

ИНСТИТУТ «КИБЕРНЕТИЧЕСКИЙ ЦЕНТР» ТПУ
ОТДЕЛ ПРОБЛЕМ ИНФОРМАТИЗАЦИИ ТИЦ СО РАН

А.Ф. Тузовский, С.В. Чириков, В.З. Ямпольский

СИСТЕМЫ
УПРАВЛЕНИЯ ЗНАНИЯМИ
(методы и технологии)

Под общей редакцией
В.З. Ямпольского



Томск – 2005

УДК 004.83

T 817

Тузовский А.Ф., Чириков С.В., Ямпольский В.З. Системы управления знаниями (методы и технологии) / Под общ. ред. В.З. Ямпольского. – Томск: Изд-во НТЛ, 2005. – 260 с.

ISBN 5-89503-241-9

В монографии рассматривается новое научное направление и новое направление менеджмента организаций – «Управление знаниями». Анализируется процесс работы со знаниями, описываются технологии, методы и формы работы с явными и неявными знаниями. Возникающие проблемы и задачи рассматриваются на уровне корпоративных систем управления знаниями, а также на уровне их важнейших подсистем, таких, как корпоративные хранилища знаний, Web-порталы, информационно-программное обеспечение.

Обобщен отечественный и зарубежный опыт по созданию систем управления знаниями, структурированы и систематизированы основные понятия и определения, используемые в данной предметной области, приведены описания существующих в мировой практике программных систем и инструментов для работы со знаниями.

Основное внимание в монографии уделено использованию семантических методов реализации систем, таких, как онтологии, семантические метаописания и дескриптивная логика.

Книга может быть полезна разработчикам систем управления знаниями, руководителям компаний, широкому кругу специалистов по проблематике управления знаниями и интеллектуальной собственностью, преподавателям и студентам университетов, специализирующихся на информационных и интеллектуальных системах, а также всем, кто интересуется этим новым научным направлением.

УДК 004.83

ISBN 5-89503-241-9

© А.Ф. Тузовский, С.В. Чириков,
В.З. Ямпольский, 2005

ВВЕДЕНИЕ

Накопление, распространение и передача знаний от поколений к поколениям во все времена определяли развитие человеческой цивилизации. В последние десятилетия знания, интеллектуальные ресурсы приобрели особую значимость в социальном и экономическом развитии общества. Это связано с действием ряда фундаментальных факторов и, прежде всего, с информационной революцией и возникновением новой экономики.

Информационная революция, вызванная появлением многих поколений компьютеров и программных систем, их проникновением практически во все сферы деятельности, и последовавшая за ней коммуникационная революция привели к небывалому прогрессу в обработке, хранении и передаче информации, упростили и ускорили взаимодействие между объектами и субъектами экономической, социальной и политической жизни разных стран, привели к глобализации рынка.

Новой экономике – экономике, основанной на знаниях, свойствен стремительный рост наукоемкости товаров и услуг, сокращение их жизненного цикла, интеллектуализация используемых технологий, обеспечивающих кратное повышение производительности труда, возникновение крупного сегмента рынка собственно интеллектуальных продуктов и услуг (патенты, лицензии, транзакции, консалтинг), быстрый темп обновления знаний и необходимость их постоянного пополнения.

Программные компании продают продукцию, которая, по существу, является идеями, интеллектуальной собственностью, встроенной в компьютерные коды. Можно классифицировать программное обеспечение как услуги: набор функций, предоставляемых в цифровом формате. Становится понятным, почему Microsoft так старательно занимается наймом умных работников. Неудивительно, что программный бизнес является новым видом экономики, основанной на знаниях, но даже традиционные производственные фирмы все более становятся как пользователями, так и продавцами знаний. Возник и новый вид управленче-

ской деятельности – управление знаниями, крупным и растущим активом современных компаний. Происходит определенная трансформация организационных механизмов и структур в сторону целевых команд, матричных, виртуальных и сетевых и других структур с целью повышения эффективности в использовании знаний. Следует при этом отметить, что при всей актуальности проблемы теория управления знаниями находится еще на начальной стадии, а все более возрастающая доля нематериальных активов компаний регулируется и управляется пока слабо и не системно. Когда-то традиционно производственные фирмы теперь выделяются из среды конкурентов тем, что предлагают «умную» продукцию: от программно-управляемых хлебопечек и стиральных машин до программно-управляемых станков и автомобилей, которые сами соблюдают дистанцию в потоке или паркуются без водителя. В новых товарах содержится все больший объем новых знаний.

Знания, интеллектуальный капитал, нематериальные активы, интеллектуальная собственность рассматриваются ими как новый источник богатства, как важный фактор приобретения конкурентных преимуществ [Wiig K.M., 1993; Румизен М.К., 2004].

В структуре собственности ряда известных транснациональных компаний, рыночная стоимость которых составляет десятки миллиардов долларов, интеллектуальный капитал, нематериальные активы составляют значительную и постоянно увеличивающуюся часть.

Нематериальные активы, как известно, включают:

- *человеческие активы* (human assets) – знания, опыт, мастерство, творчество;

- *интеллектуальные активы* (intellectual assets) – корпоративная культура, философия управления, стратегии и программы, бизнес-процессы;

- *активы интеллектуальной собственности* (intellectual property assets) – патенты, авторские и издательские права, торговые марки, коммерческие секреты;

- *структурные активы* (structure assets) – распределение полномочий, связи и отношения внутри и вне компании, правила и процедуры принятия решений, система стимулирования персонала, стандарты и регламенты;

- *бренд-активы* (brand assets) – известность, репутация, доброе имя компании.

Понимание растущей роли человеческих ресурсов (human resource), организационного управления (organizational behaviour) и информационных технологий (information technologies) в повышении эффективности современных компаний явилось основной причиной того, что со второй половины 90-х годов специалисты различных научных направлений стали активно исследовать и обсуждать проблематику управления знаниями (knowledge management). В недрах компаний, специализирующихся на разработке и производстве персональных ЭВМ, аппаратных и программных средств для информационных систем и телекоммуникаций, стали создаваться научные лаборатории по системам управления знаниями и даже научно-исследовательские институты, как это имело место в компании IBM.

Прошло немногим больше десяти лет с тех пор, как известный специалист по искусственному интеллекту К. Вииг положил начало научным публикациям по управлению знаниями [Wiig K.M., 1993; Wiig K.M., 1999]. За этим последовали сотни работ – монографий статей, отчетов, содержащих результаты исследований и практический опыт. Среди них необходимо выделить исследования Л. Прусака и Т. Давенпорта [Davenport T., Prussak L., 1999], И. Нонака и Х. Такеучи [Krogh G., Ichijo K., Nonaka I., 2000; Нонака И., Такеучи Х., 2003], Ч. Дисперса и Д. Чавела [Despres Ch., Chauvel D., 2000], У. Буковича и Р. Уильямса [Букович У., Уильямс Р., 2002] и Т. Стюарта [Stewart T.A., 1998]. Проводятся исследования и в российских научных центрах. С каждым днем появляется все больше информации о разрабатываемых проектах и программных системах в управлении знаниями, в создании корпоративных систем управления знаниями.

Таким образом, можно констатировать, что актуальность и важность проблематики управления знаниями осознается большинством специалистов, занимающихся корпоративным управлением и ИТ-технологиями для целей управления. Концептуальное единство распространяется и на существование процесса управления знаниями, важнейшими элементами (этапами) которого признается создание, хранение, поиск, передача (распространение) и использование знаний. Однако существуют два значительно отличающихся подхода к построению систем управления знаниями (СУЗ).

Первый можно назвать классическим, когда СУЗ строится на основе комбинирования существующих, уже зарекомендовавших себя технологий для поддержки различных подпроцессов работы со знанием. Речь идет о стандартных и широко используемых ИТ-технологиях, таких, как

E-mail, доски объявлений, дискуссионные форумы, общие каталоги документов, порталы, метаданные, а также о специфических технологиях, тяготеющих к инструментарию искусственного интеллекта, таких, как автоматическая классификация, автоматическое аннотирование документов, распознавание образов и речи и т.п.

Второй подход правильно определить как семантический. Он основан на использовании взаимосвязанного набора методов и технологий по работе со смыслом, семантикой данных, информацией и знаниями. В их числе онтологии предметных областей, технологии их построения и сопровождения, семантические метаданные, семантический поиск, системы логического вывода, семантическое профилирование знаний экспертов, семантические порталы и сети и т.п. И все это с соответствующей технологической поддержкой в части языков описания, моделей, программных инструментов и систем.

При этом семантический подход не отвергает классический. Большинство элементов и инструментов классического подхода зачисляется в арсенал средств развитых корпоративных информационных систем, которые могут применяться и, по существу, применяются для повышения уровня работы с данными и информацией.

Однако существенным отличительным признаком СУЗ от корпоративной ИС следует, очевидно, считать работу с семантикой и использование семантически ориентированных подходов, методов и технологий.

Авторы данной монографии отдают предпочтение семантическому подходу. Это предпочтение нашло свое отражение и в структуре монографии. Наиболее подробно (главы IV, V, VI) в ней рассматриваются онтологические описания, онтологические языки, программные инструменты для работы с онтологиями, дискрептивные логики и средства обеспечения логического вывода, семантические Web-порталы. При этом не забыты и компоненты классического подхода. Однако они представлены в монографии более сжато.

Новизна, сложность и междисциплинарный характер проблемы, участие в ее разработке различных научных школ и специалистов разного профиля породили, с одной стороны, широту взглядов на проблему и, с другой – терминологическую неоднородность, многообразие подходов, концепций и структур, положенных в основу построения систем управления знаниями (Knowledge Management Systems, KMS). Естественно, что такая ситуация создает трудности в работе специалистов соответствующего профиля и нуждается в анализе, структуризации и обобщении.

В данной монографии предпринята попытка обобщить отечественный и зарубежный опыт по созданию систем управления знаниями, структурируются и систематизируются основные понятия и определения, культивируемые в данной предметной области, излагаются теоретические основы и практические рекомендации по построению систем управления знаниями, в том числе выработанные авторами при исследовании проблематики СУЗ и в процессе создания реальных систем.

Книга может быть полезна не только разработчикам систем управления знаниями, но и руководителям компаний, широкому кругу специалистов по проблематике управления знаниями и интеллектуальной собственностью, преподавателям и студентам университетов, специализирующихся на информационных и интеллектуальных системах, а также всем, кто интересуется этим новым научным направлением.

Глава 1

ЗНАНИЕ КАК ПОНЯТИЕ

Термин «знание» не является новым. Он давно и широко используется в образовании, науке и в других сферах человеческой деятельности. Не существует каких-либо расхождений в его обиходном (неформальном) значении. Термин «знание» знаком и, в общем-то, понятен каждому культурному человеку. Вместе с тем, когда речь идет о системах управления знаниями, этот общеизвестный термин нуждается во всестороннем рассмотрении. От его понимания и толкования зависят как цель создания такого рода систем, так и, в определенной мере, их функциональная структура и состав.

Очевидно, что понятие «знание» тесно связано и с такими общеизвестными понятиями, как «информация», «данные». Между этими понятиями нет четких границ. Можно даже утверждать о наличии некоего взаимного проникновения. Однако в этом следует разобраться, ибо путаница с тем, что является данными, информацией и знаниями, в чем их различие, часто приводила к огромным расходам на технологические проекты, которые не давали нужного результата.

Рассел Аккоф, один из классиков исследования операций, предложил следующую, вполне убедительную иерархию:

[данные – информация – знания – понимание – мудрость].

Данные по Р. Аккофу – это некоторые неупорядоченные символы, рассматриваемые безотносительно к какому-либо контексту.

Информация – это выделенная и упорядоченная часть базы данных, обработанная для использования, то есть отвечающая на вопрос: «Кто?, Что?, Где?, Когда?»

Знание – это выявленные тенденции или существенные связи между фактами и явлениями, представленные в информации.

Понимание – это осознание закономерностей, содержащихся в разрозненных знаниях, позволяющее ответить на вопрос: «Почему?»

Мудрость – взвешенное, оцененное понимание закономерностей с точки зрения прошлого и будущего.

Продвижение по уровням иерархии понятий от «данных» к «мудрости» не есть механическое суммирование данных, информации, знаний.

Каждое из приведенных понятий является основой для последующего, материалом для получения элементов нового более высокого качества знаний. При этом, как считает Р. Аккоф, первые четыре понятия имеют дело с прошлым или с тем, что уже известно, а «мудрость» касается будущего.

Остановимся подробнее на таких базисных и широко используемых понятиях, как «данные», «информация», «знания».

1.1. Данные

Данные – это набор объективных фактов об объектах, событиях, явлениях, процессах, это все то, что регистрируется, описывается и воспринимается человеком. Данные могут быть цифровыми (факты, результаты измерений), графическими, аудио, видео и т.п. Они могут описываться на различных языках (символьном, математическом, графическом и т.п.).

Качественными мерами для данных являются своевременность, соответствие и точность. Организации иногда накапливают данные, потому что они основаны на фактах и поэтому создают иллюзию научной точности. Существует мнение, что если собрать достаточно много данных, то объективное точное решение будет автоматически предлагать себя само. Это ошибочное мнение по двум причинам. Во-первых, слишком много данных может затруднить определение и понимание подходящего. Во-вторых, и что наиболее важно, не существует врожденного смысла в данных. Данные описывают объекты, явления, факты и процессы лишь частично. Они не предоставляют оценок или интерпретаций и поэтому не всегда являются приемлемой основой для деятельности. Хотя материал для принятия решений может включать и данные, они не скажут вам, что делать. Данные ничего не говорят об их важности и соответствии. Но данные важны для организации в основном потому, что они являются исходным материалом для создания информации.

1.2. Информация

Информация – это данные в определенном контексте (необходимые пользователю, полезные для решения). Или [Drucker P., 1993] информация – это «данные, наделенные значимостью и целями». Не лишено оснований определение, что информация – это данные плюс метаданные, содержащие их описание (данные о данных). Продолжая эту логическую цепь рассуждений, знания можно определить как информацию плюс метаинформацию (информация об информации).

В дискуссиях по проблемам интеллектуальных ресурсов обсуждаются различные точки зрения на эти широко используемые понятия. Теория информации определяет информацию как меру неопределенности, устраняемую сообщением. В информационных технологиях под информацией понимаются данные, которые могут быть зафиксированы приборами или людьми, обработанные и представленные в той или иной форме. В коммуникационном маркетинге информацию определяют как содержание сообщения. То есть сфера применения понятия налагает некоторый существенный отпечаток на определение понятия информация.

Закон Российской Федерации «Об информации, информатизации и защите информации» (1995 г.) закрепил следующее определение: «Информация – “сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления”».

Как уже отмечалось, в отличие от данных, информация имеет смысл, соответствие и цель. Информация передается по организации посредством жестких (формальных) и мягких (неформальных) сетей. К первым относятся электронные письма и документы, традиционные письма и документы, посылки и пакеты Интернет-сети. Мягкая сеть менее видима и устойчива. Она возникает от случая к случаю во время встреч, дискуссий и т.п. К количественным мерам информации можно, кроме объема, отнести связность и число обменов. К качественным мерам – информативность и полезность.

Данные преобразуются в информацию различными способами. Рассмотрим некоторые из них [Davenport T., Prusak L., 1998]:

- *контекстуализация*: известно, для какой цели данные были собраны;
- *категоризация*: известны единицы анализа или ключевые компоненты данных;
- *вычисляемость*: данные могут быть проанализированы математически или статически;

- *корректировка*: ошибки убраны из данных;
- *сжатие*: данные могут быть обобщены в более сжатую форму.

Отметим, что компьютеры могут помочь добавить полезность и трансформировать данные в информацию, но пока они редко могут помочь с контекстом, и люди должны участвовать в категоризации, обработке и сжатии.

В общефилософском плане информация представляет собой некую субъективную реальность. В объективном мире существуют разнообразные свойства и отношения между субстанцией и энергией. Часть их воспринимается нашими органами чувств и созданными нами приборами, распознается и субъективно воспринимается как информация. Информация является, таким образом, семантической трансформацией изображения модели или объективной реальности. Информация реально существует только в представлении субъекта.

Информация, существующая в глубине сознания, именуется памятью. Память – информационная система, непрерывно занятая приемом, видоизменением, хранением, извлечением информации и возвращением ее в область сознания. Сознание есть субъективный образ объективного мира, то есть сознание – субъективная реальность.

1.3. Знание

В определении и систематизации знаний особое место занимают воззрения Карла Поппера, известного философа XX века [Popper K.M., 1972].

В соответствии с этими воззрениями, различают три типа (три мира) знаний:

- *мир знаний 1* – кодированные знания в физических системах (например, генетический код ДНК), который позволяет объектам адаптироваться к окружающей среде (знания в неодушевленных объектах);
- *мир знаний 2* – убеждения и склонности к убеждениям, представления о мире, прекрасном, справедливости, сохраняющиеся в испытаниях, эволюции и опыте (знания в одушевленных объектах);
- *мир знаний 3* – совместно используемые лингвистические формулировки, единицы знаний о мире, прекрасном и справедливости, которые помогают выжить в испытаниях и эволюции личности, группе, сообществу, команде, организации, обществу, приобретающему, формулирующему и оценивающему утверждения о знаниях (знания, существующие самостоятельно как артефакты).

То есть, мир 1 – физика, мир 2 – психология, мир 3 – продукты человеческого разума. Следуя рассуждениям К. Поппера (Popper K.R., Eccles J.C., 1977), эволюция знаний началась с развития биологических существ, которые достигали своих целей посредством ограниченных адаптивных и обучающихся способностей. Они имели мозг, но не имели разума. Разум развивался как система управления для мозга. Разум позволил биологическим существам создать субъективные проекции (модели) реальности и представления для совершенствования механизмов адаптации.

Эволюция пошла дальше. Она создала существа, которые не только имеют мозг, разум и сознание, но также существа, которые имеют язык и культуру. Эти существа используют язык и культуру для создания проекций (моделей), отражающих совместное представление о реальности. То есть язык и культура позволяют создать более объективные представления (утверждения о знаниях, модели), которые налагают ограничения на личностные, субъективные представления.

Специалисты по теории научного познания, начиная с Декартовой эпистемологии рациональности, тратят жизнь, пытаясь понять, что означает знать. Потому наши попытки будут носить прагматический характер, ибо нас интересует некое рабочее определение.

В Большой Советской Энциклопедии [Большая Советская Энциклопедия, 1972] приводится следующее определение:

«Знание – это проверенный практикой результат познания действительности и верное ее отражение в сознании человека».

Другого определения – «Знание – это обоснованное истинное мнение» – придерживаются многие философы, особенно эмпирики, кто полагает, что знание может быть проверено фактами.

«Знание – это информация в контексте». То есть именно контекст делает информацию знанием, если она (информация) согласуется и полезна в ситуационном контексте.

Ниже приводится несколько других определений. Они исходят прежде всего из того очевидного факта, что знания проистекают из информации, так же как информация выводится из данных.

«Знание – это способность превращать информацию и данные в эффективные действия» [Applehans W., 1999].

«Знание – это умение сотрудников компании решать стоящие перед ними проблемы».

Проблемы могут иметь отношение к разным направлениям и аспектам деятельности компании, таким, как: плановая, производственная, технологическая, финансовая, маркетинговая, юридическая и т.д. Знания накапливаются в компании по мере приобретения ее сотрудниками опыта, выполнения научно-исследовательских работ, обучения и повышения квалификации персонала, разработки стандартов и руководящих материалов компании.

«Знание – это совокупность информации, понимания, осведомленности, полученных в результате опыта» [Oxford Advanced Learners Dictionary, 1982].

Сходное определение дается в словаре [The American Heritage Dictionary of the English Language, 1992].

«Знание – это осведомленность, компетентность или понимание, достигнутое в результате опыта или обучения». И далее «Знание – это сумма или набор того, что воспринято, обнаружено или выучено».

В приведенных определениях доминирует образовательная компонента в формировании знания.

Большинство исследователей проблемы интеллектуальных ресурсов отмечают, что знания неотделимы от человека. Знания могут осознаваться владельцем (то есть человек может пояснить логику своих умозаключений и принимаемых решений), а могут и не осознаваться (то есть человек действует рационально, эффективно, но пояснить, почему он так действует, не в состоянии). Таким образом, знание – это смесь осознанного и неосознанного в решениях, действиях и поведении человека. Отчасти это происходит еще и потому, что до сих пор остается неизвестным, как знания хранятся в нейронных структурах головного мозга и как они используются человеком. Наблюдаем не процесс мышления, а его результат, который может фиксироваться и изменяться.

«Знание – это наличие в мозге человека определенных моделей действительности. Увеличение знания, то есть возникновение новых моделей действительности в мозге – это процесс познания» [Турчин В.Ф., 2000].

С учетом изложенного, представляет интерес следующее определение:

«Знание – это сложная сеть понятий и многообразных отношений (оценки, мнения, причинно-следственные и пространственно-времен-

ные связи и зависимости) между ними, которая сознательно (логически) или бессознательно используется нейронной сетью головного мозга при необходимости выработки новых суждений или принятия разнообразных решений».

Эта сеть понятий частично передается человеку генетически, как результат эволюционного развития, но в большей степени формируется в результате приобретения жизненного опыта, а также формального или неформального обучения. Эта сеть понятий пополняется новыми понятиями и связями в результате обучения и на основе индивидуальной, часто неосознанной, работы мозга человека.

С учетом сделанных пояснений знания специалиста можно представить в виде модели черного ящика (рис. 1.1).

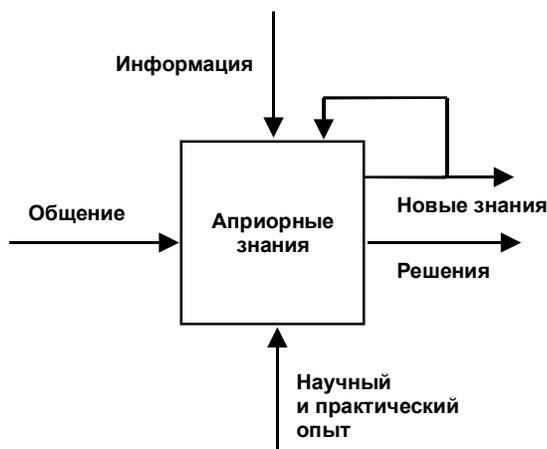


Рис. 1.1. Знания специалиста как модель черного ящика

Завершая данный раздел, имеет смысл привести рабочее определение понятия «знание», приведенное в Европейской Схеме по управлению знаниями [European Guide to good Practice in Knowledge Management, Part1]:

«Знание – это комбинация данных и информации, к которым добавлено мнение, мастерство и опыт эксперта, что в результате дает ценный актив, который может быть использован для оказания помощи в принятии решений».

1.4. Классификация знаний

Несмотря на многообразие и некоторую расплывчатость в определении столь сложного понятия, как «знание», многие исследователи единодушны в том, что вся совокупность знаний может быть разделена на следующие классы:

- эмпирические знания;
- теоретические знания;
- личностные знания;
- организационные знания;
- неявные знания;
- явные знания.

К *эмпирическим знаниям* относятся наблюдения, наблюдаемые явления.

К *теоретическим знаниям* относятся законы, теории, абстракции, обобщения.

По существу, наличие этих двух классов вытекает из теории познания. Гносеологическая цепочка, как известно, включает:

<факт – обобщенный факт – эмпирический закон – теоретический закон>.

Теория – это не только стройная система обобщения научного знания, это также способ производства новых знаний. Основными методологическими критериями научности знания являются: внутренняя согласованность, объективность, системность, историзм.

К *личностным профессиональным знаниям* относятся:

- знания познавательного плана («знаю, что») – глубокое владение предметом и сферой деятельности, достигаемое профессиональным путем интенсивного обучения (теоретического, эмпирического) и сертификации;
- прикладное мастерство («знаю, как») – способность применять правила и методы, относящиеся к предмету и сфере деятельности, для решения возникающих задач и проблем. Прикладное мастерство – наиболее распространенный профессионализм, создающий ценности;
- системное представление («знание, почему») – глубокое понимание всей системы взаимосвязей и взаимоотношений, причин и следствия, лежащих в основе определенной сферы деятельности;
- личная мотивация творчества («хочу знать, почему») – активная настроенность, внутренняя заинтересованность и воля к достиже-

нию успеха. Организации, культивирующие у своих сотрудников мотивацию к творчеству, имеют лучшие предпосылки для оперативной реакции на изменения в бизнес-среде, успешнее конкурируют на рынках новых товаров, продуктов и услуг.

К *организационным знаниям* относятся стратегические доктрины и программы, теоретические знания конкретной сферы деятельности, технологические и производственные стандарты, правила и инструкции, коммерческие знания. Организационные знания являются совокупностью знаний каждого из сотрудников, но не простой суммой, а специально отобранной и тщательно проверенной в практике деятельности групп, подразделений, отделений и компании в целом.

1.4.1. Неявные знания

К *неявным знаниям* (*tacit knowledge*), с учетом вышеизложенных определений, относятся опыт, мастерство, культура мышления, интуиция, хранящиеся в нейронных структурах головного мозга как результат генетической наследственности, образования и приобретенного жизненного опыта. Неявное знание – это способность человека к адаптации в меняющихся условиях.

Неявные знания существуют в умах специалистов, развиваясь во времени, через опыт, почерпнутый из профессиональной деятельности, книг, наставничества, а также обучения. Неявные знания зависят от жизненных ресурсов личности, от ее биофизических свойств и психологического потенциала.

Опыт (*experience*) и эксперт (*expert*) – связанные слова и понятия. Оба они происходят от латинского глагола, означающего «подвергнуть проверке». Эксперты, люди с глубокими знаниями по предмету, были обучены и проверены опытом.

Одним из преимуществ опыта является его опора на историческую перспективу, с позиций которой рассматриваются и понимаются новые ситуации и события. Знание, рожденное опытом, распознает схожие ситуации и обнаруживает связь с тем, что было.

Применение опыта может быть таким простым, как выявление в бизнесе сезонных спадов в продажах, но может быть и таким сложным, как ощущение ученого, что новое направление исследований сулит полезные результаты.

Если знания специалиста и эксперта не пополняются на основе непрерывной научной и практической деятельности, если эксперт пере-

стает проверять себя и развиваться, его опыт становится неким устоявшимся мнением или догмой.

Неявное знание работает через эвристики – некие гибкие руководства к действию, формирующиеся посредством проб и ошибок в результате наблюдений и эволюции мыслительной деятельности мозга. Весьма часто с помощью эвристик специалист приходит к ответу интуитивно, не осознавая и не понимая, как это происходит.

Опыт, интуиция и культура мышления, передающиеся по наследству и будучи сформированными, отчасти образованием, обеспечивают знанию способность преодолевать сложность. Известное искушение – найти простые ответы на сложные вопросы, найти простые решения сложных проблем, работать с неопределенностью как будто ее не существует, – чаще всего не ведет к успеху. Применительно к данному контексту, здесь уместно привести известное определение такой современной науки в области оптимизации управления, как исследование операций, данное Томасом Саати: «Исследование операций – это наука о том, как в сложных случаях давать посредственные советы, когда иными способами могут быть даны советы заведомо более худшие».

1.4.2. Явные знания

К *явным знаниям* (*explicit knowledge*) относятся описания теорий, методов, методик, технологий, механизмов и машин, конструкций, систем и т.п.

Явные знания хранятся на реальных физических носителях (в книгах, бумажных документах, рисунках, схемах, фильмах, аудио и видео записях, магнитных и электронных файлах и базах данных и т.п.).

То есть к явным знаниям относятся многочисленные компоненты информационных систем компании, такие, как:

- **данные** (файлы с данными, базы данных, базы инструкций и правил регламентного характера);
- **документы** (файлы с текстами в разных форматах);
- **программы** (расчетные, аналитические, управления данными, графические, экспертные), реализующие разнообразные алгоритмы решения задач;
- **адреса ресурсов и ссылки**, фиксирующие местонахождение различных информационных ресурсов в архивах компании и в сетях интранет и Интернет.

К явным, четко структурированным и проверенным знаниям относятся и описания объектов интеллектуальной собственности (ИС) [Борохо-

вич Л., Монастырская А., Трохова М., 2001; Калятин В.О., 2000]. Применительно к научно-технической и производственной сфере патентуемыми объектами интеллектуальной собственности являются изобретения, полезные модели, промышленные образцы и товарные знаки. К неохраняемым патентами объектам ИС относятся ноу-хау, техническая документация, результаты НИОКР. Объектами авторского права являются программы ЭВМ, базы данных, топологии интегральных схем.

Устоявшаяся классификация объектов ИС научно-технической и производственной сферы приведена на рис 1.2.



Рис. 1.2. Классификация объектов интеллектуальной собственности

Иначе говоря, явные знания компании – это все то, что остается в ее офисах, когда все сотрудники уходят домой, и без чего в значительной мере также невозможна их эффективная деятельность.

Явные знания, несмотря на стремительный рост их объемов в книгах, журналах, электронных базах и сетях, являются лишь частью всех знаний. Обучение по книгам, через интранет и Интернет – это работа только с явными знаниями. И не случаен в этом отношении определен-

ный протест педагогического сообщества, и школьного, и вузовского, относительно чрезмерного увлечения этими формами в ущерб общению с учителем, в ущерб работе в группе, в ущерб дискуссиям и сотрудничеству. При общении у собеседника может неожиданно возникнуть какой-то вопрос, который он неосознанно сформулировал, а у отвечающего может возникнуть объяснение, которое он не планировал дать, но которое появилось в связи с обсуждением, ответами на вопросы и комментариями окружающих. Таким образом, сотрудничество приводит к формированию новых знаний.

1.4.3. Связь между явными и неявными знаниями

Между явными и неявными знаниями происходит непрерывный обмен и трансформация [Нонака И., Такеучи Х., 2003]. Этот циклический процесс показан в таблице.

Преобразование знаний между явной и неявной формами

Используем	Получаем	
	Неявные (tacit)	Явные (explicit)
Неявные (tacit)	Обобществление (Socialization)	Отчуждение (Externalization)
Явные (explicit)	Усвоение (Internalization)	Комбинирование (Combination)

Преобразование знаний в пределах одной формы или при переходах между формами происходит в результате следующих процессов:

- *Обобществление* (tacit to tacit) – включает формирование и передачу неявных знаний в коллективных формах (дискуссии, семинары, команды и т.п). При этом чаще всего это происходит без создания явных знаний.
- *Отчуждение* (tacit to explicit) – происходит путем концептуализации неявных знаний, извлечения и выявления их и, в конечном счете, их формулировании и фиксации в той или иной форме как итог дискуссий, семинаров, мозгового штурма и т.п.
- *Комбинирование* (explicit to explicit) – осуществляется в ходе распространения явных знаний через E-mail, в процессе составления обзоров и сводных отчетов. Увеличение явных знаний происходит здесь за счет пополнения баз данных коллективного пользования, классификации и систематизации файлов и документов и т.п.

- *Усвоение* (explicit to tacit) – осуществляется в процессе чтения и изучения документов из баз данных, журналов и книг. Это приводит к усвоению знаний, которые были созданы другими (возникновению новых неявных знаний у познающего субъекта), а также, возможно, к созданию новых знаний в результате мыслительной деятельности в процессе познания.

Перечисленные процессы не происходят изолированно, а выполняются вместе в различных комбинациях в обычных производственных ситуациях. Например, знания создаются в результате взаимодействия сотрудников и явных знаний. Посредством разнообразных взаимодействий неявные знания материализуются и распределяются. Хотя индивидуумы (например, служащие) повседневно участвуют во всех этих процессах, с точки зрения управления знаниями и, следовательно, с точки зрения организации наибольшая ценность возникает от их комбинации, так как при этом создаются новые знания, распространяются и усваиваются другими служащими, которые могут использовать их и таким образом формировать новый опыт и неявные знания. Последние, в свою очередь, могут использоваться совместно с другими служащими и т.д.

Траекторию развития описанного процесса, имея в виду его многоплановость и слабую предсказуемость, видимо, лучше всего описывает спиралеобразный характер движения.

Существуют различные точки зрения относительно доли явных и неявных знаний в различных организациях. На рис. 1.3 приведены результаты исследований состава знаний представительной группы компаний США, осуществленных Delphi Group в 2000 г.

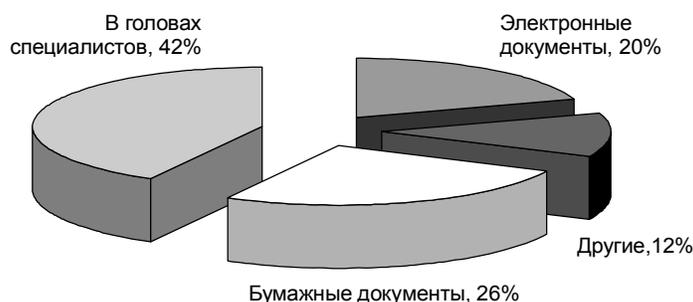


Рис. 1.3. Состав знаний компаний USA (результаты исследования Delphi Group, 2000 г.)

Несмотря на то, что приведенные данные относятся к 2000 г. и что это результаты исследований весьма продвинутых в смысле IT-технологий американских компаний, доля знаний, хранящаяся на электронных носителях, невелика (20%), доля явных знаний (с учетом остальных носителей) лишь немногим превышает половину. Очевидно, что в большинстве других стран, включая Россию, доля знаний, хранящаяся в головах специалистов, будет существенно выше. И это все касательно корпоративных знаний, которые были основным объектом исследования Delphi Group. Если же иметь в виду всю совокупность знаний, которыми обладают сотрудники любой организации с учетом их образования, предшествующего научного и практического опыта, то станет очевидным еще более высокий удельный вес неявных знаний.

Приведенная классификация необходима с точки зрения понимания свойств отдельных категорий знаний. Вместе с тем необходимо понимание того, что знания находятся, главным образом, в голове человека, мозг которого обладает уникальной способностью создавать и работать со знанием.

При выполнении интеллектуальной работы человек усваивает данные и информацию, пытаясь найти решение некоторой проблемы. Проблема может носить внешний по отношению к индивидууму характер или внутренний, если, например, речь идет о повышении объема и уровня его знаний. По мере усвоения новых данных и информации человек (осознанно или неосознанно) пытается встроить их в общую структуру своих знаний. При этом в структуре его представлений могут возникать противоречия и сложности, которые человек тем или иным способом преодолевает. Упрощение сложности такого рода достигается в результате мыслительной деятельности человека путем новых обобщений и уточнения (развития) связей. Эти обобщения и связи позволяют упорядочить хранение и упростить понимание новой информации, а также ее запоминание и использование.

Человек в определенной мере ограничен в объемах запоминания разрозненных данных и сведений. Ему гораздо проще работать с упорядоченной информацией, со связанными наборами понятий, с цельными структурами знаний. Как разлитую жидкость сила притяжения собирает в более низких местах, так и головной мозг человека собирает отдельные элементы понятий и отношений в слитые, связанные структуры.

Если связывания не происходит, мозг уточняет понятия, достраивает структуры, связи, которые, в конце концов, позволяют ему достигнуть

цельного представления. Человек находится в состоянии творческих мук до тех пор, пока не наступает момент формирования ясного понимания явления, ситуации, проблемы.

Современные информационные технологии оказывают существенную помощь человеку в преодолении имеющихся ограничений мозга в части объемов и скорости поиска и связывания информации и понятий из многих разнообразных источников в единую сеть, и в этом, очевидно, состоит их постоянно возрастающее значение.

1.5. Свойства знаний

Знания представляют особый вид интеллектуальных ресурсов, обладающий рядом специфических свойств.

Знания долговечны, ибо они нематериальны. Здесь нет парадокса. Даже такие прочные и долговечные творения человечества, как египетские пирамиды подвержены эрозии и разрушаются от времени. А вот геометрическая модель пирамиды нетленна.

Знания инвариантны к пространству. Они легко распространяются, в особенности по современным цифровым сетям телекоммуникаций. Причем это происходит практически без затрат по сравнению с материальными ресурсами (нефть, уголь, руда и т.п.). И потому глобализация знаний по существу является свершившимся фактом.

Знания – постоянно увеличивающийся ресурс. Его расширенное воспроизводство обеспечивают наука, технический прогресс, образование. В последнее время благодаря развитию региональных и международных связей, новых коммуникационных возможностей, которые представляют современные системы связи и транспорта, а также деятельности транснациональных корпораций темпы прироста знаний и их использование кратно возросли.

Как не без основания утверждает Paul [Romer P, 1995]: «знания – единственный неограниченный ресурс, единственный актив, который увеличивается по мере его использования». Идеи рождают новые идеи, переданные (проданные) знания остаются с теми, кто ими делится (продает).

Знания можно продавать многократно, ибо они слабо отчуждаемы. У того, кто их продает, остается не меньше. Следует признать, что в коммерциализации знаний имеет место специфическая проблема. Мало кто готов покупать знания, пока не поймет, в чем их суть, а познав, – теряет желание к их приобретению по понятным причинам.

Знания чувствительны к фактору времени. Их ценность сильно связана с временем тогда, когда речь идет о принятии решений. Имеет место и старение знания, по аналогии с моральным износом оборудования.

В США даже установлена своеобразная единица устаревания знаний – период полураспада знаний – время после обучения, в течение которого профессионалы теряют половину первоначальной компетентности. К знаниям с длительным периодом полураспада относятся фундаментальные знания.

Знания социальны, являясь одновременно и общественным, и частным благом. Знания, не востребованные рынком, существуют в форме общественного блага (культура, образование, фундаментальная наука и т.д.). Знания превращаются в частное благо, когда они востребованы рынком как экономический ресурс.

Знание – орудие конкуренции. Интеллектуальный капитал (intellectual capital), менеджмент знаний (knowledge management) признаются сегодня важнейшими инструментами и ресурсами индустриальных и сервисных компаний в обеспечении эффективности их деятельности и, следовательно, в обеспечении их преимуществ в конкурентной борьбе. Отсюда и повышенный интерес активных компаний к созданию офисов и технологий управления знаниями, к созданию и развитию систем управления знаниями.

1.6. Знания как движущая сила новой экономики

Знания, интеллектуальная собственность, интеллектуальные ресурсы осознаются в последние десятилетия в качестве важнейшей движущей силы новой экономики, экономики «третьей волны» по образному определению Э. Тоффлера [Тоффлер Э., 2002]. В нарисованной им метафорической картине «трех волн» экономического развития человечества к «первой волне» относятся общества с аграрной экономикой, ко «второй волне» – общества с индустриальной экономикой и, наконец, «третья волна» – это общества с экономикой, основанной на расширенном воспроизводстве знаний.

Большинство стран мира, в том числе высокоразвитых, пока купаются в водах «второй волны». И лишь немногие страны – лидеры мировой экономики – находятся на подъеме «третьей волны», в стадии перехода от индустриальной экономики к экономике, основанной на знаниях.

Появление термина «интеллектуальный капитал» относится к первой половине 90-х годов. В его популяризации заметную роль сыграла книга Тома Стюарта «Интеллектуальный капитал – главное богатство вашей компании» [Sewart T.A., 1998]. Следует отметить, что к тому времени и, тем более, в последующие годы интеллектуальный капитал ведущих национальных и транснациональных компаний стал составлять заметную и постоянно растущую долю в их рыночной стоимости.

Под рыночной стоимостью или стоимостью акционерного капитала понимается балансовая стоимость материальных активов компании плюс стоимость ее нематериальных активов. К последним, как отмечалось во введении, относятся человеческие активы, интеллектуальные активы, интеллектуальная собственность, структурные и бренд-активы.

На рис. 1.4 в качестве иллюстрации приведены данные о структуре рыночной стоимости трех известных в мире компаний.

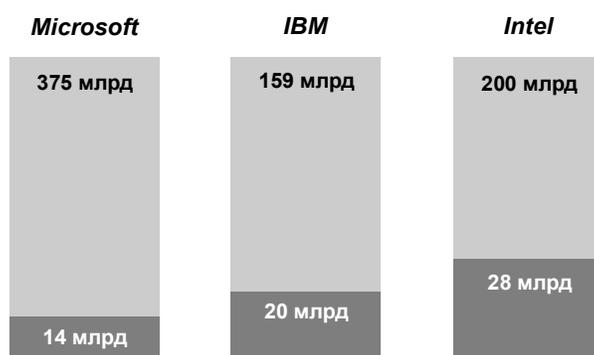


Рис. 1.4. Структура рыночной стоимости компании, внизу балансовая стоимость (по данным 1999 г.) [Tiwaka A., 2000]

Известно, что Microsoft, IBM и Intel – это компании, работающие в сфере высоких компьютерных, информационных и телекоммуникационных технологий, и высокий удельный вес интеллектуальных активов естествен для них.

Однако, как показывают экономические исследования, существенное превосходство рыночной стоимости компании над балансовой – это не исключение, а скорее правило, характерное для многих высокотехнологичных компаний. Не балансовая стоимость, а уровень текущих и будущих доходов определяет цену, которую готовы заплатить за нее

покупатели пакетов акций либо бизнеса в целом. Превышение рыночной стоимости компании над балансовой стоимостью активов тем значительней, чем выше корпоративная культура и знания ее менеджмента, чем перспективней товар и продукт компании, чем выше уровень используемых ею технологий, чем выше конкурентные преимущества компании.

Экономика, основанная на знаниях, относится к ресурсной экономической теории, являющейся наиболее современной, в составе других экономических концепций. Предметом изучения каждой из них является определенная и весьма важная сторона деятельности фирмы: производство (неоклассическая теория), транзакции (теория транзакционных издержек), инновации (теория эволюционной экономики), ресурсы (ресурсная теория). С точки зрения ресурсной теории, фирма – больше, чем административная единица. Она есть совокупность производительных ресурсов, распределенных между различными пользователями с помощью административных и экономических решений. Культивирование ресурсов в компетентной фирме подразумевает постоянный поиск, приобретение, производство, развитие и получение экономической ренты от эффективного использования ресурсов, важнейшим из которых признается человеческий капитал и нематериальные активы.

В своей работе организация всегда опиралась на знания того, что делать и как делать, считая это само собой разумеющимся. Возрастающий экономический вес знаний и их основополагающая роль в конкурентной борьбе увеличили их роль. Компании осознают заинтересованность в управлении знаниями и потому начинают наращивать инвестиции в их создание, сохранение, использование так же, как они традиционно наращивали, сохраняли и использовали другие свои ресурсы. И даже более того, ибо знания осознаются как важнейший стратегический ресурс в обеспечении конкурентных преимуществ.

Эффективность производства инновации – две основных сферы приложения знаний. Эффективность достигается за счет новых подходов, методов и технологий в обеспечении роста объемов производства и снижения затрат. Инновации обеспечивают создание новых продуктов, товаров и услуг, новых бизнес-процессов и предприятий.

Реальные дивиденды приносят компаниям патенты и авторские свидетельства, торговые марки, сведения о профессиональных качествах сотрудников, информация о клиентной базе, сети лояльных поставщиков, партнеров и потребителей и т.п.

Особенность знания как экономического блага состоит в том, что оно чаще всего присутствует в приобретаемых товарах и услугах, материализуясь в них. Значительно реже знание является предметом продажи само по себе.

Реально существует ряд путей, по которым знание преобразуется в экономические результаты.

Во-первых, как утверждает П. Друкер [Drucker P., 1993], «в основе любого бизнеса лежит особое (профессиональное) знание».

Во-вторых, с помощью знания модернизируются существующие и создаются новые продукты и услуги, пользующиеся спросом.

В-третьих, новое знание повышает эффективность производства и управления.

Наконец, в-четвертых, новые знания обеспечивают рост компетентности персонала до уровня, соответствующего уровню развития технологий, изделий и услуг.

Для эффективного использования знаний и интеллектуальных ресурсов компании важное значение имеет концепция обучающейся организации (learning organization). П. Сендж в своей книге «Пятая дисциплина» [Senge P., 1990] определяет обучающуюся организацию как организацию, в которой невозможно не учиться, ибо обучение включено в саму жизнь. При этом выделяются следующие основные составляющие обучающейся организации: системное мышление, персональное мастерство, ментальные модели, общее видение, групповое обучение.

Системное мышление состоит в осознании всеми служащими компании зависимости общих результатов от индивидуальных.

Персональное мастерство заключается в достаточном уровне знаний, умений, компетенций.

Модели ментальности представляют глубоко укоренившиеся обобщения и представления, влияющие на понимание мира и совершаемых действий.

Общее видение компании **реализуется в ее стратегии и вытекающих из нее планов для структурных единиц.**

Групповое обучение заключается в активных формах совместной деятельности, в решении конкретных проблем и задач по повышению ее эффективности.

Эффективное формирование и использование интеллектуальных ресурсов во многом зависит от организационной среды, в которой проте-

кают эти процессы. Важнейшей характеристикой такой среды является корпоративная культура.

В понятии «корпоративная культура» объединяются миссия организации, культивируемые ценности и убеждения, стиль руководства, нормы, правила поведения и принятие решений.

Предпринимаются различные попытки анализа и сопоставления корпоративных культур, в том числе с помощью следующих дихотомических признаков:

- интеллектуально-ориентированные или интеллектуально-безразличные культуры;
- культуры, ориентированные на приобретение знаний или на создание знаний;
- культуры, ориентированные на накопление знаний или на их распространение.

Представляется достаточно очевидным, что с точки зрения эффективного управления интеллектуальными ресурсами наиболее прогрессивным является интеллектуально-ориентированный тип корпоративной культуры, оптимальным образом сочетающий в себе приобретение знаний и технологий со стороны и их собственное производство и обеспечивающий организации конкурентные преимущества и высокую мобильность в меняющихся условиях внешней среды.

Глава 2

ЖИЗНЕННЫЙ ЦИКЛ ЗНАНИЙ

Все без исключения организации (компании) взаимодействуют со средой для достижения своих целей.

Основной целью любой организации является производство и предоставление продукции и/или услуг потребителям. Для достижения этой цели руководители и сотрудники организации используют собственные знания, знания партнеров, поставщиков и потребителей в собственных бизнес-процессах, а также в процессах взаимодействия и сотрудничества.

Организационные (социальные) системы адаптивны по своей природе, то есть приспособляются, обучаются, развиваются.

Любая организация зарождается на определенном уровне знаний своих сотрудников, функционирует, используя и пополняя корпоративные знания (о продукте, о технологиях, о потребителях и рынке), являясь одновременно и получателем, и источником такого рода знаний.

В организации и вне ее происходит непрерывный процесс движения информации и знаний. Знания поступают в организацию в разнообразных формах (набор персонала, повышение квалификации, изучение специальной литературы, приобретение лицензий), формируются и развиваются в процессе проектной, инженерно-технической, производственной и маркетинговой деятельности, передаются вместе с товарами и услугами.

Знания движутся в форме человеческого капитала при перемещении рабочей силы, при оказании или получении консалтинговых услуг, а также в виде интеллектуальных продуктов. К последним относятся: результаты фундаментальных исследований, результаты НИОКР, техническая документация, лицензии и т.п.

Таким образом, в каждой организации, вне зависимости от сферы ее деятельности, вне зависимости от степени ее взаимодействия с внешним миром, осуществляется непрерывный процесс движения знаний

большей или меньшей степени интенсивности, то есть осуществляется некий процесс производства, обобщения и распространения знаний, в том числе через продукцию и услуги потребителям.

2.1. Основные этапы жизненного цикла знаний

В соответствии с методологией системного анализа [Перегудов Ф.И., 1976], в жизненном цикле знаний (ЖЦЗ), как и в жизненном цикле любой сферы деятельности, можно выделить три основных процесса (этапа):

- выявление потребности в знаниях;
- производство (создание) знаний;
- потребление (утилизация) знаний.

На рис. 2.1 приведен жизненный цикл знаний, предложенный [McElroy M.W., 2003], встроенный в бизнес-среду организационной системы.

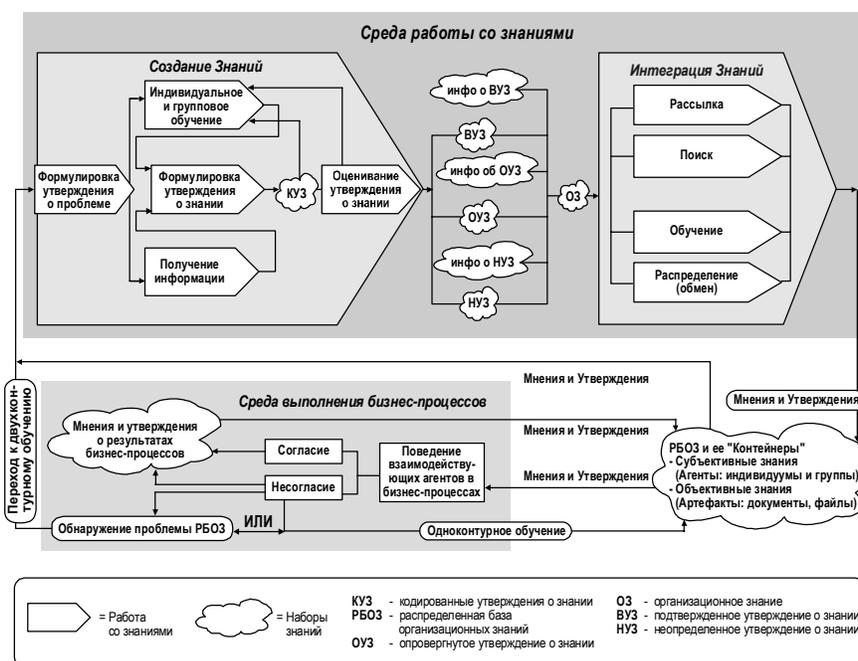


Рис. 2.1. Жизненный цикл знаний

Данная концептуальная модель нуждается в пояснениях относительно различных вариантов и условий ее функционирования.

- Персонал компании склонен участвовать в пополнении знаний в результате осознания разрыва в их текущем и целевом (желательном) состоянии.

- Выявление таких разрывов (проблем) индивидуумами или подразделениями компании инициирует в системе деятельность по обучению, которая ведет к появлению догадок, суждений, теорий и предложений (знаниевых утверждений – learning claims) относительно способов решения проблем.

- По мере обучения и генерации новых знаниевых утверждений формируются группы коллективного обсуждения, где происходит неформальный обмен идеями и их критический доброжелательный анализ.

На индивидуальном и групповом уровне этот процесс может продолжаться сколь угодно долго, пока новые идеи и утверждения не будут использованы в практической деятельности. Как правило, этому предшествует проверка истинности идей на организационном уровне компании. Процессы формулировки «утверждений о знаниях» и их оценивание имеют прямое отношение к созданию знаний.

- Не все «утверждения о знаниях», сформулированные индивидуумами или группами, будут успешными на организационном уровне. Положительно оцененные относятся к категории выживших (успешных); те, которые не будут оценены, попадут в одну из двух категорий: нерешенные «утверждения о знаниях» или ошибочные «утверждения о знаниях».

Информационные отчеты об оценках «утверждений о знаниях» также являются знаниями и результатами процесса создания знаний. Они, по сути, являются метаутверждениями (т.е. утверждения об утверждениях).

- После оценки «утверждений о знаниях» на организационных уровнях, менеджерами компании будут предприняты попытки доведения их содержания до других групп и структур. Этот процесс управляемого обмена и распространения знаний является, по существу, процессом интеграции (потребления, использования) знаний.

- Интегрированное знание хранится в двух формах:
 - ментально хранимое знание индивидуумов или групп (неявное знание);

- объективно хранимое знание в форме явных лингвистических выражений и артефактов (документов, файлов, баз данных и т.п. – явные знания).

Объединение неявных и явных знаний организации может рассматриваться как распределенная база (корпоративных) знаний организации (DOKB, organization's distributed organizational knowledge base).

К субъективным формам знаний DOKB относятся мнения (beliefs) или предрасположенность к мнениям, содержащиеся в умах индивидумов, групп, команд, сообществ, подразделений и т.п. К объективным формам знаний DOKB относятся лингвистические выражения и/или кодированные знания, содержащиеся в файлах, документах, электронных БД, на дисках, видеозаписях, аудиолентах, в книгах, статьях, докладах, презентациях, лекциях, музыке и др.

- ЖЦЗ начинается с обнаружения в практической деятельности (в бизнес-процессах) дефицита знаний о том, как осуществить переход из текущего в целевое (желаемое) состояние, и заканчивается после выбора проверенных «утверждений о знаниях» в DOKB. В следующем затем этапе использования знаний в реальных бизнес-процессах могут возникнуть новые проблемы. Их разрешение осуществляется ЖЦЗ по линиям связей сред обработки (создания) знаний и выполнения бизнес-процессов, как это показано на рис. 2.1.

Остановимся поподробнее на сути основных этапов ЖЦЗ и соответствующих им процессах.

В обширной литературе по этой проблеме, а также в Европейской Концепции по управлению знаниями [European Guide to good Practice in Knowledge Management, Part 1] выделяют пять этапов (процессов) жизненного цикла (рис. 2.2).

Между системной структуризацией ЖЦЗ и приведенной на рис. 2.2 нет принципиальных различий, если иметь в виду, что на рис. 2.2 некоторые из них детализированы. Так, в процессе «производство знаний» выделено два подпроцесса – «создание знаний» и «хранение знаний». В процессе «потребление знаний» также выделено два подпроцесса – «распространение знаний» и «использование знаний».

Как следует из рис. 2.2, жизненный цикл знаниевого процесса, содержащий пять основных этапов (выявление, создание, хранение, распространение и использование), обеспечивает поддержку бизнес-процессов организации, имея в виду и всех его внешних участников: потребителей продукции и услуг, клиентов, партнеров.



Рис. 2.2. Европейское представление ЖЦЗ

К способствующим факторам успешных бизнес-процессов необходимо отнести персональные способности сотрудников компании и способности организации как системы – организационные способности. К персональным способностям сотрудников относятся амбиции (стремления), мастерство, опыт, корпоративное поведение и т.п. К организационным знаниям и способностям относятся понимание миссии, стратегии, структуры и процессов деятельности, а также инфраструктуры и технологий деятельности.

