на пользователя, словарь библиотеки онтологий — на терминологию проблемной области. Эти две онтологии связываются прямыми правилами отображения.

2. Принципы, закладываемые в системы

Можно выделить следующие требования к онтологиям:

1. с одной стороны, онтология должна быть ограничена проблемной областью, чтобы можно было договориться об используемых понятиях, с другой стороны, для поддержки процесса интеграции знаний онтология должна выходить за пределы этой области;
2. онтология должна содержать концептуальные, а не эпизодические знания;
3. онтология не должна зависеть от естественного языка, используемого для описания знаний;
4. онтологии должны быть совместимы друг с другом на предмет дальнейшей интеграции и обмена содержащейся в них информацией;
5. онтология должна быть полезной для решения задачи в области, для которой данная онтология предназначена;
6. онтология должна поддерживать точные аксиомы вывода (должна быть хорошо специфицирована) и внутренне согласована со структурой, именами и содержимым для всех определенных в ней понятий;
7. онтология должна быть совместима с другими источниками/ресурсами знаний;
8. онтология должна быть хорошо структурированной;
9. онтология должна быть простой для понимания и для поиска понятия;
10. онтология должна запоминать и предоставлять информацию о своих предыдущих состояниях по мере развития;
11. онтология должна поддерживать следующие функции: возможность трансляции с одного языка/формата в другой, предоставлять возможность многократного использования, быть редактируемой, быть дополненной или до-

полнять другую онтологию, обеспечивать возможность совместного доступа пользователей.

Между исследователями достигнуто соглашение, что в мире есть *объекты*; у них есть *свойства*, которые имеют *значения*. Объекты могут быть в различных отношениях друг к другу; свойства и отношения изменяются во *времени*; в различные *моменты времени* возникают *события*; существуют *процессы*, в которых участвуют объекты и которые возникают во времени; мир и его объекты могут находиться в различных *состояниях*; события могут вызывать другие события, т.е. могут давать *эффект*; объекты могут состоять из *частей*. Способы организации фактов мира — это понятия *классов*, *экземпляров классов* и *подклассов*, где понятие класса ассоциируется с совместно используемыми свойствами. Таким образом, отношение *is-a*, указывающее на отношения подклассов, — фундаментальное для представления онтологий [16].

С учетом данного соглашения и на основании перечисленных требований был разработан KIF-формат для представления знаний [2], в котором сформулированы основные понятия и принципы построения онтологий. Под онтологией понимается «подробно разработанная спецификация структуры определенной проблемной области». Онтология включает в себя словарь, позволяющий описать рассматриваемую область знаний, и набор логических высказываний, описывающих ограничения, существующие в проблемной области, и устанавливающие используемые значения слов из словаря.

Данный формат охватывает известные системы представления знаний и останавливается на операциях, которые эффективно поддерживаются большинством систем. Формат опирается на фреймовое представление знаний, поддерживает операции наследования, наложения и проверки ограничений, документирования. Формат не включает команды запросов к базе данных или команды по обработке баз данных. В данном формате классы и подклассы представляются фреймами, их атрибуты — наследуемыми полностью или частично слотами, накладываемые на значения слотов ограничения — фасетами, экземпляры классов характеризуется конкретными значениями слотов.

Элементарные операции, используемые в системах, ориентированных на работу с онтологиями, подробно описаны в [24]. Они включают операции по связыванию баз знаний; операции работы с базами знаний (открытие, создание и т.п.); операции над фреймами, включающие операции над классами, экземплярами классов и частностями; операции над слотами и фасетами; операции работы со списками; интерфейс типа «вопрос/ответ»; поведенческие операции и операции работы с процедурами (вызов процедуры, регистрация, создание и т.п.). Перечисленный набор операций достаточен для описания более сложных механизмов и операций (табл. 3), поддерживаемых в системах основанных на знаниях.