2.2. Базовые виды деятельности, связанные со знанием

Как уже отмечалось в предыдущем параграфе, ЖЦЗ включает пять основных процессов, пять базовых видов деятельности. Как правило, все они встречаются в организациях, культивирующих методы и технологии управления знаниями. Они выявлены в результате специального обследования более чем 150 европейских компаний и фирм [Mertins K., Heisig P., Vorbeck J. (eds), 2003].

Далее остановимся на описании каждого из базовых видов деятельности, связанных со знаниями.

2.2.1. Идентификация (выявление) знаний

Речь в данном случае идет как об идентификации знаний, так и о выявлении потребности в них для конкретной предметной области и конкретного вида деятельности.

Первичным в данном виде деятельности является анализ стратегических целей в производстве конкурентоспособных товаров и услуг (традиционных или принципиально новых), выявление знаний, необходимых для достижения этих целей. На этом этапе важно осуществить «анализ разрыва», то есть выявить, какие существующие знания доступны и какие знания отсутствуют. Анализ необходимой информации и знаний выполняется как на уровне организации и ее основных структурных подразделений, так и на персональном уровне, то есть на уровне отдельных специалистов и профессиональных групп.

Идентификация существующих и необходимых знаний является весьма важным элементом в обосновании решений для обеспечения инновационного развития компании. При этом следует учитывать не только внутренние процессы и потребности компании, но и измеренные в количественном и качественном отношении требования потребителей.

К методам и инструментам, поддерживающим этот вид деятельности в организациях, относятся: системный анализ, семантический поиск, мозговой штурм, методики построения карт знаний, систематизированные опросы потребителей.

2.2.2. Создание новых знаний

Существует много способов создания новых знаний. Создание новых знаний может происходить при выполнении фундаментальных и прикладных НИР, в процессе деятельности аналитических и экспертных групп, при проектировании сложных объектов, процессов и систем. Заказчиком такого рода исследований, анализов, экспертиз, проектов может быть организация или ее структурные подразделения, преследующие определенные цели инновационного развития бизнеса.

На персональном или групповом (командном) уровне новые знания чаще всего являются результатом обучения, повышения квалификации, целенаправленного усвоения новой информации путем чтения профессиональных журналов и книг в традиционной или электронной форме, продуктом общения и взаимодействия при решении практических проблем и задач, результатом логического вывода и метода проб и ошибок.

При этом максимально должен быть задействован накопленный ими опыт, их явные и неявные (скрытые) знания, из которых проистекают новые знания (явные и неявные). Их часто фиксируют в виде лучших практических решений (ЛПР).

Другим источником новых знаний является прием на работу (на временной или постоянной основе) специалистов недостающего профиля или специалистов, имеющих более высокий уровень знаний и опыт работы в других организациях. В качестве еще одного источника новых знаний может стать покупка другой компании вместе с ее специалистами, знаниями, технологиями и опытом.

2.2.3. Хранение знаний

Хранение знаний в организациях необходимо для формирования и наращивания активов знаний (интеллектуального капитала, баз знаний), питающих, как это следует из рис. 3.1, знаниевый и бизнес-процессы.

Ввиду существенного различия явных и неявных знаний различаются и способы их хранения.

Явные знания – текстовые документы, электронные таблицы, базы данных, Web-страницы, чертежи, схемы, почтовые сообщения и т.п. хранятся в специально создаваемых для этой цели репозиториях знаний. Поскольку техническую и технологическую основу такого рода хранилищ составляют компьютеры и информационные технологии, то применяются и соответствующие способы помещения, пополнения и извлечения знаний из репозитариев.

Перед помещением явных знаний в репозитарий осуществляется их описание с помощью определенных языков высокого уровня (например, XML). Осуществляется также классификация и систематизация знаний, без чего немислимо их эффективное хранение с целью обеспечения эффективного поиска.

Как правило, репозитарии знаний оснащены эффективными средствами (программными, техническими) для трансформации форм представления знаний. Дело в том, что по мере технического прогресса появляются новые и совершенствуются традиционные физические носители информации. Соответственно множатся и формы представления и хранения знаний, использующие электронные, магнитные, лазерные, аудио и видео носители. Использование в репозитариях знаний разнообразных носителей информации диктуется, с одной стороны, удобст-

вом и эффективностью их применения, с другой – надежностью и долговечностью хранения.

Для того чтобы потенциал репозитария знаний использовался в основных бизнес-процессах организации, необходимо осуществлять актуализацию и верификацию хранящихся в нем знаний применительно к новым задачам и меняющимся условиям бизнес-среды.

Проблема заключается в том, что, следуя только путями традиционного управления документами, даже с помощью лучших программных систем такого рода, управление знаниями может воспользоваться только результатами их явной или неявной классификации. Ограниченность такого подхода в части выделения контента и семантического анализа документов побуждает к использованию экспертных систем, которые, со своей стороны, нуждаются в очень строгой структуризации и кодировании всех знаний, что также не всегда достижимо либо сопряжено со значительными издержками. Определенный компромисс между повторным использованием знаний, уровнем формализованности и стоимости кодирования достигается при построении депозитариев с использованием метаданных, метаописаний и онтологий, о которых речь пойдет в последующих главах.

Неявные знания хранятся, прежде всего, в нейронных структурах головного мозга сотрудников компании. Кроме того, они «хранятся» в группах, в командах, в организационных формах ежедневной деятельности (в правилах выполнения рутинных работ и процессов), которые описаны либо даже не описаны явно. Поскольку сотрудники, группы и команды постоянно присутствуют в компании, их знания доступны ей и используются многократно во всех без исключения бизнес-процессах. Они актуализируются и пополняются при решении новых задач или традиционных задач в меняющихся условиях бизнес-среды.

2.2.4. Распространение знаний

Целью данного вида деятельности (процесса) является передача знаний в нужное место, в нужное время, с нужным качеством.

Распространение знаний происходит многими способами. Знания могут распространяться через документы, книги и журналы, путем пополнения баз данных и баз знаний с телекоммуникационным доступом. Это так называемый «складской подход», когда потребители обращаются к легко доступному репозитарию знаний.

Но большая часть знаний лучше всего передается от человека к человеку посредством общения, сотрудничества конференций и семинаров, тренингов и наставничества. Такая передача знаний именуется «потокосовым подходом». Здесь особенно важным является создание благоприятной обстановки для обмена идеями и опытом.

К методам и инструментам, которые поддерживают обмен знаниями, относятся интранет-порталы, распределенные базы данных и упоминавшиеся выше семинары, обучение, тренинги, ротация кадров.

Существенным в распространении знаний является наличие на них спроса, продиктованного потребностями инновационного развития бизнес-процессов компании, интересов профессионального роста и развития личности специалиста.

2.2.5. Использование знаний

Знания и управление знаниями – не самоцель. Знание как интеллектуальный ресурс только тогда может что-то добавить к стоимости, если оно используется в компании.

Множество знаний остаются неиспользованными либо не используются повторно. Главной задачей данного процесса, данного вида деятельности является создать условия, чтобы все усилия, затраченные в предыдущих видах деятельности, окупались.

Использование знаний является завершающей стадией процесса. Одновременно, в ЖЦЗ, и формально, и фактически он является первым по важности, отправной точкой для того, чтобы знания создавались, хранились и распространялись. Именно в ходе использования знаний обнаруживаются разрывы в знаниях, на практике проверяется истинность «знаниевых утверждений», коммерческая полезность знаний в смысле повышения стоимости продукта или снижения затрат на его производство, приобретаетсЯ новый опыт как источник новых знаний.

Использование знаний зависит, прежде всего, от того, насколько сотрудники компании способны и желают это делать. В этом процессе недостаточно полагаться на профессиональную любознательность и добросовестность сотрудников. Компания должна стимулировать (побуждать) использование новых знаний различными формами явного признания и материальными поощрениями, продвижениями по службе.

2.3. Рынок знаний

В соответствии с ЖЦЗ и соответствующими ему основными видами деятельности в любой организации осуществляется процесс движения знаний, использующихся с той или иной степенью продуктивности.

Движение знаний во многом определяется формами и процедурами обмена знаниями, в котором принимают участие структурные подразделения и многочисленные сотрудники различного профиля.

Многие инициативы по управлению знаниями заканчиваются безуспешно, ибо основываются на утопических представлениях, что знания перемещаются без трения и мотивов, что специалисты будут делиться ими, не заботясь о том, что они получают или потеряют при этом.

Там, где есть обмен, имеет смысл говорить о рынке [Davenport T., Prusak L., 1998]. Подобно рынку товаров и услуг, имеет место и рынок знаний, на котором совершаются знаниевые транзакции (сделки). Только потому, что объект обмена не осязаем, не означает, что рыночные силы менее сильны.

Люди ищут знания, так как ожидают, что они помогут преуспеть в работе. Знания являются действенным средством для снятия неопределенности, без чего невозможно принятие обоснованных решений. Когда сотрудник или подразделение передают свои знания, то ожидают получить определенную выгоду.

В пределах одной организации денежная форма оплаты обычно не используется при знаниевых транзакциях. Однако это не означает, что не действуют иные побудительные мотивы и ценности, мотивирующие и балансирующие обмены знаниями. Рынок знаний, подобно любому другому рынку, является системой, в которой участники обмениваются дефицитными ресурсами за настоящую или условную стоимость.

Прежде чем приступить к теме управления знаниями, имеет смысл рассмотреть состав основных участников рынка знаний, их интересы и цели, а также факторы, воздействующие на ценовые параметры и механизм знаниевых транзакций.

Основными участниками рынка знаний являются покупатели, брокеры и продавцы знаний.

Покупатели или искатели знаний – это специалисты или организации, которые пытаются решить проблемы, сложность и неопределенность которых выше их априорных знаний. Покупатели знаний ищут интуитивные подходы, оценки, понимание проблемы. Они ищут знания

потому, что они имеют исключительную ценность для них. Приобретенное знание позволит найти нужное решение проблемы, повысит эффективность деятельности специалистов, группы, команды, менеджеров того или иного уровня.

Продавцы знаний – это, как правило, специалисты (эксперты) с высокой репутацией на внутреннем рынке компании, обладающие значительными ресурсами знаний по определенным проблемам и темам.

Среди них есть те, кто имеют квалификацию, однако не способны четко выразить свои знания в явной, товарной форме. Другие имеют глубокие, но очень узкоспециализированные знания, ограниченной сферы применения, что существенно снижает их стоимость на рынке знаний. Есть и такие, кто полагает остаться в выигрыше, если оставит свои неявные (скрытые) знания при себе. Последняя точка зрения не лишена в отдельных случаях основания. Если знания – это сила, то сила владельца знаний может иссякнуть, если другие специалисты овладеют этим знанием. Естественно, это не отвечает его интересам.

Одной из стратегий разумного рыночного поведения в отношении продавцов знаний является гарантирование того, что продажа знаний вознаграждается более, чем их скрытое накопление.

Брокеры знаний выполняют роль связующего звена между теми, кто нуждается в знаниях, и теми, кто их имеет, то есть между покупателями и продавцами.

Им доставляет удовольствие исследовать организации, выяснять компетенцию и специализацию сотрудников, их предметную область и уровень знаний. То есть они обладают информацией, к кому и по какой проблематике можно обратиться в поиске недостающих знаний.

Некоторые неформальные брокеры знаний являются, по сути, антрепренерами по знаниям. Они намеренно становятся экспертами по дислокации знаний в организации. И очень часто они «продают» эти знания и опыт не за деньги, а расположение и репутацию, обслуживая тем самым внутренний рынок знаний компании.

Естественными брокерами являются библиотекари, если библиотека и/или электронная библиотека обслуживает организацию в целом. Они аккумулируют статистику спроса на знания сотрудников и подразделений компании, владеют традиционными и компьютерными методами поиска новых знаний. Они располагают информацией о доступности знаний – мерой времени и усилий, которые покупатель должен будет затратить для получения искомого знания.

Что является ценовой системой рынка знаний? Какой валютой обмениваются участники знаниевых транзакций?

Когда компании покупают знания у внешней организации, они чаще всего расплачиваются деньгами. Примерами могут служить заказные НИР, приобретение лицензий, оплата консалтинговых услуг, экспертиз и т.п.

Внутри компаний средством обмена редко являются деньги, но существуют обусловленные валюты («сущности» – в терминах теории обмена), которые используются рынком знаний. К такого рода сущностям относятся взаимность, репутация и альтруизм.

Взаимность (взаимный обмен) зиждется на том, что продавец будет тратить время и усилия для обмена знаниями, если он ожидает, что покупатель будет столь же старательным продавцом, когда он обратится к нему за его знаниями.

Время, энергия и знания у конкретного сотрудника конечны. Это очень дефицитные ресурсы. И они расходуются в том и только в том случае, если расходы приносят или принесут прибыль. Кроме того, продавец осознает, что информация о его доброй воле в обмене знаниями может положительно повлиять на других сотрудников компании, побудит их охотнее делиться знаниями. И это его также вдохновляет.

Репутация хоть и является слабо осязаемой категорией, очень часто может и дает осязаемые результаты. Наличие у сотрудника репутации знающего человека, готового делиться знаниями, создает ему такие преимущества, как надежность рабочего места, ускоренное продвижение по службе, более частые поощрения, приобретение неформального статуса гуру в компании. Хотя продавец напрямую не получает денег, но опосредованно, через премии и более высокую зарплату, через бонусы, есть и денежная компенсация со стороны компании. Действенность репутации как обменной сущности в значительной мере зависит от корпоративной культуры, точнее, насколько компания культивирует знание как конкурентное преимущество и поддерживает стремление сотрудников к обмену знаниями.

Многие владельцы знаний частично мотивируются к обмену любовью к теме и до некоторой степени **альтруизмом**, стремлением содействовать «процветанию фирмы» или естественным желанием помочь ближнему. Ведь всегда есть люди, которым нравится помогать другим.

Наставничество является распространенной формой передачи знаний, частично основанной на альтруизме. Формальное признание на-

ставнических связей, выделение знающим сотрудникам времени на передачу знаний и само понимание, что опыт сотрудников – это ценнейший знаниевый ресурс, – все это формы поощрения альтруизма. При всем при этом вряд ли следует переоценивать роль альтруизма, полагаться только на добрую волю при культивировании в компании обмена знаниями.

Важным фактором активного использования упомянутых сущностей (взаимности, репутации, альтруизма) на рынке знаний является обстановка доверия в компании. Без обстановки доверия, весьма важного во многих отношениях элемента корпоративной культуры, инициативы, связанные со знанием, будут обречены на неудачу вне зависимости от уровня их организационной и технологической поддержки.

Чтобы рынок знаний действовал в организации:

1. *Доверие должно быть видимым.* Не достаточно декларации о важности доверия в миссии компании. Должны быть прямые и очевидные признаки доверия, что продавцы знаний имеют высокую репутацию, что сотрудники и подразделения им отвечают взаимностью.

2. *Доверие должно быть повсеместным.* Если часть внутреннего рынка знаний компании не заслуживает доверия, то рынок сжимается, становится асимметричным и, следовательно, менее эффективным.

3. *Доверие должно начинаться с верхов.* Образцы поведения руководства верхнего уровня часто определяют нормы и ценности компании. Если топ-менеджеры компании заслуживают доверия, то доверие будет просачиваться и олицетворять компанию в целом. Если же они цинично будут эксплуатировать знания других в личных целях, то недоверие будет распространяться по всей организации.

Даже сделки, оформленные письменными контрактами, требуют определенной степени доверия. Внутренний рынок знаний компании, без письменных контрактов и возможности судебных исков, по сути, базируется на доверии. Если его нет, то не может быть и рынка знаний.

Глава 3

УПРАВЛЕНИЕ ПРОЦЕССОМ РАБОТЫ СО ЗНАНИЯМИ

Как отмечалось в предыдущей главе, работа со знанием – это достаточно сложный многоэтапный процесс, имеющий собственный жизненный цикл. Как и любой процесс, и тем более сложный, он нуждается в осознанном управлении для обеспечения должной последовательности, согласованности, эффективности.

3.1. Подходы к управлению знаниями

Управление знанием (УЗ) – это, по сути, управление процессами, связанными со знанием, или управление процессами работы со знанием. Существует два сформировавшихся подхода к управлению знаниями, существенно отличающихся друг от друга [McElroy M.W., 2003].

Первый подход (подход первого поколения УЗ) исходит из того, что ценные знания в организации уже существуют, то есть они уже созданы и все, что требуется сделать, – это должным образом зафиксировать их, закодировать и распространить. В соответствии с этой системой взглядов, деятельность по УЗ начинается после того, как знание произведено. Следовательно, целью УЗ в этом случае является не совершенствование процесса производства (создания) знаний, а лишь его использование (применение в практической деятельности).

Сторонники второго подхода (второго поколения УЗ) придерживаются мнения, что знания не существуют в готовом виде. Фактически, они непрерывно производятся в процессе обработки знаний (knowledge processing). Использование знаний в бизнес-процессах проверяет знания и выявляет новые проблемы, формируя спрос на новые знания и его удовлетворение.

Взаимосвязь сред обработки знаний и выполнения бизнес-процессов, в соответствии со вторым подходом, показана на рис. 3.1 [McElroy M.W., 2003].

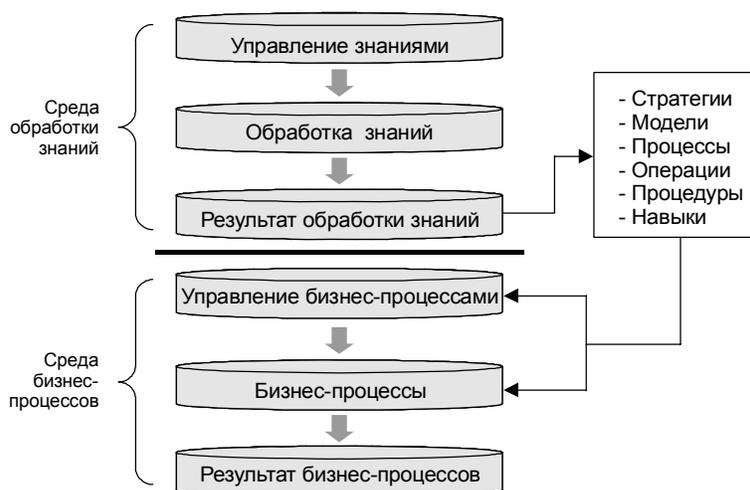


Рис. 3.1. Связь между процессами работы со знаниями и бизнес-процессами

Для дальнейшего уяснения существа различий первого и второго подходов дадим некоторые пояснения. В подходе первого поколения не рассматривается ЖЦЗ, нет системного рассмотрения процесса работы со знанием как сложного внутрикорпоративного социального процесса. Основное внимание уделялось тому, как лучше зафиксировать, закодировать и распространить организационные знания, в том числе путем максимизации перевода неявных знаний в явные. По сути, речь в нем идет об эффективной системе доставки имеющихся знаний потребителю (лицу, группе лиц), то есть об их оперативном информировании относительно существующих в организации знаний. На самом деле, это мало чем отличается от традиционных действий по управлению информационными ресурсами компании.

Управление знаниями отличается от управления информацией за счет наличия метаинформации, знаниевых утверждений и утверждений об утверждениях (метаутверждений). Именно метаутверждения (оценки

достоверности, история применения, обобщения) могут дать пользователю доступ к аргументам и доказательствам, которые стоят или не стоят за этими утверждениями. В обеспечении таких возможностей и состоит основная цель подхода второго поколения в УЗ.

Комментарии к недостаткам или, точнее, к неполноте подхода первого поколения отнюдь не означают, что обработка информации и управление информацией не играют позитивной роли в обработке знаний. Эффективный доступ к информационным ресурсам – важный элемент в работе со знаниями. Но не следует путать обработку информации и обработку знаний друг с другом, и не нужно считать, что задача менеджера по знаниям заключается только в облегчении доступа потребителей к информации за счет использования новейших информационных технологий.

Таким образом, можно констатировать, что в каждой компании есть бизнес-процессы и есть процессы работы со знанием, взаимосвязанные между собой, как это схематично показано на рис. 3.2.

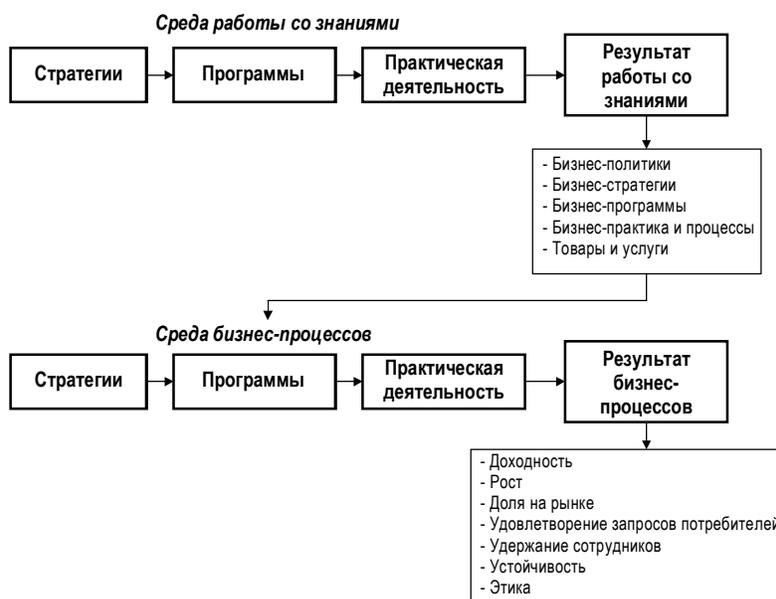


Рис. 3.2. Бизнес-процессы и процессы работы со знанием

Есть сторона, запрашивающая знания (demand-side), и есть сторона, предлагающая (поставляющая) знания (supply-side). Обеспечение баланса спроса и предложения (demand and supply) является своего рода критерием эффективности социального процесса управления знаниями в конкретной компании, обобщенная схема которого приведена на рис. 3.3.



Рис. 3.3. Жизненный цикл обработки знания

Исходя из перечисленных целевых установок и критерия эффективности, в компании необходимо обеспечивать управление работой со знаниями, так же, как необходимо обеспечивать управление производством.

На рис. 3.4 приводится фрагмент реального бизнес-процесса, иллюстрирующего использование знаний (явных и неявных) при разработке проектно-технологической документации (ПТД) для нефтегазового месторождения (НГМ).

Разработка и эксплуатация НГМ – сложный и длительный процесс, охватывающий временной горизонт в несколько десятков лет. Началу разработки НГМ предшествует весьма трудоемкий процесс разработки ПТД, ее экспертизы и утверждения в установленном государством порядке. Эту работу выполняют специализированные проектно-технологические институты, укомплектованные квалифицированными специалистами (геологами, нефтяными инженерами, математиками и др.) с использованием мощных компьютеров и программных систем ведущих мировых производителей (Schlumberge, Landmark).

Основу ПТД составляют цифровые трехмерные (3D) геологические и гидродинамические модели НГМ. Для их построения используются различные данные о нефтегазовом пласте, в том числе результаты гидродинамических исследований скважин (ГДИС).

Априорные данные по ГДИС, как это показано на рис. 3.4, находятся в базе исходных данных (БД для ПТД НГМ). Ведущий аналитик отдела

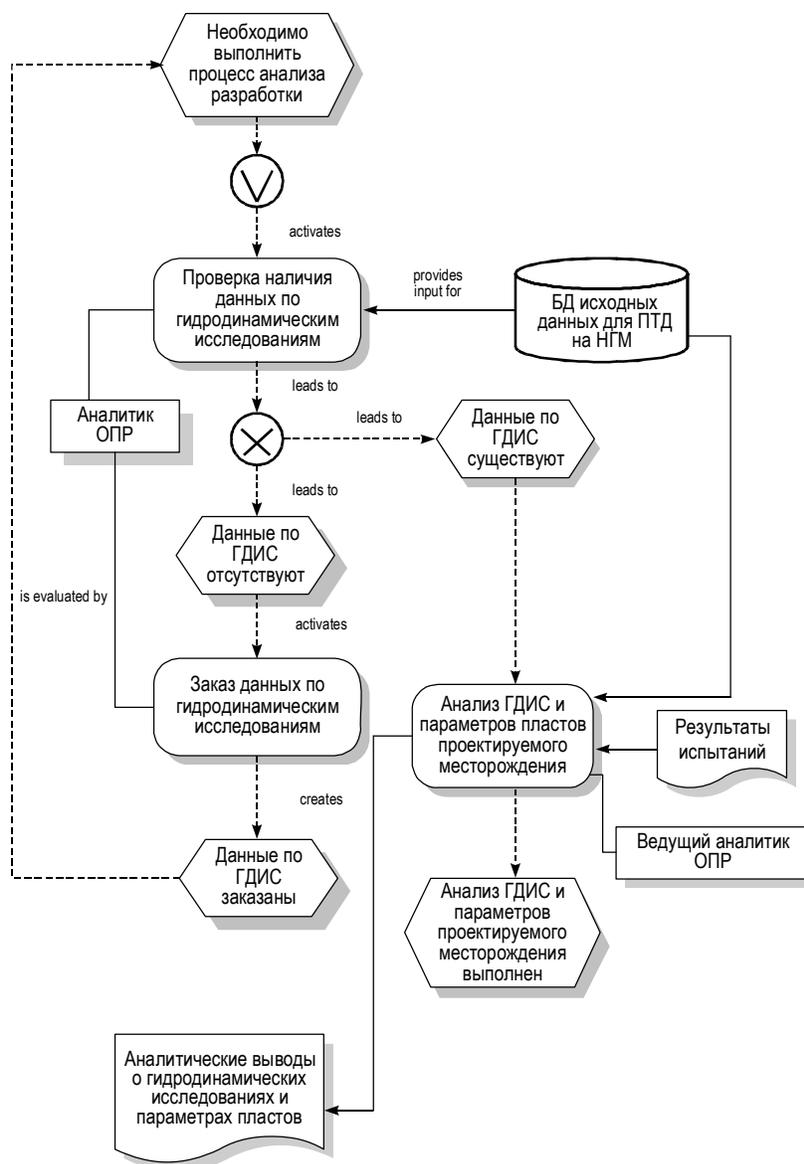


Рис. 3.4. Пример использования знаний в бизнес-процессе разработки проекта нефтегазового месторождения

проектирования разработки (ОПР) удостоверяется в наличии данных по ГДИС для определенной скважины и, если они имеются в наличии, осуществляет анализ ГДИС и соответствующих параметров проектируемого месторождения. Если данные по конкретной скважине отсутствуют, аналитик отдела ОПР оформляет заказ на проведение ГДИС данной скважины (см. рис. 3.4) сервисной организации, владеющей соответствующими методиками, приборами и технологией. Полученные данные по ГДИС, обработанные на ЭВМ, вместе с результатами испытаний скважин поступают ведущему аналитику ОПР, который и формирует аналитические выводы о параметрах пластов НГМ.

Подпроцесс анализа гидродинамических исследований скважин и параметров пластов в большом процессе разработки ПТД расписан на рис. 3.4 столь детально потому, что к нему в процессе составления ПТД обращаются многократно. Построение геологической и гидродинамической модели НГМ, как правило, осуществляется по множеству скважин. Параметры каждой из них активно участвуют в итерационном процессе адаптации и верификации упомянутых моделей. Конечный результат, фиксируемый в ПТД, является, таким образом, продуктом итеративного использования данных, информации и знаний, а также их генерации и проверки в ходе анализа различных вариантов.

Исходя из основных положений второго поколения УЗ, в настоящей главе будет рассмотрена обобщенная модель создания нового корпоративного знания, а также ряд организационных механизмов УЗ, таких, как: команды, создающие знания, сообщества по интересам, эксперты и их роль в процессе принятия решений, система обучения персонала, а также корпоративная культура как важнейший фактор содействия эффективному управлению знаниями.

3.2. Модель процесса создания нового знания организацией

Из перечисленных этапов (процессов) ЖЦЗ (см. рис. 2.2) процесс создания знания организацией является, вероятно, наиболее сложным и потому заслуживает отдельного, более обстоятельного обсуждения. На рис. 3.5 представлена пятифазная модель процесса создания знания организацией, предложенная в [Нонака И., Такеучи Х., 2003].

Процесс создания организационного (корпоративного) знания начинается с распространения неявных знаний (фаза 1), что в общих чертах

напоминает *обобществление*, поскольку концентрированное и пока неиспользуемое индивидуальное знание должно, прежде всего, распространяться в пределах организации. Во второй фазе распространенное неявное знание преобразуется некоторой командой (стихийно или осознанно организованной) в виде новой концепции в явное знание. Процесс этот соответствует *отчуждению* знаний. В третьей фазе созданная концепция подвергается проверке, в ходе которой организация определяет, действительно ли предложенная концепция имеет право на существование. Получившая «путевку в жизнь» концепция в четвертой фазе преобразуется в архетип, который может принять форму прототипа при разработке некоторого материального продукта или организационного элемента (структуры, процесса) в случае, если инновация носила нематериальный характер. Последняя фаза создания знания предусматривает распространение знания по некоторому отделу, по разным отделам или среди дочерних компаний, потребителей, дистрибьюторов и даже в университетах. Тем самым осуществляется переход знания на новый уровень, и в том нет ничего удивительного, ибо создающая знания компания – это открытая система, постоянно обменивающаяся знаниями с актуальной средой.

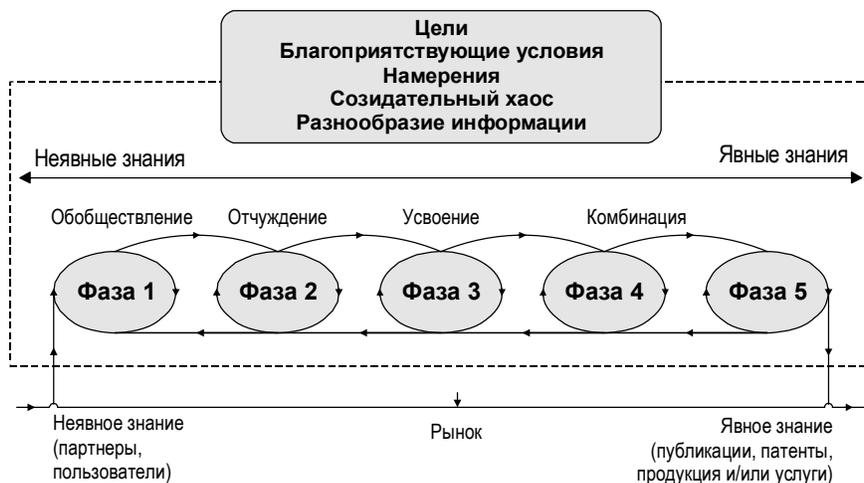


Рис. 3.5. Пятифазная модель процесса создания знания организацией: фаза 1 – распространение неявного знания; фаза 2 – создание концепции; фаза 3 – проверка концепции; фаза 4 – построение архетипа; фаза 5 – переход знания на новый уровень

Поскольку процесс создания нового знания сложен и, отчасти, таинствен, имеет смысл подробнее остановиться на описании каждой из упомянутых фаз.

Распространение неявного опыта (фаза 1) начинается со сбора неявных знаний, являющихся в большинстве организаций слабо задействованным, хотя и богатым источником нового знания. Неявные знания весьма сложно распространить или передать, поскольку оно, в основном, получено из опыта и с трудом поддается даже вербализации.

Дело также не в том, что получателями неявных знаний являются множество индивидуумов с различным образованием и предшествующим опытом, отличающихся точками зрения и мотивацией. Для достижения успеха в *обобществлении* знаний эмоции и интеллектуальные модели индивидуумов следует сделать общими, то есть создать некоторое общее «поле», на котором индивидуумы смогут взаимодействовать в благоприятной среде, дискутировать, обмениваться опытом, синхронизировать свою физическую и интеллектуальную активность. Типичное общее поле взаимодействия – команда, члены которой собраны, а лучше сами пришли из различных отделов, но склонны работать сообща и стремятся к единой, увлекающей их цели.

Самоорганизующаяся команда создает новое организационное знание благодаря разнообразию своего состава, вовлекая избыточную информацию и разнообразие видения цели и средств ее достижения. Руководство организации преднамеренно создает созидательный хаос постановкой труднодостижимых целей, представляя каждому члену команды значительную самостоятельность. Команда сама определяет целевые горизонты и границы задач и, как особый отдел с широкими полномочиями, начинает взаимодействовать с актуальной средой, накапливая неформализованное и формализованное знание.

Создание концепции (фаза 2) осуществляется в режиме наиболее интенсивного взаимодействия неявных и явных знаний. Как только в поле взаимодействия выкристаллизовывается некая общая интеллектуальная модель, команда приступает к ее формализации в процессе постоянного диалога, индивидуальных и коллективных размышлений. Эта фаза превращения неформализованной интеллектуальной модели в формализованную более всего соотносится с этапом *отчуждения* знаний. В этом процессе используются разнообразные методы мышления, такие, как дедукция, индукция и абдукция. Особая роль в данной фазе принадлежит абдукции, использующей язык метафоры и аналогий. Коэффициент

полезного действия от взаимодействия членов команды повышается при использовании диалектики, ибо создание знаний – это идущий по спирали процесс синтеза, вытекающий из дискуссий, противоречий и парадоксов.

Автономия, а она является весьма ценным активом на этом этапе, позволяет членам команды свободно обмениваться мыслями с целью выявления направления интеллектуальной деятельности. Для создания концепции разработчикам порой приходится в корне пересматривать свои первоначальные представления. В этом им помогает разнообразие информации и множественность точек зрения членов команды. Встряски и хаос внешнего и внутреннего происхождения также активно воздействуют на изменение образа мыслей, что также способствует творчеству.

Проверка концепции (фаза 3) является важной фазой в создании знания, ибо новое организационное знание должно быть *обоснованным истинным убеждением*, имеющим определенную ценность как для организации, так и для общества. Специалисты проверяют информацию, концепции или знания постоянно и даже бессознательно по мере их получения или в процессе их создания. Однако организация должна осуществлять проверку более строго и всесторонне для установления соответствия концепции своим намерениям и целям.

Для бизнес-организации общепринятыми критериями достоверности концепции являются затраты на ее реализацию, прибыль и вклад, вносимый результатами в развитие фирмы. Критерии оценки в зависимости от типов концепции могут быть количественными, качественными либо смешанными.

Определение системы критериев и желательных диапазонов их значения является прерогативой руководства компании. При этом оно должно учитывать не только корпоративные интересы и ценности, но и систему ценностей профессионального и регионального сообщества, общества в целом, его запросы и т.п.

Построение архетипа (фаза 4), которым может быть прототип новой продукции или технология предоставления новой услуги, осуществляется путем сочетания только что созданного нового формального знания с уже существующим формальным знанием, то есть этой фазе можно поставить в соответствие процесс *комбинирования* знаний.

Для построения, например, прототипа вместе собираются специалисты с различными знаниями и опытом (из отделов НИОКР, маркетинга, производства, технического контроля и др.) и совместно добиваются

конкурентных характеристик нового продукта. Аналогично происходит и с моделью новой оргструктуры или технологией оказания услуг.

При извлечении знаний в процессе межличностного сотрудничества полезно использовать коммуникативные методы, а именно:

- пассивные (наблюдения, протоколы «мыслей вслух», лекции);
- активные (игры, диалоги, круглые столы);
- групповые («мозговой штурм», ролевые игры);
- индивидуальные (анкетирование, интервью, экспертиза).

Поскольку настоящая фаза носит весьма комплексный характер, в ней необходимо обеспечить активное сотрудничество различных отделов и служб организации. Должна быть также обеспечена избыточность информации, разнообразие опыта, технологических инноваций, стимулирование межличностного и межструктурного сотрудничества.

Переход знания на новый уровень (фаза 5) – следствие того очевидного факта, что создание нового знания – процесс бесконечный и самовоспроизводящийся. Новая концепция создается, выверяется, моделируется и затем восходит к новому циклу создания знаний, который находится в иной онтологической плоскости. Этот интерактивный, развивающийся по спирали процесс происходит как в пределах одной организации, так и на межорганизационном уровне. На данном этапе имеет место процесс *усвоения* знаний.

Реализованное или принявшее форму архетипа внутриорганизационное знание способно инициировать новый цикл создания знания, распространяясь в организации как по горизонтали, так и по вертикали. На межорганизационном уровне знание, созданное организацией, способно посредством активного взаимодействия мобилизовать знания в других дочерних компаниях, в клиентской среде, в среде поставщиков и конкурентов и в других компонентах среды. Например, введение нового способа бюджетирования в одной компании неизменно приводит к изменениям в бюджетировании дочерних компаний, а это, в свою очередь, может вызвать новый цикл инноваций. Положительный опыт рано или поздно станет достоянием других фирм, в том числе конкурентов.

Для эффективности процесса в этой фазе особенно важна автономность структурных единиц и отделов, позволяющая им абсорбировать и применять знания внешнего происхождения. Избыточность информации, ротация персонала здесь также весьма желательны. Все это может способствовать перекрестному обогащению знаниями в компании.

3.3. Команда, создающая знания

Создание нового корпоративного знания требует участия сотрудников, играющих разные роли и занимающих разные позиции в иерархической лестнице компании. Вне зависимости от принадлежности сотрудников к той или иной категории, ценность их вклада определяется главным образом важностью представляемой информации, их способностями и навыками в работе со знанием и в командах.

Следуя рекомендациям И. Нонака, Х. Такеучи [Нонака И., Такеучи Х., 2003], команда, создающая знание, должна включать в себя практиков, организаторов и идеологов знания.

Практиками знания чаще всего являются рядовые сотрудники и менеджеры низших звеньев. *Организаторами* выступают менеджеры средних звеньев, а *идеологами* – высшее руководство.

Основная обязанность практиков знания – воплощение знания. Они накапливают, генерируют и обновляют как неявное, так и явное знание, выступая в роли ходячих архивов. Поскольку большинство из них рядовые сотрудники, находящиеся в постоянном контакте с внешней средой, они обладают новейшей информацией о состоянии рынка и маркетинговых стратегиях конкурентов, о новых технологиях, продуктах и услугах и т.п. Поставляя эту актуальную информацию и знания в команду, они в то же время нуждаются в целеуказании и постановке задачи со стороны идеологов и организаторов знания.

Практики знания подразделяются на две взаимодополняющие группы: операторы и специалисты. *Операторы* знания собирают, комбинируют и генерируют обширное неявное знание в виде навыков, основанных на опыте. В эту группу входят сотрудники торговых и сбытовых организаций компании, квалифицированные рабочие и мастера, накапливающие неявное знание посредством непосредственного контакта со средой и личного опыта. *Специалисты* знания также накапливают, генерируют и обновляют знания. Но в отличие от операторов, они мобилизуют хорошо структурированное явное знание в виде технических, научных и других источников, то есть знание, готовое к вводу в компьютер. К группе специалистов знания относятся ученые из отделов НИ-ОКР и заводских лабораторий, конструкторы, программисты, сотрудники отделов кадровой политики, маркетинга, финансовых и юридических служб.

В идеале практик знания должен обладать следующими качествами:

- высокими интеллектуальными показателями;
- выраженной убежденностью и волей по преобразованию окружающего мира;
- разнообразным опытом;
- навыками собеседника, способного вести диалог с коллегой и потребителем;
- открытостью в дискуссиях.

Организаторы знания играют роль посредника между умозрительными идеалами высшего руководства и зачастую хаотичными воззрениями рядовых сотрудников. По сути, они перебрасывают мост между тем «что есть» и тем «что должно быть».

Организаторы знания изменяют реальность, трансформируя знание. Используя разные способы трансформации, они уделяют основное внимание преобразованию неформализованных образов и представлений в формализованные, воплощая их в новые действия, технологии, продукты, услуги.

Организаторы знания – менеджеры среднего звена – должны обладать следующими способностями:

- координировать работу и управлять проектами;
- работать с гипотезами с целью создания новых концепций;
- интегрировать различные методы создания знания;
- владеть навыками делового общения для организации бесед членов команды;
- завоевывать доверие членов команды;
- предвидеть направление действий на основе накопленного опыта.

Идеологи знаний, как правило, занимающие в компании высшие руководящие должности, осуществляют общее руководство процессом создания корпоративных знаний. Они участвуют в создании знаний путем формулировки базисной концепции развития компании, определяют видение или декларируют политику компании, устанавливают критерии оценки создаваемого знания.

Если практики знания должны знать «что есть», то идеологи знания должны знать «что должно быть». Важной обязанностью идеологов знания является определение и систематизация видения системы ценностей компании. Именно система ценностей во многом определяет корпоративную культуру и поведение сотрудников, их восприятие идеалов высшего руководства и вовлеченность в процесс генерации идей.

Идеологи знания, в идеале, должны обладать следующими качествами:

- способностью формулировать видение компании, определяя тем самым спрос на те или иные знания;
- способностью донести ценности и видение компании до создателей продукции и услуг;
- способностью оценить созданное знание со стратегических позиций;
- талантом выявления потенциальных руководителей проектов;
- умением создать атмосферу «созидательного хаоса» в команде;
- способностью направлять и координировать процесс создания знания.

Эффективность деятельности команд зависит от ряда факторов: технических, организационных, культурных, а также, в значительной мере, от личностей, входящих в команду, от их технических знаний, опыта, способностей эффективно и продуктивно взаимодействовать друг с другом.

3.4. Сообщества по интересам

В организациях, где уделяется большое внимание обмену неявными знаниями, поддерживается создание различных сообществ, которые являются также механизмом согласования взглядов и интересов индивидуальностей [European Guide to good Practice in Knowledge Management, Part 2].

К их числу относятся сообщества по практическим интересам (Communities of Practice – CoPs), которые именуют также интеллектуальными сообществами (knowledge communities), сетями знаний (knowledge networks), обучающимися сообществами (learning communities), сообществами по интересам и тематическими группами.

Сообщества состоят из группы сотрудников, имеющих различные знания и умения, историю исследований и разработок, базовый опыт. Сообщества работают совместно для достижения совместных целей.

Эти группы отличаются от команд и групп специалистов для решения конкретных задач (task forces). Участники CoPs могут выполнять одну и ту же работу, сотрудничать над совместной задачей в процессе решения проблемы или работать вместе над каким-то продуктом. Они являются равными в выполнении «реальной работы». Совместно их удерживает общее понимание цели, реальная потребность знать то, что

знают другие, а также желание совместно расширить знание и понимание проблем.

Обычно в одной компании существует много сообществ по практическим интересам, причем сотрудники могут участвовать более чем в одном сообществе. В таких сообществах индивидуальность не только не подавляется, но признается и даже поощряется, обеспечивая тем самым продвижение знаний через функциональные и структурные границы подразделений.

Существуют три основных типа сообществ:

- *Сообщества по интересам*, то есть сообщества по конкретной теме.
- *Сообщества по практической деятельности* (СПД), объединяющие людей по конкретной сфере деятельности.
- *Сообщества по целям*, создающиеся на ограниченные временные горизонты, до момента достижения объявленной цели. К последним относятся проектные группы, экспертные комиссии и т.п.

Многие сообщества развиваются естественно, начиная с группы энтузиастов. Там, где организация решила поощрять и легализовать сообщества, необходимо также разработать механизм или процедуру учета их мнений. Единственно, что плохо приживается, это когда сообщество создается по указке сверху.

Отдавая должное разнообразным сообществам в сохранении и передаче знаний, необходимо иметь в виду, что рабочие группы и сообщества порождают феномен и культуру так называемого «группового мышления». Сильные групповые убеждения и чрезмерное групповое давление может ограничивать свободу творчества у индивидуума. К числу недостатков группового мышления можно отнести иллюзию неуязвимости, («если мы едины, мы не победимы»), иллюзию моральности (правильно то, что соответствует морали группы), групповые стереотипы и укоренившиеся мнения (игнорирование источников информации и знаний, противоречащих взглядам группы), иллюзию единодушия, самоцензуру и т.п.

Сообщества существуют как в больших, так и в малых организациях. Повышение эффективности использования как формальных сообществ (функциональных и целевых групп, проектных команд), так и неформальных (групп по интересам, групп по общности целей, групп курьерщиков, завсегдаев кофе-брейков и т.п.), как правило, содействует успеху в обмене и производстве знаний. То есть наличие в таких груп-

пах физически реализованных и виртуальных рабочих мест увеличивает возможность общения, способствует раскрепощению и обмену мнениями индивидуумов.

Содействие со стороны организации в становлении и функционировании конкретного сообщества может выражаться:

- в выявлении или выборе координатора;
- в создании инфраструктуры сообщества (например, информационной среды взаимодействия: E-mail, интранет, проектная зона);
- в привлечении новых членов путем распространения информации об интересах, приоритетах и намечающихся действиях;
- в согласовании усилий актива, координатора и спонсора сообщества и их взаимодействия с его членами;
- в оценке и признании результатов, праздновании и оповещении об успехах как внутри, так и вне общества и т.п.

Процесс создания сообщества не должен быть поспешным. Важно, чтобы в этом процессе зародилась саморегулирующие механизмы, которые впоследствии и обеспечат его живучесть. Общими стадиями, через которые проходят сообщества, являются:

- восхищение новизной;
- неразбериха с целью;
- прояснение интересов и направлений;
- рост доверия и уважения;
- создание устойчивого сообщества.

Независимо от степени достижения цели создания сообщества неформальный характер отношений между его членами должен сохраниться как непреходящая ценность, обеспечивающая обмен и создание нового знания через функциональные и структурные границы подразделений.

Сообщества в процессе своей деятельности используют ряд общеизвестных методов управления знаниями. Речь, прежде всего, идет о таких методах работы с неявными знаниями, как «мозговой штурм» (групповая генерация идей, путей реализации идей, выбор способа решения задачи), метод непосредственного общения, метод перевода проблемы в задачу и др.

Мозговой штурм (Brainstorming) – это процесс, включающий группу людей, которые встречаются для рассмотрения проблемы и затем намеренно предлагают так много необычных решений, насколько это возможно. Участники высказывают идеи, как только они возникают у них,

затем развивают идеи, высказанные другими. Все идеи фиксируются и не критикуются. Только когда сеанс мозгового штурма завершается, идеи оцениваются. Мозговой штурм помогает в решении проблем и создании новых знаний из имеющихся. Для успешного проведения мозгового штурма полезны следующие рекомендации:

- Лидер должен контролировать сеанс и поддерживать его проведение.
- В начале должна быть определена проблема, которую нужно решить, и критерии, которым должно соответствовать полученное решение.
- Лидер должен поощрять энтузиазм, не критическое отношение между участниками штурма, стимулировать и поощрять активное участие всех членов команды. Сеанс продолжается в течение фиксированного времени, и лидер должен следить, чтобы мыслительная деятельность участников продолжалась не слишком долго. Лидер должен стараться удержать мозговой штурм на заданной теме и направлять его на выработку некоторых практически значимых решений.
- Желательно, чтобы участники процесса мозгового штурма были бы представителями различных областей знаний, обладающими разным опытом работы. Разнообразие их априорных знаний и опыта способствует зарождению большего количества творческих идей в течение сеанса мозгового штурма.
- Участники мозгового штурма должны быть раскрепощены, пребывать в хорошем и даже веселом настроении, что также способствует зарождению новых идей как сугубо практических, так и совершенно непрактичных.
- Идеи не должны критиковаться или оцениваться в ходе сеанса мозгового штурма. Критика тормозит генерацию идей, вводит элемент риска для членов группы в высказывании своих идей. Все это сдерживает творчество и мешает непринужденному ходу сеанса мозгового штурма.
- Участники мозгового штурма должны не только предлагать свои идеи, но также должны поддерживать чужие, «воспламеняться» от ассоциаций с идеями других участников и развивать их.
- Ход сеанса должен записываться либо в виде отдельных существенных заметок, либо полностью на магнитофон (диктофон). Эта запись в дальнейшем должна анализироваться, а предложенные

идеи оцениваться. В ходе мозгового штурма полезно записывать и исследовать идеи на доске, с тем чтобы они были видны всем участникам сеанса.

Непосредственное общение (Face-to-Face Interaction) является традиционным, широко используемым методом передачи неявных знаний (socialisation), которыми владеют сотрудники организации. Такое общение является неформальным и весьма плодотворным. Личное взаимодействие способствует увеличению коллективной памяти организации, развивает доверие и поощряет сотрудников к обмену знаниями. Общение усиливает в организации социальные связи, повышает коллективное понимание цели и смысла решаемых задач и проблем.

Непосредственное общение приводит к появлению консенсуса в том, что является верным знанием, и к созданию (выявлению) новых знаний. Оно создает в организации среду, в которой участники видят фирму как сообщество людей, способных разносторонне анализировать информацию (т.е. создавать знания).

Постпроектный анализ (Post-Project Reviews) – это сессии, собрания, семинары с разбором результатов выполненной работы. Они используются для выделения приобретенного опыта (уроков) в ходе выполнения проекта в процессе деятельности организации. Такие критические анализы и обзоры способствуют фиксации знаний, причин успехов и неудач, лучших практических решений. Постпроектный анализ повышает эффективность обучения сотрудников, создает возможность повторного использования знаний в последующих проектах.

Для эффективного использования этого метода должно выделяться специальное время для сессий, собраний, семинаров. Важно также, чтобы постпроектный анализ осуществлялся сразу после завершения проекта, когда все плюсы и минусы свежи в памяти, а участники событий и работ не отошли от дел и не потеряли к ним интерес.

3.5. Эксперты, экспертиза, консультации

В успешно функционирующей организации имеется значительная доля кадров, обладающих теоретическими знаниями и практическим опытом. Все они относятся к категории квалифицированных специалистов, обладают определенным формальным статусом в иерархии управления бизнесом, участвуют в выработке стратегии развития компании, в анализе проблемных ситуаций, в обосновании и принятии решений.

Естественно, что как участники соответствующих бизнес-процессов они вовлечены в той или иной степени в процесс поиска и накопления знаний, в процесс обмена знаниями, в генерацию новых идей и т.п. Именно в той или иной степени, ибо вовлечение в знаниевые процессы определяется не столько формальным статусом специалиста, сколько тем, насколько организация и ее сотрудники признают энциклопедичность знаний и опыта определенного специалиста, широту кругозора, глубину теоретических познаний и культуру его мышления. Именно таких специалистов чаще других привлекают к анализу проблемных ситуаций, к разработке стратегий и оценке альтернатив, к обобщению и распространению передового опыта, и именно таких специалистов принято относить к категории экспертов.

Итак, эксперт – *это квалифицированный специалист, выработавший в процессе научного и/или практического опыта определенные знания и суждения об оцениваемых объектах (явлениях, процессах) и руководствующийся ими в практической деятельности.*

Понятно, что количество экспертов в организации существенно меньше общего числа квалифицированных специалистов. Их профили компетентности должны быть описаны и представлены в базах знаний и должны быть доступны поисковым системам. Весьма желательно, чтобы выделение эксперта из общего числа квалифицированных специалистов осуществлялось на основе некоторой процедуры и сопровождалось присвоением ему определенного статуса. Для того чтобы эксперт делился своими знаниями в процессе экспертиз и/или консультаций, его интеллектуальные усилия должны морально и материально вознаграждаться. Статус эксперта должен учитываться как весомый фактор управлением кадровой политики в карьерном росте сотрудников и кандидатур на замещение вакансий в руководящих органах компании.

Одним из основных способов эффективного использования экспертов – привлечение их в качестве консультантов при решении сложных проблем, с которыми сталкиваются сотрудники компании.

Эксперты могут включаться в состав команд, создающих знания. Эксперты могут участвовать в мозговых штурмах, где их главное предназначение – оценка предложенных идей, стратегий, альтернатив после того, как сеанс мозгового штурма завершен. Участие экспертов весьма полезно в дискуссиях (очных, виртуальных) на заданную тему, постпроектном анализе, в обобщении полученных здесь знаний и опыта.

Эксперты могут вносить весомую лепту в разделы базы знаний, посвященные лучшим практическим решениям.

Весьма полезным направлением использования экспертов является консультирование. Оно может быть индивидуальным и групповым, очным и заочным с использованием современных телекоммуникационных технологий.

Консультации – это один из стандартных видов занятий в системе образования и повышения квалификации. Проходят они, как правило, в форме беседы консультанта с учащимися или слушателями курсов и имеют целью расширение и углубление их знаний. Консультации широко используются в высших и средних учебных заведениях, особенно в заочных и вечерних.

Консультации проводятся и при подготовке к семинарам, коллоквиумам, курсовым и государственным экзаменам, по вопросам учебной и производственной практики, по курсовому и дипломному проектированию (курсовым и дипломным работам), по самостоятельно разрабатываемым студентами или слушателями проблемным и научным темам.

Выявление экспертов в компании является одной из наиболее сложных проблем. Очевидно, в качестве экспертов необходимо использовать тех людей, чьи суждения и опыт наиболее помогут принятию адекватного решения. Но как выделить, найти, подобрать таких людей? Нужно прямо сказать, что не существует методов подбора экспертов, гарантированно обеспечивающих успех экспертизы или консультации пользователя.

Часто предлагают использовать методы взаимооценки и самооценки компетентности экспертов. С одной стороны, кто лучше может знать возможности эксперта, чем он сам? С другой стороны, при самооценке компетентности нельзя исключить, что оценивается скорее степень самоуверенности эксперта, чем его реальная компетентность.

При использовании метода взаимооценки, помимо возможности проявления личностных и групповых симпатий и антипатий, играет роль неосведомленность экспертов о способностях друг друга. В современных условиях достаточно хорошее знакомство с работами и способностями друг друга бывает лишь у специалистов, много лет работающих совместно. Однако и здесь таится некоторая опасность, ибо взаимооценка осуществляется в таком случае на основе сильно схожих точек зрения, знаний и опыта.

Использование формальных показателей (должность, ученые степень и звание, стаж, число публикаций и т.д.) может иметь место, но

должно, видимо, носить вспомогательный характер при выделении экспертов из числа сотрудников компании.