3. Заключение

В статье рассмотрены основные подходы к разработке онтологий, перечислены требования к онтологиям, операции работы с онтологиями и методы их реализации. Рассмотренные принципы, используемые при проектировании онтологий, позволят усовершенствовать процесс интеграции знаний, опирающийся на приведение различных способов представлении знаний к общему виду, предлагаемому онтологиями.

Таблица 3. Основные операции над онтологиями

Название операции	Цель	Методы реализации	Примечание
Операции по редактированию			
Создание (Creating) [25]	Создание новой онтологии	<ul style="list-style-type: none"> создание новой онтологии, постепенно определяя и уточняя понятия, связи и ограничения, описывающие представляемые знания; создание новой онтологии на основе имеющихся онтологий 	
Расширение (Extension) [26]	Возможность определения новых терминов для конкретных целей на основании существующего словаря способом, не требующим исправления существующих определений	<ul style="list-style-type: none"> достижение минимального онтологического соглашения (определение минимального словаря, способного описать все необходимые понятия); концептуализация, специфицированная на уровне знаний, не должна зависеть от конкретного способа кодирования 	
Редактирование (Editing) [25]	Обеспечение доступа к редактированию онтологий, хранящихся на сервере онтологий, из удаленных приложений или через Internet		
Поддержка (Maintaining) [2, 25]	Обеспечение возможностей пополнения онтологии и ее модификации		
Изменение (Modifying) [26]	Определение, что должно быть изменено, при внесении изменений в часть онтологии	Декларативная спецификация, дающая точную характеристику связям среди различных множеств ограничений, используемых для описания проблемы	
Просмотр (Browsing) [2, 25]	Обеспечение возможностей просмотра		
Сохранение (Saving) [2]	Сохранение предыдущих состояний онтологии в ходе ее разработки		
Выпуск (Publishing) [2, 26]	Предоставление свободного доступа к онтологии		
Алгебра онтологий [27, 28]			
Пересечение (Intersection) [27, 28]	Создание новой онтологии — онтологии-подмножества, состоящей из совместно используемых компонент	Создание списка терминов, которые определены правилами отображения в качестве принадлежащих двум проблемным областям. Требуется интерпретация правил отображения. Она может	Здесь же решается задача определения минимального пересечения

Название операции	Цель	Методы реализации	Примечание
		включать вторичные термины, если они необходимы для наложения ограничений на первичные термины	
Объединение (Union) [27, 28]	Создание непротиворечивой новой онтологии	Объединение исходных терминов, для которых не было найдено пересечения, и списков терминов, являющихся результатом операции пересечения	Включает операции по разрешению конфликтов, описанных в операции "соединение"
Отличие (Difference) [27, 28]	Создание несимметричных подмножеств, заменяет отрицание	Создание списка терминов, которые не вошли в результате операций intersection и union в онтологию	
Отображение (Map) [28]	Преобразования внутри онтологии, позволяющие проверять большее количество терминов на совпадение и использовать их для определения целесообразных пересечений	Внутреннее отображение назначает владельца конкретной проблемной области ответственным за правила отображения вместо менеджеров посредников, имеющих доступ к разнообразным проблемным областям	
Операции по созданию новой онтологии из существующих			
Отсечение (Pruning) [29]	Удаление терминов, которые не являются необходимыми для данной области	<ul style="list-style-type: none"> рекурсивное удаление всех подпонятий удаляемого понятия и ссылающихся на удаляемые понятия (см. LOOM [30]); избирательное удаление предложений, содержащих удаляемый термин в своих определениях, и обобщение данных определений таким образом, чтобы ссылка на удаляемый термин могла быть устранена (см. PowerLoom [30]) 	
Выборка (Slicing) [31]	Выбор части входной онтологии для использования в новой прикладной онтологии	<ul style="list-style-type: none"> максимальная выборка (MaximalSlice): (шаг 1) определение всех значимых классов; (шаг 2) определение восходящего домена в графе таксономических связей. Подробный алгоритм описан в [31]; выборка с оптимизацией (MinimalMaximalSlice): на основании критериев оптимизации выбираются только классы, представляющие интерес для конкретного приложения. Подробный алгоритм описан в [31] 	
Извлечение (Extraction) [29]	Перебор и организация терминов, необходимых для включения в модель данной проблемной области из онтологии широкого диапазона	Отображение терминов, рассматриваемых, как ключевые, в термины онтологии, ориентированной на лексические отношения (например, WordNet). При помощи вероятностных и структурных эвристик выявляются термины, связанные с определенными на первом этапе, и извлекаются для создания приближенной онтологии, приспособленной для данной проблемной области [32]	Включает операции по разрешению конфликтов, описанных в операции "соединение"
Приращение (Aug-	Приращение некоторой онтологии	Если в онтологии проблемной области обнаружена ссылка на	Включает операции по