Успешность участия в предыдущих экспертизах – хороший критерий для выбора человека в качестве эксперта. Именно по этой причине в системе управления знаниями целесообразно иметь технические инструменты, позволяющие пользователям оценивать работу экспертов в виде проставления им оценки после проведения экспертиз, консультаций и т.п. Целесообразно также хранение такого рода информации в накопительной форме.

Накопление и систематизацию оценок не сложно осуществлять при сериях однотипных экспертиз, таких, как судейство в соревнованиях, оценка проектов и т.п. Здесь становится возможным исчисление оценок методами корреляции рангов коэффициентов конкордации оценок эксперта по отношению к оценкам группы экспертов. Однако наиболее интересны и важны для практики уникальные экспертизы больших проектов, не имеющих аналогов.

В выявлении экспертов определенным интерес представляет метод «снежного кома», при котором от каждого специалиста, привлекаемого в качестве эксперта, получают несколько фамилий тех, кто может быть экспертом по рассматриваемой тематике. Очевидно, некоторые из этих фамилий встречались ранее в этой деятельности, а некоторые – новые. Процесс расширения списка останавливается, когда новые фамилии перестают встречаться. В результате получается достаточно обширный и все же конечный список возможных экспертов.

При всем многообразии проблем в подборе и организации труда экспертов немаловажным является стимулирование экспертов к коллективному использованию и распространению знаний.

В параграфе, посвященном рынку знаний, отмечалось, что основными участниками рынка знаний являются покупатели, продавцы и брокеры знаний. Эксперты, вне всякого сомнения, относятся к продавцам знаний, так как они обладают значительными и, порой, даже избыточными ресурсами знаний по определенным проблемам и темам. Для того чтобы эксперты делились своими знаниями, они должны быть убеждены, что компания вознаграждает передачу знаний более весомо, чем их скрытое накопление в головах экспертов. Вознаграждение экспертов не обязано при этом быть исключительно материальным. В качестве разменной валюты могут использоваться взаимность в обменах знаниями, рост репутации эксперта и ее учет при его карьерном росте, приобрете-

ние неформального статуса весьма эрудированного и глубоко образованного специалиста-аналитика, что представляет немалую ценность для альтруистов и честолюбивых людей.

Помимо стимулирующих факторов, имеет смысл указать и ряд препятствий на пути эффективного использования экспертов.

Кроме компаний, основной сферой деятельности которых является экспертиза и консалтинг, экспертами, как правило, являются сотрудники, не свободные от другой систематической работы. В таких случаях занятость эксперта может стать препятствием на пути его стремления или обязанности осуществлять экспертизу или консультации.

У эксперта могут возникать подозрения и страх, что необходимость в нем как в квалифицированном сотруднике отпадет, как только его знания и опыт станут общим достоянием. У эксперта может пропасть желание к общению, если процесс общения затруднен (командировки, написание емких отчетов и наставлений и т.п.) и если среда общения не поддерживается современными IT-технологиями.

Подводя итог, следует отметить, что эксперты компании являются весьма ценным и весьма плодотворным ресурсом в обнаружении, обобщении и распространении знаний, который следует накапливать и эффективно расходовать в интересах совершенствования бизнес-процессов.

3.6. Система обучения персонала

Необходимым атрибутом конкурентоспособного предприятия является непрерывная система обучения персонала.

Традиционная система профессионального обучения, включающая профессиональные училища, колледжи, институты и университеты, обычно решает задачу априорной (докорпоративной) подготовки специалистов широкого профиля с базовым набором фундаментальных и прикладных знаний. Корпоративная система обучения должна в относительно короткие сроки обеспечивать адаптацию фундаментальных и прикладных знаний специалиста к профилю производства конкретного предприятия, к постоянно меняющимся требованиям конкретных рабочих мест при внедрении новых технологий, освоении новых товаров и услуг.

Существенное расширение доступа к знаниям, ставшее возможным благодаря современным информационным технологиям, в значительной мере видоизменяет характер отношений между профессионалом и не-

профессионалом, организацией и работником, источником и получателем знаний.

Тем самым получает мощную поддержку концепция и практика непрерывного образования, дающая возможность человеку учиться на протяжении всей жизни, как альтернатива традиционному образованию (возможно, и весьма достойного) один раз на всю оставшуюся жизнь.

Для построения системы непрерывного обучения персонала, как одного из важнейших элементов создания и распространения знаний, возможны различные формы. Классическая форма «преподаватель – группа учащихся», культивируемая в учебных заведениях, вряд ли приемлема для корпоративного обучения, ибо ей свойственны определенные недостатки:

- При групповом обучении большинство обучающихся будут ощущать дефицит индивидуального общения с преподавателем.
- Средний уровень и средняя интенсивность процесса передачи знаний не позволяют передать максимум знаний за относительно короткое время.
- Как правило, преподаватель настраивает процесс передачи знаний на средний уровень. Сотрудники с низкими и высокими интеллектуальными способностями в этих условиях не получают должного удовлетворения, хотя и по разным причинам.

Для обеспечения интенсивного (по объему и уровню) процесса передачи знаний может быть использована форма «преподаватель – индивидуальный ученик». Однако эта форма обучения при массовом охвате сотрудников компании требует привлечения большего отряда квалифицированных преподавателей, что далеко не всегда достижимо по разным причинам, в том числе по сумме затрат.

При современном уровне развития информационных технологий, электронных систем дистанционного обучения оптимальным является способ, основанный на синтезе рассмотренных традиционных форм [Вебер А.В., Данилов А.Д., Шифрин С.И., 2002]. Речь идет о построении системы непрерывного корпоративного обучения персонала на основе создания *сетевой среды обучения* (ССО). Блок-схема такой системы приведена на рис. 3.6.

Основными элементами ССО являются:

- Хранилище знаний компании, корпоративная память, содержащие структурированную совокупность явных знаний в виде баз данных и знаний с соответствующими метаописаниями.

- Сервер сетевой среды, содержащий правила и методические рекомендации, обеспечивающие рациональное управление процессом освоения знаний.
- Web-портал компании с развитым доступом в международные информационные ресурсы.
- Рабочее место (или некоторая совокупность рабочих мест) преподавателей-экспертов по различным корпоративным программам.
- Рабочая станция администратора ССО.
- Рабочие места обучаемых, количество которых может достигать общего числа сотрудников компании, подключенных к корпоративной информационной системе.

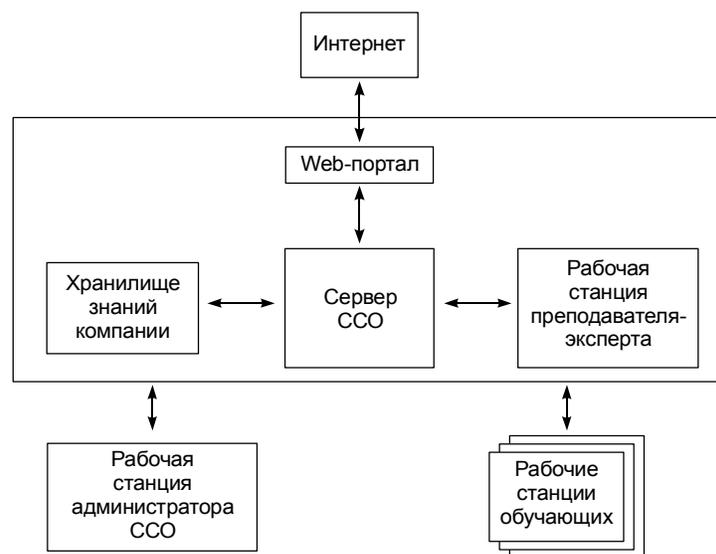


Рис. 3.6. Структурная схема ССО компании

Естественно, что ССО должна быть постоянно развивающейся средой, поддержку которой осуществляют преподаватели-эксперты по соответствующим образовательным программам. Кроме того, развитие ССО обеспечивается, с одной стороны, общим процессом создания и применения знаний в компании, результаты которого систематически отражаются в базах знаний компании, а с другой – развитием Web-

портала компании и обеспечиваемым им доступом к знаниям ресурсам сети Интернет.

Такого рода сетевая среда обучения, развивающаяся вместе с развитием корпоративной системой управления знаниями, позволяет:

- Обеспечить массовый и в то же время персонализированный процесс обучения сотрудников.
- Обеспечить личную образовательную и личную познавательную траекторию каждому обучающемуся с учетом его интеллектуальных способностей и индивидуальных потребностей в знаниях.
- Сократить время обучения и соответственно отрыв специалистов от производства с учетом асинхронного характера взаимодействия сотрудников с ССО.
- Сократить общие затраты компании на обучение, так как ССО, по сути, является лишь еще одной подсистемой корпоративной системы управления знаниями, создаваемой для удовлетворения многоцелевых интересов персонала.

Одной из первых компаний, разработавшей и выпустившей в 1996 г. на рынок образовательных услуг первую версию ССО, является компания Lotus. Позднее в Канаде была разработана ССО WEB CN (www.webct.com), которая используется во многих учебных заведениях и университетах мира.

В настоящее время Европейский Союз и многие входящие в него страны уделяют большое внимание развитию ССО. Применительно к корпоративным ССО, имеющим ряд отличий от ССО для учебных учреждений, в нашей стране в ФГУП «Адмиральские верфи» была разработана интегрированная сетевая среда обучения «НИКА» [Вебер А.В., Данилов А.Д., Шифрин С.И., 2002].

Эта система обеспечивает обучающимся удаленный доступ к сетевым учебным материалам и разнообразной информации корпоративного хранилища знаний под контролем администратора ССО, интерактивный обмен информацией между преподавателем-экспертом и обучающимся, анализ и обработку результатов обучения в реальном времени, поддержку базы данных (БД) требований к знаниям, умениям и навыкам применительно к конкретным рабочим местам, хранение результатов тестирования обучающихся, а также ранжирование специалистов в соответствии с критериями, устанавливаемыми управлением кадровой политики.

Перечисленные и другие возможности интегрированных сетевых сред обучения позволяют считать их эффективным средством быстрого

и массового распространения уникальных профессиональных знаний, которые, как уже отмечалось, являются важным компонентом обеспечения конкурентоспособности современного предприятия.

Другим важным элементом непрерывной системы повышения квалификации персонала является тренинг и наставничество. Тренировка (натаскивание) и наставничество (воспитание) стали в последние годы весьма распространенными методами передачи знаний.

Ученики часто работают с мастерами (наставниками), обучаются ремеслу путем наблюдения, имитации и практической работы. Все это направлено, в основном, на повышение квалификации индивидуумов так, чтобы они могли позже выполнять свои задачи сами. Процесс накопления мастерства требует от учеников непрерывной практики до тех пор, пока они не достигнут требуемого уровня.

Грамотно построенная тренировка требует, чтобы тренер обеспечил развитие компетенции ученика, что трудно достигнуть без глубокого понимания психологии ученика, его потенциала и опыта. При этом использование элементов группового обучения также не исключается. Наставничество предполагает соединение новых и малоопытных сотрудников с более опытным старшим поколением, для того чтобы явные и неявные знания предприятия или организации могли быть эффективно переданы. Связка между наставником (mentor) и его протеже ускоряет процесс профессионального роста новых сотрудников. Такой метод передачи знания особенно полезен в организациях с существенной долей сотрудников, достигающих пенсионного возраста, или там, где имеет место высокая сменяемость персонала. С помощью наставничества в этих условиях достигается преемственность знаний, их передача от поколения к поколению.

Тренировка и наставничество позволяют также более опытному персоналу сполна «отплатить» организации, то есть вернуть ей то, что она вложила в своего сотрудника за весь период действия трудового договора.

3.7. Корпоративная культура

Корпоративная культура (КК) является одним из наиболее важных факторов содействия эффективному управлению знаниями. Существуют множество различных определений КК с различных точек зрения.

«Корпоративная культура – это совокупность коллективных базовых представлений, обретаемых организацией при разрешении проблем адаптации к изменениям внешней среды и внутренней интеграции, эф-

фektivность которых оказывается достаточной для того, чтобы считать их ценными и передавать новым членам организации в качестве правильной системы восприятия и разрешения названных проблем» [Похолков Ю.П., Чучалин А.И., 2004].

В контексте УЗ: «Корпоративная культура – это приобретенный способ восприятия, постижения, мышления, совместно используемый и передаваемый между членами организации» [Schein E.H., 1985].

КК включает в себя такие элементы, как:

- Историю, традиции, ритуалы, правила.
- Способ распределения и осуществления власти.
- Степень формализации и стандартизации управления процессами.
- Язык профессионального общения.
- Возможности для творческого самовыражения индивидуумов и групп.
- Систему учета мнений персонала.

К выводу о том, что КК относится к числу важнейших факторов успеха в эффективном управлении знаниями, приводят не только аргументы концептуального характера, но и результаты опросов, проведенных в 104 компаниях [Mertins K., Heisig P., Vorbeck J., 2003] (рис. 3.7).



Рис. 3.7. Пять решающих факторов для достижения успеха в управлении знаниями

В числе пяти решающих факторов для достижения успеха в управлении знаниями, а именно «Развитие в топ-менеджменте», «Мотивация и квалификация персонала», «Информационные технологии», «Корпоративная культура», «Структуры и процессы» в 49 из 104 компаний первое по важности место отдается «Корпоративной культуре». Далее степень важности перечисленных факторов с небольшим градиентом убывает вплоть до «Информационных технологий», важность которых признают 29 из опрошенных компаний. Известно, однако, что этому эмпирическому выводу далеко не всегда следуют на практике, предпочитая IT-технологии кропотливой работе по другим менее осязаемым, но не менее простым направлениям.

Существуют разные типы КК: культура иерархии, культура конкуренции, культура адхократии, культура семьи и др.

Культура иерархии, в соответствии с которой организация сосредоточена на внутренних проблемах, стабильности, управляемости, контроле. Ее деятельность структурирована и формализована. Целостность организации поддерживается формальными правилами. Процедуры, правила, инструкции диктуют людям, что нужно делать. Поощряется высокая исполнительность. Работника могут не поощрить за успех, достигнутый с нарушением процедур, и не наказать за ущерб, если все процедуры выполнены по инструкциям. Изменения связаны, прежде всего, с изменением процедур. Лидеры гордятся тем, что они хорошие организаторы и умеют программировать деятельность подразделений.

Культура конкуренции, в соответствии с которой организация сосредоточена на взаимодействии с внешней средой, стабильности, управляемости, контроле, достижении измеряемых рыночных результатов. Главная задача организации в целом и каждого работника в отдельности – достижение намеченных целей в установленные сроки. Стиль организации – жесткая конкуренция внутри и снаружи. Успех определяется в экономических терминах и терминах завоевания рынка. Лидеры – хозяева, нацеленные на конкурентную борьбу. Они требовательны и лишены снисходительности.

Культура адхократии (творчества) ориентирует организацию на внешние факторы, на высокую степень гибкости и индивидуализма. Для данной культуры характерны динамичные, творческие, поощряющие предприимчивость и индивидуальные результаты условия работы. Люди склонны к инициативе и риску. Поощряются независимость и свобода сотрудников. Лидеры – новаторы, экспериментаторы – пользу-

ются уважением за творчество. Главная задача организации в целом и каждого работника в отдельности – быть в авангарде, занимать лидирующие позиции. Критерием успеха является обладание уникальными технологиями, продуктами или услугами. Базовыми предположениями для адхократической культуры являются хаотичность и сложность окружения, поощрение инноваций, а сверхзадача управления – культивирование предприимчивости и творческого подхода. Акценты делаются на формирование зримого будущего, творческого хаоса, структурированного воображения.

Культура семьи. Семейная или клановая культура воспроизводит организацию как большую семью с пожизненным наймом, слабо иерархичной структурой, неформальным подходом к работе и акцентом на групповые и командные структуры. Организация сосредоточена на внутренних проблемах, для нее характерны гибкость, забота о людях, чувствительность к проблемам заказчика, который рассматривается как партнер. По существу, организация представляет собой многочисленную профессиональную семью, некое дружественно-ориентированное место для работы, где люди оставляют часть себя. Лидер организации воспринимается как отец семейства, с неограниченными правами и ответственностью. Целостность организации поддерживается традициями и лояльностью к семейным ценностям. Поощряются работа в командах, соучастие и единодушие. Большое значение придается развитию человеческих ресурсов, сплоченности, моральному климату. Декларируется, что благоразумная свобода и вовлечение персонала в бизнес ведут к единству и прибыльности.

На самом деле, при доминировании некоторого конкретного типа КК, в каждой организации можно обнаружить следы нескольких культур, которые воздействуют тем или иным способом на ценности, мотивацию и поведение сотрудников. Такая ситуация не является следствием некоего культурного хаоса, а может быть следствием предшествующего опыта, накопленного сотрудниками или группами при их работе в других организациях, в других условиях либо при другой команде топ-менеджеров. Кроме того, в одной организации могут существовать персональные, командные и корпоративные намерения и программы, содержащие конкурирующие или даже конфликтующие устремления. Все это может приводить к наличию в организации наряду с основной культурой неких субкультур, свойственных отдельным группам, подразде-

лениям, филиалам, к появлению в отдельных частях организации так называемого феномена «бункерного мышления».

Наличие субкультур в организации далеко не всегда является недостатком. Применительно к УЗ, разнообразие культур способствует многообразию взглядов и оценок, придаваемых различным типам знаний, расширению языка и способов профессионального общения, приверженности тем или иным стилям управления и технологиям. В любом случае «культурный аудит» – выявление и идентификация корпоративной культуры и субкультур организации – необходимый и ответственный этап, предшествующий и в некоторой степени определяющий последующие рекомендации по совершенствованию системы управления знаниями.

Способ распределения и осуществления власти как элемент корпоративной культуры во многом предопределяется структурой организации. К традиционным, слабо приспособленным для УЗ типам структур относятся многоуровневые иерархические структуры, декомпозированные по функциональному и/или территориальному принципам, матричные структуры, в основе которых лежит как функциональный, так и процессный принцип выделения и группировки структурных элементов.

В экономике, основанной на знаниях, в новых условиях, предоставляемых современными ИТ-технологиями, имеет место другая тенденция – *разделение крупной организации на малые самоуправляемые структуры*. Так, крупный завод разделяется на малые «целевые фабрики», которые производят небольшие партии разнообразных товаров или комплектующие для более сложной продукции. Большие организации, где работают служащие, подразделяются на малые офисы, которые предоставляют определенный вид услуг. Этот переход от крупномасштабных систем к малым подразделениям минимизирует численность бюрократического аппарата, наделяет взаимодействующие подразделения большими полномочиями и делает изменения более быстрыми и адекватными требованиям внешней среды.

Интеграция операций в единое целое осуществляется при этом с помощью информационных систем. Компьютерно-интегрированное производство опирается на мощные персональные компьютеры для информационного обеспечения всех фаз деятельности – от проектных разработок до производства, управления запасами, распределения ресурсов и сбыта продукции. Коммуникационные системы позволяют локальным структурам и отдельным сотрудникам «работать на расстоянии» при

любой степени их удаленности. Организация становится совокупностью отдельных гибких производственных систем, взаимодействующих и управляемых с помощью знаний и компьютеров. В результате такого способа распределения власти и ответственности обеспечивается более высокая эффективность производства, создается обстановка взаимного доверия и взаимной ответственности. Партнерство здесь менее формально. Информационные сети устанавливают эффективные и быстро действующие связи между целевыми фабриками. Это – прямой путь к появлению в будущем структур, именуемых так называемыми *горизонтальными корпорациями*.

Отличительные особенности в осуществлении властных полномочий имеет и *сетевая организация*, для которой характерно гибкое, иногда временное взаимодействие между производителями, поставщиками и потребителями. Это динамичная структура, в которой основные компоненты могут быть «смонтированы» или «размонтированы» согласно изменившимся конкурентным условиям.

Существует несколько разных терминов для описания совсем уж новой организационной модели. Некоторые считают ее большой *паутиной* с главным подразделением в центре, которая работает как централизованная организация. Этот центр связан с разными участниками, каждый из которых выполняет специализированную функцию, и все взаимосвязаны друг с другом, что очень напоминает паука со своей паутиной.

К числу значимых примеров комбинации многочисленных и разнообразных знаний, достижений в технике и технологиях относится так называемая *виртуальная корпорация*. Она используется для формирования временной *сетевой структуры* независимых организаций, связанных информационной технологией и объединяемых в разных сочетаниях для эффективного использования в стремительно меняющихся условиях.

В связи с возрастающей международной конкуренцией, увеличением затрат на научные исследования, стремлением минимизировать риск, связанный с созданием и коммерциализацией технологии, компании объединяются в консорциумы, *инновационные стратегические альянсы*.

В консорциумах и стратегических альянсах, когда принятие решения может осуществляться вне границ традиционной метрополии, управление знаниями сосредотачиваются на таких задачах, как:

- создание, сохранение и распределение активов, находящихся в совместной собственности;

- выработка процедур принятия решений, необходимых для противостояния кризисам в любой части либо сфере деятельности сети;
- использование многообразных форм удовлетворения потребностей местных рынков.

В наши дни применительно к экономике, основанной на знаниях, предлагается множество организационных структур (моделей), причем большинство из них тяготеет к различным вариантам рабочих групп. В их числе уже упоминавшаяся «адхократия» (от латинского «ad hoc» – на данный случай), а также «бесконечно плоская организация», «перевернутая пирамида», «протуберанец» (спутник) и др. К числу наиболее пригодных структур для производства и использования знаний, полученных в результате синтеза бюрократической структуры и структуры по принципу рабочих групп, относится также *гипертекст-организация*, предложенная Нонакой и Такеучи [Нонака И., Такеучи Х., 2003] (рис. 3.8). Метафора «гипертекст», пришедшая из мира компьютеров, как нельзя лучше отражает возможность одномоментного доступа к различным уровням такой структуры, построенным по принципу бизнес-систем, проектных команд, баз знаний.

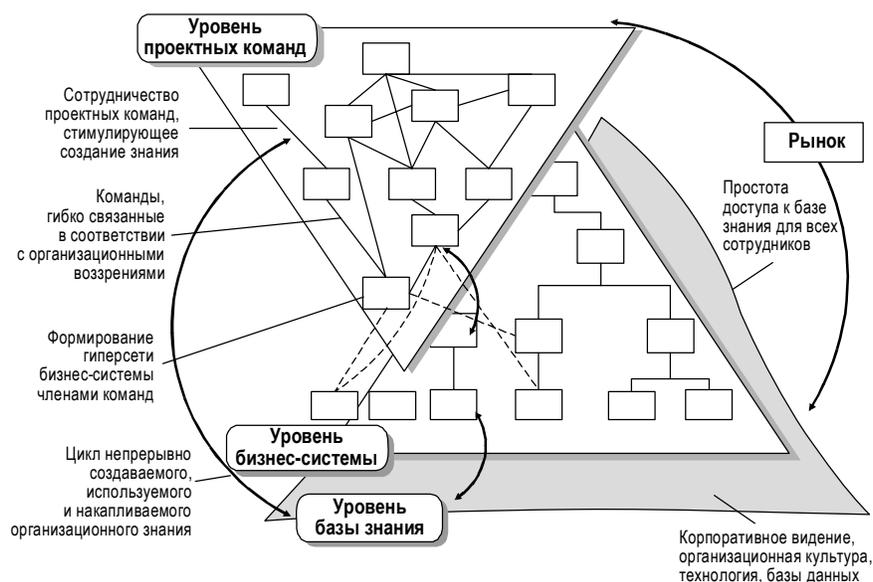


Рис. 3.8. Гипертекст-организация [Нонака И., Такеучи Х., 2003]

Гипертекстовая организация предполагает кросс-иерархическую, самоорганизующуюся структуру, которая работает в тандеме с иерархической формальной структурой. По мере того как бизнес организации растет в масштабе и сложности, гипертекст-организация наращивает гибкость и оперативность как на корпоративном, так и на локальном уровне.

Корпоративная культура, как и организационная структура, может быть стимулирующей системой, мотивирующей сотрудников искать новые, разумные способы поведения и работы. Культуры, ориентированные на создание и использование знаний, требуют как макроуправления (политика и программы, охватывающие всю организацию), так и микроуправления (учет субкультур и индивидуальных различий). Чтобы знание создавалось и применялось, оно должно передаваться и совместно использоваться, а это в значительной мере зависит от характера отношений между индивидуумами и группами. Не лишено оснований утверждение, что внедрить технику и технологию для эффективного доступа к корпоративным информационным ресурсам для информационного взаимодействия, групповой работы гораздо легче, чем развивать культуру сотрудничества. Поэтому так важно поддерживать любые движения в этом направлении и организационные формы. Более высокий уровень доверия между сотрудниками усиливает и ускоряет потоки знания. Если сотрудники тяготеют к неформальному обмену знаниями в чувствительной к контексту обстановке, то акцент на корпоративные хранилища данных, базы знаний и электронные коммуникации могут оказаться не лучшим решением. Создание и поддержка в этом случае тематических сообществ различной направленности может в ряде случаев принести большую пользу. При этом не следует и противопоставлять одно другому. Успех состоит в их гармоничном сочетании.

Корпоративная культура формируется и является результатом совокупного действия индивидуумов. По сути, КК – это накопленная сумма вкладов индивидуумов, прошлых и настоящих сотрудников компании. «Психологический контракт» индивидуумов (их мнения о том, что они должны организации и что организация должна им) заставляет их, а не только руководство компании искать, находить и видоизменять КК для лучшего обслуживания из психологического контракта. При этом вовсе не обязательно ожидать торжества эгоистических и своекорыстных мотивов. Достаточно вспомнить девиз «психологического контракта», ко-

торый кандидат в президенты Д.Ф. Кеннеди предложил американскому обществу: «Не спрашивай, что страна сделала для тебя. Спроси себя, что ты сделал для страны». Общественные и корпоративные интересы, обязанности и заботы также могут быть сильным стимулом для отдельных индивидуумов, групп и организаций.

В процессах обмена неявными знаниями, в установлении границ между личным и корпоративным знанием имеют место определенные проблемы. Прогрессивная система управления знаниями организации должна в этом отношении исходить из понимания, что неявное знание остается в головах индивидуумов и является их личным знанием, в то время как результат его применения – инновационные идеи, стратегии, решения, продукция, создаваемая структурными подразделениями в течение оплачиваемого времени, становятся корпоративными знаниями ресурсами, организационным знанием.

Корпоративная культура действует, таким образом, как посредник между личностным и организационным знанием (рис. 3.9), определяя, какие знания принадлежат организации, а какие – личностям и группам.

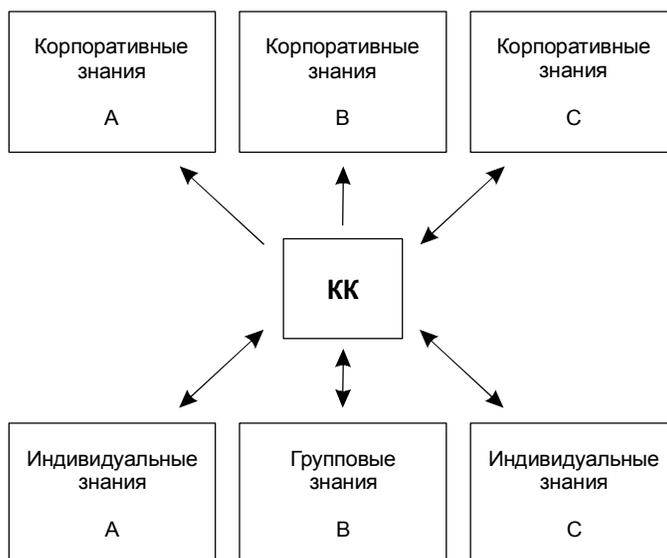


Рис. 3.9. КК как знаниевый посредник

Правила, по которым действует этот посредник, должны минимизировать ситуации, когда сотрудники не склонны передавать свои знания в корпоративные БЗ, опасаясь потерять на них право собственности. Ни в части авторских прав, ни в части моральных и материальных вознаграждений носители явных знаний не должны чувствовать себя ущемленными. В этих вопросах корпоративная культура должна выступать гарантом справедливости.

Глава 4

ОНТОЛОГИЧЕСКИЕ МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Существует множество обстоятельств, которые затрудняют распространение и обмен знаниями между людьми, столь необходимыми для их систематического пополнения и накопления.

Прежде всего, это обстоятельства количественного характера, связанные с быстрым ростом численности населения, с вовлечением новых поколений в различные сферы деятельности, требующие постоянно растущего уровня знаний, умений и навыков.

Особым обстоятельством являются фундаментальные отличия и множественность национальных языков народов, населяющих планету Земля. По данным ЮНЕСКО на нашей планете существует более 2700 языков, народов и народностей.

Но дело не только в том, что народы мира думают, говорят и пишут на разных языках. Многие проблемы в обмене и создании знаний связаны с неоднозначным или неадекватным восприятием смысла данных, информации, знаний различными участниками знаниевого процесса. Дело в том, что в цепи передачи знаний (рис. 4.1) отправитель и получатель знания зачастую пользуются различными представлениями, различной терминологией и понятийным аппаратом. Из-за различий в образовании и в предшествующем опыте они могут руководствоваться различными моделями деятельности и культурой мышления. Много неструктурированных и полуструктурированных информационных источников доступно в сети Web и на различных корпоративных порталах, основное содержание которых представлено на естественном языке в формате HTML. Эта информация не понимается машиной, что приводит к множеству проблем при поиске необходимой информации. Значительная проблема заключается в том, что пользователи до сих пор должны фильтровать результаты работы по-

исковых систем, так как контекст результата поиска может в значительной мере не соответствовать контексту запроса. Если поисковые системы не понимают смысла запроса или информации в источниках, то результат будет выбираться, например, на основе встречаемости ключевых слов в индексируемых информационных источниках, что приводит зачастую к неэффективным результатам поиска. Также возможно, что подходящие документы не будут обнаружены поисковыми системами, потому что ключевые слова в запросе не точно соответствуют терминам в документе.

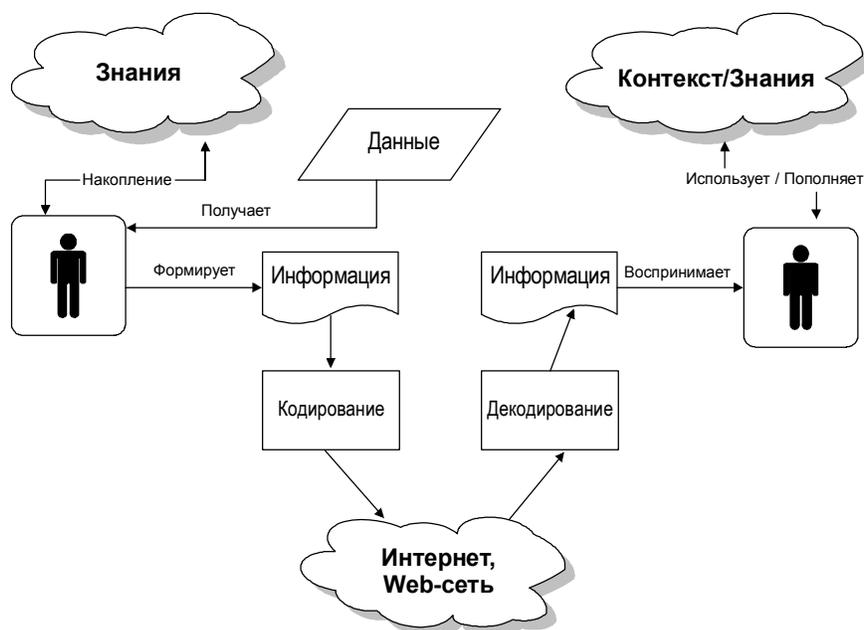


Рис. 4.1. Движение знаний между отправителем и получателем

Вследствие этих объективных обстоятельств весьма желательно, чтобы информация и знания были структурированы и описаны таким образом, чтобы получатель (пользователь) был способен понять и текст, и контекст (смысл) сообщения. В идеале, сообщение (знаниевая сущность) должна структурироваться таким образом, чтобы компьютер, а

не только образованный человек был способен «понять» его. Под словом «понять» здесь имеется в виду, что компьютер будет способен обработать документ (знаниевую сущность) посредством использования известных ему правил с помощью некоторого логического языка, а также будет способен вывести новые факты и знания из данного документа.

4.1. Модель интеллектуального пространства

Основными способами повышения эффективности использования явных знаний являются их систематизация (упорядочение) и использование настроенных на эту систематизацию средств поиска объектов, содержащих требуемые знания. Данная задача является очень сложной. Однако существуют подходы и методы, которые могут быть использованы для ее решения. Их изложению посвящено содержание данной главы.

В предлагаемом авторами подходе предпринята попытка систематизировать объекты, содержащие различные виды знаний, и инструментарий для работы с ними в виде *интеллектуального пространства организации* (или пространства знаний).

Модель многомерного пространства является признанной в разных областях науки абстракцией, которая используется для работы с различными и не только математическими описаниями объектов. Известно, что описание любого пространства включает такие элементы, как:

- выбор системы координат;
- задание способа описания положения объектов в выбранной системе координат;
- задание метрики (способа вычисления) близости объектов в данном пространстве.

В соответствии с этим пространство знаний (интеллектуальное пространство) организации предлагается [Тузовский А.Ф., Ямпольский В.З. Интеллектуальное пространство, 2004] описывать следующим образом:

1. В качестве системы координат использовать *онтологию предметной области*.

2. Описания объектов, содержащие знания, задавать в виде их *метаописаний*, составленных из основных понятий онтологии.

3. В качестве меры близости объектов (метрики) использовать семантическую *близость их метаописаний*.

Схема модели интеллектуального пространства приведена на рис. 4.2.

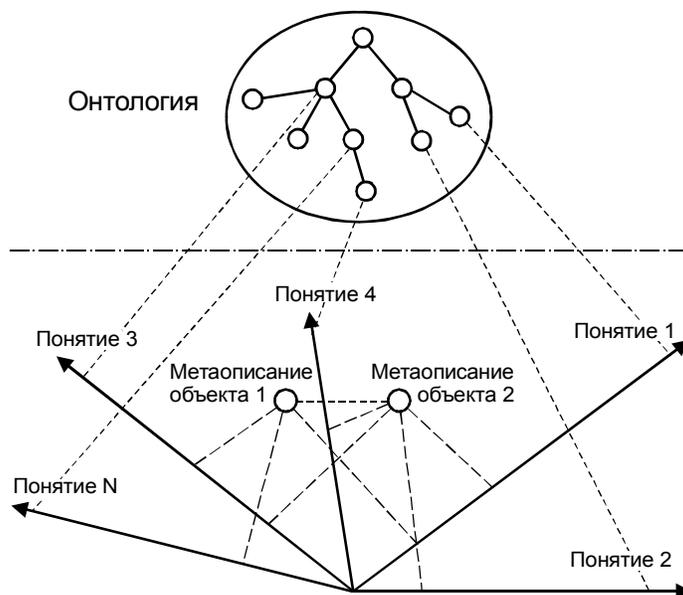


Рис. 4.2. Схема интеллектуального пространства

Остановимся далее на методах и способах описания и использования основных элементов интеллектуального пространства.

4.2. Онтологии предметных областей

Онтологии предметной области описывают явные знания, которые имеются в компании (или в отдельных ее частях). Описанием знаний уже достаточно давно занимается такая дисциплина, как «Искусственный интеллект» (ИИ), а также такие ее разделы, как «Представление знаний» и «Инженерия знаний». Учитывая, что ИИ занимается работой со знаниями с 50-х годов, в данной дисциплине накоплен достаточно большой опыт в области представления (моделирования) знаний. В вопросе описания знаний дисциплина «Управления знаниями» имеет общие интересы с ИИ. У них общий объект исследования – знания, но цели его исследования в этих дисциплинах разные.

Цель ИИ заключается в создании моделей и методов работы со знаниями, которые позволят их использовать без участия (или почти без участия) человека. Например, в [Люггер Д.Ф., 2003] дается следующее

определение: «ИИ можно определить как область компьютерной науки, занимающейся автоматизацией разумного поведения».

Целью УЗ является организация эффективной работы со знаниями (повышение эффективности процессов преобразования знаний на предприятии, создание, сбор, накопление, распространение, использование), при этом использование знаний выполняет человек, сотрудник организации. УЗ направлено на повышение эффективности создания, хранения и использования знаний, но не на замену человека компьютером. Наоборот, в УЗ считается, что единственным источником новых знаний и основным их потребителем является человек (специалист).

В настоящее время существуют и развиваются разные методы представления и описания знаний, например, такие, как: продукционные модели, семантические сети, фреймы, онтологии.

Продукционная модель или модель, основанная на правилах, позволяет представлять знания в виде предположения типа «if – then»: если (условие), то (действие). Под «условием» понимается некоторое предложение – образец, по которому осуществляется поиск в базе знаний, а под «действием» – выполняемые в результате успешного поиска действия.

Продукционные модели чаще всего применяются в промышленных экспертных системах при фиксации совокупности правил поведения персонала в некотором множестве ситуаций.

Семантическая сеть означает «смысловая» сеть, а, собственно, семантика – это наука, устанавливающая отношения между символами и объектами, которые они обозначают. Иначе говоря, семантика – это наука, определяющая смысл знаков [Люггер Д.Ф., 2003].

По своей структуре семантическая сеть – это ориентированный граф, вершины которого – понятия, а дуги – отношения между ними. Характерной особенностью семантических сетей является наличие трех типов отношений:

- отношение класс – элемент класса;
- отношение свойство – значение свойства;
- отношение фрагмент – элемент класса.

Чаще всего в семантических сетях используются следующие отношения:

- связи типа часть – целое (например, элемент – класс);
- функциональные связи (определяются глаголами типа «производит», «влияет»...);
- количественные отношения (больше, меньше, равно);

- пространственные отношения (далеко от, близко от...);
- временные отношения (раньше, позже, в течение ...);
- атрибутивные связи (иметь свойство, иметь значение);
- логические связи (И, ИЛИ, НЕ);
- лингвистические связи и др.

Фрейм (от англ. frame – каркас, рамка) – это абстрактный образ для представления некоторого стереотипа восприятия. По существу, фрейм – это некоторый абстрактный обобщенный образ совокупности объектов. Например, фрейм «комната» порождает у человека образ жилого помещения с четырьмя стенами, полом, потолком, окнами и дверью. Фреймом может называться и некоторая формализованная модель для отображения образа.

Различают фреймы-образцы или прототипы, хранящиеся в базах знаний, и фреймы-экземпляры, с помощью которых отображаются реальные физические ситуации на основе поступающих данных. Универсальность модели фрейм позволяет иметь множество конкретных разновидностей фреймов: фреймы-структуры, фреймы-роли, фреймы-сценарии, фреймы-ситуации и др.

В последние десятилетия в качестве наиболее перспективной модели представления знаний рассматриваются онтологии.

Онтология (от древнегреч. онтос – сущее, логос – учение, понятие) – термин, определяющий учение о сущем, бытии, в отличие от гносеологии – учение о познании. В философском смысле, а этот термин заимствован из философии, онтология есть определенная система категорий, являющихся следствием определенных взглядов на мир.

Термин «онтология» был использован рядом исследовательских сообществ по ИИ вначале в области инженерии знаний, в обработке естественных языков, а затем в представлении знаний. В конце 1990-х годов понятие онтологии также стало широко использоваться в таких областях, как интеллектуальная интеграция информации, поиск информации в Интернет и управление знаниями [Jos de Bruijn, Fensel D., Staab S., Studer R., 2004]. Позже онтологии стали рассматриваться в качестве ключевого элемента в проекте Семантической Сети – нового этапа развития сети WWW (Word Wide Web). Если существующая Web-сеть – это огромное множество документов, которые связаны перекрестными ссылками, то создаваемая Семантическая Сеть должна добавить к существующей сети множество онтологий и метаописаний знаний, содержащихся в документах Web-сети (включая стандарты и программные инструменты) [Berners-Lee T., Hendler J., Lassila O., 2001].

Онтологии были разработаны для облегчения *обмена и повторного использования знаний* [Jos de Bruijn]. Они являются по существу формальными словарями, совместно используемыми группами специалистов, работающих в конкретных (возможно, и весьма широких) прикладных областях.

Определение онтологии, которым руководствуются многие исследователи в данной области, было дано в [Gruber T.A., 1995].

«Онтология – это формальное, явное, точное определение (спецификация) совместно используемой концептуализации». *Концептуализацией* именуется абстрактное упрощенное представление мира, которое формируется для некоторых целей. Онтология является *точным определением (спецификацией)* потому, что она представляет концептуализацию в конкретной форме. Она является *явной*, потому что все используемые в ней ограничения явно определены. Слово *формальная* означает, что онтология должна пониматься машиной. Слово *совместно используемая* указывает на то, что онтология содержит согласованные знания.

Рабочим и более приближенным к управлению знаниями можно признать определение, приведенное в [Гаврилова Т.А., Хорошевский Ф.В., 2001]: «Онтология – это базы знаний специального типа, которые могут «читаться» и пониматься, отчуждаться от их разработчика и /или физически разделяться их пользователями».

Онтология состоит из терминов, организованных в таксономию, их определений, атрибутов, а также связанных с ними аксиом и правил вывода.

Онтология, таким образом, соединяет человеческое и компьютерное понимание символов. Эти символы, также называемые терминами (точными определениями понятий), могут интерпретироваться как людьми, так и машинами. Термин понятен для человека, так как это слово, написанное на естественном языке. Понятны человеку и связи между терминами типа «суперпонятие – подпонятие» (род – вид), обычно обозначаемые как *is-a* (являться). Эта связь обозначает тот факт, что одно понятие (субпонятие) является более общим, чем другое (подпонятие). В качестве примера возьмем такое понятие, как компьютер, которое является менее общим, чем понятие машина (автомобиль, трактор, танк и т.д.).

На рис. 4.3 показан пример *is-a* иерархии (таксономии), где более общие понятия расположены выше менее общих (специализированных) понятий. Используя даже столь простые связи, можно сделать ряд выводов. В частности, можно утверждать, что корпус (подпонятие) может

принадлежать суперпонятиям компьютер и автомобиль. Этот вывод может быть сделан как людьми, так и компьютером.

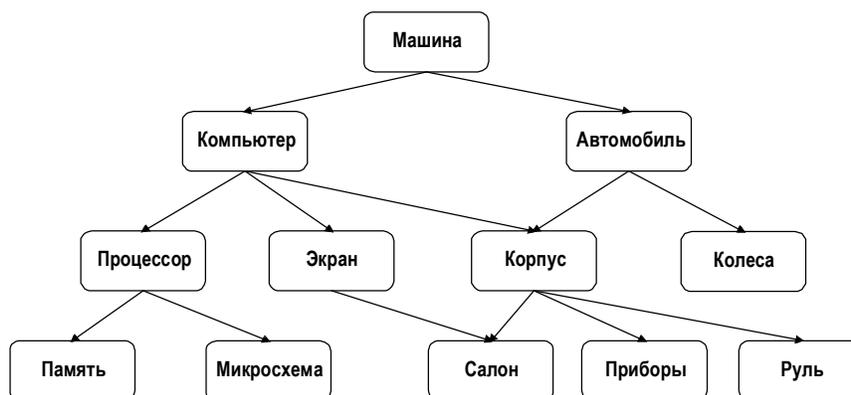


Рис. 4.3. Пример *is*-а иерархии (таксономии)

Причем человек делает этот вывод из практики, а компьютер – из формального описания приведенной схемы связей. Понятия описывают набор объектов реального мира. Связи фиксируют отношения между ними. Компьютер, не имея человеческого «понимания», обрабатывает кодированные представления понятия и связей между объектами и, таким образом, способен сделать аналогичные выводы, что и человек на основе логических рассуждений.

4.3. Формальная модель онтологии

В общем виде формальная модель онтологии может быть описана следующим кортежем [Meadche A., Zacharias V., 2002]:

$$O = \{L, C, F, G, H, R, A\},$$

где

- $L = L^C \cup L^R$ – словарь онтологии, содержащий набор лексических единиц (знаков) для понятий L^C и набор знаков для отношений L^R ;
- C – набор понятий онтологии, причем для каждого понятия $c \in C$ в онтологии существует по крайней мере одно утверждение;
- F и G – функции ссылок такие, что $F: F^{LC} \rightarrow 2^C$ и $G: F^{LR} \rightarrow 2^R$. То есть F и G связывают наборы лексических единиц $\{L_j\} \subset L$ с набо-

рами понятий и отношений, на которые они соответственно ссылаются в данной онтологии. При этом одна лексическая единица может ссылаться на несколько понятий или отношений и одно понятие или отношение может ссылаться на несколько лексических единиц. Инверсиями функций ссылок являются F^{-1} и G^{-1} ;

- H – фиксирует таксономический характер отношений (связей), при котором понятия онтологии связаны нерефлексивными, ациклическими, транзитивными отношениями $H \subseteq C \times C$. Выражение $H(C_1, C_2)$ означает, что понятие C_1 является подпонятием C_2 ;
- R – обозначает бинарный характер отношений между понятиями онтологии, фиксирующие пары области применения (domain)/области значений (range), то есть пары (D, R) с $D, R \in C$;
- A – набор аксиом онтологии.

Более простая модель онтологии (без словаря и без определения типа отношений) приведена в [Гаврилова Т.А., Хорошевский Ф.В., 2001]. Под онтологией в этой работе понимается упорядоченная тройка вида

$$O = \langle C, R, F \rangle,$$

где

- C – конечное множество концептов (понятий, терминов) предметной области, которую представляет онтология O ;
- R – конечное множество отношений между концептами заданной предметной области;
- F – конечное множество функций интерпретации (аксиоматизации), заданных на концептах и/или отношениях онтологий O .

Естественным ограничением, налагаемым на множество C , является его конечность и непустота. Иначе обстоит дело с множествами R и F . При этом R и F тоже должны быть конечными множествами.

В работе [Гаврилова Т.А., Хорошевский Ф.В., 2001] анализируются интересные граничные случаи, связанные с их пустотой, которые позволяют лучше понять онтологии.

Пусть $R = \emptyset$ и $F = \emptyset$. Тогда онтология O трансформируется в простой словарь:

$$O = V = \langle C, \{\}, \{\} \rangle.$$

Такая вырожденная онтология может быть полезна для спецификации, пополнения и сопровождения словарей. Однако онтологии-словари имеют ограниченное применение, поскольку не вводят эксплицитно смысла терминов. В отдельных случаях, когда используемые термины

принадлежат очень узкому (например, техническому) словарю и их значение устоялось в пределах некоторого профессионального сообщества, такие онтологии также могут применяться на практике.

Иная ситуация возникает при использовании терминов естественного языка или когда общаются программные агенты. В этом случае необходимо характеризовать смысл элементов словаря с помощью подходящей аксиоматизации с целью исключения неоднозначной интерпретации всеми пользователями онтологии.

Другой вариант соответствует случаю, когда $R = \emptyset$, но $F \neq \emptyset$. В этом случае каждому элементу множества терминов C может быть поставлена в соответствие функция интерпретации $f \in F$. Формально это может быть записано следующим образом:

Пусть $C = C_1 \cup C_2$,
причем $C_1 \cap C_2 \neq \emptyset$,

где C_1 – множество интерпретируемых терминов; C_2 – множество интерпретирующих терминов.

Тогда

$$\exists(x \in C_1, y^1, y^2, \dots, y^k \in C_2)$$

такие, что $x = f(y^1, y^2, \dots, y^k)$, где $f \in F$.

Пустота пересечения C_1 и C_2 исключает циклические интерпретации, а введение в рассмотрение функции k аргументов призвано обеспечить более полную интерпретацию. Вид отображения f из F определяет выразительную мощь и практическую полезность этого вида онтологии. Например, если предположить, что функция интерпретации задается оператором присваивания значений ($C_1 = C_2$, где C_1 – имя интерпретации C_2), то онтология в данном случае трансформируется в пассивный словарь V^P :

$$O = V^P = \langle C_1 \cup C_2, \{\}, \{:=\} \rangle.$$

Такой словарь пассивен, так как все определения терминов и C_1 берутся из уже существующего, фиксированного и явно прописанного множества C_2 . Практическая ценность его выше, чем просто словаря V , но явно недостаточна, например, для представления знаний в обработке информации в Интернет в силу мультипредметного характера этой среды.

Ценность словаря возрастет, если часть интерпретирующих терминов из множества C_2 будет задаваться процедурно, а не декларативно. То есть их смысл будет «вычисляться» при каждой интерпретации.

Для создания онтологии, которой по силам решение разнообразных задач обработки информации, можно отказаться от предположения $R = \emptyset$.

В [Гаврилова Т.А., Хорошевский Ф.В., 2001] рассмотрены следующие варианты формирования множества R .

Вначале определен специальный подкласс онтологий – простая таксономия:

$$O = T^0 = \langle C, \{is-a\} \rangle.$$

Под таксономической структурой понимается иерархическая система понятий, связанных между собой отношением *is-a* (быть элементом класса). Отношение *is-a* имеет априори фиксированную семантику и позволяет строить структуру понятий онтологии типа дерево.

Продолжая цепь рассуждений, можно обобщить частные случаи онтологий таким образом, чтобы обеспечить:

- представление множества концептов C в виде сетевой структуры;
- использование расширенного множества отношений R , включающего не только таксономические, но и отношения, отражающие специфику конкретной предметной области, плюс средства расширения множества R ;
- использование декларативных и процедурных интерпретаций и отношений, включая возможность построения новых интерпретаций.

4.4. Типы онтологий

Существуют много разных онтологий, построенных для различных целей и приложений. Здесь, как и в любых других случаях, когда необходима систематизация множества объектов, возможно использование различных признаков для классификации.

Выделение типов онтологий будет приведено далее на основе таких признаков, как универсальность и выразительность. Уровень универсальности (*generality*) определяет масштаб онтологии, а выразительность (*expressiveness*) – детальность ее описания.

По уровню универсальности выделяют три типа онтологий:

- *Онтологии верхнего уровня*, или *метаонтологии*, описывающие общие понятия, независимо от задач конкретного домена. Примером такой онтологии может служить WordNet [Fellbaum C., 1998]. Масштаб WordNet весьма обширный – весь английский язык с описанием каждого термина, его синонимов и гипер/типо (бо-

лее/менее) общих терминов и отношений между ними. В то же время уровень детальности в WordNet очень низкий, имеются лишь описания на естественном языке терминов, которые не могут быть поняты машиной, и зафиксированы только самые простые отношения между ними.

- *Онтологии предметных областей и онтологии задач* описывают относительно общие понятия для общих задач. В какой-то мере она относится к онтологиям верхнего уровня, так как ее можно использовать во множестве предприятий различных предметных областей.
- *Онтологии приложений* описывают понятия, зависящие как от домена, так и от решаемой задачи.

Взаимосвязь между различными онтологиями формальной модели онтологической системы показана на рис. 4.4 [Гаврилова Т.А., Хорошевский Ф.В., 2001].

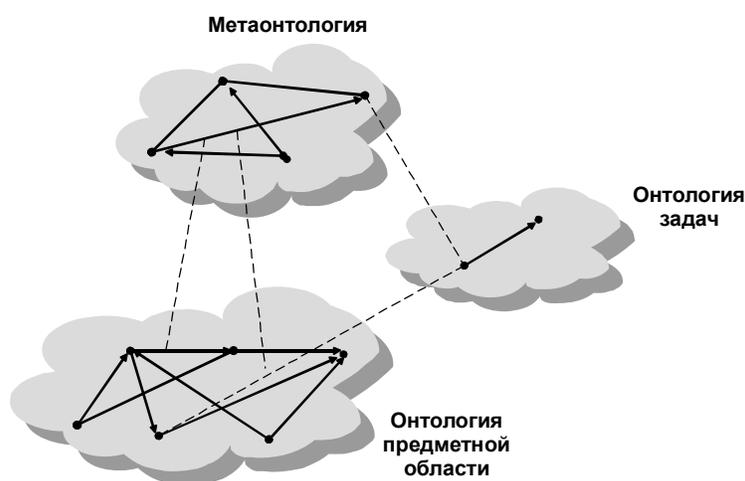


Рис. 4.4. Взаимосвязь между различными типами онтологий

По степени выразительности McGuinness [McGuinness D. L., 2003] выделяет следующий спектр онтологий:

- *Контролируемые словари*: список терминов.
- *Тезаурусы*: предоставляют связи между терминами, такие, как синонимы.

- *Неформальная таксономия*: существует явная иерархия, но нет строгого наследования. Экземпляр подкласса не обязательно является также экземпляром суперкласса.
- *Формальная таксономия*: существует строгое наследование.
- *Фреймы*: описание классов и их свойств.
- Описание классов с заданными ограничениями на их свойства.
- Описание классов с простыми логическими или математическими ограничениями на свойства и отношения.
- Описание классов со сложными логическими отношениями на свойства и отношения (логики первого порядка и отношений типа дизъюнкция, инверсия, часть – целое и т.п.).

Как уже отмечалось, существуют и другие признаки для классификации онтологий и соответственно типы онтологий. Существуют, в частности, легковесные и тяжеловесные онтологии. Следуя аргументам [Corcho O., Fernández-López M., 2003], RosettaNet (<http://www.rosettanet.org/>) и Yahoo! Dictionary (<http://www.yahoo.com/>) иногда именуют онтологиями, так как они представляют согласованную концептуализацию некоторой предметной области, но в действительности они немногим более чем таксономии, и поэтому являются легковесными онтологиями. Тяжеловесные онтологии, кроме понятий, свойств и отношений между понятиями, таксономии, включают также аксиомы и ограничения. Приведенные выше суждения относительно спектра выразительности онтологий иллюстрируются на рис. 4.5.

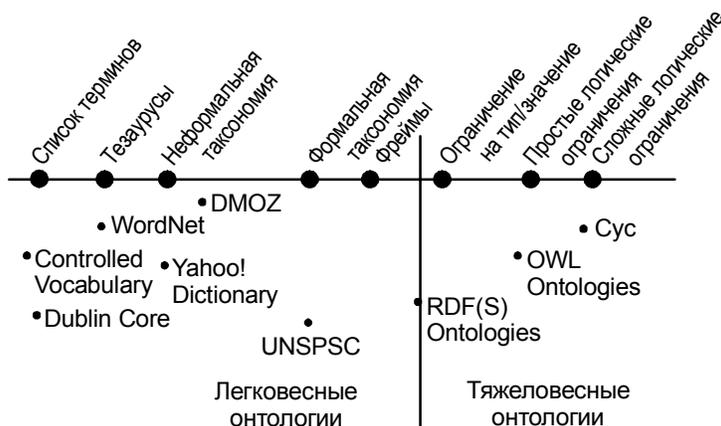


Рис. 4.5. Спектр выразительности онтологий

В качестве примера онтологии предметных областей приведем краткое описание онтологии предприятия «The Enterprise Ontology» (далее EO), разработанную Эденбургским университетом совместно с такими партнерами как IBM, Lloyd's Register и др. [Uschold M., King M., 1998; TOVE Ontology Project].

Целью создания EO было обеспечение предприятию возможности успешно справляться с быстро изменяющейся внешней средой. Основным средством для достижения этой цели признано совершенствование бизнес-планирования на основе моделирования, улучшения коммуникаций и интеграции информационных и бизнес-процессов.

Интеграция предполагает получение различных точек зрения на ситуацию и процессы, объединение задач для их использования взаимно увязанных информационных ресурсов и универсальных методов моделирования процессов и решения задач.

Коммуникация между сотрудниками предприятия, между различными функциями и задачами обеспечивает комплексное использование информации, а также согласование модели и алгоритмов действия.

Гибкость призвана обеспечивать адаптацию предприятия к изменяющимся целям и состояниям внешней среды путем предоставления сотрудникам выбора в способах реализации бизнес-процессов.

В более общем смысле, предназначением EO является оказание действенной помощи в области эффективного обмена информацией и знанием между различными пользователями, задачами и системами.

Формы EO, отражающие и некоторые этапы ее создания (предыдущие версии), приведены на рис. 4.6.

EO содержит версию на естественном языке Informal EO, формальную версию Formal EO, описанную с помощью языковых средств системы Ontolingua, предыдущие версии Pre – Implementation EO, а также конечную версию Implemented EO, состоящую из двух основных частей: Capability ontology и Knowledge Space ontology.

Важным элементом EO, как впрочем и любой другой онтологии, являются термины. Каждый термин вводится в онтологию путем тщательного отбора. Определение каждого термина является необходимым и достаточным настолько, насколько это возможно на естественном языке. Термин определяется с помощью базовых слов вместе с грамматическими вариациями (например, ДОСТИГАТЬ, ДОСТИЖЕНИЕ (ACHIEVE, ACHIEVEMENT)).

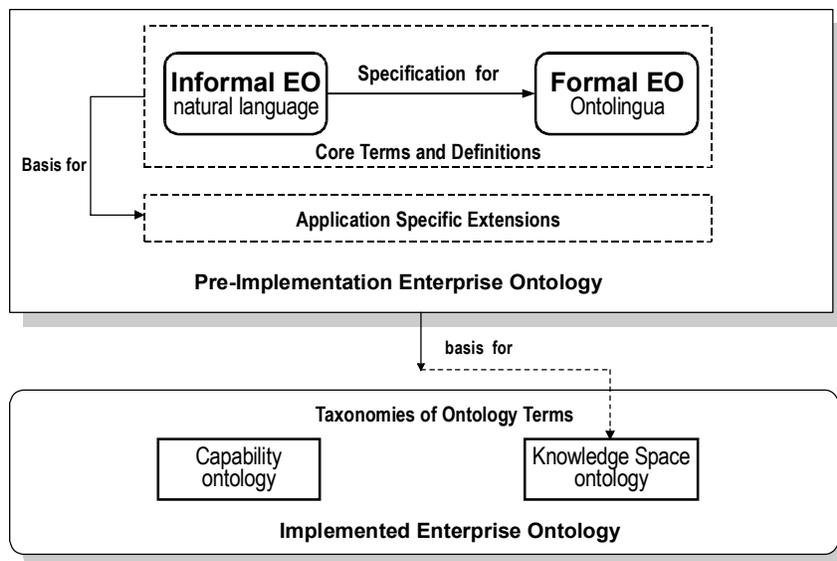


Рис. 4.6. Формы ЕО

Любой официальный термин представлен в ЕО большими буквами. Термины в Мета-Онтологии только начинаются с больших букв, далее следуют малые (например, Деятельность, а не ДЕЯТЕЛЬНОСТЬ).

В ЕО могут перечисляться и некоторые связанные термины: синонимы (Synonyms), близкие термины (Borderline Terms) и др. В ЕО включены лишь синонимы, причем лишь те из них, которые широко используются на предприятиях. Используемый в ЕО язык Ontolingua имеет удобный механизм для определения синонимов.

В целом онтология ЕО состоит из следующих секций:

- Meta Ontology and Time (Мета-Онтология и Время);
- Activity, Plan, Capability, Resource (Деятельность, План, Способность, Ресурс);
- Organisation (Организация);
- Strategy (Стратегия);
- Marketing (Маркетинг).

В табл. 4.1 содержатся основные понятия ЕО, упорядоченные по вышеупомянутым секциям.

Таблица 4.1

Основные понятия ЕО, сгруппированные по секциям

ДЕЯТЕЛЬНОСТЬ и др.	ОРГАНИЗАЦИЯ	СТРАТЕГИЯ	МАРКЕТИНГ	ВРЕМЯ
Деятельность	Человек	Цель	Продажа	Временная Линия
Спецификация Деятельности	Машина	Удерживаемая Цель	Потенциальная Продажа	Временной Интервал
Выполнять	Корпорация	Намечаемая Цель	Для Продажи	Временная Точка
Спецификация Выполняемой Деятельности	Партнерство	Держатель Цели	Предложения по Продаже	
Время Начала	Партнер	Стратегическая Цель	Продавец	
Время Конца	Юридическое Лицо	Стремление	Реальный Покупатель	
ПредУсловие	Организационная единица	Видение	Потенциальный Покупатель	
Эффект	Управление	Миссия	Покупатель	
Исполнитель	Уполномочивать	Цель	Посредник	
Под-Активность	Управленческая Связь	Help Achieve	Продукция	
Полномочие	Законный Владелец	Стратегия	Запрашиваемая Цена	
Владелец Деятельности	Не Законный Владелец	Стратегическое Планирование	Цена Продажи	
Событие	Собственность	Стратегическое Действие	Рынок	
План	Владелец	Решение	Сегментная Переменная	
Под-План	Актив	Предположение	Сегмент Рынка	
Планирование	Заинтересованное Лицо (Stakeholder)	Критическое Предположение	Исследование Рынка	
Спецификация Процесса	Контракт Найма	Не Критическое предположение	Бренд	
Способность	Акция	Фактор Влияния	Имидж	
Мастерство	Акционер	Критический фактор Влияния	Особенность	
Ресурс		Не критический фактор Влияния	Потребность	
Размещение Ресурса		Критический фактор Успеха	Потребность Рынка	
Заменитель Ресурса		Риск	Продвижение	
			Конкурент	

Связи и отношения между приведенными и другими понятиями также содержатся в описании ЕО, однако нами не приводятся, так же как структура и состав Мета-Онтологий. В качестве иллюстрации рассмотрим связи терминов и понятий в секции Деятельность и Процессы.

Центральным термином здесь является ДЕЯТЕЛЬНОСТЬ (ACTIVITY). Он предназначен для фиксации обозначения чего-либо, что включает действительное выполнение, в частности действие. ДЕЯТЕЛЬНОСТЬ может происходить в прошлом, а может происходить и в настоящем. Термин может также использоваться для ссылки на гипотетическую будущую ДЕЯТЕЛЬНОСТЬ. Однако существует потребность ссылаться явно на спецификацию (детальное описание) или план ДЕЯТЕЛЬНОСТИ. Это называется СПЕЦИФИКАЦИЕЙ ДЕЯТЕЛЬНОСТИ (ACTIVITY SPECIFICATION). Подобно рецепту, спецификация определяет на некотором уровне детальности одну или более возможных ДЕЯТЕЛЬНОСТЕЙ. СПЕЦИФИКАЦИЯ ВЫПОЛНЯЕМОЙ ДЕЯТЕЛЬНОСТИ должна иметь соответствующую ДЕЯТЕЛЬНОСТЬ, которую объект выполняет.

Понятие ДЕЯТЕЛЬНОСТЬ близко связано с понятием ИСПОЛНИТЕЛЬ, который ВЫПОЛНЯЕТ СПЕЦИФИКАЦИЮ ДЕЯТЕЛЬНОСТИ путем выполнения детально определенных (специфицированных) ДЕЯТЕЛЬНОСТЕЙ. ИСПОЛНИТЕЛЬ может быть ЧЕЛОВЕКОМ, ОРГАНИЗАЦИОННОЙ ЕДИНИЦЕЙ или МАШИНОЙ. Эти термины определены в секции Организация, и на них можно ссылаться как на ПОТЕНЦИАЛЬНЫХ АКТЕРОВ. Способность ПОТЕНЦИАЛЬНОГО АКТЕРА быть ИСПОЛНИТЕЛЕМ ДЕЯТЕЛЬНОСТИ обозначается СПОСОБНОСТЬЮ (или МАСТЕРСТВОМ, если ИСПОЛНИТЕЛЬ – это ЧЕЛОВЕК). АКТЕРЫ могут иметь другие Роли по отношению к ДЕЯТЕЛЬНОСТИ, такую, как ВЛАДЕЛЕЦ ДЕЯТЕЛЬНОСТИ.

Также тесно связанной с ДЕЯТЕЛЬНОСТЬЮ является РЕСУРС, который является чем-то, что может использоваться или потребляться в ДЕЯТЕЛЬНОСТИ. ДЕЯТЕЛЬНОСТЬ также имеет результаты или ЭФФЕКТЫ. ДЕЯТЕЛЬНОСТЬ связана с ВРЕМЕННЫМ ИНТЕРВАЛОМ. ДЕЯТЕЛЬНОСТЬ может занимать короткое или долгое время и может быть простой или сложной. Сложная ДЕЯТЕЛЬНОСТЬ может разделяться на ПОД-ДЕЯТЕЛЬНОСТИ.

СПЕЦИФИКАЦИЯ ДЕЯТЕЛЬНОСТИ с НАМЕЧАЕМОЙ ЦЕЛЬЮ называется ПЛАНом. Понятие «быть способным многократно ВЫПОЛНЯТЬ тот же самый ПЛАН» зафиксировано в термине СПЕЦИ-

ФИКАЦИЯ ПРОЦЕССА. Контроль выполнения ДЕЯТЕЛЬНОСТИ очень важен для предприятий. Для этого описано ПОЛНОМОЧИЕ быть вправе (Актеру) выполнять одну или более ДЕЯТЕЛЬНОСТЬ (так как это определено в ПЛАНЕ).

4.5. Метаописания

Метаописания (описания об описаниях) – это особо структурированная информация, характеризующая содержание документов, информационных ресурсов и баз знаний, профилей компетенции специалистов и т.п., которая может быть полезна как пользователям, так и самой системе управления знаниями. Метаописания отражают различные свойства и характеристики объекта, такие, как статус, формат, семантика и др.

Разделение описания объекта на информацию и метаописание – процесс не однозначный и зависит от целей описания. То, что может быть метаописаниями для одних целей, может являться частью содержания (информации) для других.

Процесс создания метаописаний иногда именуют аннотированием. Аннотирование может происходить как с участием человека, так и без него, с помощью специальных программно реализованных алгоритмов. Результатом аннотирования является набор метаописаний, который может помещаться в хранилище метаописаний.

В метаописаниях выделяют три типа:

- Системные (служебные) метаданные.
- Структурные метаданные.
- Семантические метаописания.

Системные метаданные предназначены для функционирования информационных систем и систем управления знаниями. Они включают имена файлов и баз, даты их создания, тип и формат, размер файла и вид носителя и т.п.

Структурные метаданные содержат, как правило, справочную информацию об объектах. Это могут быть наименование, статус, структурная принадлежность, профиль и т.п. То есть описания, использующиеся при идентификации и категоризации объектов в тех или иных целях.

Семантические метаописания – особый вид описаний, включающий концептуальное (аннотированное) изложение содержания и смысла информации об объекте.

С учетом изложенного, имеет место следующая иерархия метаописаний (рис. 4.7):



Рис. 4.7. Иерархия метаописаний

В практике использования информационных ресурсов и знаний метаописания весьма полезны.

Добавление метаданных к электронным ресурсам системы создает возможность более точного определения местоположения информации об объектах, улучшает механизм фильтрации и отбора знаний, упрощает и ускоряет процессы доступа к необходимым программам, серверам, ресурсам дисковой памяти и т.п.

Использование метаописаний в международных информационных сетях, в частности многоязыковых метаописаний, позволяет существенно повысить оперативность и эффективность поиска необходимых объектов и информации о них. Объединение метаописаний со ссылками на документы, журналы, БД и БЗ широко культивируется в настоящее время как в научных областях, так и во многих областях практической деятельности. Так, например, издателями ведущих стран мира создана Служба справочных ссылок издателей, обеспечивающая перекрестный доступ к издательским серверам, на основе действующих лицензионных соглашений, что резко повысило результативность использования поисковых систем сети Интернет.

Трудно переоценить роль метаописаний в библиотечном деле, в учебном процессе всех видов образовательных систем, включая электронные системы дистанционного обучения в части оперативности и полноты доступа к знаниям.

Семантические метаописания являются частью онтологии, при построении которой максимально учитывается семантика объектов некоторой предметной области.

При этом семантические метаописания отражают не всю семантику объекта, так же как онтология не может охватить всю описывающую ее предметную область (рис. 4.8).

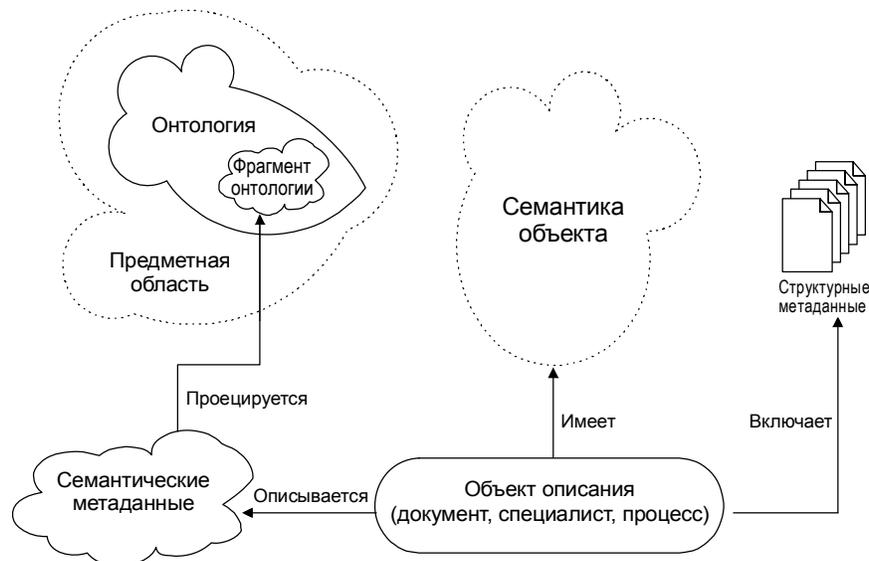


Рис. 4.8. Отражение смысла объекта в семантических метаописаниях

Из этой связи вытекают определенные требования к метаописаниям, такие, как:

- полнота представления знаний, закрепленных в онтологии;
- возможность использования метаописаний объектов в других системах;
- использование общепринятого стандарта, воспринимаемого другими системами.

Под метаданными объекта O_i будет пониматься следующее выражение:

$$MD = C_i \cup I_i,$$

где C_i – множество понятий онтологии O , имеющих отношение к объекту i , содержащихся в информации об объекте (документы, базы данных и знаний и т.п.) и в интересе пользователя. С каждым понятием связан свой весовой коэффициент K_i ; I_i – множество экземпляров понятий онтологии O с экземплярами отношений между ними.

Исходя из этого, а также из формальной модели онтологии, для совокупности объектов некоторой предметной области можно следующим образом определить структуру метаданных (метаописаний) [Maedche A., Staab S., 2002]:

$$MD = \{O, I, L, \text{inst.c}, \text{inst.r}, \text{inst.l}\},$$

состоящую из:

- онтологии, включающей C и R ;
- набора I , элементы которого являются экземплярами идентификаторов;
- набора значений литералов L ;
- функции $\text{inst.c}: C \rightarrow 2^I$, именуемой конкретизацией понятий (concept inst);
- функции $\text{inst.r}: R \rightarrow 2^{I \times I}$, именуемой конкретизацией отношений (concept inst);
- функции $\text{inst.l}: R \rightarrow 2^{I \times L}$, именуемой конкретизацией атрибутов понятий, которая связывает экземпляры онтологии со значениями литералов.

Конкретизация понятия заключается в определении соответствия между понятием и экземпляром. Результатом конкретизации может быть не полностью определенное понятие, то есть само понятие без значений атрибутов. У неопределенного понятия в качестве литерала будет использоваться степень связи понятия с описанием объекта $[0, 1]$.

4.6. Измерение близости объектов в интеллектуальном пространстве

Формализованное представление онтологий, а также метаописаний объектов создает возможность для измерения близости (подобия) объектов в интеллектуальном пространстве.

Например, подобие между метаданными $\text{Sim}(MD_i, MD_j)$ может быть определено через подобие входящих в них экземпляров понятий:

$$\text{Sim}(\text{MD}_i, \text{MD}_j) = \sum_{I_i \in \text{MD}_i} \sum_{I_j \in \text{MD}_j} \text{sim}(I_i, I_j),$$

где $\text{Sim}(\text{MD}_i, \text{MD}_j)$ – величина близости метаописания объекта i и объекта j ; $\text{sim}(I_i, I_j)$ – величина близости экземпляров понятий I_i и I_j , входящих в сравниваемые метаописания.

Можно выделить следующие составляющие измерения подобия двух экземпляров понятий:

- 1) таксономическое (по близости в иерархии онтологии, $\text{TS}(I_i, I_j)$);
- 2) реляционное (по сходству отношений экземпляров, $\text{RS}(I_i, I_j)$);
- 3) атрибутивное (по близости значений атрибутов, $\text{AS}(I_i, I_j)$).

Общая величина подобия $\text{sim}(I_1, I_2)$ экземпляров I_1 и I_2 определяется формулой

$$\text{sim}(I_i, I_j) = \frac{t \times \text{TS}(I_i, I_j) + r \times \text{RS}(I_i, I_j) + a \times \text{AS}(I_i, I_j)}{t + r + a},$$

где t , r , a – веса различных измерений подобия, которые могут быть подобраны в зависимости от важности учета разных измерений подобия, например в [Maedche A., Staab S., 2002] используется вес, равный 2, для реляционного подобия, так как наиболее важная часть информации онтологии и связанных метаданных содержалась в отношениях.

Оценить подобие можно разными способами. Рассмотрим один из предлагаемых вариантов их вычисления [Maedche A., Zacharias V., 2002].

Таксономическое подобие (близость)

Таксономическое подобие между экземплярами I_i и I_j , такими, что $C_i(I_i)$ и $C_j(I_j)$, вычисляется с учетом положения соответствующих им понятий C_i и C_j в таксономии H^C . Для вычисления семантического расстояния в иерархии понятий используется множество UC (upwards cotory), которое содержит все вышележащие по иерархии H^C понятия и само исследуемое понятие:

$$UC(C_i, H^C) = \{C_j \in C \mid H^C(C_i, C_j) \vee C_j = C_i\}.$$

Используются семантические характеристики H^C : рассмотрение ограничивается суперпонятиями заданного понятия C_i и рефлексивным взаимоотношением C_i к самому себе. Основываясь на определении UC , можно следующим образом определить таксономическое подобие:

$$\text{TS}(I_i, I_j) = \frac{|UC(C_i, H^C) \cap UC(C_j, H^C)|}{|UC(C_i, H^C) \cup UC(C_j, H^C)|}.$$

Реляционное подобие (подобие отношений)

Вычисление реляционного подобия основывается на предположении, что если два экземпляра имеют одни и те же отношения с третьим экземпляром, то они с большей вероятностью подобны, чем два экземпляра, которые имеют отношения с совершенно различными экземплярами. Поэтому подобие двух экземпляров зависит от подобия экземпляров, с которыми они имеют отношения. Подобие экземпляров, с которыми есть отношения, также вычисляется с помощью рассмотренного выше таксономического подобия. Пусть сравниваются два экземпляра I_1 и I_2 , $I_1, I_2 \in I$. Из определения онтологии известно, что существует набор отношений P_1 , которые позволяют использовать экземпляр I_1 или как домен (domain, I_1 входит в область понятий, для которых допустимо использование отношения), или как интервал (range, I_1 входит в область допустимых значений отношения), или как и то и другое. Аналогично, существует набор P_2 для I_2 . Только пересечение $P_{co} = P_1 \cap P_2$ будет интересным для оценки подобия отношений, так как различия между P_1 и P_2 определяются таксономическими отношениями, которые уже учтены таксономическим подобием. Набор P_{co} отношений разделяется на набор отношений P_{co-I} , допускающих экземпляры I_1 и I_2 в интервалы допустимых значений, и P_{co-O} , которые допускают экземпляры I_1 и I_2 в качестве домена.

Для дальнейших пояснений определим иерархию транзитивно связанных понятий H^{trans} , а также множества входящих $P_{co-I}(I_i)$ и исходящих отношений $P_{co-O}(I_i)$ следующим образом:

- $H^{trans} := \{(a, b) : (\exists a_1 \dots a_n \in C : H^c(a, a_1) \dots H^c(a_n, b))\}$,
- $P_{co-I}(I_i) := \{R : R \in P \wedge ((C(I_i), range(R)) \in H^{trans})\}$ (это такие отношения онтологии, которые являются отношениями между понятием экземпляра с их допустимыми значениями, которые входят в транзитивную иерархию);
- $P_{co-O}(I_i) := \{R : R \in P \wedge ((C(I_i), domain(R)) \in H^{trans})\}$ (это такие отношения онтологии, которые являются отношениями между понятием экземпляра с областями их применения, которые входят в транзитивную иерархию).

Тогда множества входящих и исходящих отношений пар экземпляров (I_i, I_j) определяются следующим образом:

- $P_{co-I}(I_i, I_j) := P_{co-I}(I_i) \cap P_{co-I}(I_j)$ (пересечение множеств входящих отношений экземпляров);
- $P_{co-O}(I_i, I_j) := P_{co-O}(I_i) \cap P_{co-O}(I_j)$ (пересечение множеств исходящих отношений экземпляров).

Далее будем рассматривать только P_{co-0} , но все сказанное может применяться также и к P_{co-1} .

Прежде чем продолжить, отметим: для заданной онтологии с отношением P_x существует минимальное подобие, больше 0, между любыми двумя экземплярами, которые являются источником или целью отношения экземпляров – $MinSim_s(P_x)$ и $MinSim_t(P_x)$. (Интервал и домен, определяющие понятие и любые два экземпляра этого понятия или одного из его подпонятий, будут иметь таксономическое подобие больше чем 0.) Игнорирование этого будет увеличивать подобие двух экземпляров, у которых заданы отношения с совершенно различными экземплярами, в сравнении с двумя экземплярами, для которых просто не определены такие отношения. Особенно это создает проблему при работе с отсутствующими значениями. Для каждого отношения $P_n \in P_{co-0}$ и каждого экземпляра I_i существует набор отношений экземпляра $P_n(I_i, I_x)$. Экземпляры I_x называются ассоциированными экземплярами A_s :

$$A_s(P, I) = \{I_x : I_x \in I \wedge P_n(I, I_x)\}.$$

Таким образом, задача сравнения экземпляров I_i и I_j относительно отношения P_n будет сводиться к сравнению $A_s(P_n, I_i)$ с $A_s(P_n, I_j)$. Это делается путем определения подобия по одному отношению следующим образом:

$$OR(I_i, I_j, P) = \begin{cases} MinSim_{t(P)}, & \text{если } A_s(P, I_i) = 0 \vee A_s(P, I_j) = 0, \\ \frac{\sum_{a \in A_s(P, I_i)} \max\{\text{sim}(a, b) | b \in A_s(P, I_j)\}}{|A_s(P, I_i) = 0|}, & \text{если } |A_s(P, I_i) = 0| \geq |A_s(P, I_j) = 0|, \\ \frac{\sum_{a \in A_s(P, I_j)} \max\{\text{sim}(a, b) | b \in A_s(P, I_i)\}}{|A_s(P, I_j) = 0|}, & \text{иначе.} \end{cases}$$

И, наконец, результаты для всех $P_n \in P_{co-0}$ и $P_n \in P_{co-1}$ объединяются путем вычисления их арифметического среднего для определения реляционного подобия:

$$RS(I_i, I_j) = \frac{\sum_{p \in P_{co-1}} OR((I_i, I_j, p)) + \sum_{p \in P_{co-0}} OR((I_i, I_j, p))}{|P_{co-1}| + |P_{co-0}|}.$$

Последней оставшейся проблемой является рекурсивная природа процесса вычисления подобий, что может приводить к бесконечным циклам. Но она может быть легко решена заданием максимальной глубины для рекурсии. После достижения этого максимума глубины возвращается арифметическое среднее таксономического и атрибутивного подобия.

Атрибутивное подобие (Attribute Similarity)

Атрибутивное подобие основывается на подобии значений атрибутов для определения подобия между экземплярами. Так как атрибуты очень сходны с отношениями (например, в RDF атрибуты являются отношениями с интервалами (область допустимых значений), которые содержат литералы), то большая часть того, что было сказано об отношениях, также применимо и здесь.

Для вычисления атрибутивного подобия вначале определим набор сравниваемых атрибутов для двух экземпляров:

$$P_{A_i}(I_i) := \{A : A \in A\},$$

$$P_A(I_i, I_j) := P_{A_i}(I_i) \cap P_{A_j}(I_j),$$

а также значения их атрибутов:

$$A_s(A, I_i) := \{L_x : L_x \in L \wedge A(I_i, L_x)\}.$$

Только члены наборов A_s , определенные ранее, являются не экземплярами, а литералами, и требуется новый метод оценки подобия для сравнения литералов. Так как атрибуты могут быть именами, датами рождения, количеством жителей страны, доходом и т.п., то сравнивать их разумным способом очень трудно. Можно попытаться разобрать значения атрибутов на известные типы данных (например, даты и числа) и выполнять сравнение, используя разобранные значения. (Для простых строковых типов данных можно использовать подобие строк: редакторское расстояние (edit distance), введенное Levenshtein [Levenshtein, 1966], является общепринятым методом для взвешивания разницы двух строк. Оно измеряет минимальное количество вставок, удалений и замен текстовых частей (token), которое требуется сделать для преобразования одной строки в другую, с помощью алгоритма динамического программирования. Например, редакторское расстояние (ed) между двумя лексическими единицами «ГенеральныйДиректор» и «Генеральный_Директор» равно 1, ed («ГенеральныйДиректор», «Гене-

ральный_Директор») = 1, так как только одна операция вставки меняет строку «ГенеральныйДиректор» на «Генеральный_Директор».)

Если невозможно разобрать все значения специфических атрибутов, то они игнорируются. Но даже если сравниваются числа, преобразование числовой разницы в меру подобия $[0,1]$ может быть затруднительным. Например, при сравнении атрибута, содержащего население страны, разница на 4 должна давать подобие, очень близкое к 1, а при сравнении атрибута «среднее количество детей у женщин» та же самая числовая разница должна дать меру подобия, равную 0. Для того, чтобы учитывать это, вначале находится максимальная разница между значениями этого атрибута, а затем вычисляется подобие как

$$1 - (\text{Difference} / \text{MaxDifference}).$$

Подобие литералов может быть определено как

$$\text{slsim}(A_1, A_2) \rightarrow [0, 1],$$

$$\text{mlsim}(A) = \max \{ \text{slsim}(A_1, A_2) : A_1 \in A \wedge A_2 \in A \},$$

$$\text{lsim}(A_1, A_2, A) = \frac{\text{slsim}(A_1, A_2)}{\text{mlsim}(A)}.$$

Следует отметить, что, в отличие от отношений, минимальное подобие при сравнении атрибутов всегда равно 0.

Подобие для одного атрибута равно

$$\text{OA}(I_1, I_2, A) = \begin{cases} 0, & \text{если } (A_s(A, I_1) = 0) \vee (A_s(A, I_2) = 0), \\ \frac{\sum_{a \in A_s(A, I_1)} \max \{ \text{lsim}(a, b, A) \mid b \in A_s(A, I_2) \}}{|(A_s(A, I_1))|}, & \text{если } |A_s(A, I_1)| \geq |A_s(A, I_2)|, \\ \frac{\sum_{a \in A_s(A, I_2)} \max \{ \text{lsim}(a, b, A) \mid b \in A_s(A, I_1) \}}{|(A_s(A, I_1))|}, & \text{если } |A_s(A, I_1)| < |A_s(A, I_2)|. \end{cases}$$

И, наконец, подобие атрибутов вычисляется следующей формулой:

$$\text{AS}(I_i, I_j) = \frac{\sum_{a \in P_{A(I_i, I_j)}} \text{OA}(I_i, I_j, a)}{|P_{A(I_i, I_j)}|}.$$

4.7. Дескриптивные логики как формальные модели онтологий

Построение, использование и эволюция онтологий в значительной степени зависит от их семантических свойств и инструментов логического вывода. Важное значение имеют при этом формальные модели описания онтологий и языки представления онтологий, которые бы поддерживали эту модель. Идеальными кандидатами для построения формальных моделей онтологий являются дескриптивные логики (ДЛ) [Baader F. et al., 2003].

ДЛ являются семейством языков представления знаний (ПЗ), которые могут быть использованы для записи знаний предметной области (домена) формальным способом. ДЛ описывают знания прикладной проблемной области (домена), вначале вводя подходящие понятия домена (его терминологию), а затем используют эти понятия для точного описания свойств объектов и индивидуумов (экземпляров), встречающихся в домене (описание мира).

Ключевой особенностью ДЛ, отличающей их от предшественников, таких, как семантические сети и фреймы, является то, что это логики, т.е. формальные языки с хорошо определенной семантикой.

Другой их отличительной особенностью является упор на логический вывод как основной способ использования. Логический вывод позволяет выявить неявно представленные знания из знаний, которые явно содержатся в базе знаний. ДЛ поддерживают шаблоны логического вывода, которые встречаются во многих вариантах интеллектуальных систем обработки информации и также используются людьми для структурирования и понимания мира: классификация понятий и их представителей (индивидуумов).

Так как ДЛ являются формализмами ПЗ и так как в ПЗ обычно предполагается, что система ПЗ всегда должна отвечать на запросы пользователей за разумное время, исследователи процедур логического вывода (ЛВ) в ДЛ заинтересованы в процедурах принятия решений (*decision procedures*). В отличие, например, от алгоритмов доказательства теорем логик первого порядка (ЛПП), эти процедуры должны всегда завершаться, как при положительных, так и при отрицательных ответах. Так как гарантия ответа в ограниченное время не предполагает, что ответ будет дан через приемлемый интервал времени, то исследование вычислительной сложности в конкретной ДЛ с разрешимыми задачами логического вывода является важной проблемой. Разрешимость и слож-

ность задач вывода зависит от выразительной силы конкретной ДЛ. С одной стороны, очень экспрессивные ДЛ сталкиваются с задачами вывода высокой сложности, которые могут быть даже неразрешимыми. С другой стороны, очень слабые ДЛ (с эффективными процедурами вывода) могут не обладать достаточной выразительностью для представления важных понятий конкретной прикладной задачи. Определение баланса между выразительностью ДЛ и сложностью их задач вывода является одной из наиболее важных проблем в исследованиях ДЛ.

В дескриптивных логиках:

- Синтаксическими строительными блоками являются атомарные понятия (унарные предикаты), атомарные роли (бинарные предикаты) и представители (индивидуумы, константы). Роли (свойства, отношения) являются самостоятельными элементами, которые затем могут быть связаны с понятиями.
- Понятия и роли можно объединяться в выражения для описания более сложных понятий с помощью конструкторов (операций).
- Между понятиями и между ролями можно задать отношения (какие понятия (роли) являются тождественными, и какие понятия (роли) включают другие понятия (роли)).
- Выразительная мощь языка ограничивается тем, что он использует достаточно малый набор конструкторов для построения сложных понятий и ролей.
- Неявные знания о понятиях и индивидуумах могут быть выведены из явных автоматически с помощью процедур вывода. В частности, важную роль играют отношения включения (родовидовые отношения, *subsumption relationships*) между понятиями и отношениями экземпляров.

ЛПП в общем случае является неразрешимой, т.е. нельзя в общем случае гарантированно выполнить логический вывод (автоматически доказать теорему). Задача гарантированного получения результата – это задача о разрешимости. ДЛ можно рассматривать как область разрешимых частей логики первого порядка.

4.7.1. Базовые формализмы ДЛ

ДЛ представляют знания о предметной области в виде баз знаний (БЗ). БЗ состоит из двух компонент: ТВох и АВох.

- ТВох описывает (содержит) *терминологию*, т.е. словарь интересующей предметной области (домена);

- АBox содержит *утверждения* о конкретных представителях (individuals) в терминах этого словаря.

Пример терминологии TBox (описаний онтологии) семейных отношений показан на рис. 4.9, а описание конкретных экземпляров предметной области (ABox) показан на рис. 4.10:

Женщина \equiv Человек \sqcap женскийРод
 Мужчина \equiv Человек \sqcap \neg женскийРод
 Мать \equiv Женщина \sqcap \exists иметьДетей.Человек
 Отец \equiv Мужчина \sqcap \exists иметьДетей.Человек
 Родитель \equiv Мать \sqcup Отец
 Бабушка \equiv Мать \sqcap \exists иметьДетей.Родитель
 МногодетнаяМать \equiv Мать \sqcap ≥ 3 иметьДетей
 МатьБезДочери \equiv Мать \sqcap \forall иметьДетей. \neg Женщина
 Жена \equiv Женщина \sqcap \exists иметьМужа.Мужчина

Рис. 4.9. Терминология (TBox) понятий о семейных отношениях

МатьБезДочери (МАРИЯ) Отец (ПЕТР)
 иметьДетей (МАРИЯ, ПЕТР) иметьДетей (ПЕТР, ИВАН)
 иметьДетей (МАРИЯ, СЕМЕН)

Рис. 4.10. Описание мира (ABox)

Можно считать, что TBox соответствует онтологии предметной области, а ABox состоит из метаданных описываемых объектов системы.

Словарь (TBox, онтология) включает понятия (классы, concept), которые обозначают некоторые термины предметной области (классы элементов, понятия), и роли (отношения, свойства, role), которые обозначают бинарные отношения между различными элементами. В дополнение к атомарным понятиям и ролям (имена понятий и ролей) все ДЛ позволяют строить описания сложных понятий и ролей.

ABox (метаданные) может использоваться для назначения имен сложным описаниям (descriptions).

Язык для построения описаний является характерной особенностью каждой системы ДЛ, и различные системы различаются их языком описания. Дескриптивный язык имеет теоретико-модельную семантику. Поэтому утверждения в TBox и в ABox могут отождествляться с формулами в ЛПП или, в некоторых случаях, незначительными их расширениями.

Системы ДЛ позволяют не только описать терминологию и утверждения, но также предоставляют возможности по выполнению логического вывода с их помощью.

Основными задачами логического вывода о терминологии (ТВох) являются:

- определение выполнимости (*satisfiable*) описания (т.е. не противоречивости);
- определение того, что одно описание (*description*) является более общим, чем другое, т.е. что одно включает в виде категории другое (*subsumes*).

Важными задачами для АВох являются:

- определение непротиворечивости (*consistent*) набора содержащихся утверждений, т.е. имеет ли она модель;
- следует ли из утверждений, содержащихся в АВох, заключение о том, что конкретный представитель является экземпляром заданного описания понятия.

4.7.2. Дескриптивный язык

Элементарными описаниями ДЛ являются *атомарные понятия* и *атомарные роли*. На их основе с помощью *понятийных (концептуальных) конструкторов (concept constructors)* могут быть построены сложные описания. В абстрактном обозначении используют буквы А и В для атомарных понятий, букву R для атомарных ролей и буквы С и D для описания сложных понятий. Для обозначения всеобщего, универсального понятия, т.е. понятия, которое включает все другие понятия, используется знак Т. Пустое понятие обозначается как \perp (табл. 4.2).

Т а б л и ц а 4.2

Обозначение	Описание
А, В	Атомарные понятия, которые не определяются через другие понятия
С, D	Сложные понятия, которые могут быть определены через другие понятия
R, S	Роли, отношения между двумя элементами (понятиями, ролями, экземплярами)
Т	Всеобщее, универсальное понятие, т.е. понятие, которое включает все другие понятия
\perp	Пустое понятие

В описаниях логик могут использовать следующие основные конструкторы для понятий и ролей (табл. 4.3 и 4.4).

Таблица 4.3

Некоторые понятийные конструкторы дескриптивной логики

№ п/п	Название	Синтаксис	Семантика	Символ
1	Вершина (Top)	\top	Δ^I	AL
2	Основание (Bottom)	\perp	\emptyset	AL
3	Пересечение	$C \sqcap D$	$C^I \cap D^I$	AL
4	Объединение	$C \sqcup D$	$C^I \cup D^I$	U
5	Отрицание	$\neg C$	$\Delta^I \setminus C^I$	C
6	Ограничение на значение	$\forall R.C$	$\{a \in \Delta^I \mid \forall b. (a,b) \in R^I \rightarrow b \in C^I\}$	AL
7	Квантификатор существования	$\exists R.C$	$\{a \in \Delta^I \mid \exists b. (a,b) \in R^I \wedge b \in C^I\}$	ϵ
8	Неуточненное (безусловное) числовое ограничение	$\geq n R$ $\leq n R$ $= n R$	$\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I\} \geq n\}$ $\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I\} \leq n\}$ $\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I\} = n\}$	N
9	Уточненное (условное) числовое ограничение	$\geq n R.C$ $\leq n R.C$ $= n R.C$	$\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} \geq n\}$ $\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} \leq n\}$ $\{a \in \Delta^I \mid \{b \in \Delta^I \mid (a,b) \in R^I \wedge b \in C^I\} = n\}$	Q
10	Именной	I	$I^I \in \Delta^I \text{ с } I^I = 1$	O

Таблица 4.4

Некоторые ролевые конструкторы дескриптивной логики

№ п/п	Название	Синтаксис	Семантика
1	Универсальная роль (Universal role)	U	$\Delta^I \times \Delta^I$
2	Пересечение (Intersection)	$R \sqcap S$	$R^I \cap S^I$
3	Объединение (Union)	$R \sqcup S$	$R^I \cup S^I$
4	Дополнение (Complement)	$\neg R$	$\Delta^I \times \Delta^I \setminus R$
5	Инверсия (Inverse)	R^-	$\{(b,a) \in \Delta^I \times \Delta^I \mid (a,b) \in R^I\}$
6	Транзитивное замыкание	R^+	$\bigcup_{n \geq 1} (R^I)^n$

Различные дескриптивные языки различаются используемыми наборами конструкторов для понятий и ролей, а также ограничениями на интерпретацию ролей. Минимальным языком, имеющим практический интерес, является язык AL (атрибутивный язык, *attributive language*). Все другие языки семейства ДЛик являются расширениями языка AL.

4.7.3. Базовый дескриптивный язык AL

Описание понятий в AL образуется в соответствии со следующим синтаксическим правилом построения сложных выражений C или D:

$$C, D \rightarrow A \mid \top \mid \perp \mid \neg A \mid C \sqcap D \mid \forall R.C \mid \exists R. \top$$

Здесь знак \rightarrow обозначает операцию подстановки, т.е. вместо C или D можно записать выражения, стоящие справа, знак \mid обозначает операцию «или» (или A, или, \top или ...)

Отметим, что в AL отрицание может быть применено только к атомарным понятиям, и только универсальное понятие разрешено в квантификаторе существования после роли. По историческим причинам усеченный (*sublanguage*) язык AL, полученный путем запрещения атомарного отрицания (*atomic negation*), называется FL и усеченный язык FL⁻, полученный запрещением ограниченного квантификатора существования называется FL₀.

Для того, чтобы дать пример, что может быть выражено на языке AL, предположим, что Человек и женскийПол являются атомарными понятиями. Тогда

Человек \sqcap женскийПол и Человек \sqcap \neg женскийПол – это понятия языка AL, описывающие, интуитивно, тех людей, которые являются женщинами, и тех, которые не являются женщинами. Если, дополнительно, мы полагаем, что иметьРебенка, является атомарной ролью, то можно формировать следующие понятия:

Человек \sqcap ЭиметьДетей. \top

и Человек \sqcap \forall иметьДетей.Женщина,

обозначающее тех людей, кто имеет детей, и тех людей, все дети которых являются девочками.

Используя пустое понятие (*bottom concept*), можно описать людей, которые не имеют детей, понятием Человек \sqcap \forall иметьДетей. \perp .

Семантика ДЛ обычно задается с использованием теории Тарского [Tarski A., 1956].

Для определения формальной семантики понятий языка AL рассматриваются интерпретация $I = (\Delta^I, \bullet^I)$, которая состоит из не пустого набора Δ^I (домен интерпретации) и функции интерпретации \bullet^I , которая назначает атомарному понятию A множество $A^I \subseteq \Delta^I$ и каждой атомарной роли R бинарное отношение $R^I \subseteq \Delta^I \times \Delta^I$. Функция интерпретации расширяется до описаний понятий с помощью следующих индуктивных описаний:

$$\begin{aligned} \top^I &= \Delta^I \\ \perp^I &= \emptyset \\ (\neg A)^I &= \Delta^I \setminus A^I \\ (C \sqcap D)^I &= C^I \cap D^I \\ (\forall R.C)^I &= \{a \in \Delta^I \mid \forall b. (a, b) \in R^I \rightarrow b \in C^I\} \\ (\exists R.C)^I &= \{a \in \Delta^I \mid \exists b. (a, b) \in R^I\}. \end{aligned}$$

Два понятия C и D считаются эквивалентными, $C \equiv D$, если $C^I = D^I$ для всех интерпретаций I . Например, возвращаясь к определению семантики понятий, легко проверить, что $\forall \text{иметьДетей.Женщина} \sqcap \forall \text{иметьДетей.Студент}$ и $\forall \text{иметьДетей.}(Женщина \sqcap \text{Студент})$ эквивалентны.

4.7.4. Терминологии

Для задания отношений о том, как роли и понятия связаны друг с другом, используются *терминологические аксиомы*. В наиболее общем случае терминологические аксиомы имеют следующий вид:

$$C \sqsubseteq D \ (R \sqsubseteq S) \ \text{или} \ C \equiv D \ (R \equiv S),$$

где C, D – это понятия (и R, S – это роли). Аксиомы первого вида называются включениями (*inclusions*), а аксиомы второго вида – равенствами (*equalities*). Для упрощения описания в дальнейшем используются только аксиомы, включающие понятия.

Семантика аксиом определяется достаточно очевидным способом. Интерпретация I удовлетворяет включению (inclusion) $C \sqsubseteq D$, если $C^I \sqsubseteq D^I$, и она удовлетворяет равенству $C \equiv D$, если $C^I = D^I$. Если T – это набор аксиом, то тогда I удовлетворяет T , если и только если I удовлетворяет каждому элементу T . Если интерпретация I удовлетворяет аксиоме (соответственно, набору аксиом), то тогда говорится, что I является моде-

лью этой аксиомы (соответственно, набора аксиом). Две аксиомы или набора аксиом *эквивалентны*, если они имеют одинаковые модели.

Равенство, в котором левая часть является понятием, – это определение (*definition*). Определения используются для введения символических имен для сложных определений. Например, с помощью аксиомы

$$\text{Мать} \equiv \text{Женщина} \sqcap \exists \text{иметь} \text{Детей} \cdot \text{Человек}$$

связывается описание, расположенное с правой стороны с именем Мать. Символическое имя может использоваться как сокращение в других описаниях. Если, например, определить понятие Отец так же, как и Мать, то можем определить понятие Родитель следующим образом:

$$\text{Родитель} \equiv \text{Мать} \sqcup \text{Отец}.$$

Конечный набор определений T называется **терминологией** или $TBox$, если символические имена определяются не более чем один раз, т.е. если для каждого атомарного понятия A существует не более чем одна аксиома в T , в левой части которой стоит A .

Если T – терминология, то можно атомарные понятия, встречающиеся в T , разделить на два набора: *именные символы* (*name symbols*) N_T , которые встречаются с левой стороны некоторых аксиом, и *базовые символы* B_T , которые встречаются только с правой стороны аксиом. Именные символы часто называются определенными понятиями, а базовые символы примитивными понятиями. Считается, что терминология определяет именные символы в терминах базовых символов, которые теперь становятся более точными.

4.7.5. Обозначение дескриптивных логик

Имеется несколько возможностей по расширению языка AL для того, чтобы получить более выразительные (экспрессивные) ДЛ. Наиболее известными методами являются три:

1. Добавление дополнительных конструкторов для понятий.
2. Добавление конструкторов для ролей.
3. Формулирование ограничений на интерпретацию ролей.

Для этих расширений определена схема формирования имен. Обычно, каждому расширению назначается буква или символ:

- для понятийных конструкторов буквы/символы записываются после начальных символов AL ;
- для ролевых конструкторов пишутся буквы/символы в качестве верхних индексов;

- для ограничений на интерпретацию ролей пишутся буквы/символы в виде нижних индексов.
Например, ДЛ $ALCQ^{-1}_{R^+}$ расширяет язык AL
- понятийным конструктором отрицания (C) и уточненными числовыми ограничениями (Q),
- ролевым конструктором инверсии ($^{-1}$) и
- ограничением, что некоторые роли являются транзитивными ($_{R^+}$).

Для того, чтобы избежать очень длинных имен выразительных языков ДЛ, была введена аббревиатура S вместо записи наименования языка ALC_{R^+} . Т.е. S – это базовый язык (AL), в который добавлено отрицание для сложных понятий (C) и транзитивные роли ($_{R^+}$).

Известными представителями семейства языков S являются:

- SIN – расширение ALC_{R^+} числовыми ограничениями и инверсными ролями;
- SHIF – расширение ALC_{R^+} иерархией ролей, инверсными ролями и числовыми ограничениями в форме $\leq 1 R$;
- SHIQ – расширение иерархией ролей, инверсными ролями и уточненными числовыми ограничениями в форме $\leq 1 R$.

Кроме этого, к обозначениям используется дополнение (D), для того, чтобы показать, что в данной ДЛ используются некоторые стандартные типы данных.

4.7.6. Логический вывод

ДЛ предоставляют не только средства для описания терминов и экземпляров (онтологий и метаданных), но также и возможность выполнять рассуждения различного вида. БЗ, включающие компоненты TBox и ABox, имеют семантику, которая делает ее эквивалентной набору аксиом в логике предикатов первого порядка. Поэтому, подобно любому другому набору аксиом, она содержит неявные (implicit) знания, которые могут быть сделаны явными с помощью логического вывода. Например, из TBox на рис. 4.9 и ABox на рис. 4.10 можно сделать заключение, что Мария является Бабушкой, хотя это знание не записано явно, в виде утверждения.

Различные виды рассуждений, выполняемые в ДЛ, определяются как логический вывод. Оказывается, что разные задачи логического вывода сводимы друг к другу и решение одной задачи позволяет решить другую. Рассуждения могут выполняться с TBox и ABox. Далее будет рассматриваться только логический вывод над понятиями.

4.7.7. Задачи логического вывода

При моделировании предметной области создается терминология T , определяющая новые понятия, в терминах других понятий, которые были определены ранее. В ходе этого процесса важно выяснить, имеет ли смысл вновь определенное понятие или оно противоречит тому, что уже было сказано. С логической точки зрения, понятие имеет смысл, если в некоторой существующей интерпретации, которая удовлетворяет аксиомам T (модели T), оно обозначает не пустой набор. Говорят, что понятие с таким свойством выполнимо (*satisfiable*) по отношению к T и невыполнимо (*unsatisfiable*) иначе.

Проверка выполнимости понятий является основной задачей вывода. Множество других важных задач вывода для понятий может быть сведено до проверки выполнимости.

Например, для того чтобы проверить, является ли модель домена корректной (или для оптимизации запросов, которые формулируются как понятия), требуется узнать, является ли одно понятие более общим, чем другое: это задача определения включения (*subsumption problem*). Понятие C включается (*subsumed*) понятием D , если в каждой модели T набор, обозначенный C , является подмножеством множества, обозначенного D . Алгоритмы, которые проверяют включения, также используются для построения таксономии понятий, входящих в компонент $TBox$. Другими важными отношениями между понятиями являются эквивалентность (*equivalence*) и непересекаемость (*disjointness*). Эти свойства формально определяются следующим образом. Пусть T – это $TBox$.

Выполнимость (Satisfiability): Понятие C является выполнимым по отношению к T , если существует модель I для T , такая, что C^I не пустое множество. В этом случае мы также говорим, что I – это модель C .

Включение (Subsumption): Понятие C включается в понятие (*subsumed*) D по отношению к T , если $C^I \subseteq D^I$ для каждой модели I для T . В этом случае мы можем записать $C \sqsubseteq_T D$ или $T \models C \sqsubseteq D$.

Эквивалентность (Equivalence): Два понятия C и D являются эквивалентными по отношению к T , если $C^I = D^I$ для каждой модели I для T . В этом случае мы пишем $C \equiv_T D$ или $T \models C \equiv D$.

Непересекаемость (Disjointness): Два понятия C и D не пересекаются (*disjoint*) по отношению к T , если $C^I \cap D^I = \emptyset$ для каждой модели I для T .

По отношению к ТВох на рис. 4.9 Человек включает понятие Женщина, и как Женщина, так и Родитель включают понятие Мать, а Мать включает понятие Бабушка (как мать, у которой ребенок тоже мать). Более того, Женщина и Мужчина, Отец и Мать являются непересекающимися (disjoint) понятиями. Отношения включения (subsumption) следуют из определений, т.е. семантики конструкторов «П» и «Л». То, что понятие Мужчина не пересекается с понятием Женщина, следует из того факта, что Мужчина является категорией отрицания Женщина.

Традиционно, системы, основанные на ДЛ, обеспечивают решение задачи включения (subsumption) понятий. Фактически, решение этой задачи достаточно для решения также других задач, т.е. все другие задачи сводятся к задаче включения.

Первые ДЛ использовали *структурные алгоритмы включения* (structural subsumption algorithms), которые вначале нормализовали описания понятий и затем рекурсивно сравнивали синтаксические структуры нормализованных определений. Эти алгоритмы обычно очень эффективны (полиномиальны), но они имеют тот недостаток, что являются полными для очень невыразительных ДЛ, т.е. для более выразительных ДЛ они не могут обнаружить все существующие отношения включения/экземпляра (subsumption/instance).

Позже был предложен новый подход к разработке алгоритмов, так называемые алгоритмы, основанные на таблицах (tableau-based algorithms). Для того, чтобы определить согласованность БЗ, алгоритм, основанный на таблицах (tableau-based algorithm), пытается построить ее модель путем разбиения понятий в БЗ, накладывая, таким образом, новые ограничения на элементы этой модели. Алгоритм останавливается из-за неудачи всех попыток построить модель, которые приводят к явным противоречиям или к получению «канонической» («canonical») модели.

В последнее десятилетие были разработаны эффективные процедуры логического вывода для очень выразительных ДЛ (например, SHIQ), основанных или на табличном подходе (tableau-approach), или на трансляции в модальные логики. Высокооптимизированные системы (Fact, Racer и Dlp) [Haarslev V., Moller R., 2001; Patel-Schneider P.F., Horrocks I., 1999] показали, что алгоритм, основанный на таблицах (tableau-based algorithm), для выразительных ДЛ показывает хорошее практическое поведение системы даже на больших БЗ.

Глава 5

ЯЗЫКИ ОПИСАНИЯ ЗНАНИЙ

Языки представления знаний (Knowledge Representation Language), или языки описания онтологий (Ontology Language), имеют весьма важное значение в управлении знаниями. От степени выразительности, точности и универсальности языка представления знаний во многом зависит полезность онтологии как инструмента оперирования с информационными ресурсами и знаниями. С учетом концептуального, структурного и прикладного разнообразия онтологий естественным является выдвижение ряда общих требований к такого рода языкам.

5.1. Требования к онтологическим языкам

К числу основных требований к онтологическим языкам (ОЯ) можно отнести следующие:

- ОЯ должен иметь четко определенный синтаксис, что делает его машиночитаемым.
- ОЯ должен быть эпистимологически адекватным, то есть быть способным выразить все понятия и отношения в любом заданном домене.
- ОЯ должен обладать достаточной выразительной силой, чтобы быть пригодным для описания множества предметных областей.
- ОЯ должен обладать строгостью кодирования для однозначной машинной интерпретации понятий, связей, аксиом и иных знаниевых сущностей.
- ОЯ должен основываться на логике. В зависимости от уровня выразительности онтологии ОЯ может основываться на логике первого порядка, дескриптивной логике и др.
- Формальность ОЯ должна соответствовать формальности онтологии.

- Современные ОЯ должны представлять определенные семантические возможности, основанные на продукционных и фреймовых моделях дескриптивной логики.
- ОЯ должен в ряде случаев предоставлять возможность выполнять логический вывод, основываясь на структурных компонентах языка.
- ОЯ должен обладать возможностью быть использованным для концептуального моделирования.
- ОЯ должен обладать функциональной полнотой, необходимой для решения различных задач при операциях с онтологиями, а именно: построение и совместное использование онтологий, обмен знаниями, взаимодействие с онтологиями.
- Наконец, немаловажным требованием к ОЯ является недвусмысленное понимание семантики при совместном использовании онтологий, а также однозначная интерпретация смысла понятий и знаниевых сущностей не только создателями онтологии, но и внешними агентами (пользователями, программами и т.п.).