Название операции	Цель	Методы реализации	Примечание
ment) [29]	или базы знаний посредством извлечения и включения выбранных частей из независимо разработанных онтологий	задачу, решаемую в родовой онтологии, то производится приращение этой родовой онтологии, предоставляющей дополнительные средства решения проблемы, находящиеся на более высоком уровне абстракции и расширяющие возможности информационной системы [33]	разрешению конфликтов, описанных в операции “соединение”
Соединение (Merging)	Соединение двух независимо разработанных баз знаний (БЗ), учитывающее разрешение конфликтов между именами понятий и структурным представлением	<p>Проблемы перемещения аксиом из одной онтологии в другую:</p> <ul style="list-style-type: none"> ◦ необходима доработка формальной <i>контекстной теории</i> [29]; ◦ включение аксиом из одной онтологии в другую, оставляя ссылки на словарь включаемой онтологии [25]. <p>Разрешение семантических и синтаксических конфликтов:</p> <ul style="list-style-type: none"> • конфликт при структурном противоречии класс—слот—значение (различные представления для одного и того же действия), выявляемый на семантической фазе соединения [31]: <ul style="list-style-type: none"> ◦ выбор одного из представлений; ◦ добавление аксиом, определяющих эквивалентность двух представлений. В общем, эта фаза может включать значительные интерактивные вмешательства (см. Chimaera [37], SMART [34]); • конфликты между определениями: соединение определений и предоставление пользователю возможности устранения конфликтов по мере их возникновения; • противоречия в названиях: <ul style="list-style-type: none"> ◦ использование таблицы отображений, способной делать переводы между общими соглашениями по именованию; ◦ перевод понятий из двух онтологий в одну общую (например, WordNet) и с ее помощью корректировка имен [32]; ◦ противоречия в названиях понятий решаются автоматически со временем в результате операции “slicing” — выбор понятий из общей онтологии широкого диапазона [31]; ◦ сбор всех терминов со всех значимых онтологий, приведение их к соглашению и подготовка определений, которые приемлемы для данных онтологий. Когда определения будут окончательно сформулированы, издается документ с определениями и предполагается, что все разработчики откорректируют свои понятия в соответствии с данными определениями [35]; ◦ предполагается, что разрозненные понятия всегда озна- 	Операции перевода, выборки и преобразования осуществляются до операции соединения