Языки описания онтологий появились в начале 90-х годов в составе ряда известных систем разработки и управления онтологиями. Ярким и широко известным проектом такого рода явилось создание системы Ontolingua [Gruber T.R., 1993]. Используемый в этой системе язык представления знаний KIF (Knowledge Interchange Format) обеспечивал не только внутренние потребности этой онтологической системы, но и взаимодействовал с другими языками, такими, как KL-ONE, LOOM и CLASSIC. Современные языки онтологий основываются на дескриптивной логике [Baader F., Horrocks I., 2003], в то время как KIF основывается на логике первого порядка. Язык KIF использовался, кроме того, и в таких важных прикладных онтологических проектах, как TOVE (Toronto Virtual Enterprise) [TOVE Ontology Project] и The Enterprise Ontology [Uschold M., King M., 1998].

В конце 90-х годов онтологии и соответствующие им языки (SHOE, F-Logic) стали применяться в WWW для аннотирования HTML-документов [Heflin J., Hendler J., 2003; Kifer M., Lausen G., 1989]. В 1999 г. был разработан язык RDF [W3C, «Resource Description Framework (RDF)»] для аннотирования Web-страниц метаданными, которые можно обрабатывать автоматически. RDF и его последующие семантические расширения содержат множество базовых примитивов для моделирования тяжеловесных онтологий, содержащих классы и свойства.

Последовавшие затем разработки в области языков описания онтологий (DAML-ONT, DAM-OIL и др.), а также объективные потребности мегапроекта Семантическая Сеть (Semantic Web) привели к созданию языка OWL (Web Ontology Language), который рассматривается в World Wide Web Consortium (W3C) как новый стандарт для Семантической Сети [W3C, «Web Ontology Language (OWL) Reference Version 1.0»].

Далее перейдем к рассмотрению современных языков, используемых для описания онтологий и метаданных. Из множества разработанных и используемых языков остановимся на тех из них, которые определены в качестве стандартов организацией W3C при разработке систем управления знаниями. Именно они находят широкое применение, что в сочетании с их свойствами обеспечивает предпосылки для информационной и программной совместимости различных систем.

В данной главе рассмотрены следующие языки:

1. Расширяемый язык разметки – XML (eXtensible Markup Language) [W3C, «Extensible Markup Language (XML)»], который предназначен для описания иерархической структуры информации с помощью линейного синтаксиса. Моделью данных этого языка является иерархия (дерево). В последние годы технологии, основанные на языке XML, переживают этап взрывного развития. Появляется множество продуктов, ориентированных на XML или поддерживающих его тем или иным образом. Вокруг XML формируются разнообразные стандарты (DDT, XML Schema, XSLT, XPath, XPoint, XLink) [Дейтл Х.М., Дейтл П.Дж., 2001]. В настоящее время насчитывается более 450 стандартов, построенных на основе данного языка [Холман К., 2000]. Разработано большое количество программ, которые позволяют работать с документами на этом языке, – грамматические анализаторы языка (parsers). Средства для работы с XML встроены практически во все современные системы разработки программного обеспечения.

2. Язык структуры (шаблона) описания ресурсов – RDF (Resource Description Framework), предназначен для представления информации о ресурсах в сети WWW, но также используется для описания различных метаданных любых объектов (документов, пользователей и т.п.). Моделью данных этого языка является ориентированный граф, который соответствует модели представления знаний как многомерной сети понятий и отношений между ними. Существуют различные варианты тек-

стовой записи (сериализации) утверждений языка RDF, основным из которых является RDF/XML.

3. Язык описания RDF – RDF Vocabulary Description Language (кратко RDF Schema, или совсем кратко RDFS), предназначен для простых онтологий, которые описывают понятия и отношения, используемые для записи метаданных на языке RDF. Язык RDF Schema имеет ограниченные выразительные возможности и использует RDF для описания словарей понятий и отношений (свойств).

4. Язык Web-онтологий – OWL (Ontology Web Language), основан на дескриптивной логике и имеет большие выразительные возможности. Данный язык является расширением языка RDF Schema.

Логическая зависимость перечисленных языков приведена на рис. 5.1.

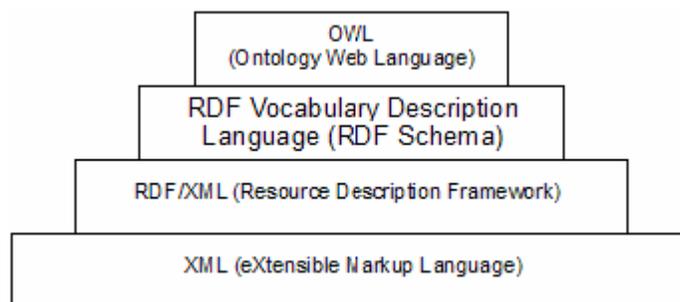


Рис. 5.1. Логическая зависимость ОЯ

5.2. Язык описания документов XML

XML (eXtensible Markup Language) – это основанный на метках (тэгах) язык для описания иерархической структуры с помощью линейного синтаксиса [Дейтл Х.М., Дейтл П.Дж., 2001; Холман К., 2000; W3C, «Extensible Markup Language (XML)»]. XML – это метаязык в том смысле, что он может использоваться для определения других языков, называемых XML-приложениями. XML основывается на простой идее представления документов в виде деревьев. К достоинствам языка XML относится его независимость от платформы, отделение содержания от формата, возможность определения соответствия документов структурным правилам. Он является синтаксически ограниченным подмножест-

вом более раннего и более сложного языка Standard Generalized Markup Language (SGML), стандарта используемого сообществом по обработке текстов. XML сохранил основные особенности языка SGML, но не включает те его особенности, которые редко используются или трудны для реализации.

Рассмотрим простой пример – набор данных, который имеет иерархическую структуру (рис. 5.2).



Рис. 5.2

Синтаксис XML имеет сходство с синтаксисом языка HTML (Hyper Text Markup Language), на основе которого создана система гипертекстовых документов WWW в сети Интернет. Это не удивительно, так как HTML также является приложением языка SGML, языка, на основе которого был создан XML. Документ, записанный на языке XML (так же как и HTML-документ), представляет собой обычный текст, в который включены специальные коды. Эти коды называются метками (или тэгами), и они записываются между угловыми скобками (знаки меньше < и больше >), например: <Организация> или <ОтделА>. Метки предоставляют дополнительную информацию о тексте и позволяют структурировать и уточнять текстовый документ. Однако в отличие от HTML, XML ничего не говорит о смысле этих тэгов. Поэтому тэг <P> в одном документе может обозначать параграф, а в другом – что-либо другое.

Пример версии XML-документа, описывающего структуру организации, приведен в листинге 5.1.

Как видно из листинга 5.1, XML-документ начинается со специальной инструкции <?xml version="1.0"?> (объявления XML). Она указывает версию (в данном случае version="1.0") и может содержать информацию о используемой системе кодирования символов.

Листинг 5.1

Пример XML-документа с описанием структуры организации

```
<?xml version="1.0"?>
<организация>
  <отделА штат=5>
    <сотрудник>
      <фио>ИвановАП</фио>
      <должность>завотдела</должность>
      <датаРождения>1965</датаРождения>
    </сотрудник>
    <сотрудник>
      <фио>ПетровСМ </фио>
      <должность>инженер</должность>
      <датаРождения>1970</датаРождения>
    </сотрудник>
  </отделА>
  <отделВ штат=12>
    <сотрудник>
      <фио>СидоровПС</фио>
      <должность>работник</должность>
      <датаРождения>1975</датаРождения>
    </сотрудник>
  </отделВ>
</организация>
```

Существует три вида тэгов в XML: начальный тэг, конечный тэг и тэг пустого элемента. Начальный тэг состоит из окруженного угловыми скобками имени и может включать атрибуты. Каждый атрибут состоит из пары имя/значение, разделенных знаком равенства (штат=12). Конечный тэг содержит имя ранее записанного начального тэга, но перед ним ставится символ наклонной вправо черты (слэш), и он не может содержать атрибуты. Каждый начальный тэг должен иметь только один соответствующий ему конечный тэг. Пустой элемент походит на начальный тэг, но не имеет соответствующий ему конечный тэг. Вместо этого, пустой элемент показывается символом слэш перед закрывающейся скобкой. Например, тэг <тип класс="станок" /> является тэгом пустого элемента.

Данные от начального до конечного тэга составляют *элемент*. Элемент может включать другие элементы, свободный текст или их комби-

нацию между его начальным и конечным тэгами. Правильно сформированный XML-документ содержит только один элемент верхнего уровня, но может иметь произвольное количество вложенных в него элементов. В вышеприведенном примере корневой элемент данного документа «организация» обозначен начальным <организация> и конечным </организация> тэгами. Все данные включены в элементы, вложенные в этот корневой элемент. Правильно сформированный XML-документ создает сбалансированное дерево вложенных наборов именованных элементов.

Одни и те же понятия на языке XML можно описать различными способами. Например, в листинге 5.2 показаны возможные представления имени человека. Одним из вариантов выбора является, будет ли информация о фамилии, имени и отчестве (ФИО) сотрудника строкой или элементом со своей собственной структурой. Другим вариантом является представление ФИО как элемента или как атрибута. Нет какого-то специального смысла в использовании атрибута или части содержания. Часть содержания может использоваться для описания свойств объекта или группы, которые связаны с элементом, в то время как атрибуты могут использоваться для задания дополнительной информации или единичных значений.

Листинг 5.2

Различные варианты описания одних и тех же данных

```
<!-- ФИО является элементом с текстовым содержанием -->
<сотрудник>
  <фio>Иванов Петр Семенович</фio>
</сотрудник>
<!-- ФИО является элементом, включающим другой элемент -->
<сотрудник>
  <фio>
    <фамилия>Иванов</фамилия>
    <имя>Петр</имя>
    <отчество>Семенович</отчество>
  </фio>
</сотрудник>
<!--ФИО является атрибутом элемента СОТРУДНИК -->
```

```
<сотрудник фιο="Иванов Петр Семенович">
```

Пример XML-документа показан в листинге 5.3. Корневой элемент «сотрудник», обозначен начальным <сотрудник> и конечным </сотрудник> метками (тэгами). Все данные включены в элементы, вложенные в этот корневой элемент. Если XML-документ из листинга 5.3 послан от одной программы другой, то программа, которая получает документ, может обработать его только в том случае, если ей известна схема документа (т.е. точная синтаксическая структура) и если она знает, как обрабатывать данные, включенные в документ (т.е. смысл элементов).

Листинг 5.3

XML-документ «сотрудник»

```
<сотрудник>
  <имя>ПетровИА</имя>
  <должность>
    заведующий отдела
  </должность>
  <датаРождения>
    1979-06-23
  </датаРождения >
</сотрудник>
```

Листинг 5.4

XML-документ «завотдела»

```
<завотдела>
  <имя> ПетровИА</имя>
  <датаРождения>
    23-6-1979
  </датаРождения>
</завотдела>
```

В листингах 5.3 и 5.4 можно видеть, что оба XML-документа описывают одну и ту же информацию. Данные в обоих документах имеют тот же самый смысл (т.е. ту же самую семантику), но способы его представления (синтаксис или структура) различаются. Например, там, где документ в листинге 5.4 использует элемент «завотдела» для обозначения сотрудника заведующего отделом, документ в листинге 5.3 использует элемент «сотрудник», вместе с элементом «должность», содержащим данные «заведующий отдела», для обозначения сотрудника заведующего отделом. Человек, читающий эти документы, может понять, что они оба представляют того же самого сотрудника. Компьютер, обрабатывающий документ, однако, не видит сходства между этими двумя документами, так как программа не имеет средств связывания имен в различных структурах.

Когда программа работает с XML-документом, то она видит его в линейном формате (сериализованным):

```
<?xml version="1.0"?> <организация> <отделА штат=12>  
<сотрудник> <фio> ИвановАП </фio> <должность>  
завотдела</должность> <датаРожения> 1965  
</датаРожения> </сотрудник> <сотрудник> <фio>  
ПетровСМ </фio> <должность> инженер </должность>  
<датаРождения> 1970 </датаРождения> </сотрудник>  
</отделА> <отделВ штат=12> <сотрудник> <фio>  
СидоровПС </фio> <должность> работник </должность>  
<датаРождения> 1975 </датаРождения> </сотрудник>  
</отделВ> </организация>
```

Но использование тэгов позволяет определить структуру и зависимости отдельных элементов.

Имена тэгов могут быть выбраны произвольно создателем XML-документа. Это означает, что XML-документ может быть понят читающим его человеком, если имена тэгов и структура выбраны автором обдуманно и читатель понимает язык, используемый автором (отметим, что компьютеры, в общем случае, не могут понять эти XML-документы, если им заранее не было дано точное указание, что делать с каждым тэгом). То есть XML-документ решает две проблемы: синтаксическую и семантическую (но семантическую только для человека). Но даже в этом случае может возникнуть неоднозначность, так как разные люди имеют разное понимание терминов (разные ассоциации).

Язык XML часто используется как формат обмена данными, и в этом случае все стороны, принимающие участие в обмене, должны согласиться на общую структуру для XML-документов (заданную с помощью языков DTD или XML Schema, как поясняется ниже). Другое возможное использование XML, как это сделано в Semantic Web, заключается в его использовании как языка сериализации для других языков. Это имеет преимущество в связи с существованием большого количества XML-грамматических анализаторов (parsers), которые могут повторно использоваться для новых языков, которые используют XML в качестве их языка сериализации.

Разработчики различных XML-приложений могут использовать совпадающие имена для атрибутов и элементов – словари. Часто одни приложения используют имена из других приложений. В этом случае, для того чтобы различать имена различных приложений, была разработана спецификация пространств имен (XML Namespace specification). В соответствии с этой спецификацией группе имен тэгов и атрибутов при-

ложения разработчики задают глобально-уникальное имя (URI), и полные имена всех элементов такой группы (пространства имен) состоят из имени пространства и краткого имени элемента (или атрибута). Обычно URI являются достаточно длинными последовательностями символов. Для удобства работы с ними именам пространств можно присваивать короткие префиксы и использовать эти префиксы, с именами разделяя префикс и короткое имя элемента символом ‘:’, <dc:creator>. Указание используемых пространств имен в XML-документе и задание им префиксов выполняется с помощью атрибута xmlns в одном из элементов (предпочтительно в корневом). Когда XML-документ разбирается, то все имена расширяются именами пространства и могут быть уникально идентифицированы XML-приложением.

```
<организация xmlns = «kms.cctpu.edu.ru/ontology»
              xmlns : dc =
http://purl.org/dc/elements/1.1/>
```

Как видно из примера, каждое объявление пространства имен задает короткое имя (например, префикс dc для URI <http://purl.org/dc/elements/1.1/>) для каждого фрагмента URI. Это короткое имя указывается перед именем тэгов <dc:author>.

Универсальным идентификатором ресурса URI может быть любая уникальная строка. Обычно в качестве URI используется универсальный указатель ресурсов URL, так как его уникальность гарантируется правилами назначения имен доменов в сети Интернет. Эти URL не разбираются синтаксическими анализаторами, они всего лишь представляют собой уникальную последовательность символов для различения имен, и ничего более. Указанные URL могут даже не существовать или не соответствовать действительности. Так как XML-документ формально структурирован с помощью тэгов, то возможно обрабатывать документы с помощью компьютера, если существует соглашение о структуре документов (используя DTD или XML Schema, как описано ниже). Однако компьютер не может «понимать» смысл XML-документа, так как тэги выбраны автором произвольно.

Следует еще раз отметить, что язык XML практически (кроме начальной инструкции <?xml version="1.0"?>, определения используемых пространств имен xmlns и обозначений тэгов (<’ и >’)) не вводит никаких терминов. Все термины определяются разработчиками специальных языков – XML-приложений.

5.2.1. Document Type Definitions (DTD)

Хотя гибкость XML позволяет пользователю описать произвольное содержание быстро и легко, эта легкость может быть проблематичной для компьютерной обработки. Так как язык XML не может выразить смысл тэгов, то большинство программ, обрабатывающих такие документы, требуют наборы тэгов, смысл которых согласуется некоторыми стандартами или соглашениями. Для того чтобы помочь программной обработке, XML позволяет определить грамматику тэгов. Эта информация содержится в определении типа документа (document type definition (DTD)) [Спенсер П., 1999], который точно описывает правильно записанные элементы, содержание этих элементов и какие атрибуты могут уточнять элемент. Здесь не приводятся детали языка DTD, которые достаточно сложны. Можно сказать, что они по существу определяют контекстно-независимую грамматику. XML-документ, который имеет связанный с ним DTD и соответствует правилам, определенным в нем, называется правильным (*valid*) XML-документом.

Конкретное описание на языке DTD определяет ограничения на структуру XML-документа, для того чтобы можно было проверить его правильность в соответствии с этим конкретным описанием. DTD описывает структуру класса XML-документов. Соглашение по использованию DTD-описаний между различными сторонами позволяет обмениваться документами, которые соответствуют этому DTD. Однако отметим, что только договоренности об особенностях DTD недостаточно для понимания всех правильных XML-документов, так как описывается только структура (разрешенные тэги), но не смысл этих тэгов. Простой DTD, описывающий класс XML-документов, к которому принадлежит документ, показанный в листинге 5.3, приведен в листинге 5.5.

Листинг 5.5

DTD для примера в листинге 5.3

```
<!ELEMENT сотрудник (имя, должность, датаРождения)>
<!ELEMENT фио (#PCDATA)>
<!ELEMENT должность (#PCDATA)>
<!ELEMENT датаРождения (#PCDATA)>
```

5.2.2. Язык XML Schema

Точно так же, как и DTD, XML Schema [W3C, «XML Schema Part 0: Primer Second Edition»] используется в качестве языка для описания классов XML-документов. Точно так же, как и DTD, структура XML-документа предписывается схемой. Основное различие между DTD и XML Schema заключается в том, что XML Schema имеет XML-синтаксис, широкий набор (простых) типов данных [W3C, «XML Schema Part 2: Datatypes Second Edition»] и возможность формировать сложные типы данных. Пример XML Schema схемы для класса XML-документов, к которым принадлежит документ, показанный в листинге 5.3, приведен в листинге 5.6.

Листинг 5.6

XML Schema для примера «сотрудник»

```
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="сотрудник">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="имя" type="xs:string"/>
        <xs:element name="должность" type="xs:string"/>
        <xs:element name="датаРождения" type="xs:date"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Из приведенного примера видно, что определение XML Schema является значительно более многословным, чем DTD для одного и того же класса документов. Если DTD требуется только 5 строк для задания структуры XML-документа, то XML Schema требуется 12 строк. Это происходит потому, что каждый документ с XML Schema также является XML-документом и XML имеет очень многословный синтаксис. Угловые скобки и необходимость в закрывающихся тэгах занимает в записях основное пространство. Далее, способ кодирования атрибутов, при-

нятый в XML, также очень многословный, так как каждый раз имя атрибута должно быть записано перед заданием его значения. Кроме этого, мы видим в документе, что кроме самих описаний XML-элементов имеется тэг «schema», который указывает, что это XML Schema, а не экземпляр XML-документа. Потребность в элементах «complexType» и «sequence» существует в связи с тем, что XML Schema обеспечивает намного более сложные описания структуры, чем DTD. Сложность языка схем XML Schema по сравнению с DTD и его выразительность являются связанными понятиями. Менее выразительный язык DTD является также менее сложным.

Атрибут «type» в листинге 5.6 обозначает тип данных, содержащихся в элементе. В примере показаны два типа данных, а именно xs:string и xs:date. Префикс xs: указывает, что используются XML Schema типы данных. Этот пример XML Schema уже более выразителен из-за ограничений типа данных xs:date на элемент «датаРождения», что требует дату, закодированную стандартным способом ('уууу-мм-дд'). Так как XML Schema также позволяет спецификацию многих типов ограничений, она была бы более подходящим кандидатом для языка онтологий, чем DTD. Основная проблема в том, что XML Schema описывает физическую структуру (синтаксис) XML-документа, а не смысл (семантику) данных.

5.3. Язык описания метаинформации RDF

RDF является языком описания метаинформации о ресурсах в WWW, отсюда и аббревиатура – Resource Description Framework (Архитектура Описания Ресурсов) [W3C, «Resource Description Framework (RDF)»]. Фактически RDF (или вернее RDF/XML) является специфическим XML-приложением с набором имен тэгов и атрибутов в пространстве имен «<http://www.w3.org/1999/02/22-rdf-syntax-ns#>», которое позволяет описывать с помощью языка XML размеченные ориентированные графы.

RDF состоит из двух частей: спецификации RDF-модели данных (ориентированные графы) и основанный на XML синтаксис записи этих моделей. Описание модели данных является основной частью спецификации. Преимуществом рассмотрения RDF-модели данных таким способом является ее независимость от конкретного синтаксиса.

Основными строительными элементами структур знаний RDF являются ресурсы (**resources**) и литералы (**literals**). Ресурсом может быть все, о чем можно что-либо утверждать (документы, Web-страницы, люди и т.п.), в то время как литерал является сущностью, которая описывается

своим именем, т.е. некоторое значение. Ресурсы являются уникально идентифицируемыми и представляются URI-ссылками. Между этими элементами задаются бинарные отношения. В RDF две сущности и бинарное отношение между ними (фактически ребро графа с вершинами) называются утверждением. При графическом представлении исходный элемент отношения называется субъектом (subject), сама помеченная дуга называется предикатом (predicate, также называемая свойством) этого утверждения и конечный элемент отношения называется объектом (object) утверждения. Субъект и предикат утверждения всегда являются ресурсами, в то время как объект может быть ресурсом или литералом.

Рассмотрим простой пример, показанный на рис. 5.3. Данный граф может быть представлен в виде перечня ребер (листинг 5.7). Каждое ребро записано в виде тройки (метка начальной вершины, метка ребра, метка конечной вершины). Это модель данных языка RDF.

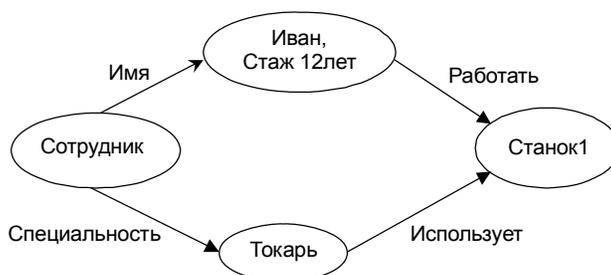


Рис. 5.3. Пример размеченного ориентированного графа

Листинг 5.7

Запись графа в виде описания ребер в форме троек

```

(сотрудник, имя, Иван)
(сотрудник, специальность, токарь)
(Иван, работать, станок1)
(токарь, использует, станок1)
  
```

RDF/XML-синтаксис [W3C, «RDF/XML Syntax Specification (Revised)»] позволяет записать граф в последовательной форме в виде XML-документа. Этот синтаксис достаточно гибок – он допускает различные формы записи одного и того же графа, различные сокращенные формы.

Описание графа, показанного на рис. 5.3, в виде XML-документа приведено в листинге 5.8.

Листинг 5.8

Запись графа в виде RDF-документа

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#"

xmlns:kms="http://kms.cctpu.edu.ru/ontology#">
  <rdf:Description about="kms:сотрудник">
    <kms:имя>kms:Иван</kms:имя>
  </rdf:Description>
  <rdf:Description about=kms:сотрудник>
    <kms:специальность rdf:resource = "kms:токарь" />
  </rdf:Description>
  <rdf:Description about=kms:Иван>
    <kms:работать rdf:resource = "
      http://kms.cctpu.edu.ru/
      ontology#Станок1" />
  <kms:стаж
rdf:datatype="http://www.w3.org/2001/XMLSchema#integ
er">
    12
  </ kms:стаж>
  </rdf:Description>
  <rdf:Description about=kms:Иван>
    <kms:работать rdf:resource = "kms:Станок1">
  </rdf:Description>
</rdf:RDF>
```

Метки <rdf:RDF> и </rdf:RDF> указывают, что их содержание представляет RDF-документ.

В метке rdf:RDF также выполняется определение используемого пространства имен XML, например:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#"

xmlns:kms="http://kms.cctpu.edu.ru/ontology/">
```

В общем виде запись ('объект', 'свойство', 'субъект') в RDF/XML может быть записана несколькими способами, основным из которых является следующий:

```
<rdf:Description rdf:about = объект>
  <свойство> субъект </свойство>
</rdf:Description>
```

В RDF можно использовать сокращенную запись для предопределенного свойства `rdf:type` (иметь тип, быть экземпляром класса). При использовании этого свойства вместо обычной записи

```
<rdf:Description rdf:about = объект>
  <rdf:type> класс </rdf:type>
</rdf:Description>
```

можно записать

```
<класс about = объект />
```

Это можно сделать только в том случае, если объект относится к одному классу.

Атрибут `rdf:ID` (или `rdf:about`, при этом `rdf:about="#x"` аналогично `rdf:ID="x"`) в метке `rdf:Description` обозначает субъект утверждения, имя тэга, вложенного элемента, обозначает свойство утверждения и значение атрибута. Атрибут `rdf:resource` обозначает объект утверждения. Альтернативно, объект может также быть вложенным элементом содержания или значением атрибута `rdf:ID` вложенного элемента. RDF предоставляет другие способы кодирования графов. Например, синтаксически возможно использовать идентификатор ресурса объекта вместо тэга `rdf:Description` для создания элементов фиксированной формы `rdf:type(subject,object)`. Например, мы можем закодировать в RDF/XML утверждение `rdf:type("Станок1",Оборудование)` следующим образом:

```
< Оборудование rdf:ID="Станок1" />
```

Утверждения могут вкладываться, что приводит к ресурсам без идентификаторов – анонимным ресурсам. Анонимные ресурсы создаются, если идентификатор ресурса опускается в утверждении RDF/XML-описания.

Основными элементами пространства имен `rdf` являются:

- `rdf:RDF` – элемент, определяющий начало `rdf`-описания;
- `rdf:Description` – элемент, с которого начинается описание каждого ребра графа; в виде атрибута данного элемента задается идентифи-

катор описываемого ресурса (с помощью атрибутов `rdf:about` или `rdf:ID`). В данный элемент включаются свойства и значения, которые относятся к указанному ресурсу;

- `rdf:about` – атрибут (для `rdf:Description`), указывает на ресурс, который будет описываться в элементе `rdf:Description`;
- `rdf:ID` – атрибут (для `rdf:Description`), указывает на ID ресурса, который будет описываться в элементе `rdf:Description`;
- `rdf:datatype` – атрибут в элементе свойства, который указывает, что значение свойства имеет тип (обычно один из типов XML Schema, например, `rdf:datatype="xsd:decimal"`, где `xsd` – префикс пространства имен XML Schema);
- `rdf:type` – предопределенное свойство, указывающее, что ресурс относится к определенному классу, заданному в онтологии. Для этого свойства есть сокращенная запись, в которой имя класса указывается вместо тэга `rdf:Description`;
- `rdf:Resource` – атрибут, который показывает, что значением свойства является другой ресурс, обозначаемый URI;
- `rdf:value` – предопределенное свойство, задает основное значение свойства, если значением свойства является структура;
- `rdf:parseType` – атрибут свойства, который указывает, что содержание элемента должно интерпретироваться специальным способом (например, как коллекция или структура).

Рассмотрим пример описания метаданных документа, которые показаны на рис. 5.4.

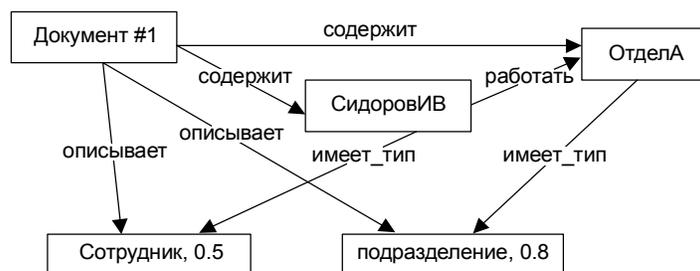


Рис. 5.4. Метаданные документа #1

Описание этих метаданных на языке RDF показано в листинге 5.9.

Листинг 5.9

Описание на языке RDF метаданных документа #1

```
<?xml version="1.0" encoding="windows-1251" ?>
<rdf:RDF xmlns:rdf=http://www.w3.org/1999/02/22-rdf-
syntax-ns#
  xmlns:http://www.w3.org/2001/XMLSchema#
  xmlns:kms="http://kms.cctpu.edu.ru/ontology">
  <rdf:Description rdf:about="#kms:Документ#1">
    <kms:описывает rdf:parseType="Resource">
      <rdf:value
rdf:datatype="&xsd;decimal">0.5</rdf:value>
      <kms:concept rdf:resource="kms#сотрудник"/>
    </kms:описывает>
  </rdf:Description>
  <rdf:Description rdf:about="#kms:Документ#1">
    <kms:описывает rdf:parseType="Resource">
      <rdf:value
rdf:datatype="&xsd;decimal">0.8</rdf:value>
      <kms:concept
rdf:resource="kms#подразделение"/>
    </kms:описывает>
  </rdf:Description>
  <rdf:Description rdf:about="#kms:Документ#1">
    <kms:содержит rdf:resource = "kms:СидоровИВ"/>
    <kms:содержит rdf:resource = "kms:ОтделА"/>
  </rdf:Description>
  <kms:сотрудник rdf:about="#kms:СидоровИВ">
    <kms:работает rdf:resource = "kms:ОтделА"/>
  </rdf:Description>
  <kms:подразделение rdf:about="#kms:ОтделА"/>
</rdf:RDF>
```

Здесь для задания весовых коэффициентов используются типы данных из XML Schema.

Язык RDF также имеет следующие дополнительные возможности:

- работа с контейнерами (rdf:Bag, rdf:Seq, rdf:Alt);
- работа с коллекциями (rdf:parseType="Collection");
- утверждения об утверждениях (реификация).

5.4. Язык описания онтологий RDFS

Как уже отмечалось, множество названий классов (понятий) и множество названий свойств (отношений) RDF-описаний составляют словарь конкретного документа на языке RDF. Если словарь RDF не известен, то смысл документа нельзя определить. Для программы названия классов являются бессмысленными строками. Требуется определять словари (наборы терминов), которые предполагается использовать в утверждениях языка, особенно для того, чтобы показать, что они описывают специальные виды или классы ресурсов и будут использовать специфические свойства в описаниях этих ресурсов. Фактически требуется некоторая онтология терминов, описывающих знания предметной области.

Язык RDF позволяет описывать метаданные, но для работы с ними требуется также описывать онтологию, на основе которой строятся метаданные и которая определяет связи между понятиями и отношениями, используемыми в метаданных. Для этих целей предлагается использовать язык RDF Schema (RDFS) [W3C, "RDF Vocabulary Description Language 1.0"].

Язык описания словаря RDF является простым языком моделирования над RDF. Язык RDFS описывает онтологию средствами RDF, предоставляя набор предопределённых ресурсов и свойств с обозначенной для них смысловой нагрузкой, которые могут быть использованы для описания новых RDF-словарей.

Средства самого языка RDF Schema предоставлены в форме словаря RDF, то есть как специализированный набор предопределённых RDF-ресурсов с их собственными специфическими смыслами. Эти ресурсы в словаре RDF Schema имеют ссылочный URI с префиксом <http://www.w3.org/2000/01/rdf-schema#> (обычно связываемый с кратким префиксом `rdfs:`). Описания словарей, записанные на языке RDF Schema, являются правильными RDF-графами.

RDFS предназначен для представления примитивов, которые требуются для описания словаря, используемого в конкретных RDF-моделях. Такое описание получается путем выражения набора объектов для расширений классов и свойств. Поэтому RDFS использует классы, родовидовые отношения (включения, *subsumption*) как над классами, так и над свойствами и глобальные ограничения на области применения (*domen*) и области значений (*range*). При сравнении с обычными языками объ-

ектно-ориентированного моделирования RDFS показывает несколько отличное понимание объектной ориентированности: RDFS рассматривает свойства как полноправные элементы, т.е. они могут существовать независимо от классов.

RDFS позволяет определить классы ресурсов и свойства как элементы словаря и специфицировать, какие свойства, с какими классами могут быть использованы. RDFS выражает эти словари средствами RDF, предоставляя набор предопределённых ресурсов и свойств с обозначенной для них смысловой нагрузкой, которые могут быть использованы для описания новых RDF-словарей. Таким образом, любое RDFS-описание представляет собой «обычные» RDF-данные – данные о классах и свойствах. RDFS позволяет определить уникальные (идентифицируемые URI) классы ресурсов, представляющие концептуальную модель конкретной предметной области и уникальные (идентифицируемые URI) свойства, интересующие нас в этой области.

Принадлежность ресурса к конкретному классу задается с помощью свойства `rdf:type`. Описываемые в словаре классы сами являются экземплярами предопределённого класса `rdfs:Class`, свойства же являются экземплярами `rdf:Property`. RDFS позволяет указать, каким классам присущи заданные свойства (`rdfs:domain`) и ресурсы какого класса могут появиться в качестве значения заданного свойства (`rdfs:range`). RDFS позволяет связать классы (`rdfs:Class`) отношениями множественного наследования (`rdfs:subClassOf`).

В RDFS-модели, как и в обычном объектном подходе, классам свойствен полиморфизм, то есть экземпляр подкласса всегда может сыграть роль экземпляра своего суперкласса и появиться как субъект или объект свойства, для которого в качестве соответственно `range` или `domain` был указан суперкласс. Свойства также могут быть связаны отношениями множественного наследования (`rdfs:subPropertyOf`). Наследование свойства означает более узкую специализацию этого свойства, уточнение смысла и сужение границ использования.

Иерархия основных понятий (тэгов) языка RDFS показана на рис. 5.5.

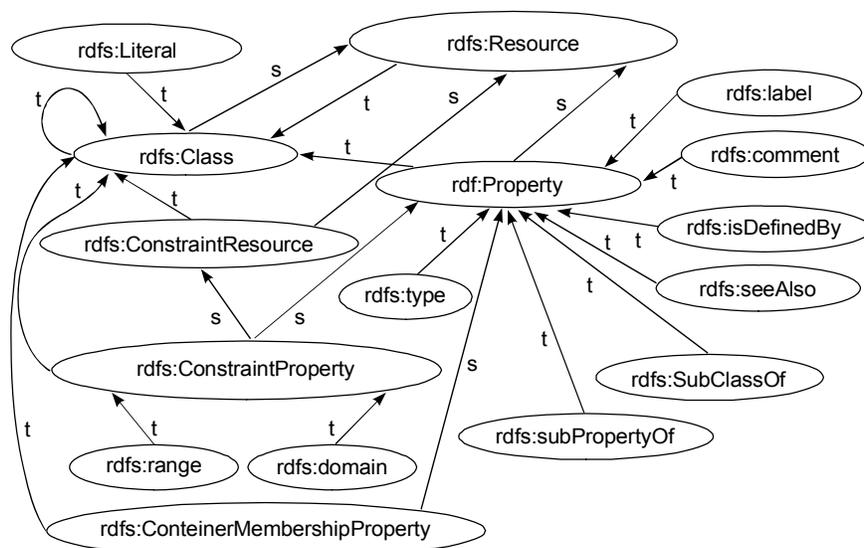


Рис. 5.5. Иерархия классов для RDF Schema
(здесь: s = rdfs:subClassOf; t = rdf:type)

Ниже приводится полный набор элементов rdf-описаний:

- rdfs:Class;
- rdfs:SubClassOf;
- свойство rdf:Property – класс для всех свойств;
- свойство rdfs:subPropertyOf – указывает, что свойство включается другим свойством;
- свойство rdfs:range и rdfs:domain – для задания ограничения на значения свойств;
- класс всех классов языка rdfs:Resource;
- свойство rdfs:label – для определения имен, удобных для человека;
- свойство rdfs:comment (например, fatherOf-property включается parentOf-property, так как любой отец также является родителем);
- свойства rdfs:seeAlso и rdfs:isDefinedBy используются для указания на связанные ресурсы;
- классы rdfs:ConstraintResource и rdfs:ConstraintProperty.

Рассмотрим пример описания простой онтологии (рис. 5.6). Вначале выполняется описание классов (понятий). При этом каждое понятие

описывается как экземпляр суперкласса `rdfs:Class` с помощью стандартного элемента `rdf:Rescription` и встроенного свойства `rdf:type`.

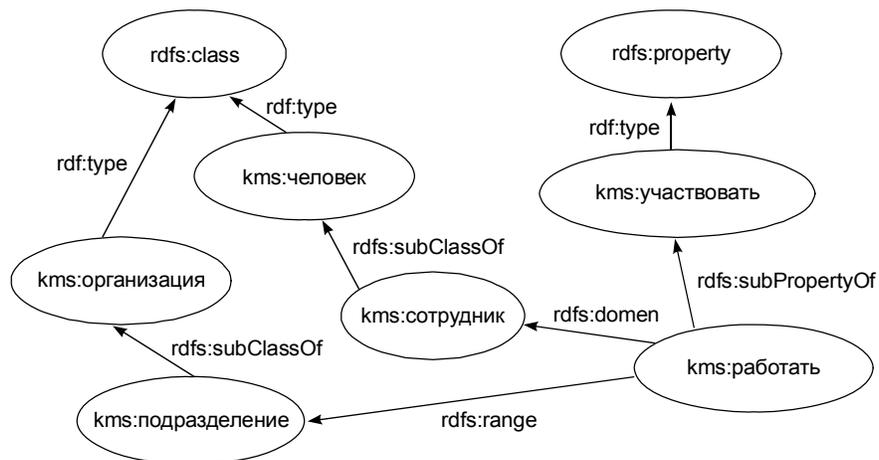


Рис. 5.6. Пример записи простой онтологии с использованием RDF Schema

Отношения наследования между классами задается с помощью элемента `rdfs:subClassOf`:

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  kms="http://kms.cctpu.edu.ru/schemas/">
  <rdf:Description rdf:ID="Организация">
    <rdf:type rdf:resource="#Class"/>
  </rdf:Description>
  <rdf:Description rdf:ID="Подразделение">
    <rdf:type rdf:resource="#Class"/>
    <rdfs:subClassOf rdf:resource="#Организация"/>
  </rdf:Description>
  . . .
</rdf:RDF>
```

Используя сокращенную запись для свойства `rdf:type`, это описание можно записать более компактно:

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://example.org/schemas/vehicles">
  <rdfs:Class rdf:ID="Организация"/>
  <rdfs:Class rdf:ID="Подразделение">
    <rdfs:subClassOf rdf:resource="#Организация"/>
  </rdfs:Class>
```

Свойства (отношения) онтологии определяются путем указания их типа – `rdf:type` и задания (если имеются) ограничений на их применимость в определенных классах (`rdfs:domain`) и возможные их значения (`rdfs:range`). Отношения между классами задаются с помощью встроеного свойства `rdfs:subPropertyOf`:

```
<rdf:Property rdf:ID="kms:участвовать"/>
<rdf:Property rdf:ID="kms:работать">
  <rdfs:domain rdf:resource="#Сотрудник"/>
  <rdfs:range rdf:resource="#Подразделение"/>
  <rdfs:subPropertyOf
rdf:resource="#kms:участвовать"/>
</rdf:Property>
```

RDFS предоставляет лишь базовые возможности для описания словарей предметных областей, но он легко может быть расширен дополнительными примитивами моделирования, более детально и специализированно описывающими нужные аспекты классов и свойств. Расширяемость позволяет RDFS служить фундаментом для более выразительных языков описания онтологий предметной области, которые позволяют адаптировать к Web системы логики и обеспечить семантическую обработку данных.

5.5. Онтологический язык OWL

Как отмечалось ранее, OWL (Ontology Web Language) основан на дескриптивной логике и имеет большие выразительные возможности [W3C, «OWL Web Ontology Language Overview»; Antoniou G., Frank van Harmelen, 2004]. В этом разделе будет описано как ядро OWL, его син-

таксис, свойства ограничения, так и различные его модификации (виды OWL).

5.5.1. Синтаксис языка

Язык OWL строится на основе языков RDF и RDFS и использует XML-синтаксис RDF (RDF/XML). Наряду с ним OWL может использовать и другие синтаксические формы OWL:

- Основанный на XML синтаксис, который не следует соглашениям RDF. Такой синтаксис уже является более легким для чтения пользователям.
- Абстрактный синтаксис, который используется в документе спецификации языка. Этот синтаксис является более компактным и читаемым, чем XML-синтаксис или RDF/XML-синтаксис.
- Графический синтаксис, основанный на соглашениях языка UML (Universal Modelling Language). Так как UML широко используется, то он облегчит пользователям знакомство с OWL.

Документы, написанные на языке OWL, обычно называются *OWL-онтологиями* и являются RDF-документами. Поэтому корневым элементом OWL-онтологии является `rdf:RDF`-элемент, который также определяет набор пространств имен. Например:

```
<rdf:RDF xmlns:owl
="http://www.w3.org/2002/07/owl#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#">
```

OWL-онтология может начинаться с набора утверждений, назначением которых является организация работы с онтологиями. Эти утверждения группируются под элементом `owl:Ontology`, который содержит комментарии, контроль версий и включение других онтологий.

Единственным утверждением, оказывающим влияние на смысл онтологии, является `owl:imports`, которое перечисляет другие онтологии, содержание которых будет частью текущего документа. Отметим, что в то время как задание имен пространств предназначено для ликвидации двусмысленностей, импортированные онтологии предоставляют определения, которые могут быть использованы. Обычно импортируется элемент для каждого используемого пространства имен, но также возможно им-

портировать дополнительные онтологии, например онтологии, которые предоставляют определения без введения каких-либо новых имен.

Отметим также, что *owl:imports* является транзитивным свойством: если онтология А импортирует онтологию В и онтология В импортирует онтологию С, то онтология А также импортирует онтологию С.

5.5.2. Элементы, определяющие классы

Классы определяются с помощью элемента *owl:Class* (этот элемент является подклассом *rdfs:Class*). Например, можно определить класс *associateProfessor* следующим образом:

```
<owl:Class rdf:ID="associateProfessor">
  <rdfs:subClassOf
    rdf:resource="#academicStaffMember"/>
</owl:Class>
```

Можно определить, что этот класс не пересекается с классами «professor» и «assistantProfessor» с помощью элементов *owl:disjointWith*. Эти элементы могут быть включены в записанное выше определение или добавлены с помощью ссылок на id класса с помощью *rdf:about*. Этот механизм наследуется от RDF.

```
<owl:Class rdf:about="associateProfessor">
  <owl:disjointWith rdf:resource="#professor"/>
  <owl:disjointWith
    rdf:resource="#assistantProfessor"/>
</owl:Class>
```

Эквивалентность классов может быть определена с помощью элемента *owl:equivalentClass*:

```
<owl:Class rdf:ID="faculty">
  <owl:equivalentClass
    rdf:resource="#academicStaffMember"/>
</owl:Class>
```

И, наконец, в языке имеется два предопределенных класса – *owl:Thing* (сущности) и *owl:Nothing* (ничего). Первый элемент (*owl:Thing*) является наиболее общим классом, который содержит все (все является сущностями), а второй (*owl:Nothing*) является пустым классом. В связи с

этим каждый класс является подклассом owl:Thing и суперклассом (т.е. включает его) owl:Nothing.

5.5.3. Элементы, определяющие свойства (отношения)

В OWL существует два типа свойств:

- Объектные свойства (object properties), которые задают отношения между объектами (связывают одни объекты с другими). Например, «обучатьсяКемТо», «руководитьКемТо» и т.п.
- Типовые свойства (datatype properties), которые задают отношения между объектами и значениями типов данных. Например, «номер телефона», «название», «возраст» и т.п. OWL не имеет каких-либо предопределенных типов данных и не предоставляет специальных возможностей их определения. Вместо этого он позволяет использовать типы данных, определенные в XML Schema, и таким образом использует многоуровневую архитектуру Semantic Web.

Ниже показан пример типового свойства:

```
<owl:DatatypeProperty rdf:ID="age">
<rdfs:range rdf:resource=
"http://www.w3.org/2001/XMLSchema#nonNegativeInteger
"/>
</owl:DatatypeProperty>
```

Для этих свойств задается тип возможных значений, в данном случае не отрицательные целые значения (<http://www.w3.org/2001/XMLSchema#nonNegativeInteger>). В качестве допустимых значений могут быть и определенные пользователями типы. Определенные пользователями типы данных обычно записываются в документе XML Schema и затем используются в OWL-онтологии.

Ниже показан пример объектного свойства:

```
<owl:ObjectProperty rdf:ID="isTaughtBy">
<owl:domain rdf:resource="#course"/>
<owl:range rdf:resource="#academicStaffMember"/>
<rdfs:subPropertyOf rdf:resource="#involves"/>
</owl:ObjectProperty>
```

Каждое объектное свойство может иметь следующие основные свойства (свойства у свойства):

- домен (owl:domain) – классы, у объектов которых имеется это свойство (область возможного применения);
- диапазон (owl:range) – классы, объекты которых могут быть значениями свойства (область допустимых значений);
- наследование (rdfs:subPropertyOf) – свойство, которое является базовым для описываемого свойства.

Можно объявить более чем один домен или диапазон. В этом случае используется пересечение доменов или диапазонов.

OWL позволяет связать «инверсные (противоположные) свойства». Типичным примером является «обучатьсяКемТо» (isTaughtBy) и «учитьКогоТо» (teaches):

```
<owl:ObjectProperty rdf:ID="teaches">
  <rdfs:range rdf:resource="#course"/>
  <rdfs:domain rdf:resource="#academicStaffMember"/>
  <owl:inverseOf rdf:resource="#isTaughtBy"/>
</owl:ObjectProperty>
```

В действительности домен и диапазон (domain и range) могут наследоваться от инверсного свойства (при этом они меняются местами).

Эквивалентность свойств может быть определена с помощью элемента owl:equivalentProperty:

```
<owl:ObjectProperty rdf:ID="lecturesIn">
  <owl:equivalentProperty rdf:resource="#teaches"/>
</owl:ObjectProperty>
```

5.5.4. Ограничение свойств

С помощью элемента rdfs:subClassOf можно указать, что класс *C* является подклассом другого класса *C'*. Тогда каждый экземпляр класса *C* также является экземпляром класса *C'*. В случае, если вместо этого требуется определить, что класс *C* удовлетворяет некоторым условиям (т.е. что все экземпляры класса *C* удовлетворяют этим условиям), то очевидно, что это эквивалентно определению, что *C* является подклассом класса *C'*, где *C'* включает все объекты, которые удовлетворяют заданным условиям.

Именно это делается в языке OWL, как будет показано далее. Отметим, что в общем случае *C'* может оставаться анонимным, как будет пояснено ниже.

В следующем примере определяется, что курсы первого года обучения студентов (first year courses) преподавались *только профессорами* (в соответствии со спорным мнением, что более старые и более старшие по званию преподаватели обучают лучше):

```
<owl:Class rdf:about="#firstYearCourse">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#isTaughtBy"/>
      <owl:allValuesFrom rdf:resource="#Professor"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Элемент owl:allValuesFrom используется для того, чтобы указать класс возможных значений, которые свойство, определенное в элементе owl:onProperty, может принимать (другими словами: все значения свойства должны быть элементами заданного класса). В данном примере только объекты класса «professor» разрешены в качестве значений свойства «isTaughtBy».

Объявить, что все математические курсы преподает David Billington, можно следующим образом:

```
<owl:Class rdf:about="#mathCourse">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#isTaughtBy"/>
      <owl:hasValue rdf:resource="#949352"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Элемент owl:hasValue задает конкретное значение, которое свойство, указанное owl:onProperty, должно иметь.

Также можно следующим образом описать, что все члены преподавательского состава (academic staff members) должны преподавать, по крайней мере, один курс студентам первых четырех лет обучения (undergraduate course):

```
<owl:Class rdf:about="#academicStaffMember">
  <rdfs:subClassOf>
    <owl:Restriction>
```

```

<owl:onProperty rdf:resource="#teaches"/>
<owl:someValuesFrom
rdf:resource="#undergraduateCourse"/>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

Сравним элементы owl:allValuesFrom и owl:someValuesFrom. Пример использования первого элемента (owl:allValuesFrom) требует, чтобы каждый человек, который преподает данный курс для первого года обучения, был профессором. В терминах логики это соответствует квантору всеобщности (universal quantifier).

Пример использования второго элемента (owl:someValuesFrom) требует, чтобы среди курсов первых четырех лет обучения существовал курс, который преподается членом преподавательского состава (academic staff member). Но также возможно, что тот же самый преподаватель обучал и студентов-магистрантов. В терминах логики это соответствует квантификатору существования (existential quantifier).

В общем случае элемент owl:Restriction содержит элемент owl:onProperty и одно или более ограничивающее объявление. Одним типом ограничивающих объявлений являются ограничения на виды значений, которые это свойство может принимать: owl:allValuesFrom, owl:hasValue и owl:someValuesFrom. Другим типом ограничивающих объявлений является ограничение на мощность класса (cardinality restrictions). Например, можно потребовать, чтобы каждый курс обучался хотя бы одним преподавателем:

```

<owl:Class rdf:about="#course">
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="#isTaughtBy"/>
<owl:minCardinality
rdf:datatype="&xsd;nonNegativeInteger">
1
</owl:minCardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

Отметим, что требуется указать, что литерал «1» должен интерпретироваться как nonNegativeInteger (неотрицательное целое), а не как строка,

и что используется объявление пространства имен `xsd`, которое сделано в заголовочном элементе, путем ссылки на документ XML Schema.

Или можно указать, что по практическим причинам кафедра (`department`) должна иметь по крайней мере 10 и не более 30 сотрудников:

```
<owl:Class rdf:about="#department">
  <rdfs:subClassOf>
  <owl:Restriction>
  <owl:onProperty rdf:resource="#hasMember"/>
  <owl:minCardinality
  rdf:datatype="&xsd;nonNegativeInteger">
  10
  </owl:minCardinality>
  <owl:maxCardinality
  rdf:datatype="&xsd;nonNegativeInteger">
  30
  </owl:maxCardinality>
  </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Можно указать точное значение, используя одно и то же число в элементах `owl:minCardinality` и `owl:maxCardinality`. Для удобства, OWL предлагает также `owl:cardinality`, которое позволяет точно задать мощность класса.

В завершение отметим, что `owl:Restriction` определяет анонимный класс, который не имеет `id`, не определен с помощью элемента `owl:Class` и имеет только локальную область видимости (`local scope`): он может быть использован в том месте, где появились эти ограничения.

Когда говорится о классах, то нужно иметь в виду два их смысла: классы, которые определены в элементе `owl:Class` с `id`, и локальные анонимные классы в виде коллекции объектов, которые удовлетворяют определенным ограничивающим условиям.

5.5.5. Специальные свойства

Некоторые свойства свойств элементов могут быть определены напрямую:

- owl:TransitiveProperty определяет транзитивные свойства, такие, как «быть лучше чем» ("has better grade than"), «быть выше чем» ('is taller than"), «являться предком» ('is ancestor of") и т.п.;
- owl:SymmetricProperty определяет симметричные свойства, такие, как «иметь такой же уровень как» ('has same grade as"), «быть родственником» ('is sibling of") и т.п.;
- owl:FunctionalProperty определяет свойства, которые имеют не более чем одно уникальное значение для каждого объекта, такие, как «возраст» ('age"), «вес» ('height"), «непосредственного руководителя» ('directSupervisor") и т.п.;
- owl:InverseFunctionalProperty определяет свойство, для которого два различных объекта не могут иметь одно и то же значение, например свойство «являться Номером Социального Обеспечения» ('is-TheSocialSecurityNumberfor") (один номер социального обеспечения назначается только одному человеку).

Пример записи таких свойств приведен ниже:

```
<owl:ObjectProperty rdf:ID="hasSameGradeAs">
<rdf:type rdf:resource="&owl;TransitiveProperty" />
<rdf:type rdf:resource="&owl;SymmetricProperty" />
<rdfs:domain rdf:resource="#student" />
<rdfs:range rdf:resource="#student" />
</owl:ObjectProperty>
```

5.5.6. Булевы комбинации

Можно говорить о булевых комбинациях (объединении (union), пересечении (intersection), дополнении (complement)) классов, которые могут быть определены в элементе owl:Class или в выражении класса (class expressions). Например, можно определить, что курсы (courses) и члены персонала (staff members) не пересекаются, следующим образом:

```
<owl:Class rdf:about="#course">
<rdfs:subClassOf>
<owl:Restriction>
<owl:complementOf rdf:resource="#staffMember"/>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

Эта запись определяет, что каждый «курс» (course) является экземпляром дополнения (owl:complementOf) «членов персонала» (staff members), т.е. «курс» не является «членом персонала». Отметим, что это утверждение можно также выразить, используя элемент owl:disjointWith.

Объединение классов строится с помощью элемента owl:unionOf:

```
<owl:Class rdf:ID="peopleAtUni">
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#staffMember"/>
    <owl:Class rdf:about="#student"/>
  </owl:unionOf>
</owl:Class>
```

Атрибут rdf:parseType является сокращением для явного синтаксиса по построению списка с <rdf:first> и <rdf:rest> тэгами. Такие списки требуются, так как встроенные контейнеры RDF имеют серьезные ограничения: отсутствует способ закрытия, т.е. обозначить, что «это все элементы контейнера».

Происходит это потому, что когда один граф описывает некоторые из членов, не исключена возможность, что где-то существует другой граф, который описывает дополнительные члены. Синтаксис списков (list) предоставляет как раз эту возможность, но очень многословно, что мотивирует введение сокращенного обозначения rdf:parseType. Отметим, что эта запись не утверждает, что новый класс является подклассом объединения. Она говорит о том, что новый класс эквивалентен объединению. Другими словами, эта запись объявляет эквивалентность классов. Здесь также не указывается, что классы не пересекаются (disjoint): ведь возможно, что кто-то в составе персонала является также студентом.

Пересечение (Intersection) задается с помощью owl:intersectionOf:

```
<owl:Class rdf:ID="facultyInCS">
  <owl:intersectionOf rdf:parseType="owl:collection">
    <owl:Class rdf:about="#faculty"/>
  <Restriction>
    <owl:onProperty rdf:resource="#belongsTo"/>
    <owl:hasValue rdf:resource="#CSDepartment"/>
  </Restriction>
</owl:intersectionOf>
</owl:Class>
```

Отметим, что здесь имеет место пересечение двух классов, один из которых определен анонимно: класс всех объектов, принадлежащих кафедре (department) CS (Computer Science). Далее отметим, что булевы комбинации могут произвольно вкладываться друг в друга.

В следующем примере определено, что административный персонал – это те сотрудники персонала (staffMember), которые не являются персоналом, поддерживающим кафедру ("#faculty"), не являются также и техническим персоналом ("#techSupportStaff"):

```
<owl:Class rdf:ID="adminStaff">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#staffMember"/>
    <owl:complementOf>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#faculty"/>
        <owl:Class rdf:about="#techSupportStaff"/>
      </owl:unionOf>
    </owl:complementOf>
  </owl:intersectionOf>
</owl:Class>
```

5.5.7. Перечисления (Enumerations)

Перечисление задается элементом owl:oneOf и обычно определяется классом, который перечисляет все свои элементы:

```
<owl:oneOf rdf:parseType="Collection">
  <owl:Thing rdf:about="#Monday"/>
  <owl:Thing rdf:about="#Tuesday"/>
  <owl:Thing rdf:about="#Wednesday"/>
  <owl:Thing rdf:about="#Thursday"/>
  <owl:Thing rdf:about="#Friday"/>
  <owl:Thing rdf:about="#Saturday"/>
  <owl:Thing rdf:about="#Sunday"/>
</owl:oneOf>
```

5.5.8. Экземпляры (Instances)

Экземпляры классов объявляются так же, как и в RDF. Например:

```
<rdf:Description rdf:ID="949352">
  <rdf:type rdf:resource="#academicStaffMember"/>
```

```
</rdf:Description>
```

эквивалентно

```
<academicStaffMember rdf:ID="949352"/>
```

Можно предоставить больше деталей, таких, как:

```
<academicStaffMember rdf:ID="949352">
<uni:age rdf:datatype="xsd:integer">39<uni:age>
</academicStaffMember>
```

В отличие от обычной системы баз данных, OWL не использует предположение об уникальности имен: просто из того факта, что два экземпляра имеют разные имена (или: ID), не следует, что они действительно различные индивидуумы. Например, если определить, что каждый курс преподается не более чем одним сотрудником преподавательского состава:

```
<owl:ObjectProperty rdf:ID="isTaughtBy">
<rdf:type rdf:resource="&owl;FunctionalProperty" />
</owl:ObjectProperty>
```

и далее определить, что конкретный курс преподается двумя преподавателями:

```
<course rdf:about="CIT1111">
<isTaughtBy rdf:resource="949318">
<isTaughtBy rdf:resource="949352">
</course>
```

то это не приведет к тому, машина логического вывода OWL (reasoner) сообщит об ошибке. Просто система может правильно вывести, что ресурсы "949318" и "949352" очевидно равны. Для того, чтобы гарантировать, что различные индивидуумы действительно распознавались различающимися, нужно явно объявить их уникальность:

```
<lecturer rdf:about="949318">
<owl:differentFrom rdf:resource="949352">
</lecturer>
```

Так как такого рода утверждения встречаются часто и количество таких утверждений может очень быстро расти, если требуется заявить о несовпадении большого числа индивидуумов, то OWL предоставляет

краткое обозначение для утверждения о попарном неравенстве всех индивидуумов заданного списка:

```
<owl:allDifferent>
  <owl:distinctMembers rdf:parseType="Collection">
    <lecturer rdf:about="949318">
    <lecturer rdf:about="949352">
    <lecturer rdf:about="949111">
  </owl:distinctMembers>
</owl:allDifferent>
```

Отметим, что `owl:distinctMembers` может использоваться только в комбинации с `owl:AllDifferent`.

5.5.9. Типы данных (Datatypes)

Хотя XML Schema предоставляет механизм для построения определяемых пользователями типов данных (например, тип данных «взрослыйВозраст» (`adultAge`), как целые числа более 18 или тип данных – строки, начинающиеся с номера), такие производные типы данных не могут использоваться в OWL. Фактически, даже не все из многих встроенных типов данных XML Schema могут использоваться в OWL. Справочный документ по OWL перечисляет те данные XML Schema, которые могут использоваться. Как правило, они включают большинство часто используемых типов, таких, как `string`, `integer`, `boolean`, `time` и `date`.

Ранее было показано использование элемента `owl:priorVersion` как части заголовочной информации для указания о ранних версиях текущей онтологии. Эта информация не имеет формальной модельно-теоретической семантики, но может использоваться людьми и программами в целях, отличных от управления онтологиями.

Кроме элемента `owl:priorVersion`, OWL имеет еще три утверждения для указания дополнительной неформальной информации о версиях. Ни один из них не несет какого-либо формального смысла.

- `owl:versionInfoStatement` обычно содержит строку, дающую информацию о текущей версии, например, ключевые слова `RCS/CVS`.
- `owl:backwardCompatibleWith` содержит ссылку на другую онтологию. Таким образом, определяется, что конкретная онтология является предыдущей версией онтологии, содержащейся в документе, и более того, указывается, что она обратно совместима с ней. В част-

ности, это указывает, что все идентификаторы из предыдущей версии имеют те же предполагаемые интерпретации в новой версии. Таким образом, это подсказывает авторам документа, что они могут внести изменения в документы, чтобы соответствовать новой версии (просто исправляя объявления имен пространств и утверждения `owl:imports` для ссылки на URL новой версии).

- `owl:incompatibleWith`, с другой стороны, указывает, что содержащаяся онтология является более поздней версией онтологии, на которую ссылается, но обратно не совместима с ней. По существу, это может быть использовано авторами онтологий, которые желают явно указать, что документы не могут быть обновлены (`updated`) для использования новой версии без тщательной проверки необходимости внесения изменения.

5.5.10. Разновидности языка OWL

Конфликт требований, а именно поддержка эффективного логического вывода и выразительность языка привели к тому, что Web Ontology Working Group определила OWL как три различных подязыка, каждый из которых разработан для удовлетворения столь несовместимому набору требований:

- **OWL Full:** Полный язык называется OWL Full и использует все примитивы (базовые элементы) языка. Он также позволяет объединять эти примитивы произвольным способом с RDF и RDFS. Это позволяет также изменять смысл предопределенных в RDF или OWL примитивов путем применения примитивов языка друг к другу. Например, в OWL Full можно задать ограничения мощности `owl:class`, по существу ограничивая количество классов, которые могут быть описаны в любой онтологии. Преимущество OWL Full заключается в том, что он полностью совместим с RDF как синтаксически, так и семантически: любой допустимый RDF-документ также является допустимым документом языка OWL Full и любое верное заключение для RDF/RDFS является также верным заключением для OWL Full.

Один из недостатков языка OWL Full возник из-за того, что он стал таким мощным и выразительным, что появилась проблема разрешимости, вследствие чего не обеспечивается поддержка логического вывода.

- **OWL DL:** (Description Logic – Дескриптивная логика) является подмножеством языка OWL Full, который ограничивает способ ис-

пользования конструкторов языков OWL и RDF, восстанавливая тем самым вычислительную эффективность языка. Органичение заключается в запрещении применения конструкторов друг к другу и, таким образом, гарантирует, что язык соответствует хорошо изученной дескриптивной логике. Это позволяет обеспечить поддержку эффективного логического вывода. Недостатком является то, что в OWL DL отчасти теряется совместимость с RDF: RDF-документ должен быть, в общем случае, некоторым образом расширен и некоторым образом ограничен, прежде чем стать правильным OWL DL-документом. И наоборот, каждый правильный OWL DL-документ остается правильным RDF-документом.

- OWL Lite: Является еще более ограниченным языком, чем OWL DL. Он содержит подмножества конструкторов языка OWL DL. OWL Lite не содержит перечисляемые классы (enumerated classes), утверждения о непересекаемости (disjointness statements) и произвольные мощности (arbitrary cardinality).

Преимущество этого языка в том, что его проще изучить (пользователем) и проще реализовать (разработчикам инструментов).

Разработчики онтологий, осваивающие OWL, должны решать, какое подмножество OWL лучше соответствует их потребностям. Выбор между OWL Lite и OWL DL зависит от степени потребности пользователя в более выразительных конструкторах, предоставляемых OWL DL и OWL Full. Выбор между OWL DL и OWL Full в основном зависит от степени потребности пользователя в возможностях метамоделирования языка RDFS (таких, как определение класса классов или прикрепления свойств к классам). При использовании OWL Full, по сравнению с OWL DL, поддержка логического вывода является менее предсказуемой, так как полная реализация логического вывода в OWL Full будет невозможна.

Имеется строгое понимание восходящей совместимости между этими тремя языками:

- Каждая онтология, допустимая на языке OWL Lite, является онтологией, допустимой на языке OWL DL.
- Каждая онтология, допустимая на языке OWL DL, является онтологией, допустимой на языке OWL Full.
- Каждое правильное вывод для OWL Lite является правильным выводом для OWL DL.

- Каждое правильный вывод для OWL DL является правильным выводом для OWL Full.

OWL все еще в значительной степени использует RDF и RDFS:

- Все варианты OWL используют RDF в качестве своего синтаксиса.
- Экземпляры объявляются в RDF, используя RDF описания и информацию о типах.
- OWL-конструкторы типа `owl:Class`, `owl:DatatypeProperty` и `owl:ObjectProperty` являются специализациями их двойников в RDF.

На рис. 5.7 показаны иерархия включения классов между некоторыми примитивами моделирования OWL и RDF/RDFS.

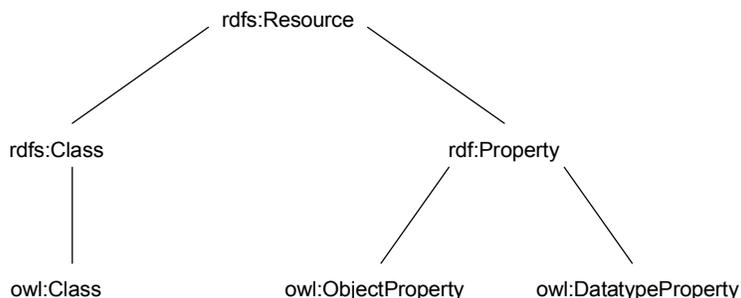


Рис. 5.7. Иерархия классов OWL и RDF/RDFS (отношения подклассов)

При проектировании OWL вначале надеялись на обратную совместимость с соответствующей возможностью повторного использования программного обеспечения (ПО) между различными уровнями. Однако преимущество полной обратной совместимости для OWL (что любой процессор, понимающий OWL, будет обеспечивать правильную интерпретацию любого документа RDFS) достижимо только для OWL Full, ценой частичной потери вычислимости.

5.5.11. Связь языка OWL с дескриптивной логикой

OWL был спроектирован на основе SH семейства ДЛ (напомним, что S – это сокращение ДЛ ALC_{R^+} (AL – базовый язык; C – отрицание для сложных понятий; R^+ – транзитивные роли), а H – это использование иерархии ролей). Конструкторы и аксиомы, поддерживаемые SH, включают булевы операции (пересечение, объединение и дополнение), ограничения на свойства, транзитивные свойства и иерархию свойств, т.е.

эквивалентны ALC ДЛ, расширены транзитивными свойствами и иерархией свойств. Иерархия свойств важна для OWL, так как это особенность RDFS (онтологическое моделирование), однако многие серьезные приложения требуют использования транзитивных свойств. Члены семейства SH включают важную SHIQ ДЛ, которая добавляет инверсные свойства и обобщенные ограничения мощности (generalised cardinality restrictions) и SHOQ(D), которая добавляет возможность определить класс (понятие) путем перечисления экземпляров (например, понятие ДниНедели = {Понедельник, Вторник, Среда, Четверг, Пятница, Суббота}) и поддержку типов данных и значений (например, типы данных целые, строки и значения, такие, как "35").

Использование языка OWL DL в стиле дескриптивной логики тесно связано с ДЛ SHOIN(D), которая сама является расширением SHOQ(D) (расширенной инверсными ролями и ограниченной до неуточненных числовых ограничений). OWL DL может формировать описания классов, типов данных, индивидуумов, используя конструкторы (табл. 5.1) и аксиомы (табл. 5.2).

OWL DL связан с очень выразительной дескриптивной логикой SHOIN(D). Эта ДЛ достаточно трудна для ее изучения новыми пользователями, так как позволяет строить сложные булевы описания, используя, например, объединение и дополнение. SHOIN(D) также трудна для выполнения логического вывода, так как ключевые задачи вывода имеют NExpTime сложность и до некоторой степени трудны для построения даже не выполняющими логический вывод инструментами из-за сложных описаний.

По этим причинам было выявлено подмножество OWL DL, которое должно быть проще по всем указанным выше показателям. Это подмножество называется OWL Lite. OWL Lite не использует объединения и дополнения, ограничивает пересечения до неявных пересечений в аксиомах, подобных фреймам классов, ограничивает встроенные описания

Таблица 5.1

Основные конструкторы понятий классов OWL

Конструктор	Синтаксис ДЛ	Пример
Thing	\top	
Nothing	\perp	
intersectionOf	$C_1 \sqcap \dots \sqcap C_n$	Human \sqcap Male
unionOf	$C_1 \sqcup \dots \sqcup C_n$	Doctor \sqcup Lawyer

complementOf	$\neg C$	$\neg \text{Male}$
oneOf	$\{x_1 \dots x_n\}$	$\{\text{john, mary}\}$
allValuesFrom	$\forall P.C$	$\forall \text{hasChild.Doctor}$
someValuesFrom	$\exists P.C$	$\exists \text{hasChild.Lawyer}$
maxCardinality	$\geq nP.C$	$\geq 2 \text{hasChild.Lawyer}$
minCardinality	$\leq nP.C$	$\leq 2 \text{hasChild.Male}$
Cardinality	$= nP.C$	$= 1 \text{hasParent.Femail}$

Таблица 5.2

Основные аксиомы OWL

Аксиома	Синтаксис ДЛ	Пример
subClassOf	$C_1 \sqsubseteq C_2$	$\text{Human} \sqsubseteq \text{Animal} \sqcap \text{Biped}$
equivalentClass	$C_1 \equiv C_2$	$\text{Man} \equiv \text{Human} \sqcap \text{Male}$
disjointWith	$C_1 \sqsupseteq \neg C_2$	$\text{Male} \sqsupseteq \neg \text{Female}$
sameIndividualAs	$\{x_1\} \equiv \{x_2\}$	$\{\text{President_Bush}\} \equiv \{\text{G_W_Bush}\}$
differentFrom	$\{x_1\} \sqsupseteq \neg \{x_2\}$	$\{\text{john}\} \sqsupseteq \neg \{\text{peter}\}$
subPropertyOf	$P_1 \sqsubseteq P_2$	$\text{hasDaughter} \sqsubseteq \text{hasChild}$
equivalentProperty	$P_1 \equiv P_2$	$\text{cost} \equiv \text{price}$
inverseOf	$P_1 \equiv P_2^{-}$	$\text{hasChild} \equiv \text{hasParents}^{-}$
transitiveProperty	$P_1^{+} \sqsubseteq P_2$	$\text{ancestor}^{+} \sqsubseteq \text{ancestor}$
functionalProperty	$\top \sqsubseteq \leq 1P$	$\top \sqsubseteq \leq \text{hasMother}$
inverseFunctionalProperty	$\top \sqsubseteq \leq 1P^{-}$	$\top \sqsubseteq \leq \text{isMother}^{-}$

до имен понятий, не позволяет индивидуумам появляться в описаниях и аксиомах классов и ограничивает мощности до 0 и 1.

Эти ограничения делают OWL Lite подобным дескриптивной логике SHIF(D). Подобно SHIF(D), ключевые логические выводы в OWL Lite в худших случаях вычисляются за экспоненциальное время (ExpTime), и уже существует несколько оптимизированных систем вывода для логических эквивалентов OWL Lite. Такое улучшение в реализации получается с относительно небольшой потерей выразительной силы – хотя синтаксис OWL Lite более ограничен, чем синтаксис OWL DL, он все еще способен выразить сложные описания, вводя новые имена классов и используя неявное отрицание с помощью аксиом непересекаемости. С помощью этих методов все OWL DL-описания могут быть записаны в OWL Lite, за исключением тех, которые содержат или имена индивидуумов, или мощности более чем 1.

Глава 6

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ДЛЯ РАБОТЫ СО ЗНАНИЯМИ

Как уже отмечалось в предыдущих главах, современное представление об управлении знаниями не сводится к применению компьютеров и современных информационных технологий. Однако справедливо и обратное утверждение. Без современных ИТ-технологий, без Интернета и интранета управление знаниями в компании не может быть поднято на должный уровень, так как именно ИТ-технологии обеспечивают расширение доступа к знаниям, облегчают и ускоряют процесс обмена знаниями между специалистами, подразделениями, предприятиями, создают технологическую основу для семантической обработки информации и получения новых знаний.

В отличие от искусственного интеллекта, ставящего своей целью создание *заменителя* человеческого интеллекта, целью управления знаниями является создание *усилителя* человеческого интеллекта. Усилителя, позволяющего человеку минимизировать затраты времени при поиске и обработке знаний, при их накоплении и распространении в пространстве дислокации подразделений, фирм, транснациональных корпораций и рынков. А с этими и другими подобными функциями год от года все лучше справляются компьютеры, телекоммуникационные сети и ориентированные на работу со знаниями программные системы и технологии.

Рассмотрению инструментальных средств для работы со знаниями и посвящается основное содержание настоящей главы.

Популярное определение, данное Ruggles [Ruggles R.L.E., 1997], утверждает, что инструменты управления знаниями – это технологии, используемые для совершенствования и увеличения возможностей реализации процессов работы со знанием, таким, как создание, накопление, распространение и др.

При этом различают «методы УЗ» (KM techniques) и «технологии УЗ» (KM technologies). Именно технологии УЗ в большинстве своем опираются на компьютерные и программные системы.

6.1. Эволюция развития инструментальных средств

Прежде чем приступить к описанию современных инструментальных средств, составляющих основу современных информационных сред компаний, а также инструментов для работы со знаниями, имеет смысл вспомнить эволюцию развития электронных вычислительных машин (ЭВМ) и подходов к обработке данных.

Сначала (пять десятилетий назад) ЭВМ научили считать, то есть ЭВМ овладела *вычислительной функцией*. Затем ЭВМ научили (алгоритмически, программно) распознавать и обрабатывать символы (языковые, графические), то есть ЭВМ овладела начальными *функциями в обработке данных*. Потом в машину вложили модели технологических процессов и алгоритмы решения задач моделирования и оптимизации, то есть ЭВМ овладела *функцией моделирования*, столь важной для управления и принятия решений. Далее ЭВМ соединили с линиями и центрами связи, то есть ЭВМ овладела *коммуникационной функцией* и стала основой для построения не только телекоммуникационных систем, но и систем связи новых поколений.

Очередным естественным этапом в функциональной эволюции ЭВМ явилось возникновение концепции информационных систем, баз данных и баз знаний.

Не менее интересна эволюция развития данных, основные этапы которой приведены на рис. 6.1.

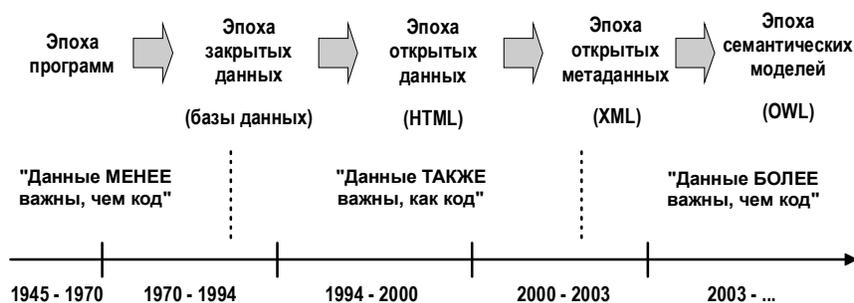


Рис. 6.1. Временная диаграмма развития данных

Как следует из приведенной временной диаграммы, на первом этапе (1945 – 1970 гг.) доминировали представления о том, что данные менее важны, чем код. Дело в том, что на заре развития ЭВМ обладали весьма скудным программным обеспечением. Организаторы вычислительного процесса в ЭВМ – программы-диспетчеры, операционные системы – появились далеко не сразу. Поэтому алгоритмизация вычислений и написание соответствующих программ считалось делом первостепенной важности. Отсюда и отличительный признак этапа – эпоха программ.

По мере развития функций ЭВМ по обработке символьной информации пришло осознание важности данных. Данные, по крайней мере, также важны, как и код. Этому достаточно затянувшемуся этапу (1994 – 2003 гг.) соответствуют две эпохи: эпоха закрытых данных, первоначально господствовавшая в информационных системах тех лет, и эпоха открытых (HTML) данных, не потерявших свою актуальность и широко применяемых в последнем десятилетии.

Вслед за эпохой открытых данных, начиная с 2000 г., наступила эпоха открытых (XML) метаданных. Эта эпоха соответствует на временной диаграмме развития данных (рис. 6.1) этапу, когда данные стали признаваться более важными, чем код. И это понятно, если вспомнить, что метаданные – это данные о данных, которые приобретают исключительную ценность при лавинообразном росте последних. Эпоха метаданных важна еще и потому, что именно она является естественным предшественником эпохи семантических моделей, для которых и создано онтологическое семейство языков описания знаний OWL, рассмотренных в гл. 5.

Технологии управления знаниями представляют собой комбинацию технических (hardware) и программных (software) инструментальных средств.

Технические средства формируют платформу для работы программных средств и среду для поиска, хранения и передачи знаний. К основным компонентам технических средств относятся:

- Персональные компьютеры, которыми оснащаются рабочие места сотрудников компании.
- Мощные серверы корпоративной сети и локальных сетей подразделения.
- Коммуникационное оборудование, включающее маршрутизаторы, коммутаторы и хабы.
- Волоконно-оптические, кабельные и радиоканалы, обеспечивающие высокоскоростную передачу данных, информации и знаний в ISDN (Integrated Services Digital Network).

- АТМ (Asynchronous Transfer Mode) – режим мультимедиа коммутации для обработки высокоскоростного трафика, включающего обычные, а также голосовые и видео данные.
- Технические средства доступа в Интернет, интранет и экстранет.

Современная информационная система компании и СУЗ опираются практически на одни и те же технические средства, на одну и ту же техническую среду. Главные отличительные признаки приходится на программные инструменты. Программные средства (инструменты), количество которых в последнее время стремительно нарастает, играют определяющую роль в управлении знаниями. Их функциональный состав и уровень работы с данными, информацией и знаниями соответственно и превращает развитую информационную систему компании в СУЗ. А СУЗ в дополнение к программам и инструментам ИС должна иметь в своем составе программные инструменты поиска и извлечения информации из данных, знаний из информации, новых семантически значимых утверждений и обобщений из доступных ей знаний.

Из сказанного следует, что программные инструменты СУЗ являются своего рода интеллектуальной надстройкой над программными инструментами развитой ИС современной компании, с краткого описания структуры и состава которых имеет смысл начать.

6.2. Программное обеспечение информационных систем

Информационная система современной компании включает техническое обеспечение, программное обеспечение и информационные ресурсы.

Состав технического обеспечения включает компьютеры и средства для их работы в составе сети. Он был кратко описан в предыдущем параграфе практически полностью, за исключением входящих в его состав средств оргтехники (принтеры, сканеры, цифровые фотокамеры и пр.).

Доминирующей архитектурой в современных информационных системах является архитектура «Клиент – Сервер» (рис. 6.2).

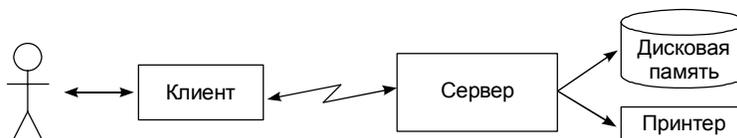


Рис. 6.2. Архитектура «Клиент – Сервер»

В этой централизованной архитектуре техническое и программное обеспечение разделяется на две части. Одна часть, называемая сервером (server – слуга), является пассивной, т.е. сама не инициирует начало действий, а в автоматическом режиме обслуживает запросы других частей системы, именуемых клиентами (client). Клиенты взаимодействуют с пользователями-людьми и потому являются активными, инициирующими различные запросы к серверам. Задача сервера – как можно более эффективно обслужить запросы клиентов (коллективное использование ресурсов технических и информационных), а задача клиента – предоставить наилучшие возможности для работы пользователя.

Как техническое, так и программное обеспечение клиентов и серверов существенно различается.

ПО, установленное на компьютерах, подразделяется на системное (базовое), которое предназначено для обеспечения эффективной работы оборудования и выполнения базовых функций, и пользовательское ПО, которое предназначено для решения конкретных задач пользователей. Такое деление достаточно условно, так как и системное ПО может решать некоторый класс задач, требуемых пользователем (например, предоставлять возможность коллективной работы с документами).

Поэтому все программное обеспечение, устанавливаемое в современной организации, можно систематизировать в соответствии с табл. 6.1.

Таблица 6.1

Состав программного обеспечения современного офиса

Программное обеспечение	Общее (серверное)	Индивидуальное (клиентское)
Системное (базовое)	Операционные системы – серверы (Windows 2003 Server)	Операционные системы компьютеров пользователей (Windows 2000, Windows XP)
Пользовательское (специальное)	Специальные серверы: - SQL Server 2000 - Web Server (Internet Information Server) - E-mail Server (Exchange Server 2000) - Порталы организаций	Программы пользователей: - Браузер - Клиент E-mail - Приложения Microsoft Office - Приложения для работы с БД - Приложения для решения специальных задач

Современные *операционные системы* (ОС) являются крупными комплексами программ, которые включают все необходимое для реализации эффективной работы компьютера, для организации работы групп компьютеров в сетях (локальных или глобальной) и обеспечения удобных интерфейсов для работы пользователя с компьютером.

Программы ОС расширяются за счет включения в их состав дополнительных систем (поддержка локальной сети, интранет, средства для работы в сети Интернет и др.).

Примерами современных ОС являются широко распространенные операционные системы семейства Windows фирмы Microsoft, такие, как Windows 2000 и Windows XP.

Специальные серверы предназначены для решения стандартных задач по обработке информации:

- Коллективные базы данных организации.
- Почтовые серверы для обмена сообщениями электронной почты.
- Серверы для работы с документами в Web-сети.

Пользовательское программное обеспечение предназначено для решения конкретных задач пользователя, таких, как:

- создание, хранение и работа с документами;
- создание, хранение и работа с данными;
- решение специфических информационных задач организации;
- обеспечение взаимодействия между пользователями компьютеров.

Основными прикладными программами (приложениями) в организациях являются *программы для создания и редактирования документов различных типов*, таких, как:

- текстовые (различных форматов, например, txt, doc, rtf, pdf);
- презентации (различных форматов, например, ppt, pdf);
- электронные таблицы;
- Web-страницы;
- графические изображения;
- мультимедиа (видео, аудио).

Для работы с этими информационными ресурсами имеются различные наборы программ, наиболее известным из которых является Microsoft Office (различных версий). Основными составляющими этого пакета для работы с документами являются:

- редактор текстовых документов Microsoft Word;
- программа работы с электронными таблицами Microsoft Excel;

- программа создания и редактирования презентаций (набор слайдов) Microsoft PowerPoint;
- программа для работы с почтовыми сообщениями Microsoft Outlook;
- программа работы с локальными базами данных Microsoft Access.

Крупной составляющей программного обеспечения являются специальные программные системы для хранения и эффективной работы с данными – *системы управления базами данных (СУБД)*. Различные версии разрабатываются и достаточно активно используются на практике. Они стандартизированы (стандартный язык запросов SQL), поддерживают различные модели данных (например, реляционную модель), обладают высокой производительностью.

Существуют СУБД для поддержания локальных (пользовательских) БД и больших (коллективных) БД. Известными примерами для работы с локальными БД являются система Microsoft Access (входящая в состав комплекта Microsoft Office), система Foxpro (Microsoft) и Engine (Borland).

Примерами коллективных СУБД являются: Oracle (одноименной компании) и SQL Server (Microsoft). Данные системы реализованы в виде серверов для обслуживания больших баз данных и большого количества запросов клиентов.

Клиентами этих серверов являются прикладные программы, которые предоставляют пользователям удобный интерфейс для ввода новых данных в БД, для корректировки хранимых данных и выборки требуемых данных. Клиентские приложения могут быть стандартными (например, 1С Бухгалтерия) или оригинальными, разработанными для специфических задач организации. Современные системы предоставляют широкий набор средств для разработки приложений под управлением соответствующих СУБД.

Обязательным элементом современной информационной системы является *система обмена почтовыми сообщениями*. Данная система позволяет выполнять обмен электронными письмами между сотрудниками самой организации и сотрудниками других организаций. Данная система также имеет архитектуру «Клиент – Сервер» и состоит из почтовых серверов и клиентских программ. Почтовые серверы, например Exchange Server (Microsoft), обеспечивают обмен электронными письмами между собой. При этом они создают и поддерживают почтовые ящики для каждого пользователя электронной почты, в которых сохраняются посту-

пающие и отправляемые письма. Для работы с почтовыми ящиками, т.е. для создания, отправки и получения писем используются клиентские программы, например Outlook Express (составная часть ОС Windows) или программа Outlook (входящая в состав комплекта Microsoft Office).

Практически повсеместным в организациях становится использование интранет. Под интранет понимается создание Web-сети организации с использованием технологий сети Интернет.

Интранет включает установку в локальной сети организации, закрытой от внешнего доступа, Web-серверов, которые обслуживают запросы по протоколу HTTP от специальных программ, входящих в состав ОС, браузеров (обозревателей). Данная система позволяет организовать коллективную работу с документами и базами данных без необходимости установки специальных программ на компьютерах пользователей.

Основная часть документов интранет записывается на языке гипертекстовой разметки HTML – Web-страницы. Эти документы объединяются в логические группы, имеющие свой адрес в сети интранет, и называются Web-сайтами. Имеются программы, которые позволяют создавать и редактировать отдельные страницы и Web-сайты в целом. Примером такого редактора является система Microsoft FrontPage, входящая в состав комплекта Microsoft Office.

В связи с большим количеством электронных документов в организациях начинают внедряться *системы управления документами* (например, система Documentum (компания EMC)). Основными функциями таких систем являются:

- централизованное хранение документов, в том числе в базах данных;
- поддержка версий документов;
- обеспечение разграничения доступа и поддержка конфиденциальности на уровне документа или его версий;
- обеспечение работы с множеством структурных метаданных по документам;
- атрибутивный и полнотекстовый поиск документов;
- поддержка составных документов и коллекций документов (папок, проектов);
- прозрачная интеграция с офисными приложениями и возможность интеграции с внешними системами;
- представление документов в разных форматах.

Системы управления документами (например, Documentum) также могут включать *систему автоматизации управления деловыми процес-*

сами (workflow). В этих системах основная функциональность направлена на обеспечение движения документов по predetermined маршрутам между исполнителями внутри компании с целью организации работ по подготовке и использованию этих документов. Системы управления предоставляют различные возможности по работе с деловыми процессами, например, такие, как:

- проектировать маршруты движения документа с помощью графических средств;
- задавать и контролировать сроки работ с документами на каждом этапе;
- контролировать текущее состояние документа относительно активного маршрута;
- автоматизировать выбор маршрута или части маршрута в зависимости от формальных условий, заданных, например, в терминах свойств документа или профиля текущего пользователя.

Такого рода системы работы с документами носят пограничный характер. С одной стороны, они являются составной частью информационных систем современных компаний, с другой – элементом СУЗ в части обработки баз данных, информации, хранящихся в репозиториях. Тем более, что в новых версиях документальных систем появляются и компоненты семантической направленности, о чем пойдет речь в последующих разделах.

6.3. Программные инструменты для построения и работы с онтологиями

Как было сказано ранее, онтологии – это способ точного описания (специфицирования) структуры знаний предметной области (домена) в виде формальной логики, спроектированной для компьютерной обработки. Имеются многочисленные ссылки на использование онтологий и таксономий разработчиками основополагающих IT-решений по интеграции приложений корпоративного уровня (enterprise-application-integration (EAI)). Предлагаются платформы, популярно обозначаемые термином «семантическая интеграция», для обмена информацией между разнородными ресурсами, такими, как действующие БД, полуструктурированные репозитории, отраслевые стандартные директории, словари типа WebXML и потоки неструктурированного контента, такого, как текст и мультимедиа. Онтологии, например, используются для управления извлечением семантического содержания из коллекций обыч-

ных текстовых документов, описывающих медицинские исследования, потребительские товары и различные связанные с бизнесом темы.

Инициативы правительств многих стран направлены на усиление возможностей ИТ, имеющихся у федеральных агентств и объединяют использование онтологий с существующими инфраструктурами для выполнения аналитических оценок информации, приходящей из различных источников. Типичными примерами этого являются: антитеррористический интеллектуальный анализ, поддержка работы центров управления и принятия решений.

Крупные сервисы по поиску в сети Web, такие, как Google и Yahoo, используют основанные на онтологиях подходы для поиска и организации содержания сети Web. Google, который приобрел Applied Semantics, Inc., один из лидеров по разработке инструментов для извлечения семантики, предсказывает быстрое наращивание роли онтологий в разнообразных информационных ресурсах.

Процесс построения онтологий охватывает:

- Подробное описание (спецификацию) предметной области (проблемы).
- Фиксацию и анализ знаний домена.
- Концептуальное проектирование онтологии в интересах профессионального сообщества.
- Тестирование и итеративное конструирование.
- Публикацию онтологии как терминологии предметной области.
- Заполнение соответствующей базы знаний экземплярами понятий онтологии.

Инструменты создания онтологий в большинстве своем требуют, чтобы их пользователи были обучены представлению знания и логике предикатов.

Обычно онтологии строятся инженерами по знаниям, которые работают со специалистами по предметной области или экспертами по рассматриваемой теме.

Для масштабирования этого подхода на большое предприятие распределенная команда экспертов по рассматриваемым темам должна быть способна независимо выполнять некоторые виды деятельности. Например, они должны быть способны вводить и корректировать знания напрямую и легко, без специальной подготовки в области представления знаний, выявлять или манипулировать ими.

Однако процесс построения онтологии, при всей его важности, не исчерпывает всех этапов жизненного цикла столь важного продукта. Специалисты по управлению знаниями сходятся в следующем понимании этапов жизненного цикла онтологии (рис. 6.3):

- Построение (creating).
- Заполнение (populating).
- Проверка (validating).
- Использование (deploying).
- Поддержка (maintaining).
- Развитие (evolving).



Рис. 6.3. Жизненный цикл онтологии

В мире в настоящее время доступны многочисленные коммерческие и бесплатные (с открытым кодом) программные инструменты и системы для построения и использования онтологий в управлении знаниями, а также для их интеграции с инфраструктурой сетей Web и БД. Многие из них относятся к категории редакторов онтологий. Редактор онтологий – программа, предоставляющая графический интерфейс для удобного описания схем онтологий, редактирования структуры онтологии, для проверки ее согласованности, для представления онтологий с использованием выбранного языка описания. Однако редакторы не покрывают весь набор функций взаимодействия пользователей с онтологиями на всех этапах жизненного цикла. Поэтому многие доступные продукты расширяют для пользователей перечень функций и сервисов,

реализуемых редактором. В качестве иллюстрации далее приводится описание Protégé 2000 (рис. 6.4).

Этот инструмент разработала Knowledge Modeling Group (KMG) Стэнфордского университета (Stanford University) как часть проекта Protégé, выполняемого в течение последних 15 лет и включающего разработку набора инструментов моделирования знаний [Gennari J., Musen M.A., Fergerson R.W., et al., 2002]. Текущая версия редактора онтологий Protégé является расширяемым приложением с открытым кодом, которое теперь доступно как бесплатное ПО по условиям открытого кода Mozilla Public License и совместимо с широким набором языков представления знаний.

Эта система позволяет пользователям строить онтологию предметной области, настраивать формы для ввода описаний экземпляров и вводить сами данные. Этот инструмент может быть легко расширен для работы с другими, основанными на знаниях, приложениями. Например, графические элементы (widgets) могут быть добавлены для таблиц и диаграмм. Protégé может также использоваться другими приложениями для получения доступа к данным. Protégé позволяет пользователям одновременно работать с понятиями (классами) и их экземплярами. Это обеспечивается единым графическим интерфейсом, верхний уровень которого состоит из перекрывающихся панелей (закладок) для компактного представления информации.

Панель «Classes» используется для определения классов и иерархий классов, слотов и ограничений на значения слотов, отношений между классами и свойств этих отношений. Панель «Instances» используется для ввода экземпляров понятий (классов), описанных в онтологии. Для ввода описаний экземпляров используются формы, основанные на типах слотов (свойств) понятий, которые были описаны. Созданная автоматически системой форма для описания экземпляров может быть изменена путем задания порядка расположения полей на экране, изменения их размера, заголовков и свойств слотов.

Основным предположением Protégé 2000 является то, что системы, основанные на знаниях, обычно являются очень дорогими для построения и поддержки. Например, предполагается, что разработка систем, основанных на знаниях, выполняется командой, которая включает как разработчиков онтологии, так и экспертов в предметной области, которые могут быть менее знакомыми с компьютерной программой. Protégé 2000 спроектирован таким образом, чтобы направлять действия разработчи-

ков и экспертов в процессе разработки системы. Он позволяет разработчикам повторно использовать онтологии и методы решения задач, тем самым сокращая время, требуемое для их разработки и поддержки. Несколько приложений могут использовать одну и ту же онтологию для решения различных задач, и один и тот же метод решения задач может использоваться с различными онтологиями.

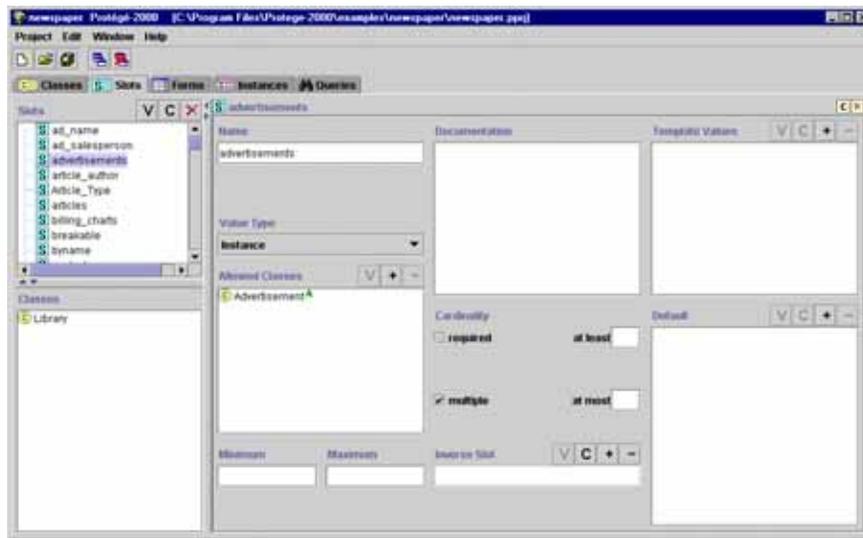


Рис. 6.4. Окно системы Protégé 2000

Основные особенности системы Protégé 2000:

- Импортируемые форматы
 - XML, RDF(S), XML Schema и OWL при использовании plug-in.
- Экспортируемые форматы
 - XML, RDF(S), XML Schema, FLogic, CLIPS and Java HTML.
- Графическое представление
 - с помощью GraphViz plug-in (просмотр классов и глобальных свойств);
 - с помощью Jambalaya plug-in (вкладываемое графическое представление).
- Проверка согласованности
 - с использованием plug-ins (PAL and FaCT).

- Ограниченная многопользовательская поддержка
 - в Protégé 2.0 были добавлены новые возможности поддержки многопользовательской работы. Это предназначалось для опытных пользователей Protégé. Группа пользователей может читать одну и ту же БД и последовательно выполнять ее изменения или вносить изменения, которые не приводят к взаимным конфликтам. Однако нет поддержки ситуации, когда несколько пользователей пытаются изменять один и тот же элемент онтологии или оповещений об изменениях, сделанных другими пользователями. Одновременное изменение одной и той же части онтологии может привести к проблемам.
- Web-поддержка
 - с помощью Protégé-OWL plug-in;
 - Protégé не предоставляет прямой возможности доступа к онтологии через Web-сеть, но это может быть просто сделано. Многие пользователи работают с удаленными онтологиями в Web-сети с использованием servlets. В этом случае Protégé может выполняться как applet. Сейчас доступна новая версия Protégé Web Browser. Он может быть загружен со страницы plug-ins системы Protégé. The Protégé Web Browser дает пользователям возможность просматривать Protégé онтологии и БЗ в их Web-браузерах, без необходимости устанавливать локально приложение Protégé.
- Дополнительные особенности:
 - слияние с помощью Anchor-PROMPT plug-in;
 - нет возможности добавить новый базовый тип;
 - расширенная архитектура plug-in.
- Хранение онтологий
 - файл и СУБД (посредством JDBC).

В качестве примера использования Protégé 2000 для описания онтологий на рис. 6.5 приводится онтология «Управление знаниями». Этот пример, с одной стороны, иллюстрирует работу программного инструмента Protégé 2000, а с другой – структуру столь сложного понятия, как «Управление знаниями», рассмотрению которого посвящены основные разделы настоящей монографии. Для создания онтологии «Управление знаниями» использовался редактор Protégé версии 3.0, разработанный в лаборатории медицинской информатики медицинского университета Стэнфорда.

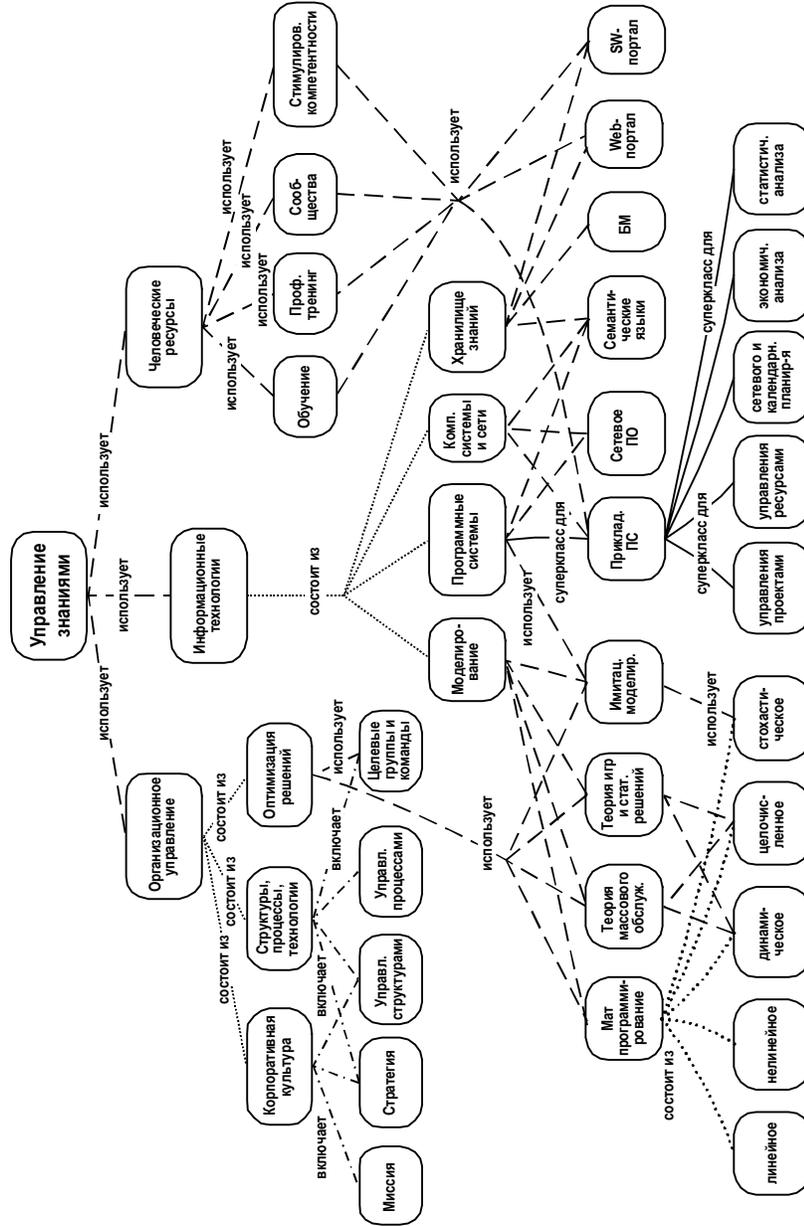


Рис. 6.5. Таксономия понятия «Управление знаниями» и используемых теорий, методов, процессов

Графический интерфейс редактора (рис. 6.6) позволяет инженеру по знаниям визуально конструировать онтологию, не задумываясь о том, как она будет храниться в памяти компьютера.

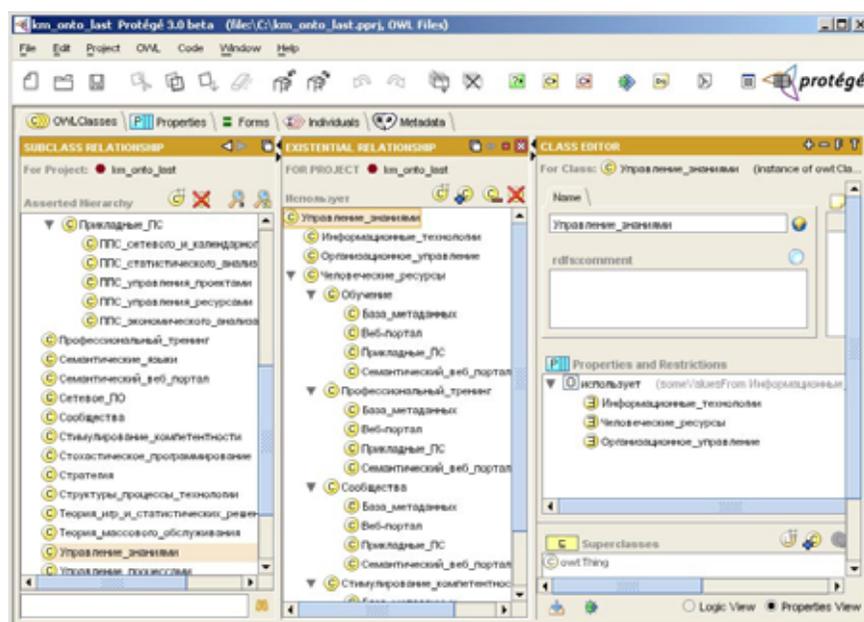


Рис. 6.6. Интерфейс редактора онтологий Protégé 3.0

Воспользовавшись соответствующей функцией Protégé, можно сохранить онтологию в одном из доступных форматов представления знаний. Для увеличения числа форматов, доступных Protégé, могут быть подключены соответствующие модули расширения редактора – плагины. В данном случае онтология была записана на языке OWL (Ontology Web Language) с использованием OWL-плагина.

Фрагмент описания онтологии на языке OWL представлен в листинге 6.1. Поскольку OWL относится к семейству языков, основанных на XML (Extensible Markup Language), в первой строке документа размещается декларация XML. Далее внутри тэга RDF перечисляются пространства имен соответствующих языков. Каждому префиксу пространства имен ставится уникальная строковая константа. Например, строка

`xmlns:owl="http://www.w3.org/2002/07/owl#"` означает, что префиксу `owl` соответствует строковая константа `"http://www.w3.org/2002/07/owl#"`.

В соответствии с концепцией пространств имен эти префиксы используются для однозначной идентификации элементов в документе XML. Запись `owl:Class` будет отсылать нас к элементу `Class`, объявленному на пространстве имен `owl` (запись `owl:Class` эквивалентна записи `http://www.w3.org/2002/07/owl#Class`).

Далее после элемента `owl:Ontology` идет непосредственное описание онтологии «Управление знаниями» в виде набора понятий (элемент `owl:Class`) и отношений между ними. Так, записи элементов `<owl:some Values From rdf: resource="# Организационное управление"/>`, `<owl:some Values From rdf: resource="# Человеческие ресурсы"/>`, `<owl:some Values From rdf: resource="# Информационные технологии"/>` свидетельствуют, что они связаны классом `<owl:class rdf: ID="# Управление знаниями"/>`.

Фрагмент описания онтологии «Управление знаниями» на языке OWL приводится в листинге 6.1. Он охватывает три уровня таксономии (декомпозиции) понятия «Управление знаниями».

Листинг 6.1

Описание онтологии «Управление знаниями» на языке OWL

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-
ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Управление_знаниями">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom
rdf:resource="#Информационные_технологии"/>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="использует"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
</rdf:RDF>
```

```

        </owl:Restriction>
    </rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:someValuesFrom
rdf:resource="#Организационное_управление"/>
        <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:someValuesFrom
rdf:resource="#Человеческие_ресурсы"/>
        <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Информационные_технологии">
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:someValuesFrom>
                <owl:Class rdf:ID="Моделирование"/>
            </owl:someValuesFrom>
            <owl:onProperty>
                <ObjectProperty rdf:ID="состоит_из"/>
            </owl:onProperty>
        </owl:Restriction>
    </rdfs:subClassOf>
<rdfs:subClassOf>
        <owl:Restriction>
            <owl:someValuesFrom>
                <owl:Class rdf:ID="Программные_системы"/>
            </owl:someValuesFrom>
            <owl:onProperty rdf:resource="#состоит_из"/>
        </owl:Restriction>
    </rdfs:subClassOf>
<rdfs:subClassOf>
        <owl:Restriction>
            <owl:someValuesFrom>
                <owl:Class rdf:ID="Хранилище_знаний"/>
            </owl:someValuesFrom>
        </owl:Restriction>
    </rdfs:subClassOf>