Название операции	Цель	Методы реализации	Примечание
		чают разное, если не существует других точных указаний. Такая информация, закодированная правилами отображения, формирует БЗ, управляемую экспертами из разных проблемных областей. Термины, которые покрываются правилами отображения, формируют новую онтологию второго уровня абстракции. Более высокие уровни абстракции могут определяться рекурсивно, оставляя ненужные абстрактные термины локальными на их абстрактном уровне [28].	
Выравнивание (Alignment) [34]	Установка связей между онтологиями	Интерактивное взаимодействие с пользователем: автоматическое создание списка начальных предположений, основанных на схожести имен классов. Из предложенного списка пользователь выбирает операцию. Для каждой выбранной операции на основании близости классов и атрибутов, используемых в операции, выполняется последовательность автоматических действий, делаются предположения, помогающие пользователю выбрать требуемую операцию, проверяется, нет ли конфликтов, и если есть, предлагаются способы их разрешения [34]	Включает операции по разрешению конфликтов, описанных в операции "соединение"
Интеграция (Integrating) [2]	Объединение существующих онтологий или их частей, построение родовой онтологии	Разработка онтологий проблемных областей. Примеры интеграции онтологий описаны в [26], § 7	Включает операции по разрешению конфликтов, описанных в операции "соединение"
Операции агрегирования/декомпозиции			
Модуляризация (Modularization) [31] или декомпозиция (Decomposition) [26]	Разбиение содержимого БЗ в понятные разделы, которые служат основой для разработки БЗ и логического вывода	Способы разбиения на модули [31]: <ul style="list-style-type: none"> основанный на предметной области: организация БЗ по предметной области, может предоставить более легкое совместное использование БЗ и разработку содержимого БЗ основанный на задаче: организация БЗ посредством <i>правил</i> и <i>экспертов</i>, имеющих отношение к данной задаче, значительно сокращая таким образом пространство поиска. Определения классов, функций и связей не влияют на пространство поиска, следовательно, нет необходимости разбивать их на модули. помещение аксиом из различных модулей в отдельные файлы 	Включает механизмы для обработки конфликтов имен [26]
Сбор (Assembling) [26]	Составление новой онтологии из существующих модулей, имеющих в библиотеке онтологий		Включает механизмы для обработки конфликтов имен

Название операции	Цель	Методы реализации	Примечание
Составление (Composing) [29]	Процесс создания базы знаний проблемной области, ориентированный на выбор подходящих моделей (микро-теорий ¹) из библиотеки онтологий и их соединение (см. LOOM [30] — формирование онтологий и баз знаний иерархическим способом).	<ul style="list-style-type: none"> композиционные теории [29]; используемые абстракции организовываются в иерархию, каждая абстракция является уточнением других [38] 	Включает механизмы для обработки конфликтов имен
Операции, выполняемые интеллектуальными агентами			
Определение терминологии (Term definitions) [2]	Обеспечение доступа к определению терминов при использовании более общей онтологии	Обмен сообщениями между агентом интерфейса пользователя и Агентом Онтологий (Ontology Agent), занимающимся онтологиями искомой проблемной области	
Поиск онтологии для совместного использования (Selecting a shared ontology) [2]	Выявление онтологий, занимающихся данной проблемой	Основная схема — обмен сообщениями между агентами «Service Provider», «Ontology Agent», «Directory Facilitator» и агентом пользователя	
Поиск (Finding) [2]	Выявление онтологий, в которых определено искомое понятие	Запрос к агенту «Directory Facilitator», который определяет доступных Агентов Онтологий. Каждый Агент Онтологий определяет список онтологий, занимающихся интересующей проблемой, и формирует к ним запросы на предмет возможности доступа к искомому термину	
Проверка эквивалентности (Testing equivalence) [2]	Проверка логической эквивалентности двух онтологий	Запрос к Агентам Онтологий, имеющим предварительные данные об эквивалентности онтологий	
Перевод терминов (Terms translation) [2]	Перевод терминологии, определенной для одной онтологии, в термины другой	Определение через «Directory Facilitator» Агента Онтологий, отвечающего за перевод терминологии между двумя выбранными онтологиями. Этот агент осуществляет перевод терминов через серверы рассматриваемых онтологий	
Операции по преобразованию			
Преобразование (Reformulation) [31]	Преобразование представления входной теории	Конвертирование функциональных предикатов в функции (преобразование отношений в функции). Метод: Jane calls Mary → call = Jane (Mary)	

¹ Использование простой абстракции, если нет необходимости в более сложном представлении. Например, моделирование пространства с помощью нескольких абстрактных моделей, каждая из которых верна, но совокупность этих моделей не является полным представлением понятия «пространства» [38]

Название операции	Цель	Методы реализации	Примечание
Создание таксономии (Taxonomy design) [31]	Создание иерархической структуры, элементы которой связаны отношением «быть экземпляром»	<ul style="list-style-type: none"> • Каждое понятие должно иметь необходимые и достаточные определения. Если существуют несколько понятий, которые систематически определяются в терминах друг друга, можно отбросить излишне определенные понятия и получить таким образом необходимый (малый) набор понятий • Имена понятий в таксономии не должны приравниваться к их значениям в естественном языке, во время создания таксономии примитивов, все толкования слова или свойства вывода должны быть представлены с использованием отдельных понятий 	Включает операции по разрешению конфликтов, описанных в операции «соединение»
Перевод (Translation)	Обеспечение доступа к онтологиям, разработанным в одном формализме, для других языков и формализмов	<ul style="list-style-type: none"> • Разработка пакета для быстрого создания трансляторов, основанных на декларативных отображениях между языком онтологии и выходным языком [29] • Разработка транслятора, обрабатывающего структурную информацию (классы, функции, отношения, связи классов, фасеты) в БЗ [31] 	Должна быть поддержка трансляции в менее выразительные языки, обеспечение простоты добавления трансляторов в дополнительные выходные языки [26]
Операции по проверке онтологий			
Оценка (Evaluating) [2]	Проверка онтологии на соответствие ее назначению	Формирование экспертами набора неформальных вопросов, на которые онтология, предположительно, должна отвечать [2], [26]	Осуществляется перед выпуском
Сравнение (Matching)	Определение степени соответствия онтологий	<p>Проверка на полноту перевода одной онтологии в другую [2]. Онтологии:</p> <ul style="list-style-type: none"> • расширяют одна другую (extension) — онтология O1 расширяет онтологию O2. Если между онтологиями O1 и O2 существует расширяющая связь, то онтология O1 расширяет или включает онтологию O2. Все понятия, определенные в онтологии O2, найдены в онтологии O1 при условии, что все свойства, определенные между сущностями в онтологии O2, сохранены в онтологии O1; • идентичны (identical) — O1 и O2 идентичны. Словарь, аксиоматизация и используемый язык представления физически идентичны. O1 и O2 могут иметь различные имена; • эквиваленты (equivalent) — O1 и O2 эквивалентны. Онтологии используют один и тот же словарь, и одинаковую логическую аксиоматизацию, но могут быть описаны различными языками представления. Используются определенные для рассматриваемых онтологий серверы с заданными воз- 	Вопрос отличия, расширяет одна онтология другую или включает ее, в [2] не рассматривается

Название операции	Цель	Методы реализации	Примечание
		<p>возможностями логического вывода;</p> <ul style="list-style-type: none"> • точно переводимые (strongly-translatable) — источник O1 точно передается в приемник O2: (1) если O1 и O2 эквивалентны, то они точно переводимы в двух направлениях, требуется только отображение для языков представления; (2) словарь одной онтологии полностью переводится в словарь другой + аксиоматизация одной онтологии поддерживается в другой + между приемником и источником не происходит потери информации + отсутствуют включения в структуру приемника и несогласованности (см. операцию проверки на непротиворечивость); • плохо переводимые (weakly-translatable) — источник O1 ненадежно передается в приемник O2. Транслирование возможно, но с потерей информации. Приемник использует подмножество терминов из словаря источника и ту же самую аксиоматизацию. Некоторые термины или связи в источнике могут упроститься, некоторые могут остаться непереуведенными. Несогласованности в результате данной трансляции не возникает; • приближенно переводимы (approx-translatable) — источник O1 приближенно передается в приемник O2. Онтологии могут быть «приближенно переводимыми», если они «плохо переводимы», имеют включения в структуру и не исключают противоречивости. Возможно, что отношения станут неверными, и ограничения не будут применены. <p>Сравнение концептуальных структур [36]. Онтологии:</p> <ul style="list-style-type: none"> • совпадающие (consensus): одинаковые понятия в разных онтологиях имеют одинаковые имена; • соответствующие (correspondence): одинаковые понятия в разных онтологиях имеют разные названия; • конфликтующие (conflict): разные понятия в разных онтологиях имеют одинаковые названия; • противоречивые (contrast): понятия и имена понятий в разных онтологиях различны. <p>Методы, используемые для сравнения структур онтологий [36]:</p> <ul style="list-style-type: none"> • Запись структуры класса базовой онтологии в терминологии составляющих слотов на элементы класса, представленных в пределах онтологии пользователя, или булевская шкала оценок, которые показывают, является или нет некоторый элемент экземпляром класса. 	