```



```

    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Человеческие_ресурсы">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom>
        <owl:Class rdf:ID="Сообщества"/>
      </owl:someValuesFrom>
      <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom>
        <owl:Class
rdf:ID="Профессиональный_тренинг"/>
      </owl:someValuesFrom>
      <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom>
        <owl:Class rdf:ID="Обучение"/>
      </owl:someValuesFrom>
      <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom>
        <owl:Class
rdf:ID="Стимулирование_компетентности"/>
      </owl:someValuesFrom>
      <owl:onProperty rdf:resource="#использует"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

<!-- Created with Protege (with OWL Plugin 1.3, Build 196) <http://protege.stanford.edu> -->

В табл. 6.2 приведен обширный, но далекий от исчерпывающего список доступных образцов инструментов и краткое описание их возможностей [Polikoff I.]. Ряд коммерческих инструментов и систем поддерживает многие, иногда все, стадии жизненного цикла онтологий. В этом случае инструменты помещены в категорию, в которой их возможности реализованы наиболее сильно.

Таблица 6.2

Инструменты и системы для работы с онтологиями

№ п/п	Инструмент	Возможности	Изготовитель	Соответствие стандартам и общим соглашениям
1	2	3	4	5
Этап жизненного цикла: Создание				
1	Protégé 2000	Создает иерархию понятий, создает экземпляры, возможен просмотр в нескольких форматах. Инструмент для одного пользователя	Stanford University KSL	Открытый исходный код (Mozilla Public License); архитектуры plug-in. Поддерживает RDF, DAML+OIL. С помощью plug-in поддерживается OWL
2	OntoEdit	Создает иерархию понятий, создает экземпляры. Интегрирована с распределенными БД. Инструмент для одного пользователя	Ontoprise	Коммерческая система. Поддерживает RDF, DAML+OIL, имеет plug-in архитектуру
3	OilEd	Создает иерархию понятий, создает экземпляры, анализирует семантическую согласованность (в соответствии с ДЛ). Инструмент для одного пользователя	University of Manchester	Поддерживает RDF, DAML+OIL. Разработана создателями языка OIL. Бесплатный редактор онтологий, интегрирован с системой логического вывода ДЛ (DL reasoner)
4	Medius Visual Ontology	Создает онтологии. Есть некоторая поддержка совместной работы пользо-	Sandpiper Software	Расширяет язык UML, требуется система Rational Rose (Enterprise

	Modeler	вателей		Ed.). Поддерживает языки RDF и DAML + OIL. Включает библиотеку онтологий, которые описывают IEEE Standard Upper Ontology (SUO)
5	Cerebra Construct	Развитый набор инструментов по созданию онтологий (с поддержкой логического вывода в реальном времени).	Network Inference	Коммерческая версия редактора онтологий OiEd, расширенная до интеграции с коммерчески доступным

Продолжение табл. 6.2

1	2	3	4	5
		Позволяет использовать начальную онтологию, поглощение существующих онтологий, таксономию баз данных и создание оберток (wrapping) с использованием онтологий		графическим редактором (Ms Visio), позволяет коллективную разработку. Первая версия вышла в марте 2003 г. Поддерживает языки RDF, DAML+OIL, OWL, SOAP интерфейс
6	LinKFactory Workbench	Среда для коллективной разработки онтологий. Первоначально проектировалась для очень больших медицинских онтологий. Имеет Java beans API и по желанию Генератор Приложений для семантического индексирования (формирования метаданных), автоматического кодирования и извлечения информации. Сравнивает и связывает онтологии с помощью базовой онтологии; связанные понятия выявляются на основе формальных отношений и лексической информации	Language and Computing	Поддерживает языки RDF(S), DAML+OIL и OWL. Имеется некоторая поддержка заполнения и сопровождения онтологий
7	K-Infinity	Среда для коллективной разработки онтологий	Intelligent Views	Модульный набор инструментов, которые

Продолжение табл. 6.2

1	2	3	4	5
				поддерживают все этапы ЖЦ. Поддерживает RDF и Topic Maps
Этап жизненного цикла: Заполнение				
8	OntoAnnotate	Выбирает экземпляры из Web-страниц. Выполняет семантическую разметку	Ontoprise	Интегрирован с браузером MS Internet Explorer. Поддерживает языки RDF, DAML+OIL
9	OntoMat	Выбирает экземпляры из Web-страниц. Выполняет семантическую разметку	AIFB University Karlsruhe	Свободно распространяется. Использовалась в проекте DARPA – OnTo-Agents построена на языке JAVA и предоставляет интерфейс для модулей расширения (plugin interface). Совместима с языком DAML+OIL
10	AeroDAML	Выполняет анализ естественного языка документа для выполнения разметки	Lockheed-Martin	Демонстрационная версия доступна в виде Web-сервиса. Совместима с языками RDF, DAML+OIL и OWL
11	CORPORUM OntoBuilder	Набор инструментов: Ontowrapper – для работы со структурированными документами. OntoExtract – строит начальные онтологии/таксономии из документов на естественном языке. Представление понятий и отношений, извлеченных из содержания документа, может быть расширено информацией из WordNet	CognIT AS	Развилась из проекта OnToKnowledge. Поддерживает языки RDF, DAML+OIL. Требуется Sesame RDF-репозиторий. Основное внимание уделяется автоматическому созданию редактируемых онтологий из документов на естественном языке. Также поддерживает создание метаописаний
12	Freedom Enterprise	Семантическое улучшение метаданных отноше-	Semagix	Поддерживает языки XML и RDF, а также

Продолжение табл. 6.2

1	2	3	4	5
	Semantic Platform	ниями и понятиями для языка, структуры и потребностей отрасли. Автоматическая управляемая онтологией классификация и семантическое аннотирование разнородного контента		развертывание и создание онтологий. Включает компоненту Knowledge Toolkit для построения онтологий
Этап жизненного цикла: Использование				
13	Orengе	Выполняет поисковые запросы на естественном языке с помощью онтологий. Поддерживает возможность поиска, основанного на понятиях	Empolis	Также имеется возможность создания онтологий. Поддерживает стандарты XML и Topic Map
14	Freedom Enterprise Semantic Platform	Категоризация и поиск с помощью онтологий. Агрегирование и нормализация контента из большого разнообразия источников. ESP является прикладной платформой для семантической интеграции разнородного контента, включая медиа и БД предприятия. Поддерживаемые возможности: Автоматическая разметка понятий, навигационный поиск, поиск, основанный на понятиях	Semagix	Также имеются возможности для создания и наполнения онтологий. Поддерживает языки XML и RDF
15	OntoBroker	Предоставляет среду для обработки правил, организованных онтологией. Поддерживаемые возможности: Система автоматизации проектирования продукции	Ontoprise	Поддерживает языки RDF, DAML+OIL
16	Semantic Miner	Создает семантически осмысленные запросы из	Ontoprise	Поддерживает языки RDF, и DAML+OIL

Продолжение табл. 6.2

1	2	3	4	5
		запросов на естественном языке		
17	OntoOffice	<p>Предоставление контента, основанного на онтологиях.</p> <p>Поддерживаемые возможности:</p> <p>Поисковая система, учитывающая контекст.</p> <p>Во время работы с MS Word/Excel/Outlook, OntoOffice проверяет введенный текст и отмечает понятия и экземпляры используемой онтологии как SmartTags. Используя SmartTag, можно проверить свойства экземпляров и найти другие экземпляры, подходящие по контексту</p>	Ontoprise	Интегрирован с Microsoft Office. RDF, поддерживает язык DAML+OIL
18	Contextia	<p>Интегрирует данные и приложения предприятия. Принимает данные во множестве форматов для нахождения соответствия и моделирования, включая XML-схемы, собственные схемы, таблицы БД и файлы с разделителями.</p> <p>Может работать самостоятельно или совместно с существующей ИТ-инфраструктурой, такой, как EAI, B2Bi, управление бизнес-процессами, брокерами сообщений и приобретенными соединителями (off-the-shelf connectors).</p> <p>Может выполнять преоб-</p>	Modulant	<p>Включает подсистему FirstStep XG для создания онтологий. Есть некоторая поддержка их анализа: Используется Экспресс-Модель (ISO 10303) для проверки правильности (validation); согласованности между онтологиями.</p> <p>Поддерживает язык XML, Web-сервисы и SOAP</p>

Продолжение табл. 6.2

1	2	3	4	5
		разование информации, включающей сложные элементы данных, вложенные структуры и неполную или конфликтующую семантику. Поддерживаемые возможности: Semantic Data Integrator, Semantic Application Integration		
19	Cerebra Server	Семантическая интеграция данных и приложений предприятия с использованием системы логического вывода Cerebra. Это коммерческая версия семантической системы логического вывода, разработанной в OilED. Планируется полная платформа инструментов, основанных на Cerebra engine. Поддерживаемые возможности: Семантическая интеграция данных, Семантический генератор форм и классификатор результатов, Knowledge Pulse	Network Inference	Также поддерживает заполнение онтологий. Включает поддержку и развитие онтологий с возможностью добавления новых аксиом/отношений без приостановки работы системы. Поддерживает языки RDF, DAML+OIL, OWL, интерфейс SOAP
20	Tucana Knowledge-Store	Распределенная БД, специально спроектированная для метаданных и управления метаданными. БД построена для хранения и поиска метаданных с очень быстрой скоростью выполнения, при этом постоянно под-	Plugged In Software	Также позволяет создавать и заполнять онтологии с помощью системы Tucana Metadata Extractor™. Поддерживает язык RDF и интерфейсы SOAP, COM и Java

Продолжение табл. 6.2

1	2	3	4	5
		держиваются целостность и санкционированность доступа		
Этап жизненного цикла: Деятельность при проверке на этапе Поддержки и Развития				
21	ConVisOr	Проверяет соответствие онтологии дескриптивной логике	Lockheed Martin	
22	OntoClean (ODE)	Анализирует согласованность онтологий	University of Madrid	Исследовательский прототип, совместим с языком DAML+OIL

Окончание табл. 6.2

1	2	3	4	5
23	OilED	Анализирует согласованность онтологий на основе дескриптивной логики	University of Manchester	В свободном доступе. Интегрируется с MS Office. Поддерживает языки RDF, DAML+OIL
24	Cerebra Inference Engine	Анализирует согласованность и делает выводы на основе дескриптивной логики	Network Inference	Коммерческая версия OilED. Поддерживает развертывание. Также позволяет поддерживать и развивать онтологии путем добавления новых аксиом/отношений без приостановки работы системы. Поддерживает языки RDF, DAML+OIL, OWL, интерфейс SOAP
Этап жизненного цикла: Деятельность по слиянию на этапе Поддержки и Развития				
25	PROMPT	Поддерживает слияние двух и более онтологий	Stanford KSL	Модуль расширения (plug-in) для редактора онтологий Protege 2000

1	2	3	4	5
26	Chimera	Позволяет обрабатывать несколько онтологий вместе, проводит анализ для определения возможности слияния онтологий	Stanford University KSL	Планируется поддержка языков RDF и OWL
27	FCA-Merge	Производит слияние онтологий снизу-вверх, основываясь на общих экземплярах понятий	AIFB, Karlsruhe	Исследовательский прототип

6.4. Программные инструменты для обеспечения логического вывода

В ходе работы системы управления знаниями необходимо выполнять логические преобразования элементов модели, логические операции над ними, а также производить порождение новых знаниевых утверждений из уже существующих. Для выполнения вышеперечисленных задач в состав системы управления знаниями включаются специальные программные инструменты логического вывода.

Как отмечалось ранее, для реализации логического вывода на формальной структуре модели предметной области используется специальный класс так называемых дескриптивных (описательных) логик. В настоящее время разработаны различные программные инструменты логического вывода. Некоторые из них являются бесплатными и свободно распространяемыми средствами, а некоторые – коммерческими. Примерами таких систем является FaCT, RACER, Cerebra Network Inference.

6.4.1. Система FaCT (Fast Classification of Terminologies)

Система FaCT [Patel-Schneider P.F., Hogrocks I., 1999] реализована на языке Common Lisp и использует интерфейс CORBA (рис. 6.7). FaCT поддерживает модели дескриптивной логики SHIQ (инверсные роли и численные ограничения на количество связей), но не работает с типизированными данными (например, Integer или String). Система поддерживает основные операции логического вывода – satisfiable, subsumes, equivalent, all_subs, all_supers. Кроме этого, поддерживаются две спецификации языка KRSS и XML-подобных (ни OWL, ни DAML+OIL не поддерживаются).

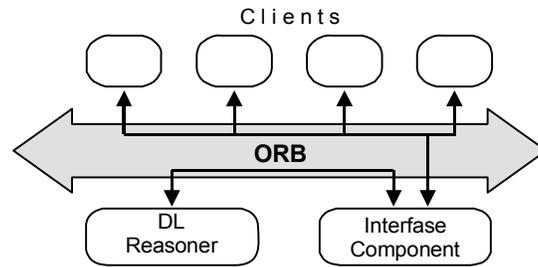


Рис. 6.7. Архитектура сервера FaCT

Имеется также версия FaCT, работающая по интерфейсу DIG (протокол взаимодействия между модулем логического вывода и клиентом). Взаимодействие осуществляется через выражения XML, передающиеся по протоколу HTTP.

Существует новая версия данной системы – FaCT++, которая реализована на языке C++ и поддерживает полную спецификацию OWL-Lite. Данная версия также включает набор новых методов оптимизации вывода.

Так как FaCT эмулирует работу CORBA-брокера, то соединение с ним возможно установить с использованием протокола HTTP. Пример работы с системой FaCT через программный код можно видеть на рис. 6.8.

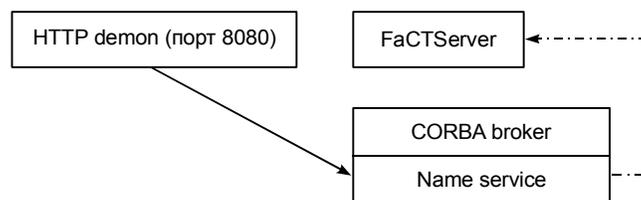


Рис. 6.8. Схема взаимодействия клиента с сервером FaCT

Листинг 6.2

Пример программной передачи примитивов в FaCT

```
chnd.begm_transaction();
chnd.clear();
chnd.defconcept("MAN");
chnd.defconcept("WOMAN");
```