Название операции	Цель	Методы реализации	Примечание
		<ul style="list-style-type: none"> • Сравнение классов, определенных в онтологиях, в терминах элементов классов: <ul style="list-style-type: none"> ◦ <i>Consensus</i> если в двух онтологиях есть два класса, названных одинаково и имеющих одинаковые элементы классов, значит, онтологии совпадают. ◦ <i>Correspondence</i> если в двух онтологиях есть два класса, названных различно, но имеющих одинаковые наборы элементов классов, то онтологии соответствуют друг другу. ◦ <i>Conflict</i> если есть два класса, названных одинаково, но имеющих различные наборы элементов, то онтологии конфликтуют. ◦ <i>Contrast</i> если нет пересечения ни между терминологиями, ни между наборами элементов — противоречие. 	
Проверка (Validation)	Подтверждение правильности новой онтологии	<p>Проверка онтологии на полноту и непротиворечивость [29], [37]:</p> <ul style="list-style-type: none"> • выявление неопределенных классов и связей (использование механизма «доски объявлений»); • выявление конфликтов и несогласованностей: (1) анализ таксономии (выявление избыточных классов и подклассов; для класса «Thing», являющегося, как правило, абстракцией верхнего уровня в таксономии классов, выявление несущественных для данного класса подклассов и экземпляров класса; (2) выявление несогласованностей между значением и типом слота, циклов в определении класса; (3) выявление конфликтов между базой знаний и онтологией на основании анализа установленных правил отображения; • проверка онтологии на полноту: (1) все термины, на которые есть ссылки, определены; (2) экземпляры классов имеют значения для всех своих атрибутов; (3) если классы объявлены несовместимыми, то экземпляры этих классов должны принадлежать только подклассу одного из несовместимых классов; (4) все иерархические связи, неявно введенные в определения классов, реализованы; • выявление семантических различий в связанных определениях: определение семантической близости между терминами в различных онтологиях или терминами в различных 	

Название операции	Цель	Методы реализации	Примечание
		<p>версиях одной и той же онтологии. Для осуществления проверки данного типа требуется разработка правил вывода, которые могут сравнивать два термина для установления между ними семантических отличий, и разработка средств для сравнения различных версий одной онтологии;</p> <ul style="list-style-type: none"> • синтаксический анализ: проверка сферы действия термина (или содержащейся в термине подстроки), выявление возможных расшифровок для аббревиатур; 	
Прочие операции			
Запрос (Query) [26]	Возможность формирования запросов к данной онтологии из удаленных приложений		

Литература

- [1] *Guarino N.* Formal Ontology and Information Systems. *Proceedings of FOIS'98*. — Trento, Italy. — Amsterdam: IOS Press, 1998 — Pp. 3–15.
- [2] FIPA 98 Specification. Part 12. Ontology Service. — Geneva, Switzerland: Foundation for Intelligent Physical Agents (FIPA), 1998. — Version 1.0.
- [3] *Smirnov A., Chandra C.* Ontology-based Knowledge Management for Co-operative Supply Chain Configuration // *Proceedings of the 2000 AAAI Spring Symposium "Bringing Knowledge to Business Processes"*. — Stanford, California: AAAI Press, 2000. — Pp. 85–92.
- [4] *Guarino N.* Understanding, Building, and Using Ontologies. A Commentary to "Using Explicit Ontologies in KBS Development" (by van Heijst, Schreiber, and Wielinga) // *International Journal of Human and Computer Studies*, 1997. — V. 46. — No 2/3. — Pp. 293–310.
- [5] *Chandrasekaran B., Josephson J. R., Benjamins V. R.* What are Ontologies, and Why Do We Need Them? // *IEEE Intelligent Systems & their applications*, January/February, 1999. — Pp. 20–26.
- [6] *Guarino N.* Some organizing principles for a unified top-level ontology // *Working Notes of AAAI Spring Symposium on Ontological Engineering*. — Stanford, 1997.
- [7] *Guarino N., Welty C.* A Formal Ontology of Properties // *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management*. — Lecture Notes on Computer Science, Springer Verlag, 2000.
- [8] *Guarino N., Welty C.* Towards a methodology for ontology-based model engineering // *Proceedings of the ECOOP-2000 Workshop on Model Engineering* (eds. Bevizin J. and Ernst J.), 2000.
- [9] *Sowa J.* Distinction, combination, and constraints // *Proceedings of IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.
- [10] *Sowa J.* Knowledge representation: logical, philosophical and computational foundation. — Boston: PWS Publishing Comp., 1998.
- [11] The Upper Cyc Ontology. <http://www.cyc.com/cyc-2-1/toc.html>
- [12] *Bateman J. A., Henschel R., Rinaldi F.* The Generalized Upper Model 2.0. — German National Research Center for Information Technology. <http://www.darmstadt.gmd.de/publish/komet/gen-um/newUM.html>
- [13] *Bateman J. A.* Upper modeling: organizing knowledge for natural language processing // *The 5th International Workshop on Natural Language Generation*. — Pittsburgh, PA, 1990. http://www.darmstadt.gmd.de/publish/komet/papers/general-d_escription.ps
- [14] *Martin P.* WebKB. — Griffith University, Australia. <http://meganesia.int.gu.edu.au/~phmartin/WebKB/>
- [15] WordNet — a Lexical Database for English. — Princeton University. Princeton, NJ. <http://cogsci.princeton.edu/~wn/>
- [16] *Chandrasekaran B., Josephson J. R., Benjamins R.* Ontology of Tasks and Methods // *Proceedings of the Eleventh Workshop on Knowledge Acquisition, Modeling and Management (KAW'98)*. — Inn, Banff, Alberta, Canada, 1998. <http://spuds.cpsc.ucalgary.ca/KAW/KAW98/chandra/>
- [17] Chimaera. — Stanford University: Knowledge Systems Laboratory. <http://www.ksl.stanford.edu/software/chimaera/>.
- [18] KACTUS. — University of Amsterdam: Department of Social Science Informatics. ESPRIT Project 8145. <http://swi.psy.uva.nl/projects/NewKACTUS/home.html>.
- [19] *Benjamins V. R.* The Ontological Engineering Initiative (KA)²: *Proceedings of the International Conference on Formal Ontologies in Information Systems (FOIS-98)* — Trento, Italy // *Frontiers in Artificial Intelligence and Applications* (eds. by N. Guarino). — IOS-Press, 1998.
- [20] *Decker S., Erdmann M., Fensel D., Studer R.* Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information // *Semantic Issues in Multimedia Systems. Proceedings of DS-8R* (eds. by Meersman et al.) — Boston: Kluwer Academic Publisher, 1999 — Pp. 351–369. <http://ontobroker.aifb.uni-karlsruhe.de/>
- [21] Ontolingua. — Stanford University: Knowledge Systems Laboratory. <http://www-ksl-svc.stanford.edu:5915/&service=frame-editor>

- [22] Protégé-2000. — Stanford Medical Informatics University. <http://smi-web.stanford.edu/projects/protege/>
- [23] Heijst G., Falasconi S., Abu-Hanna A., Schreiber G., Stefanelli M. A case study in ontology library construction // *Artificial Intelligence in Medicine*, 1995. — June. — Pp. 227–255.
- [24] Chaudhri V. K., Farquhar A., Fikes R., Fikes R., Rice J. P. Open Knowledge Base Connectivity 2.0.3. — Specification Document, 1998.
- [25] Farquhar A., Fikes R., Pratt W., Rice J. Collaborative Ontology Construction for Information Integration. — Systems Laboratory Department of Computer Science, 1995. — KSL-95-63.
- [26] Uschold M., Gruninger M. Ontologies: Principles, Methods, and Applications // *Knowledge Engineering Review*, 1996. — V. 11 — No. 2.
- [27] Wiederhold G. An Algebra for Ontology Composition // *Proceedings of 1994 Monterey Workshop on Formal Methods*. — U.S. Naval Postgraduate School, Monterey CA, 1994. — Pp. 56–61. <http://www.db.stanford.edu/SKC/publications.html>
- [28] Wiederhold G. Interoperation, Mediation, and Ontologies // *Proceedings of the International Symposium on Fifth Generation Computer Systems (FGCS94): Workshop on Heterogeneous Cooperative Knowledge Bases*. — ICOT, Tokyo, Japan, 1994. — V. W3. — Pp. 33–48.
- [29] MacGregor R., Patil R. S. Tools for Assembling and Managing Scalable Knowledge Bases. <http://www.isi.edu/isd/OntoLoom/hpkb/OntoLoom.html>
- [30] Loom Project Home Page. <http://www.isi.edu/isd/LOOM/LOOM-HOME.html>
- [31] Chaudhri V. K., Lowrance J. D., Stickel M. E., Thomere J. F., Wadlinger R. J. Ontology Construction Toolkit. Technical Note Ontology — AI Center. Report — January 2000. — SRI Project No. 1633.
- [32] Swartout B., Patil R., Knight K., Russ T. Toward Distributed Use of Large-Scale Ontologies // *Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop (KAW'96)*. — Banff, Canada, 1996. http://www.isi.edu/isd/banff_paper/Banff_final_web/Banff_96_final_2.html
- [33] Russ T. A., MacGregor R. M., Salemi B. VEIL: Combining Semantic Knowledge with Image Understanding // *Proceedings of the International Snow Science Workshop*. — Banff, Alberta, Canada, 1996. <http://www.isi.edu/isd/LOOM/papers/veil/veil-iuw96.ps.gz>
- [34] Noy N. F., Musen M. A. SMART: Automated Support for Ontology Merging and Alignment // *Proceeding of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW'99)*. — Banff, Canada, 1999. <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html>
- [35] *Proceedings of the Workshop on Application Integration Architectures* (eds. by Hodges B., Thompson C.). — Texas Instruments, NIST, 1993.
- [36] Tennison J., Shadbolt N. R. APECKS: a Tool to Support Living Ontologies // *Proceedings of the 11th workshop on Knowledge Acquisition for Knowledge-Based Systems (KAW'98)*. — Banff, Alberta, Canada, 1998. <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/tennison/>.
- [37] McGuinness D. L., Fikes R., Rice J., Wilder S. An Environment for Merging and Testing Large Ontologies // *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR'2000)*. — Breckenridge, Colorado, USA, 2000. <http://www.ksl.stanford.edu/software/chimaera/>.
- [38] Guha R. V., Lenat D. B. Cyc: A Mid-Term Report. — MCC and Stanford University, 1990. <http://www.cyc.com/tech-reports/act-cyc-134-90/act-cyc-134-90.html>