```

chnd.impliesC("<PRIMITIVE
NAME=\"WOMAN\"/>",
"<NOT><PRIMITIVE NAME=\"MAN\"/></NOT>");
chnd.end_transaction();

```

В листинге 6.2 задаются два класса MAN и WOMAN и аксиома, что WOMAN \leftarrow not (MAN). Запрос к системе производится через метод satisfiable: chnd.satisfiable("<AND><PRIMITIVE NAME=\"WOMAN\"/><PRIMITIVE NAME=\"MAN\"/></AND>"). Метод возвращает результат типа bool (в данном примере – FALSE).

Система FaCT, являясь постоянно развивающимся проектом, не свободна от ряда недостатков. Она в большей степени является инструментом для исследований и мало пригодна для использования в реальных информационных системах. В ней не поддерживаются стандарты описания знаний, такие, как OWL или DAML+OIL. Однако FaCT полностью соответствует требованиям на функциональность вышеупомянутых стандартов. Для загрузки онтологии требуется промежуточный модуль-транслятор. Ранние версии FaCT не поддерживают вывод на ABOX (данных о версии FaCT нет)

6.4.2. Система логического вывода RACER

На данный момент эта программная система поддерживает язык с максимальной выразительностью (SHIQ_R⁺(D-)) и максимальную скорость работы, по сравнению с другими известными инструментами [Haarslev V., Moller R., 2001, 2003] (рис. 6.9). RACER также поддерживает ряд функций по обеспечению многопоточного вывода на разных вычислительных установках. Обеспечивает вывод как на TBOX, так и на ABOX. RACER поддерживает формат LISP подобного языка и работает только через HTTP протокол, что влечёт простоту и надёжность в работе системы в целом. Например, на команду (taxonomy) RACER выдаёт текущую таксономию в формате триплетов.

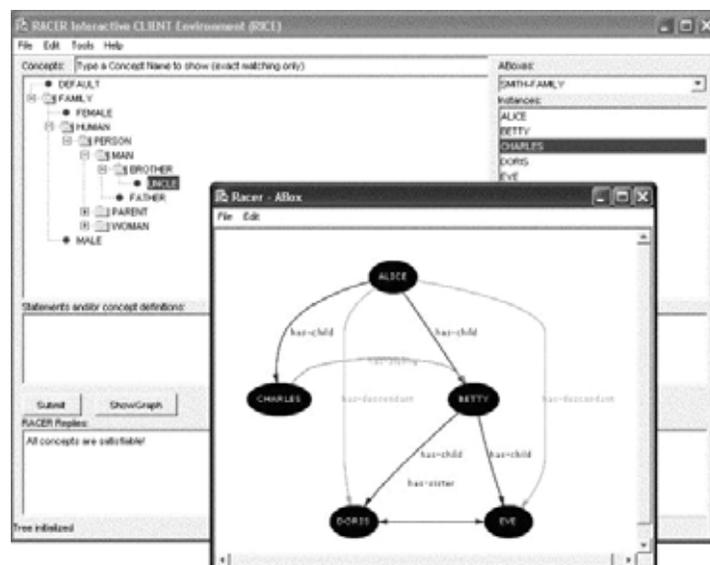


Рис. 6.9. Визуальная среда разработки и отладки RICE

RACER реализует логику большой выразительности и универсальный интерфейс DIG для взаимодействия с внешними приложениями, имеет широкий спектр операций и высокую производительность. RACER поддерживает работу через прокси для реализации распределенных вычислений. Имеет интеграцию с редактором онтологий OilEd 3.5 и выше (в виде plug-in). Имеет собственную среду отладки (RICE). Следует, однако, отметить, что реализация интерфейса DIG выполнена с некоторым отступлением от стандартов.

6.4.3. Программная среда для обеспечения логического вывода

Структура типичной платформы системы управления знаниями должна состоять из: сервера онтологий, модуля логического вывода, модуля уровня представления знаний, модуля уровня представления дескриптивной логики, ряда модулей-трансляторов и модулей-утилит. Рассмотрим состав и назначение каждого из модулей более подробно.

Сервер онтологий представляет собой отдельное автономное приложение, обеспечивающее функции хранилища онтологий, а также функции по загрузке, редактированию и получению онтологий в раз-

личных форматах. В зависимости от требований системы, возможно также реализация сервером онтологий функций по выполнению простых запросов относительно содержания онтологии. Запросы подобного рода не требуют логических рассуждений для формирования результата и реализуются путём трансформирования в один или несколько SQL-запросов к СУБД, в которой хранится онтология.

Модуль логического вывода, как описано выше, реализует базовые функции и ряд дополнительных функций. Так, например, существует возможность в ряде модулей логического вывода (например, [Haarslev V., Moller R., 2001]) после полной загрузки онтологии в память производить запросы вида ("ВСЕ ПРЯМЫЕ ПОТОМКИ", "ВСЕ ПОТОМКИ, ВСЕ ПРЕДКИ", "МЕСТО ЭКЗЕМПЛЯРА В ТОКСОНОМИИ", ...). Выполнение запросов подобного рода является неэлементарными действиями и всегда возможно на клиентской стороне, однако модуль логического вывода имеет ряд оптимизирующих алгоритмов для выполнения подобных операций.

Модуль уровня представления знаний является, вероятно, самым используемым модулем в системах управления знаниями. Роль данного модуля – представить все элементы онтологии в виде программных экземпляров классов (не путать с классами в онтологии!). Данный модуль направлен на использование программистами, создающими систему управления знаниями, и служит для облегчения работы с платформой. Рассмотрим пример: в базе знаний имеется описание нескольких классов PERSON, MAN и WOMAN. При этом классы MAN и WOMAN являются подклассами класса PERSON. Модуль представления знаний создаст экземпляры программных классов TClass, которые будут иметь следующие значения атрибутов у PERSON: Name = "PERSON", Childrens = {MAN, WOMAN}; Parents = {TOP}; у MAN: Name = "MAN"; Childrens = {}; Parents = {PERSON}; у WOMAN: Name = "WOMAN"; Childrens = {}; Parents = {PERSON};

Модуль уровня представления дескриптивной логики – это модуль для представления выражений дескриптивной логики программными классами. При этом формула, записанная при помощи данных классов, представляет собой некий односвязный граф. Разберём пример: Допустим нам необходимо записать выражение «все экземпляры, которые являются сотрудниками, работают в отделе информатизации и имеют по крайней мере два сертификата». Этому выражению будет соответствовать следующая формула в текстовом представлении:

$$\text{worker} \wedge \exists \text{work_in.IT_department} \wedge \text{atLeast}(\text{has_certification}, 2)$$

После представления формулы в виде классов эта запись будет представлена в памяти компьютера в виде следующего графа (рис. 6.10):

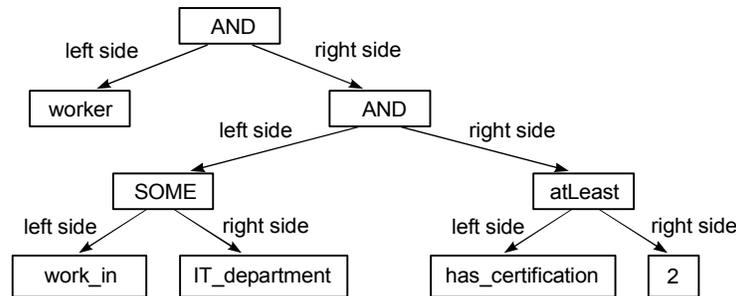


Рис. 6.10. Граф представления высказывания

Данный вид представления весьма удобен, так как алгоритмы, обрабатывающие данные структуры, весьма тривиальны. Так, например, для формирования строкового представления данной структуры необходимо использовать рекурсивный алгоритм. Необходимость трансформирования возникает в следующих случаях:

- необходимо представить структуру в виде строки для визуализации высказывания (например, во время отладки приложения);
- необходимо трансформировать высказывание в XML-выражение в соответствии со стандартом DIG для последующей передачи модулю логического вывода.

Модули - утилиты взаимодействуют с остальными модулями и предоставляют функциональность более высокого уровня. Так, например, все операции, в ходе выполнения которых необходимо составлять запросы к модулю логического вывода, трансформировать знания, представленные в онтологии, в высказывания дескриптивной логики и разбирать полученные ответы, находятся в модулях-утилитах.

В рамках исследований по построению системы управления знаниями Ralf Muller (Fachhochschule Wedel – University of Applied Sciences) и Volker Haarslev (Concordia Univ., Montreal) разработали структуру подобной системы (рис. 6.11). В ходе дальнейших экспериментов было установлено, что уже имеющиеся платформы для реализации подобных систем обладают схожей структурой. В процессе стыковки и отладки

различных частей системы управления знаниями были использованы стандартные модули и программные продукты, такие, как MS-SQL Server, RACER, RDF-DRIVER, которые облегчают реализацию блока бизнес-логики без особых усилий, связанных с реализацией базовых блоков.

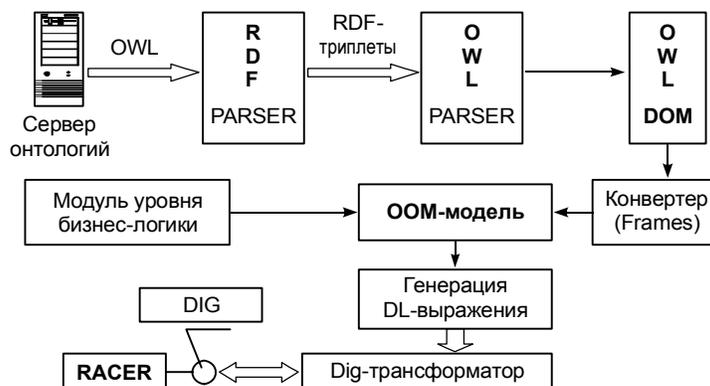


Рис. 6.11. Вариант предлагаемой структуры программной системы

Как было отмечено ранее, существует ряд готовых платформ для реализации систем управления знаниями. Одной из таких платформ, существующих в рамках исследовательского проекта WonderWeb (IST программа), является платформа KAON [Volz R., Oberle D., Staab S., Motik B., 2003].

KAON – это набор программных средств для разработки семантических распределенных информационных систем. В её состав входят как сервера приложения, так и пользовательские приложения. Упрощенная структура KAON представлена на рис. 6.12.

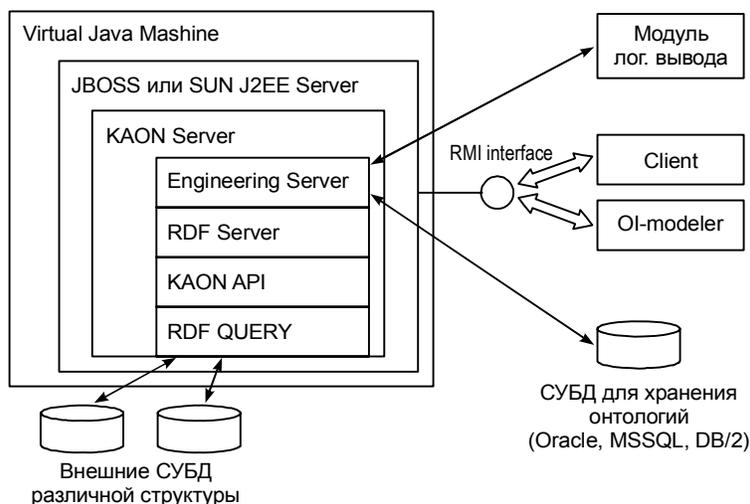


Рис 6.12. Упрощенная структура KAON

Ядром KAON можно считать KAON Server, выполняющий роль сервера приложений и имеющий отлаженный механизм для регистрации и выполнения модулей, взаимодействующих по RMI или SOAP протоколу с клиентским интерфейсом-заместителем. KAON Server позволяет регистрировать новые компоненты без перезапуска самого сервера (т.е. «на лету»), а также поддерживает загрузку новых модулей через Интернет по протоколу HTTP. Эти качества наделяют его большой гибкостью и дают возможность использовать в построении распределённых приложений, а также делают его весьма удобным инструментом, участвующим в процессе коллективной разработки приложений подобного рода. KAON Server содержит встроенную онтологию для описания модулей, располагающихся в нем, и предоставляет унифицированный доступ к серверным элементам (модулям) одной функциональной группы. По мнению разработчиков, сервер, имея в распоряжении несколько модулей, обеспечивающих схожую функциональность, но различных по реализации, принимает решения, с каким именно модулем будет работать конкретный клиент. Под клиентом здесь понимается программный модуль на стороне клиента (но не конечный пользователь). Решения принимаются на основании имеющегося описания программных модулей и требований, предоставляемых со стороны клиента. (Так, напри-

мер, клиент не запрашивает конкретный модуль RACER или FaCT, а делает запрос на модуль логического вывода. Сервер, в свою очередь, предоставляя унифицированный интерфейс, является к тому же посредником.) Возможен также «ручной» вариант выбора компонент, когда сервер предоставляет описание модулей из функциональной группы, запрашиваемых клиентом, а клиент самостоятельно решает, какой конкретный модуль использовать.

KAON предлагает базовую инфраструктуру, на основе которой возможно построить семантические приложения. В состав инфраструктуры входят: KAON ENGINEERING SERVER, KAON API, KAON QUERY.

Подсистема KAON ENGINEERING SERVER является специализированным приложением, обеспечивающим хранение, загрузку/выгрузку онтологий и ориентированным на транзакции, использующиеся при редактировании существующих онтологий. Сервер работает с любой базой данных, удовлетворяющих ряду требований, через унифицированный интерфейс ODBC (проверялась на базах: PostgreSQL, IBM DB2, Oracle 9i). Однако разработчики рекомендуют использовать систему JBOSS 3.0. В последнем случае сервер реализует возможность отслеживать в загруженной клиентом онтологии изменения, сделанные в это время другим клиентом, и уведомлять об этом клиента. Сервер обладает механизмом массовой загрузки/удаления элементов онтологии. Так, тесты показали следующее: загружались 20 онтологий, содержащих 100000 концептов, 66000 атрибутов и 1000000 экземпляров – время на обработку составило 20 с. Доступ к серверу обеспечивается через KAON API удалено, через протоколы RMI или SOAP, так что с точки зрения клиентского приложения он, фактически, работает с локальной копией сервера. KAON API успешно реализует кэширование данных на стороне клиента и механизм, ответственный за когерентность кэш и данных в БД.

В состав системы включена библиотека классов KAON API, которая предоставляет программный доступ к элементам онтологий и непосредственно описываемым объектам системы независимо от механизма хранения и представления знаний в реляционной базе данных. В системе поддерживается возможность работы с мультионтологией, состоящей из множества подонтологий (функция модуляризации). KAON API предоставляет функции по работе со знаниями, а именно: выполнение стандартных запросов на проверку когерентности (непротиворечивости) онтологии или выполнение запросов вида «subsume» и «satisfiable».

Из приведенного описания видно, что система KAON не решает производственных задач для конкретной организации и тем более не может выступать в качестве готового универсального решения. Однако, являясь набором инструментов, она выступает в качестве отправной точки для создания конечной информационно-знаниевой системы с реализованной бизнес-логикой для конкретного предприятия и пользовательским интерфейсом (например, Web-портал).

В состав KAON входят так называемые модули расширения функциональности, среди которых модуль RDF QUERY, позволяющий не только объединить на логическом уровне базы данных различных форматов и структур, но и включить их в единое информационное поле организации. В дальнейшем, при выполнении логического вывода (в процессах поиска или категоризации документов или поиска аналогичной производственной ситуации), информация, содержащаяся в базах данных и знаний корпорации, будет участвовать в процессе логических рассуждений, а значит, будет непосредственно влиять на ход выполнения всех основных процессов компании.

Глава 7

СИСТЕМЫ УПРАВЛЕНИЯ ЗНАНИЯМИ

7.1. Назначение и архитектура систем управления знаниями

В общепринятом понимании, управление знаниями – это установленный в корпорации четкий порядок работы с информационно-знаниевыми ресурсами и специалистами в определенных сферах деятельности для облегчения доступа к знаниям и повторного их использования с помощью современных информационных технологий.

Организации, способные быстро и легко распространять ценную информацию через созданную у них инфраструктуру, могут управлять запасами своих знаний. Актуальные сведения могут включать запасы явных знаний (бумажные документы, электронные документы, базы данных, сообщения электронной почты, текстовые файлы, изображения и даже видеофрагменты), полученные из любых источников, а также неявные знания, хранящиеся в головах специалистов и экспертов. Необходимо, чтобы инфраструктура управления знаниями организации позволяла эффективно и многократно использовать запасы явных и неявных знаний.

Реально управление знаниями представляет собой систематический процесс создания и преобразования индивидуального и группового, научного и практического опыта таким образом, чтобы знания могли быть перенесены в процессы, услуги и продукты, предлагаемые организацией с тем, чтобы увеличивалась их общая стоимость, а соответственно и совокупная продуктивность организации.

Ключом к управлению знаниями является, таким образом, доставка нужных знаний нужным людям в пределах группы людей и организации в нужное для эффективной реализации бизнес-процессов время. По сути своей, смысл управления знаниями заключается в том, чтобы помочь людям лучше работать вместе, используя растущие ресурсы зна-

ний и эффективно управляя ими. Результатом успешно работающей системы управления знаниями, как это уже отмечалось ранее, должна стать знающая, самообучающаяся и развивающаяся организация.

Достижение столь очевидной и кажущейся простой цели на практике сталкивается с рядом серьезных препятствий объективного характера. На пути эффективного накопления, распространения, совместного и повторного использования знаний возникает ряд естественных барьеров, схематически представленных на рис. 7.1.

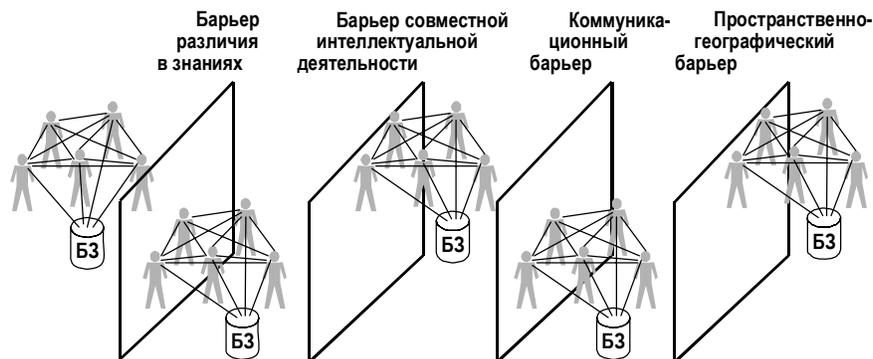


Рис. 7.1. Барьеры на пути совместной интеллектуальной деятельности

Прежде всего, это барьер существенных различий в объемах и уровнях знаний людей, который, по понятным причинам, носит латентный характер. За ним следует барьер готовности личностей и групп к совместной интеллектуальной деятельности. Серьезным препятствием является и коммуникационный барьер, без преодоления которого невозможен оперативный обмен и распространение разных форм и видов знаний даже в пределах группы, не говоря уже об организации в целом. Наконец, пространственно-географический барьер, вызванный взаимной удаленностью мест приложения труда в современных крупных вертикально или как-то иначе интегрированных компаниях, не говоря уже о транснациональных корпорациях.

Для преодоления перечисленных барьеров и достижения желаемых целей корпоративные СУЗ должны, очевидно, удовлетворять ряду требований, а именно:

- обеспечивать систематизированное описание, классификацию и агрегацию знаний;

- обеспечивать фиксацию и хранение явных и неявных знаний;
- обеспечивать накопление и обновление знаний, доступ к ним работников компании со своих рабочих мест;
- обеспечивать распространение знаний в соответствии с установленным регламентом (кому, что, когда) и по запросам;
- обеспечивать разведку знаний, семантический поиск и навигацию по знаниям;
- поддерживать в компании язык профессионального общения;
- поддерживать широкий спектр средств (среду) для профессионального общения и обмена знаниями между специалистами, экспертами, командами, проектными группами и сообществами по интересам;
- обеспечивать хранение и обновление профилей компетентности специалистов компании, поддерживать процессы экспертизы и консультирования;
- содействовать интенсификации процесса генерации новых идей, их апробации и использованию в бизнес-процессах;
- обеспечивать поддержку процесса принятия решений;
- система должна быть прозрачной для пользователей, то есть не должна создавать дополнительных трудностей во взаимодействии с СУЗ, отличных от взаимодействия с информационной системой компании.

Исходя из перечисленных требований, можно утверждать, что система управления знаниями в общем плане предназначена для усиления интеллектуального потенциала организации и повышения уровня ее корпоративной культуры. Более конкретно, функциональное предназначение СУЗ состоит в следующем:

- систематизация и накопление актуальной информации и знаний;
- обеспечение универсальной доступности сотрудников к коллективной памяти компании, к явным и неявным знаниям;
- культивирование языка профессионального общения на основе мета-данных и онтологического описания предметной области;
- обеспечение семантического поиска и фильтрации знаний в корпоративных и мировых информационных ресурсах;
- стимулирование сотрудничества высокоэффективных специалистов в процессе генерации идей и принятия решений;
- создание среды для on-line взаимодействия распределенных проектных и бизнес-групп, профессиональных сообществ и экспертов;

- создание сетевой среды для повышения квалификации специалистов в режиме distant education;
- повышение корпоративной культуры в реализации управленческих и бизнес-процессов.

Исходя из приведенного описания назначения, можно сделать вывод о том, что СУЗ существенно отличается от информационной системы организации, достигшей определенных технологических высот в своем развитии. Если предназначением информационной системы является эффективное хранение, обработка и предоставление пользователям по регламенту или запросу искомой информации, то предназначением СУЗ является увеличение объема и повышение уровня использования знаний организации за счет их эффективного поиска и фильтрации, включения в коллективную память компании профессиональных знаний и опыта, в том числе хранящегося в головах специалистов.

Информационная система рассматривает специалиста как функционера, то есть как человека, выполняющего определенную функцию, которую она (система) обязана информационно обеспечить. СУЗ рассматривает специалиста как основной источник знаний, которому она (СУЗ) обязана оказывать помощь в предоставлении недостающих знаний, в использовании имеющихся знаний в процессе генерации идей и решении задач, а также в создании новых знаний.

То есть СУЗ должна быть тесно связана со специалистами организации и обеспечивать их всеми видами знаний, как имеющимися в организации, так и за ее пределами с помощью интеллектуального интерфейса к глобальным базам и банкам знаний. По сути, СУЗ должна предоставить сотрудникам организации возможность более быстрого и качественного выполнения их работы со знаниями в бизнес-процессах, в которых они участвуют в соответствии со своими обязанностями.

Новые, отличные от информационной системы, возможности в оперировании знаниями у СУЗ появляются главным образом потому, что она предоставляет специалистам компании широкий набор средств, эффективную среду для профессионального общения на более высоком уровне – на уровне моделирования бизнес-процессов с помощью теорий, концепций, достоверных знаниевых утверждений и представлений. Это достигается наличием в СУЗ языка профессионального общения, опирающегося на метаописания и онтологии предметных областей.

Как отмечает в своих исследованиях В.Ф. Турчин [Турчин В.Ф., 2000]: «Язык профессионального общения создает общее поле знаний,

Одним из таких направлений может быть целевая и функциональная ориентация, и тогда предназначение СУЗ и состав решаемых ею задач могут быть положены в основу выделения блоков и элементов, установления связей между ними.

Еще одним и не последним по числу и важности направлением могут быть система организации информационных ресурсов компании, ибо данные, информация и знания, в том числе хранящиеся в умах специалистов, их извлечение и рациональное использование в бизнес-процессах – это то, во имя чего и создаются СУЗ.

Другим основанием для структуризации СУЗ могут быть технологии и среды, состоящие из совокупности аппаратных и программных компонентов, обеспечивающих, в конечном счете, эффективное взаимодействие специалистов как в процессе решения производственных задач, так и в процессе расширенного воспроизводства знаний.

СУЗ можно, в таком случае, рассматривать как некоторую надстройку над современной информационной системой компании, развивающую ее функциональность не только в количественном, но и в качественном отношении.

Вариант архитектуры такой надстройки приведен на рис. 7.3 [Staab S., Studer R., 2002].

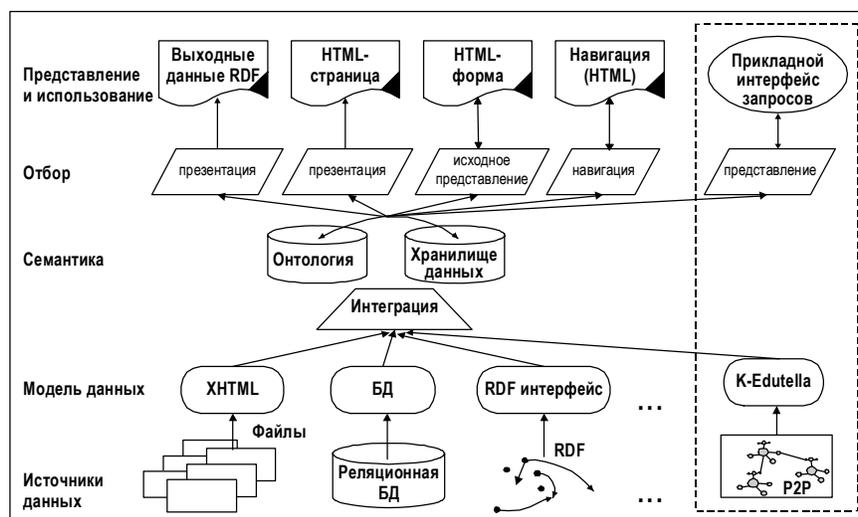


Рис. 7.3. Концептуальная архитектура системы SEAL

7.2. Корпоративная память

Корпоративная память (Corporate memory) (КП), корпоративное хранилище данных и знаний (Data Warehouse) представляют собой предметно-ориентированную, интегрированную, вариантную по времени, неразрушаемую совокупность данных, информации и знаний, предназначенную для поддержки бизнес-процессов и принятия управленческих решений.

Назначение КП как некоей подсистемы СУЗ состоит в преобразовании информационного базиса компании из аморфной эволюционирующей среды в структурированную и планомерно развивающуюся систему. Основной целью КП является предоставление информации специалистам организации, работающим со знаниями, сотрудникам, менеджерам всех уровней управления в деловом контексте и необходимом темпе времени. Потребности в данных, информации и в знаниях в современной стремительно меняющейся бизнес-среде трудно предвидеть, поэтому КП должны быть универсальными по функциям, инвариантными по типу и характеру запросов на информацию.

Хранилища данных могут быть очень внушительных размеров. Например, банк Chase Manhattan Bank имеет хранилище объемом более 560 Гбайт, компания MasterCard Online – 1,2 Тбайт. Когда все данные содержатся в едином хранилище, изучение связей между отдельными элементами данных может быть более плодотворным, а результатом анализа становятся новые знания.

Основными элементами хранения в КП являются *сущности*. Сущности (entity) обычно трактуются в корпоративных системах как объекты, вещи или события, информацию о которых необходимо хранить в КП в интересах функционирования и развития компании. Примерами сущностей могут быть объекты деятельности (месторождения, сырье, материалы), субъекты деятельности (управляющие, служащие, рабочие), средства деятельности (агрегаты, станки, инструменты), отношения (структурные, производственные, финансовые).

Помимо сущностей, в КП хранятся также метаданные. *Метаданные*, или данные о данных, обычно хранятся отдельно от данных в специальной части КП, именуемой *репозиторием*. Появившись первоначально как построчные комментарии к текстам компьютерных программ для обеспечения их сопровождения, метаданные получили в дальнейшем широкое распространение в борьбе с информационной перегрузкой.

Применительно к хранилищам данных, метаданные – это, по сути, «карта» устройства и расположения данных в хранилище.

Поскольку в КП содержатся данные и знания о многих видах сущностей, их связях и отношениях, об инструментах и средствах, с помощью которых информация и знания могут быть обнаружены, сохранены и обработаны, то отдельные части КП могут иметь собственные метаданные. Метаданными описываются сущности различных предметных областей, СУБД и программные средства доступа и обработки информации, прикладные программные системы пользователей и сферы их профессиональных интересов.

Ценность метаданных заключается в инкапсуляции информации об объектах хранимого в описании особого информационного пространства, кратно ускоряющего процесс идентификации искомого. Все это существенно облегчает исследование информации и приобретение нового знания.

На современном уровне развития информационных технологий КП, или как их еще именуют – централизованные корпоративные хранилища, имеют следующую структуру и состав (рис. 7.4) [Гаврилова Т.А., 2002].

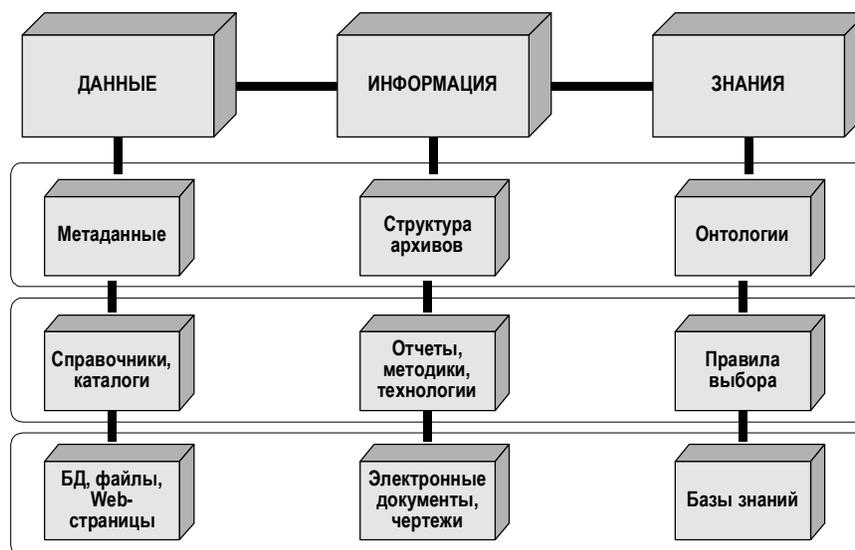


Рис. 7.4. Структура и состав корпоративного хранилища

Как следует из приведенного рисунка, в КП (в централизованном корпоративном хранилище) данные, информация и знания представляются на трех уровнях агрегации. Нижний, наиболее детальный уровень, для данных составляют базы данных, файлы, Web-страницы, для информации – электронные (оцифрованные) документы и чертежи, для знаний – базы знаний. Средний уровень – это справочники и каталоги для данных, отчеты, методики, регламенты, технологии – для информации, правила логического вывода – для знаний. Верхний уровень представляют метаданные, структура архивов (метаинформация) и онтологии для данных, информации и знания соответственно.

Предметная ориентация корпоративных хранилищ является их важным свойством. Если речь идет, например, о нефтегазовой компании, то очевидно наличие в нем данных о геологии месторождений, о нефтепроводных сетях и насосных станциях, о резервуарных парках, о технологических процессах, финансах, кадрах и т.п. Это, однако, не означает, что данные и знания обо всех этих предметных подобластях слиты воедино. Они могут и, чаще всего, имеют автономные базы данных и знаний. Принципиально, что в КП они организованы и поддерживаются таким образом, что становится возможным их совместное интегрированное использование. Интегрированный характер КП, вероятно, является ее самым важным свойством, позволяющим получать новое знание на основе совместного использования и обобщения имеющегося.

Другим важным свойством КП является наличие в ней не только текущих оперативных данных, но и данных прошлых периодов и прошлых лет, называемых историческими данными. Такое свойство появляется в результате наличия у КП больших резервов емкости и мощных средств защиты памяти. Данные и знания, помещенные в КП, уже никогда не подвергнутся каким-либо изменениям и сохраняют тем самым свою историческую достоверность вне зависимости от того, сколь часто они извлекались и использовались.

Временные рамки данных и знаний, содержащихся в хранилище, изменяются в широком временном диапазоне. Оперативно доступные данные обычно хранятся от года до пяти лет. Данные большей давности, как правило, переносятся в архив на магнитной ленте или CD ROM.

Концептуальная модель КП представлена на рис. 7.5 [Maier R., 2004].

Как следует из концептуальной модели, данные и знания извлекаются из оперативной системы и внешних источников и помещаются в хра-

нилица. Оперативные и внешние источники поставляют также данные о данных, или метаданные, в особую часть КП, именуемую репозиторием. В репозиторий пересылаются также метаданные, генерируемые моделями данных и знаний. Потребители деловой информации формируют запросы к средствам представления информации, которые, в свою очередь, генерируют запрос к хранилищу данных и знаний.

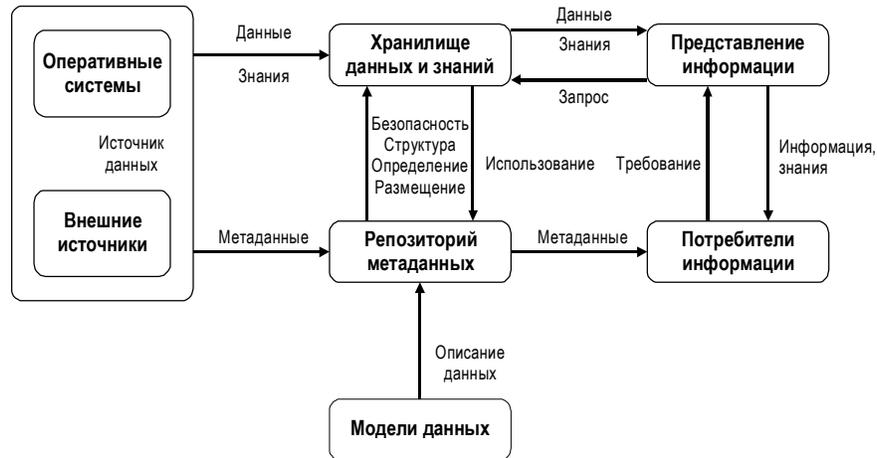


Рис. 7.5. Концептуальная модель КП

Если пользователь обладает правами доступа к соответствующим разделам КП, результат обработки запроса через блок представления информации передается потребителю. Потребителю информации могут быть доступны также структуры и модели хранения данных и знаний в КП.

Данные, информация и знания, о составе и логической модели которых говорилось выше, хранятся в некоторой физической среде. Чаще всего, с учетом уровня развития аппаратных средств современных компьютеров, они хранятся на жестких дисках большой емкости. Архитектура физических сред корпоративных хранилищ существенно зависит от применяемых ИТ-технологий.

Появление технологии типа клиент/сервер привело к использованию двух, трех и даже пятизвенных архитектур, когда речь идет о больших хранилищах и мощных системах обработки транзакций (рис. 7.6).

В больших хранилищах используются многомерные базы данных (multidimensional DB), геометрическим аналогом которых является многомерный куб. В простейшем трехмерном случае в качестве осей куба для определенного вида сущностей могут выступать, например, товар – регион – время.

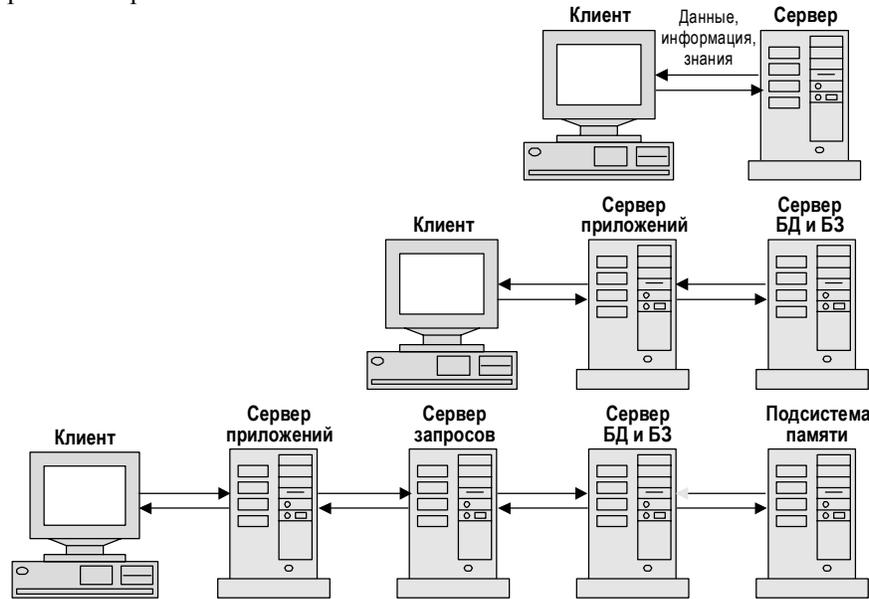


Рис. 7.6. Многозвенные архитектуры физической среды КП

Операционная система, пригодная для работы с большой КП, должна обладать рядом уникальных свойств, важнейшими из которых являются:

- Разрядность 64 и выше.
- Модель памяти VLM (Very Large Memory) вплоть до 4 Тбайт.
- Очень большие файловые системы (с размерами сетевых и локальных файлов до 1 Тбайта).
- Изменяемый размер страниц памяти.
- Параллельная обработка.

Кроме того, операционная система КП должна поддерживать стандарты открытых систем с тем, чтобы быть в состоянии интегрировать смешанные среды типа NT, NetWare, СУБД Oracle, СУБД Informix, Fast

Track Netscape, сервер приложений Web Application Server Oracle, виртуальную машину Java и др.

Весьма важным компонентом КП являются сети и средства взаимодействия. По сути, это интегрирующая технология, которая объединяет в одно целое и связывает разные элементы физической структуры корпоративного хранилища. Для обеспечения высокой пропускной способности системный поток данных может передаваться с использованием таких интерфейсов, как 100 Base-TX, Ethernet, FDDI, Gigabit Ethernet или HIPPI. При этом данные пользователей передаются по локальным вычислительным сетям типа 10/100 Base-TX Ethernet.

Рабочие станции специалистов компании и специалистов в области инженерии знаний именуется клиентами. Разнообразие клиентских функций порождает и разнообразие параметров и возможностей рабочих станций специалистов. Обычный массовый пользователь, условно именуемый *тонкий клиент*, имеет возможность работы с Web-браузером. Он не располагает большой дисковой памятью или мощной OS, поскольку большинство приложений загружается через Web. Львиная доля вычислений осуществляется на Web-сервере. Графические и мультимедийные интерфейсы тонкого клиента ограничены.

Рабочие станции среднего и тем более высокого класса (high-end-workstation) обладают оптимальными характеристиками в отношении памяти, мультимедийной обработки, графики и визуализации данных. Они могут выполнять роль сервера рабочей группы (команды), а также сложные процедуры анализа и обобщения данных, информации и знаний.

Корпоративная память, как крупная, сложная и весьма важная подсистема СУЗ, требует постоянного администрирования и управления. Не вдаваясь в детали, перечислим основные из многочисленных задач управления КП:

- Разработка и поддержание модели данных, включающей аудит данных, информации и знаний. Определение их объемов и смысловых значений.
- Управление памятью, включающее управление параметрами емкости, модификации индексов, резервного копирования и восстановления хранимого.
- Сопровождение КП, включающее архивирование, загрузку, удаление и преобразование хранимого содержания.
- Аварийное копирование и восстановление хранимого содержания.
- Контроль версий и конфигурации программного обеспечения.

- Сопровождение системы безопасности, включающее методы разрешения доступа, контроль имен и паролей пользователей, уровней доступа к хранимому содержанию.

Очевидно, что КП для того и создается, чтобы сотрудники компании имели к ее ресурсам естественный и эффективный доступ непосредственно со своих рабочих мест. Этой цели служит специальный набор инструментальных средств доступа к КП и взаимодействия с ней. Разнообразие потребностей пользователей определяет и спектр инструментальных средств.

К основным типам программных средств для доступа и взаимодействия с КП относятся:

- Средства доступа к данным.
- Средства добычи данных (data mining).
- Географические информационные системы (ГИС).
- Системы аналитической обработки в реальном времени (on-line analytical processing systems – OLAP- системы).
- Реляционные OLAP-системы (ROLAP).
- Многомерные OLAP-системы (MOLAP).
- Средства визуализации данных.
- Средства обработки статистики.
- Информационные системы руководителей (EIS).
- Средства обработки графического интерфейса пользователя.
- Электронные таблицы.
- Генераторы отчетов.
- Языки программирования четвертого поколения (4GL).
- Браузеры Интернет.
- Браузеры метаданных.

Перечисленные типы программных средств доступа и взаимодействия с КП не исчерпывают многообразия существующих и нарождающихся программных инструментов. Стремление к более интенсивному использованию информационных и знаниевых ресурсов, хранящихся в КП, привело к появлению нового класса программных инструментов, получивших название *программные агенты (ПА)*.

В общем плане агент – это аппаратная и программная сущность, способная действовать в интересах достижения целей, поставленных перед ним владельцем и/или пользователем.

Существуют мобильные агенты, автономные агенты, персональные агенты, интеллектуальные агенты и т.д. [Wooldridge M., 2002]. То есть

понятие «агент» следует трактовать как метаимя или класс, который включает множество подклассов.

Программные агенты – новый вид компонентов ПО, которое действует от лица пользователя.

Интеллектуальные программные агенты обладают рядом полезных свойств, в числе которых:

- Цели (goals) – совокупность состояний, на достижение которых направлено поведение агента.
- Базовые знания (basic knowledge) – знания о себе, о среде, о других агентах.
- Обязательства (commitments) – задачи, которые берет на себя агент по поручению владельца, пользователя и/или других агентов.
- Убеждения (beliefs) – переменная часть базовых знаний, которая может меняться во времени.
- Активность (activity) – способность генерировать цели и действовать в направлении их достижения.
- Реактивность (reactivity) – адекватная реакция на изменения в среде.
- Автономность (autonomy) – способность функционировать без вмешательства своего владельца.
- «Социабильность» (social ability) – возможность взаимодействия и связи с другими агентами.

В заключение следует отметить, что современные корпоративные хранилища данных, информации и знаний, имеющие четкие архитектурные, структурные и логические принципы построения, реализованные на мощных компьютерных системах с использованием технологии клиент/сервер, оснащенные современным системным (OS, СУБД), широким спектром прикладного и пользовательского программного обеспечения, без всякого сомнения, представляют собой весьма предпочтительную альтернативу хаотичной, неструктурированной или «бункерной» среде хранения данных, которая, к сожалению, имеет место даже в компаниях, широко использующих современные компьютеры и ИТ-технологии в повседневной практике.

7.3. Подсистема поиска знаний

Обнаружение (knowledge discovery), поиск или, как его еще называют, разведка знаний представляют собой относительно новое и быстро развивающееся направление, использующее наряду с традиционными ИТ-инструментами методы искусственного интеллекта, математической

лингвистики и статистики для обнаружения и выуживания знаний. Знания, как уже отмечалось, могут находиться в специальных и весьма емких корпоративных хранилищах (репозитариях) или информационных фондах, сайтах и порталах, доступ к которым возможен с помощью Интернет.

Термин «разведка знаний» Г. Пятецки-Шапиро и В. Фролей [Piatetsky-Shapiro G., Frawley W., 1991] определяют как «нетривиальное извлечение точной, ранее неизвестной и потенциально полезной информации из данных».

Поиск данных сегодня осуществляется с помощью различных низкоуровневых булевых поисковых механизмов. Такие средства не решают проблему управления знаниями, а только вносят свой вклад в перенасыщение информацией, заставляя пользователей плутать в океане информационного сырья.

При поиске информации найденное выдается пользователям в так или иначе организованном виде. Но при таком поиске чаще всего опускается контекст и понимание существа запроса, осложняют поиск неоднозначности толкования и синонимия. Например, невозможно провести различие между «пером» авторучки и «пером» птицы, поэтому ищущие знания пользователи вынуждены путаться в море неоднозначных терминов и понятий.

Поиск знаний (семантический) осуществляется тогда, когда поисковые механизмы высокого уровня доставляют информацию, действительно актуальную для нужд пользователей, информацию, находящуюся в резонансе с целью и контекстом исходного запроса. Для того чтобы система поиска знаний работала действительно эффективно в мире лавинообразного роста информации, она должна быть точной, масштабируемой, безопасной, расширяемой, прозрачной и простой в использовании.

Корпоративные хранилища данных, информации и знаний (репозитории) обычно содержат огромное количество записей. Так, база знаний Ford имела еще в конце 90-х годов объем, эквивалентный 30 тыс. страниц машинописного текста. И число актуальных баз знаний и их объем растут из года в год.

Поскольку большинство документов в электронной форме находятся в сетях в on-line режиме, то проблемы on-line доступа фактически трансформировались в проблему поиска материалов, релевантных решаемой задаче. Текстовый поиск, который еще каких-то десять лет на-

зад был инструментом библиотекарей, стал повседневно используемым средством большинства деловых людей, ученых, преподавателей.

Так как многочисленные базы данных и Web-сайты слабо стандартизованы, армии пользователей приходится работать с различными средствами поиска, с различными формами представления, как искомого, так и найденного.

Поэтому поиск нужной информации и знаний становится в СУЗ экстремально критической функцией, которой уделяется все большее внимание. Большинство современных методов поиска включают инструментальные средства, средства интеллектуального поиска и визуальные модели.

7.3.1. Инструментальные средства поиска

Имеется широкий набор инструментальных средств (браузеров) для информационной навигации в Интернет. К их числу относятся AltaVista, Excite, Infoseek, WebCrawler, Yahoo, Google. Многие из них адаптированы и для внутрикорпоративных нужд при работе с репозиториями знаний.

Инструментальные средства поиска содержатся также в MS SPP Server (Microsoft Share Point Portal Server), который является универсальным корпоративным порталом, средством управления документами, является также поисковой машиной. Причем поиск документов по их содержанию является одной из ключевых функций MS SPP Server. Индексировать можно не только документы, хранящиеся на сервере, но и файлы в папках Microsoft Exchange, Lotus Notes, в библиотеках SharePoint Team Services и даже произвольные сайты, в том числе из Интернета. Таким образом, поисковая система на базе MS SPP Server может охватить практически всю доступную компании информацию, а возможность автоматического выполнения заранее определенных поисковых запросов по заданному расписанию позволит, например, оперативно отслеживать ситуацию по проблеме в конкурирующих формах.

Очень полезная функция MS SPP Server – подписка. На любой список, документ, папку, библиотеку или доску обсуждений можно «подписаться» – определить, какие изменения выбранного объекта (добавление, удаление, редактирование) вас интересуют, и получать уведомления по электронной почте всякий раз, когда данные изменения происходят. Вместо почтового клиента можно использовать специальный компонент Web Part, который отображает состояние всех выбранных

подписок. Эта возможность освобождает пользователей от необходимости постоянно просматривать сайт в поисках новостей. Новости «придут» сами.

Интерфейсы обработки протоколов поиска, содержащиеся в MS SPP Server, позволяют подключаться к пользовательским источникам данных, информации и знаний. Поддержка пользовательских типов источников обеспечивается через интерфейс Ifilter. Тем самым в MS SPP Server обеспечивается расширяемость системы поиска.

К инструментальным средствам поиска относятся ряд продуктов компании Lotus. В их числе можно отметить Lotus Discovery Server (LDS) [Pohs W., Pinder G.], который обеспечивает поиск всего, в чем нуждается пользователь, путем автоматического анализа и идентификации отношений между документами, людьми и областями базы знаний. LDS базируется на следующих компонентах:

- Родственность (Affinity) – это обнаруженный LDS тип отношения между персоной и областью знаний.
- Величина родственности (Affinity Value) – количественное определение «силы» этих отношений.
- Цифровые «крошки» информации (Digital Bread crumbs), представляющие небольшие фрагменты неструктурированных знаний, которые имеются в организации, отделе или даже у конкретного человека. Они могут быть в электронных сообщениях, документах и т.д. LDS с помощью программ-пауков собирает эти фрагменты знаний и привязывает их к нужному разделу.
- Метрики (Metrics), предназначенные для увеличения точности поиска. Они отражают отношения между объектами в системе (людьми, документами, местами) и соответствующую информацию об их использовании. Сервер постоянно анализирует собираемые сведения, пересчитывая отношения, значимость документов и опыта. В результате таких операций LDS всегда предоставляет пользователю актуальную информацию, которая действительно ему необходима, вместо того, чтобы выдавать множество нерелевантных результатов, как это делает большинство известных поисковых систем.
- Программы-пауки (Spiders) – небольшие программы, перманентно просматривающие репозитории и хранилища информации, указанные в запросах на поиск знаний и доставляющих найденное на сервер.

Среди инструментальных средств поиска можно отметить продукты компании Tacit Knowledge Systems, а именно:

- KnowledgeMail, которая обеспечивает возможность поиска опыта с помощью интегрирования корпоративных подсистем электронной почты и хранилищ документов. Результат просмотра хранилища и почты используется для получения базы данных опыта.
- Expertise Services Platform, которая является мощной программной системой для быстрого обнаружения опыта и поиска в любых приложениях с помощью набора программных интерфейсов XML. Кроме того, она позволяет построить и эффективно управлять программным комплексом для работы с нечеткими знаниями.

7.3.2. Средства интеллектуального поиска

Среди растущего числа продуктов, относящихся к этому классу средств поиска, можно отметить InfoFinder, продукты компании Fulcrum, Convera Technologies Corporation, AskMeCorporation.

InfoFinder, осуществляя поиск необходимых решений, анализирует интересы пользователей по наборам классифицированных сообщений или документов, опираясь на эвристические методы.

В составе средств интеллектуального поиска выделяется набор продуктов компании Fulcrum и в их числе [Maier R., 2004]:

1. *Fulcrum KnowledgeServer*, который позволяет извлекать необходимую информацию из огромных хранилищ информации, представленной в разных формах, по одному запросу преобразуя таким образом информацию в знания, с использованием нового способа структурирования информации.

2. *Knowledge Management Workstation 1.0 (KMW)*, которая предназначена для создания и редактирования кластерных карт. В дальнейшем кластерные карты могут использоваться при построении динамических оглавлений ПО Fulcrum KnowledgeServer. Используя KMW, сотрудник, занимающийся управлением знаниями, или эксперт в той или иной предметной области может автоматически создавать рубрикаторы, отражающие структуру знаний организации.

Fulcrum выдвигает определенные требования KMW к программным и техническим средствам: операционная система – Windows NT Server 4.0; дисковое пространство – 40 Мбайт; оперативная память – не менее 64 Мбайт, предпочтительно от 128 до 384 Мбайт; процессор – минимум

Intel Pentium 200 МГц, предпочтительно Intel Pentium II 450 МГц; Web-браузер – Microsoft Internet Explorer 4.01 с установленным Service Pack 1,2 (и выше).

3. *Fulcrum WordSense*: разумный поиск. Реализованная в FWS передовая лингвистическая технология, основанная на семантических сетях, позволяет пользователям SearchServer получать точные результаты для поисковых запросов на естественном языке. Fulcrum WordSense анализирует запросы и устраняет неоднозначности в их понимании перед выполнением поиска, кроме того, он расширяет запрос посредством семантической сети. SearchServer обеспечивает пользователям многофункциональный поиск информации для широкого круга задач, включая такие дополнительные сервисы, как электронная публикация, обслуживание клиентов, техническая поддержка в реальном времени и т.п.

Основой системы и программных продуктов AskMe является 7-шаговый поисковый алгоритм:

1. Инициализация поиска. Для этого пользователь вводит строку запроса, которая может содержать ключевое слово или вопрос (например, «как я могу...?»). Запрос может содержать и логические операторы (и, или, не).

2. Обработка естественного языка. Выделяются ключевые слова в строке запроса. Используются аннотационные ключи для выделения существительных и отброса не относящихся к делу слов.

3. Нечеткое соответствие. Определяются корни ключевых слов для увеличения совпадения с результатом. Система сопоставляет ключевые слова запроса с тезаурусом, чтобы включить синонимы и акронимы в выводимый результат.

4. Сопоставление шаблонам. Система сопоставляет строку запроса с базой знаний. Можно также связать поиск с другими базами документов или знаний.

5. Применение различных бизнес-правил. Например, пользователь может указать, что его интересуют результаты только за последние семь дней. Система также смотрит, может ли данный пользователь получить доступ к каким-то конкретным результатам и т.п.

6. Ранжирование результатов поиска. Система ранжирует результаты по различным критериям (релевантности и др.).

7. Вывод результатов. Вывод результатов поиска и индексирование их для будущих запросов.

Продуктами AskMe пользуются многие ведущие компании мира, среди них даже такие гиганты, как: Procter & Gamble. Специалисты AskMe выделяют три логических уровня в системе управления знаниями компании: презентационный уровень, уровень сети знаний и уровень хранения. Уровни взаимодействуют посредством XML-интерфейсов.

Convera Technologies Corporation разработала систему Excalibur RetrievalWare. Эта промышленная информационно-поисковая система представляет собой мощное средство полнотекстового и атрибутивного поиска. К документам, с которыми работает RetrievalWare, относятся тексты в различных форматах и кодировках, электронные таблицы, базы данных, почтовые сообщения и т.п. – всего более 200 форматов плюс инструментарий, позволяющий без затрат на программирование настроить систему на поддержку специфических форматов документов заказчика.

Архитектура RetrievalWare позволяет работать с системой как через корпоративную локальную сеть, так и через Интернет. Серверная часть системы может быть установлена на всех распространенных серверных платформах, а клиентским местом может быть любая машина, имеющая графический Web-браузер.

Родившаяся в процессе моделирования сложных биологических систем, технология адаптивного распознавания образов APRP (Adaptive Pattern Recognition Processing) использует нейронные сети для обработки информации и действует как самоорганизующаяся система, которая автоматически выделяет в массиве хранимой информации и индексирует двоичные образы. Уникальные возможности технологии адаптивного распознавания образов обеспечивают семейству программных продуктов RetrievalWare преимущества при построении поисковых приложений фактически для любой информации, представленной в электронном виде, – текстов, изображений, звуков, видеоинформации. Если технология адаптивного распознавания образов APRP повышает эффективность работы с любой информацией, то технология семантического поиска ориентирована исключительно на работу со знаниями, содержащимися в текстовых документах.

7.3.3. Визуальные модели поиска

Среди новых тенденций в области систем поиска можно выделить метод визуальных моделей. Perspecta и InXight представляют различные методы визуализации знаний.

Perspecta (<http://www.perspecta.com/>) создает интеллектуальный контекст, используя метаинформацию, выделенную из исходных документов, включая структурированную информацию в БД и целевых документах или неструктурированные данные в офисных документах и Web-страницах. Для неструктурированных документов Perspecta имеет специальное средство Document Analysis Engine, которое выполняет лингвистический анализ и автоматически помечает документы. Сервер интеллектуального контекста анализирует помеченную информацию, идентифицирует взаимосвязи между документами и строит многомерное информационное пространство, используя специальный язык меток (Information Space Markup Language). С его помощью пользователь «летит» сквозь информационное пространство, манипулируя мышью. Для экономии ресурсов данные выгружаются клиенту с помощью информационного потокового протокола (Information Streaming Transport Protocol), который является расширением HTTP.

Компания InXight Software (<http://www.inxight.com>), отпочковавшаяся от Xerox PARC, выпустила собственное средство визуализации VizControl, предлагающее несколько форматов визуализации. Каждый из них развивает метод «фокус контекст», когда интересующие пользователя данные выводятся на передний план и, в то же время, неизменной сохраняется структура даже очень больших наборов данных. Одно из таких инструментальных средств, гиперболический браузер (или «рыбий глаз»), использует гиперболическую геометрию для расширения информационного пространства при работе с иерархическими структурами, которые расширяются экспоненциально с увеличением глубины. Таким образом, гиперболический браузер может показать 1000 узлов в окне размером 600x600 пикселей, в центре которого высвечивается текст довольно большого объема (для сравнения, условный 2D-браузер может показать на экране лишь около 100 узлов). Пользователь перемещается по информационному пространству, щелкая мышью на узле или передвигая указатель мыши по гиперболической плоскости.

Специфической разновидностью визуальных моделей являются онтологические описания предметных областей. В общем виде графическим отображением онтологии является сеть знаний, визуальное представление которой (рис. 7.7) способствует упрощению ее понимания.

Известны и другие варианты реализации метода визуальных моделей, облегчающие понимание пользователем поступающей информа-

ции. Так, в интерфейсе пользователя Yahoo присутствует дерево категорий, основанное на тексте.

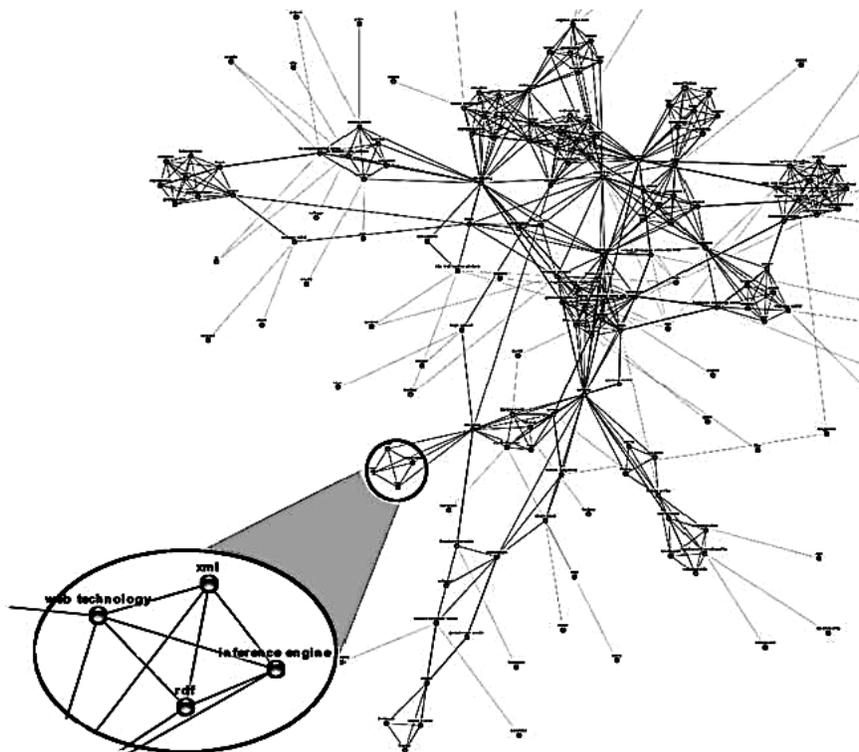


Рис. 7.7. Онтология как сеть знаний

Затененные топографические карты используются в системе Themescare в качестве метафоры представления различных предметных областей, учитывающей их расположение, связность и т.п. Метафоры карт для отображения предметных категорий используются в Visual Net. Успешно справляется с визуализацией больших таксономий и онтологий система «Cat-a-Cone».

Применение визуальных моделей позволяет усилить когнитивный аспект при работе пользователей с классификациями знаний, что важно как само по себе, так и в поиске знаний.

7.4. IT-среда для совместной интеллектуальной деятельности

IT-среда совместной интеллектуальной деятельности (СИД) предназначена для обеспечения эффективного обмена знаниями и взаимодействия в процессе СИД сотрудников и экспертов компании, территориально и географически удаленных друг от друга, объединенных в группы, команды или сообщества по интересам.

Объектом обмена могут быть данные, информация и знания (теоретические, практические, явные и неявные). Поводом для интеллектуального взаимодействия могут быть производственные задачи и проблемы, новые проекты, разрабатываемые территориально-разобщенными группами специалистов, сложные решения, принимаемые в условиях неопределенности, вызванной незнанием ряда существенных факторов и обстоятельств, а также решения, принимаемые в условиях частично не совпадающих интересов и целей.

Многообразие объектов, поводов и видов совместной интеллектуальной деятельности порождает и многообразие инструментов и программных систем, созданных для формирования IT-среды СИД. Такого рода IT-средства известны также под названием *групповое ПО* (Groupware). Групповое ПО – это достаточно широкая совокупность прикладного ПО, которая облегчает сотрудникам компании совместную работу во многих производственных ситуациях, в том числе работу в группах или командах.

Групповое ПО обеспечивает создание искусственной среды, именуемой также *виртуальным пространством*, в котором участники совместной интеллектуальной деятельности могут более эффективно сотрудничать в процессе уяснения проблем и задач, обмениваться результатами работ по совместным проектам, делиться накопленным опытом, дискутировать, заслушивать презентации, постановочные и проблемные доклады, обмениваться оценками и точками зрения.

Групповое ПО включает общение посредством электронной почты (E-mail), мгновенный обмен сообщениями (instant messaging), дискуссионные области (discussion areas), собрания в реальном времени (on-line meeting) и видеоконференции. Групповое ПО позволяет создавать на экране общее окно (whiteboard), в котором не только виден объект совместной работы, но есть также возможность в нем что-то видоизменить (текст, рисунки).

В составе группового ПО содержатся также системы совместной работы (collaboration systems), проектные зоны (project zones), доступ к которым имеют только исполнители и заказчики проекта, новостные группы (newsgroups) и подобные им форумы, открытые для всех, специализированные приложения, например, для обработки результатов мозгового штурма и т.п.

Перечень функций и компонент группового ПО может быть расширен. И он будет расширен и детализирован далее. Но прежде имеет смысл остановиться на проблеме практических знаний и умений человека, их роли в управлении знаниями вообще и в обмене знаниями в процессе совместной интеллектуальной деятельности в частности.

Приходится констатировать, что практические знания и умения человека, его профессиональные знания «о том, как делать» (ноу-хау) не могут быть выражены в виде принятых в теории точных определений, моделей и алгоритмов.

Как отмечается в философской литературе [Быстрицкий Е.К., 1990], это происходит по ряду причин. Одной из них является всеобщая настроенность ученого мира «на теорию» и, следовательно, отсутствие стремления столь же тщательно описывать практику. Другой причиной является широта и необъятность сфер человеческой жизнедеятельности, где человек действует и поступает весьма разумно и рационально, без теоретических обобщений по поводу своей многообразной практики. В сфере практических знаний и навыков есть множество вещей, которым нельзя научить извне. Сюда входят привычные орудия труда, инструменты, снаряжение, все то, что древние греки называли «pragmata», то есть то, что человек использует в быту, на работе, для передвижения, видения, измерения и т.п. По этой причине бесполезно множить «научные инструкции и трактаты» по поводу применения тех полезных вещей, которые и без них повседневно успешно используются людьми. По этим и другим причинам в сфере практических знаний и навыков не существует однозначных алгоритмов практических действий в непрерывной цепи человеческого опыта.

Возникает естественный вопрос: «Как вообще можно научиться и научить тому, что не сформулировано в четких правилах и законах?», «Как улучшить фиксацию человеческого опыта и практических знаний и как обеспечить их передачу от человека к человеку, от поколения к поколению?»

Практические знания, навыки и опыт более, чем другие виды знаний принадлежат конкретным людям, находятся в их головах и, следовательно, максимально распределены не только в рамках организационных структур, но даже в отделе, группе, команде.

Главным направлением в передаче, накоплении и использовании практического знания следует, очевидно, считать межличностное общение людей в культурной, образовательной, производственной сфере и в быту. Как утверждал К. Маркс, «обработка людей людьми» ничуть не менее чувственно-предметна, чем вещественная «обработка природы людьми».

IT-среда СИД с разнообразными программными инструментами как раз и является благоприятной средой для совместной интеллектуальной деятельности людей на основе использования как разнообразных практических знаний, накопленных специалистами и экспертами компании, так и знаний, хранящихся в централизованном корпоративном хранилище. Учитывая широкий спектр инструментальных средств, представленных групповым ПО для поддержки разнообразных форм общения специалистов и их совместной деятельности, появляются нетривиальные возможности для обработки «мозгов мозгами», в результате чего не только повышается качество принимаемых решений, но и создается новое явное и неявное знание.

Практическое знание, практическое мышление, усиленное современными IT-технологиями, приобретает особую значимость в наше время, когда в управлении бизнес-процессами современной компании перестает доминировать единоначальник и задачу управления начинают понимать как коллективную активность многих людей, опирающихся на совокупный коллективный разум.

Продукты компании Lotus появились на рынке управления знаниями одними из первых, и до сих пор они сохраняют на нем лидирующие позиции. Применительно к обеспечению среды для совместной интеллектуальной деятельности специалистов компании наибольшую известность получил комплексный продукт компании LKDS (Lotus Knowledge Discovery System). LKDS состоит из двух взаимодействующих подсистем (K-Station Portal и Knowledge Discovery Server).

K-Station Portal предоставляет набор средств для создания портала знаний, для персонализации Web-интерфейса, для получения и организации информации, а также средств для обеспечения совместной работы. В результате специалисты и эксперты получают внутри организации вирту-

альное место (пространство), где они могут взаимодействовать и управлять информацией, которая им необходима для решения своих задач. K-Station интегрирована со средствами взаимодействия в реальном времени, которые информируют о том, кто находится в сети, и обеспечивают мгновенную пересылку сообщений. Тем самым формируется рабочее пространство совместной он-лайн работы и обучения. Электронная почта также является элементом этого уровня сервиса.

Для обеспечения простоты навигации и редактирования K-Station обеспечивает единый пользовательский интерфейс доступа к настольным приложениям, Web и корпоративным системам непосредственно с рабочих мест. K-Station обеспечивает конечных пользователей средствами доступа к знаниям, компетенции, информации и связям, которые существуют внутри и вне организационных барьеров. Этот доступ начинается с персонального пространства или пространства для сообществ, в которых конечные пользователи получают доступ, генерируют и применяют знания. Конечные пользователи также могут использовать представления знаний (knowledge views) для отображения данных из приложений в областях, размеры, место и положение которых могут меняться, или на страницах с закладками, позволяющими категоризировать их содержание.

Например, представитель по продажам может создать в K-Station страницу отслеживания изменений в процессе заключения контракта. На ней он может разместить такие элементы, как календарь встреч, дискуссии с заказчиком и файлы контрактов так, что все связанные с заказчиком элементы и информационные ресурсы находятся в одном месте и постоянно доступны. Это персональное пространство может включать средство, которое в K-Station называется «информация о людях» и которое обеспечивает средство он-лайн дискуссий с необходимыми людьми. Эти возможности позволяют представителю по продажам сконцентрироваться не на технических инструментах выполнения работы, а на проблемных вопросах, которые можно теперь решать гораздо быстрее.

Пространства сообществ в K-Station аналогичны личным пространствам, однако они предназначены для совместной работы одновременно большого количества людей. Пространства сообществ могут быть публичными форумами, открытыми для сотрудников организаций, или эксклюзивными местами только для тех, кто получает соответствующее приглашение.

Пространства сообществ представляют собой повторно используемые *объекты знаний*, которые хранят в себе информацию о состоявшихся взаимодействиях между людьми, содержании обменов знаниями, которые были востребованы для выполнения специфических задач и проектов. Дискуссии, презентации, документы, планы, расписания – все они являются типичными объектами знаний, которые могут быть созданы и сохранены в K-Station. Это сохранение означает, что все усилия по выполнению какой-либо работы, включая установленные связи, созданные документы и выполненные действия, формируют элементы живой истории организации. Этот вид сохранения истории совместной работы и знаний является весьма важным в управлении знаниями и является уникальной особенностью K-Station Portal.

Lotus Knowledge Discovery System (KDS) работает с тремя основными концептами:

1. Вещи (things). Это данные, документы, бизнес-процессы, которые каталогизированы.
2. Персоны (persons). Сотрудники, партнеры, заказчики, поставщики и др.
3. Места (places). Места виртуальных собраний (с точки зрения пользователя могут представлять собой интернет-странички), где сотрудники организации, занимающиеся одним проектом или связанные каким-либо другим родом деятельности, могут обмениваться информацией, касающейся этого проекта, этого вида деятельности.

К числу функций, выполняемых Knowledge Discovery Server, относятся (рис. 7.8):

- *Knowledge Map (КМар – отображение знаний)*. Интерфейс, который отображает каталогизированные источники знаний организации.
- *Автоматическая генерация КМар*. Технология, которая анализирует и группирует схожие документы, персоны и места в категории и именуется их.
- *Редактор КМар*. Средство, которое позволяет администраторам редактировать, аннотировать, отображать содержание КМар (названия категорий, их структуру) таким образом, чтобы оно подходило терминологии и стереотипам конкретной организации.
- *Ранжирование по релевантности*. Ставит в соответствие персонам и документам числовые значения, показывающие степень отношения персоны или документа к категории, отображенной в КМар.

- *Metric Analysis*. Вычисляет ценность информационных ресурсов, которые интерпретируют частоты авторства, цитирования и использования отдельными сотрудниками.
- *Profiling – профилирование*. Средства профилирования позволяют собирать информацию об экспертах в той или иной области. Сотрудники могут найти внутри организации тех людей, которые обладают знаниями и навыками для ответа на специфические вопросы. Кроме того, оперативное взаимодействие с экспертами конкретной предметной области – один из методов обеспечения передачи неявных знаний. Связь между знанием и людьми отличает системы управления знаниями от приложений, которые поддерживают работу только с явным знанием, хранящимся в КП.

Назначением систем поиска экспертов (Expertise location) является выявление имен людей, которые имеют знания и опыт в конкретной области. Системы поиска (Search engine) экспертов хороши настолько, насколько хороши сведения, с помощью которых описаны их знания и опыт.

В LKDS используются профили компетенции экспертов, составленные самими экспертами, сведения, полученные из существующих БД и БЗ, и сведения, выведенные из ассоциации сотрудников с документами. Анализом почтовых сообщений, составлением профилей компетенции сотрудников и экспертов в LKDS занимается подсистема Knowledge Mail из Tacit Knowledge System. При наличии также результатов онлайн дискуссий разумно предположить, что четвертым источником сведений может быть содержание ответов на вопросы, которые дал сотрудник в такой системе при условии, что этот сотрудник уже проявил желание помочь кому-то. Это простой пример признака социального взаимодействия в поиске экспертов, который является важным фактором, но еще не используется в существующих системах, возможно, потому, что его трудно зафиксировать.

При этом важно отметить, что пользователь (специалист, эксперт) имеет полный контроль над содержанием своего профиля (может его редактировать и указывать степень доступности той или иной информации).

В решениях компании Lotus по управлению знаниями важное место отводится разработке средств взаимодействия в реальном времени как естественному развитию асинхронного сотрудничества. Технологии реального времени дают максимальную отдачу, когда сочетают возмож-

ности оповещения (тех, кто находится в on-line), ведения разговора (текст, аудио, видео).

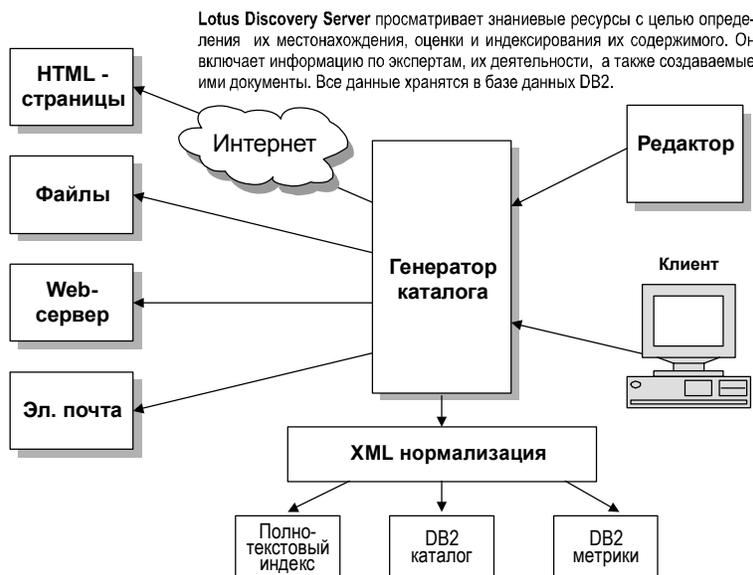


Рис. 7.8. Архитектура системы Knowledge Discovery Server

Большие возможности по организации совместной деятельности предоставляют приложения, которые поддерживают on-line собрания и сотрудничество в реальном времени. On-line собрания могут включать проведение видео или основанных на тексте конференций, а также синхронное общение и чат. Считается, что основанный на тексте чат может поддерживать обмен знаниями группы людей в разговорной форме. Такой сервис обеспечивает Lotus Sametime. В нем объединяются возможности как мгновенного обмена сообщениями (instant messaging), так и он-лайн собраний (on-line meeting). Мгновенный обмен сообщениями находится где-то между личной встречей и телефонной беседой: он менее навязчивый, чем беспокоит человека вопросом, но более эффективен, чем телефон, так как позволяет передавать сообщения группе и предлагать ответить на него позже. Кроме того, пользователи легко осуществляют в нем переход в режим чата, аудио и видеоконференции (одним щелчком мыши).

Новостные группы (newsgroups) и подобные форумы открыты для всех. В отличие от обычных дискуссий, проводимых в командах, они имеют некоторые схожие характеристики в том, что можно послать вопрос и на него можно ответить, но отличием является то, что участники обычно являются незнакомыми людьми. Тем не менее оказалось, что многие люди, которые участвуют в новостных группах, желают дать совет и помощь, ведомые смесью мотиваций, включая альтруизм, желание выглядеть экспертом или в знак благодарности и ответного отклика за то, что им ранее помогли.

Архив форума становится своего рода репозитарием полезных знаний. Ясно, что тот, кто ответил на вопрос и сделал это многократно ранее, может рассматриваться в качестве эксперта. Хотя обмен сообщениями внешне выглядит как обмен явными знаниями, эксперт должен в начале оценить проблему и затем принять решение, что требует обращения к его скрытым знаниям. Так как знания после обмена сделались явными, то сотрудник с аналогичными проблемами может найти решения, просматривая архив.

Таким образом, средства проведения таких форумов и дискуссий способствуют тому, чтобы знания извлекались у большей части сообщества, так же как и у отдельных экспертов.

Lotus Knowledge Discovery System выдвигает некоторые системные требования к аппаратной части. Для клиентской машины требование одно: MS Internet Explorer 5.0 или выше и, конечно, операционная система MS Windows 95 SP2/98/ME/NT SP5/2000/XP. Для функционирования сервера необходима Windows NT SP6 или Windows 2000 Advanced Server 1.0. Требования к серверу по оборудованию следующие: 512 Mb ОЗУ (1 Gb рекомендовано) и 30 Gb жесткий диск (100 Gb рекомендовано). Столь высокие требования обусловлены большой нагрузкой на сервер (особенно на большом предприятии).

Необходимо также отметить, что Lotus Knowledge Discovery System практически полностью реализована с использованием языка Java от компании SUN.

Программную систему MS SPP (Microsoft Share Power Point) для реализации ряда функций управления знаниями продвигает и компания Microsoft. Платформа управления знаниями Microsoft состоит из трех уровней:

- *Настольной системы.* Настольная система вашего клиента играет роль портала для доступа к знаниям компании. Компьютер с Mi-

Microsoft Office 2000, работающий под управлением операционной системы Microsoft Windows[®] 2000 Professional, Windows 9.x или Windows NT[®] Workstation 4.0, представляет собой средство управления знаниями.

- *Службы управления знаниями.* Средний уровень обеспечивает возможности поиска и доставки. Такой тип системы управления знаниями может включать в себя, например, Microsoft Site Server 3.0, Microsoft Exchange Server 2000 (или Microsoft Exchange Server 5.5) и Microsoft SQL Server[™]. Семейство Microsoft BackOffice[®] служит основой этого уровня.
- *Серверной операционной системы.* Типичная серверная операционная система для обеспечения безопасности, администрирования и хранения баз данных в системе управления знаниями – Windows 2000 Server с установленной службой Active Directory[™] или Windows NT Server 4.0 с пакетом Option Pack.

Microsoft выделяет два основных компонента систем управления знаниями:

- систему совместной работы и обмена сообщениями;
- корпоративную интрасеть.

Систему обмена сообщениями они предлагают реализовать на основе сервера Microsoft Exchange Server, включающего средства электронной почты, совместной работы, обсуждения и управления документами, а также сервера Microsoft Internet Information Server (IIS – Информационные службы Интернета), предлагающего основные службы обозревателя. Для создания корпоративной интрасети можно воспользоваться приложением Microsoft FrontPage[®] 2000 или инструментальными средствами, включенными в пакет Microsoft BackOffice[®].

То есть, по сути дела, компания Microsoft предлагает не собственную систему управления знаниями компании, а инструментарий для ее создания на основе широко распространенных продуктах компании.

Примером другого комплексного решения является система Deskartes [http://www.columnit.com/products/prod_know_deskartes.html] для поддержки совместной деятельности. Данная система, как и Lotus KDS, предлагает комплексное решение (работа с документами и сотрудниками организации). В отличие от KDS, при создании системы разработчики акцентировали внимание на следующей идее: у пользователя есть вопрос или проблема, которую он хочет решить.

Deskartes действует следующим образом. Сначала система должна попытаться найти решение проблемы в своей базе знаний, если таковое есть. Если его нет, то найти ответ в Web или получить его от экспертов (сотрудников организации) и после получения ответа и подтверждения пользователя о том, что найденное решение релевантно к поставленному вопросу, его запомнить (так происходит обучение). Для выполнения такой стратегии в системе предусмотрены следующие модули:

- *Human Interface Engine* – модуль интерфейса пользователя. Отвечает за перевод вопроса, который написан на естественном языке, в шаблон (pattern) для поиска при использовании нейронных сетей.
- *Neural Engine* – модуль нейронных сетей.
- *Discovery Engine* – модуль для формирования баз знаний, в которых затем будет проводиться поиск. База знаний представляет собой набор файлов (HTML, MS Office), где каждому файлу соответствует шаблон для поиска при использовании модуля нейронных сетей.
- *Collaboration Engine* – модуль для создания профиля пользователя и обеспечения совместной работы.
- *Interperability Engine* – модуль для обеспечения согласованности взаимодействия продуктов Deskartes.

До начала работы с Deskartes необходимо создать профили пользователей и базы знаний. Базы знаний создаются на основе выбранного массива документов (это могут быть HTML-страницы, документы MS Office). По выбранным документам создается словарь терминов. Причем у пользователя есть возможность поставить в соответствие тому или иному термину синоним (или несколько).

В системе используются два типа профилирования пользователей. Первый – это явное профилирование, когда пользователь сам вводит информацию о себе и о том, что его интересует в объединении. Под объединением (community) подразумевается некая категория, в которую входят документы и сотрудники предприятия (эксперты), относящиеся к одной тематике.

Второй вид профилирования – это профилирование, которое автоматически создается и поддерживается системой для каждого объединения, в которое вступил пользователь. Построение этого профиля осуществляется на фактических действиях пользователя (какие вопросы он задает, на какие вопросы отвечает, на что не обращает внимания, чем интересуется).

Работа с системой конечного пользователя заключается в задавании вопросов и получении на них ответов. Сначала пользователь пишет вопрос на естественном языке (система при этом осуществляет подсветку орфографических ошибок). Соответствующий язык должен поддерживаться. После того как вопрос задан, из него извлекается полезная информация (значимые слова) и по ним создается шаблон, который посылается для обработки модулю нейронных сетей. Так как поиск осуществляется не по ключевым словам, а по шаблону, то ответ не обязан содержать слова, которые были в вопросе.

После этого запускается процесс поиска в базе знаний объединения и пользователю возвращается список ответов (решений) (рис. 7.9). Каждому решению ставится в соответствие коэффициент релевантности, по которому эти решения и отсортированы. Если пользователя какой-либо из полученных ответов удовлетворил, то он может сообщить системе о том, что данное решение есть ответ на поставленный вопрос (подтвердить). В этом случае при последующем получении ответов на запрос, идентичный рассмотренному, <подтвержденному> решению будет соответствовать больший коэффициент релевантности (т. е. происходит обучение).

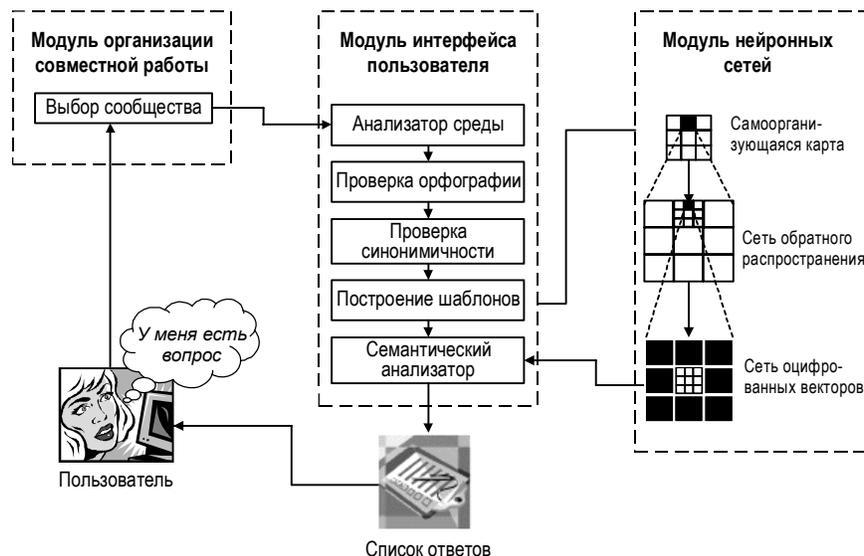


Рис. 7.9. Поиск в базе знаний

Если удовлетворительное решение не было найдено, то запускается следующий этап поиска (WEB escalation; рис. 7.10), когда вопрос пересылается поисковым серверам, которые были включены в базу знаний объединения, в котором проводится поиск. При получении ответов (т. е. Web-страниц), перед тем как предоставить их пользователю, Deskartes обрабатывает каждый из них и определяет, релевантен ли он заданному вопросу. Если пользователь был удовлетворен каким-либо из полученных решений, его URL сохраняется в базе знаний.

Если и в этом случае решение не было найдено, вопрос задается сотрудникам предприятия (экспертам), относящимся к объединению, в котором проводился поиск (knowledge community escalation; рис. 7.10). После того как пользователь получил ответы от эксперта (экспертов), он выбирает наиболее подходящий, который затем добавляется к базе знаний. Для ответа на вопрос эксперт может как непосредственно написать ответ, так и присоединить Web-страничку.

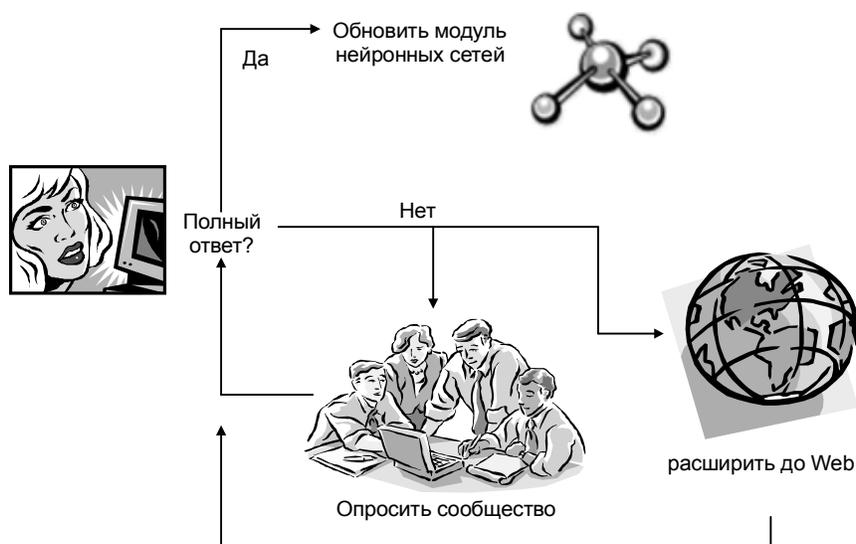


Рис. 7.10. Получение решения из Web и от экспертов

Система основана на технологии EJB (Enterprise JavaBeans – технология для создания Java-компонентов).

7.5. Web-порталы

В последние десятилетия Web-сеть сделала электронно-доступными огромное количество мировых информационных ресурсов. От узко специализированного средства для ограниченного количества пользователей в 1990 г. Web стал сегодня инструментом для огромного количества пользователей по доступу к информационному фонду более чем в 10 млрд документов. Если же учесть динамически генерируемые страницы, которые не обнаруживаются традиционными системами поиска (deep web), то количество доступных в этой сети документов оценивалось примерно в 550 млрд уже в 2001 г.

Неоспоримый успех Web-технологии обеспечен легкостью доступа к мировым информационным ресурсам различных групп пользователей: разработчиков программного обеспечения, провайдеров и потребителей контента. Вместе с тем упомянутая простота Web, обеспечившая стремительное развитие сети, имеет серьезные и, в некоторых случаях, даже критические недостатки и ограничения в части поиска, извлечения и интерпретации информации. Их преодоление потребовало новых системотехнических решений, получивших название Web-порталы [Портал как инструмент государственного управления].

Web-порталы – это особые точки доступа к информационным ресурсам через Интернет, используемые профессиональными сообществами и сообществами по интересам. Они предоставляют пользователям средства поиска необходимой информации в соответствии с предпочтениями, проблемами, темами и т.п. Порталы имеют мощную технологическую поддержку для организации коммуникации и обмена информацией, контекстного поиска и семантической обработки, а также для интерпретации данных, информации, знаний. В некоторых случаях пользователи, объединенные в сообщества по интересам, создают собственное сообщество внутри более общего сообщества, а также некий раздел портала для более оперативного предоставления и обмена информацией по специфической теме.

В настоящее время создано множество разнообразных Web-порталов, которые с достаточной степенью обоснованности можно подразделить на следующие основные разновидности:

- Информационные Web-порталы.
- Знаниевые Web-порталы.
- Семантические Web-порталы.

Логика приведенной классификации очевидна. По сути, каждая последующая разновидность отличается от предыдущей более широкими технологическими возможностями в поиске, обработке и интерпретации данных, информации и знаний. Остановимся на каждом типе порталов, под которыми, с учетом корпоративного акцента данной монографии, будем подразумевать, прежде всего, Web-порталы предприятий.

Информационный Web-портал предприятия – это специально подобранный сплав программных приложений, который открывает доступ сотрудникам к информации, хранимой внутри и вне предприятия, обеспечивает единый вход к персонализированной информации, необходимой для принятия решений. Опираясь на такие современные направления, как Data Warehouse&Mart, Data Management и др., информационный Web-портал предприятия:

- 1) использует «push» и «pull» для навигации и поиска информации посредством стандартного Web-интерфейса;
- 2) предоставляет доступ как к внешним, так и к внутренним источникам данных и информации;
- 3) объединяет разнородные приложения, включая управление данными, хранилище данных/витрина данных и другие внешние для этих приложений данные в единую систему, которая может распределять, управлять и поддерживать информацию с использованием одного центрального пользовательского интерфейса;
- 4) предоставляет «интерактивность» – способность запрашивать и делиться информацией с персонального компьютера пользователя.

Знаниевый Web-портал предприятия [Collins H., 2003; Firestone J.M., 2003] является развитием информационного Web-портала с помощью методов и инструментов управления знаниями. С их помощью становится возможным обеспечить:

- единый взгляд на организацию;
- возможности организации и поиска информации и метаданных;
- прямой доступ к корпоративной информации и знаниевым ресурсам;
- прямую связь с отчетами, анализами и запросами;
- прямую связь с релевантными данными и опытом специалистов;
- индивидуальную идентичность и персонализированный доступ к контенту;
- выявление специалистов, с которыми можно обсуждать проблемы (сотрудничать);

- выявление специалистов, с которыми можно обмениваться (делиться) знаниями;
- принятие бизнес-решений.

Знаниевый Web-портал обеспечивает удовлетворение насущных нужд интеллектуальных сотрудников предприятия путем самообслуживания, помощи самим себе и самостоятельным открытиям в различных аспектах бизнеса. Он обеспечивает доступ менеджеров к специфичным экранам (формам) или отчетам, к экспоненциально растущему количеству документов, имеющим отношение к бизнес-процессам, и полную интеграцию с различными информационными системами предприятия, а также эффективный доступ к внешним ресурсам.

Большое внимание в знаниевых Web-порталах уделяется описанию и представлению рабочих процессов пользователям, с учетом их уникальных ролей и функций, которые они выполняют в этих процессах. Каждый процесс разделяется на конкретные виды деятельности, которые включают набор задач, операций. Определенные роли назначаются и рабочим процессам. В порталах, по мере их развития, прописываются сотни рабочих процессов. Множится и количество индивидуумов, имеющих отношение к исполнению каждой роли.

Знаниевые Web-порталы имеют объектно-ориентированную архитектуру. Каждая страница является набором нескольких элементов, которые отдельно разрабатываются и динамически генерируются для пользователей, когда они обращаются к portalу. Кроме того, используя уникальные роли и права доступа пользователей, контент, загружаемый на страницы портала, различен, т.е. гарантируется, что каждый пользователь будет иметь в чем-то отличный от других пользовательский интерфейс. Схемы навигации отражают процессы, к которым пользователь имеет доступ, а окна содержания (контента) на экране показывают только ту информацию, которая специфична для определенных ролей и задач, актуальных для конкретного пользователя. Информация в окнах обеспечивается правилами обеспечения безопасности и проверки паролей доступа пользователя к приложениям и источникам данных.

Хорошо спроектированные порталы и связанные с ними потоки работ (workflow) являются критически важными для предоставления только того контента, который нужен им для выполнения работы с гарантией, что рабочие процессы будут успешно выполняться.

Таким образом, можно констатировать, что знаниевый Web-портал – это высокоразвитый персонифицированный интерфейс к он-лайн ре-

сурсам для интеллектуальных работников организации, а также для системной интеграции программных приложений, данных и знаний. Он позволяет работникам компании получать более эффективный доступ к информации и знаниям, облегчает общение друг с другом, повышает обоснованность принимаемых решений и действий по широкому кругу бизнес-процессов. Причем все это вне зависимости от местоположения работников или их принадлежности к определенным бизнес-структурам, от способа размещения информации и формата ее хранения.

Семантический Web-портал (SW-портал) – это специфический Web-сайт, предоставляющий дополнение к инструментам знаниевого Web-портала средства семантической обработки данных и информации, основанных на технологиях Semantic Web [Staab S., Maedche A., 2001; Staab S., Studer R., 2002]. Он базируется на особом способе формализации предметных областей и бизнес-процессов с помощью метаописаний и онтологий.

Формализованная таким образом семантика документов может обрабатываться компьютерами, передаваться между различными агентами, в качестве которых могут быть программы и люди. Другими словами, в семантическом портале информация становится понятной компьютеру и, таким образом, появляется возможность автоматизировать поиск, извлечение, обработку и интерпретацию информации. SW-порталы призваны существенно улучшить процесс обмена информацией, процесс создания и использования знаний, и в этом отношении они являются новым поколением Web-порталов.

Как уже отмечалось, Web-порталы различаются, прежде всего, по технологическим возможностям. Применительно к SW-порталам можно выделить три технологических уровня, приведенных на рис. 7.11 [Lausen H., Stollberg M., et al., 2004].

К трем основным технологическим уровням относятся:

- Доступ информации (Information Access) с точки зрения пользователя.
- Особенности обработки информации (Information Processing) на портале.
- Базовые технологии (Grounding Technologies).

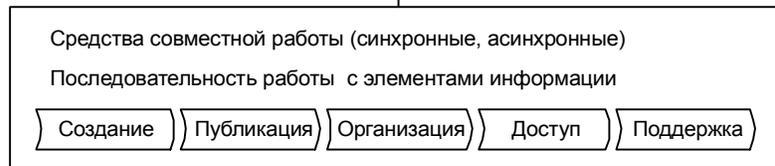
Три уровня выделены на рис. 7.11 по аналогии с трехуровневой архитектурой, которая используется в проектировании программного обеспечения информационных систем. Уровни разделены жирной ли-

ний. В приведенной трехуровневой архитектуре каждый уровень инкапсулируется определенными интерфейсами от нижележащего уровня, таким образом позволяя использовать его функциональность без необходимости понимать детали его реализации.

Уровень доступа к информации



Уровень обработки информации



Уровень базовых технологий

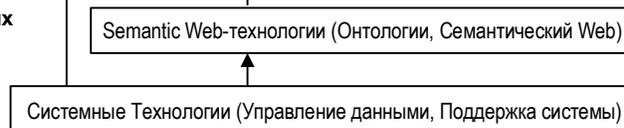


Рис. 7.11. Уровни Semantic Web-портала

Серые прямоугольники на рис. 7.11 группируют различные аспекты, стрелки обозначают взаимозависимость относительно потока информации между различными компонентами. Рабочий поток (workflow) информационных единиц соответствует этапам (шкагам) одного потока работ.

К базовым технологиям относятся:

- Системные технологии – управление данными (Database RDF repository), передача данных, поддержка (администрирование) портала, безопасность (SSL соединение или защита паролями частей SW-портала).
- SW-технологии: онтологии, логический вывод (Inference and Reasoning), управление онтологиями (редактирование, поддержка, поиск, взаимодействие), SW-сервисы.

Добавление к Web-сервисам информационных порталов Semantic Web Services создает возможность автоматической классификации, поиска, объединения и взаимодействия ресурсов и инструментов. Функциональность портала обогащается при этом как по линии интерфейса пользователя-человека, так и за счет программных агентов.

К уровню обработки информации (Information Processing) в SW-порталах относят:

- Рабочие процессы (Information Item Workflow), включающие под-процессы: создание, публикация, организация, доступ и поддержка.
- Сотрудничество (Collaboration Features), включающее помимо работы с документами сотрудничество пользователей, поддержку виртуальных групп в синхронном и асинхронном режимах.

Синхронное сотрудничество достигается посредством систем поддержки чатов (chat), интегрированных в портал и связанных со специальными темами или документами. Более того, сообщество портала может быть усилено средствами, предоставляющими возможность вводить текст в некотором общем окне (collaborative writing features), таком, как общая доска публикаций (common whiteboard area), в которой несколько пользователей могут одновременно манипулировать изображением.

Асинхронные возможности взаимодействия обычно более продвинуты, так как они не накладывают обязанность точного взаимодействия во времени распределенного сообщества, даже учитывая то, что эффект общественной связанности, получаемый с их помощью, менее интенсивен. Электронная почта (E-mail) является здесь наиболее часто используемым средством. Она может быть интегрирована в портал путем предоставления E-mail адресов в персональных профилях пользователей в качестве начального момента персонального общения двух пользователей или путем поддержки почтовых списков (mailing lists), которые могут использоваться для распределения информации. Новостные группы (News Groups) и дискуссионные форумы (Discussion Forums) также попадают в эту категорию, но они используют предоставление информации по требованию (on demand methodology), а не принудительное представление информации (interrupt driven methodology), которое применяется в почтовых листах рассылки.

Системы он-лайн-календарей групп (On-line group calendar systems), которые хорошо известны в системах групповой работы, также могут использоваться для организации сотрудничества пользователей портала. Во-первых, они могут использоваться для координации собы-

тий синхронного общения в сообществах, таких, как чаты или конференции, а во-вторых, для доставки информации о важной, связанной со временем информации, такой, как рабочие планы или сообщения об интересующих все сообщество событиях.

Уровень доступа к информации представляет собой интерфейс пользователя (front-end) SW-портала для сообщества пользователей.

Как любое Web-приложение, SW-портал должен быть удобен в использовании и доступен для того, чтобы предоставлять дружественный интерфейс для виртуального сообщества. Кроме этого, SW-портал должен предоставлять совершенную функциональность с семантическими возможностями.

Важным свойством Web-портала является удобство использования. Чтобы SW-портал был эффективным, он должен позволять пользователям выполнять их задачи наилучшим доступным способом. Удобство использования является качеством портала, которое делает его простым для понимания и запоминания, легким для использования, терпимым к ошибкам, субъективно приятным в работе.

Как отмечалось ранее, Web-порталы более высокого уровня (знаниевые, семантические) являются поступательным развитием информационных порталов. Причем это развитие осуществляется за счет использования новых технологий, которые порождают и новую функциональность. С учетом этого, можно предложить следующую классификацию Web-порталов (рис. 7.12).

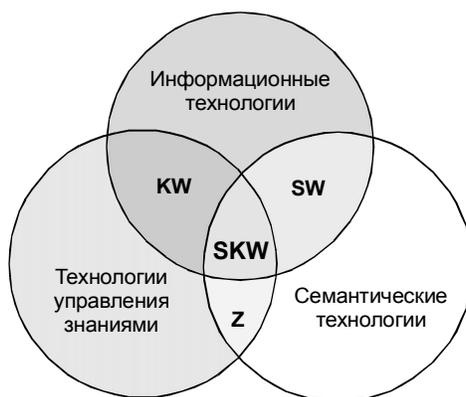


Рис. 7.12. Классификация Web-порталов на основе используемых технологий

Как следует из рис. 7.12, Web-портал управления знаниями (KW) есть системотехнический продукт, сочетающий достижения информационных технологий и технологий управления знаниями. Семантический Web-портал (SW) есть системотехнический продукт, сочетающий достижения информационных и семантических технологий. Системотехническим продуктом, сочетающим достижения всех трех групп технологий (информационных, управления знаниями, семантических), является семантический Web-портал управления знаниями (SKW).

Сегмент, помеченный знаком Z на рис. 7.12, нельзя отнести к категории Web-порталов. Web-портал не может существовать без базовых информационных технологий, так как Web является, прежде всего, информационным продуктом.

В настоящее время реализовано достаточно много Web-порталов, которые можно отнести к классам KW, SW, SKW. Остановимся далее на описании одного из них, который более всего подходит к определению SKW-портала – это Esperanto Portal.

Esperanto Portal является вариантом использования ODESeW-генератора портала знаний (ODESeW knowledge portal generator), разработанного группой по онтологиям факультета информатики Мадридского политехнического университета (Ontology Group at Facultad de Informatica, Universidad Politecnica de Madrid). Он служит как внутренней, так и внешней платформой для проекта Esperanto Европейского Сообщества. Основные характеристики портала Esperanto представлены в табл. 7.1.

Таблица 7.1

Основные характеристики портала Esperanto

Технология	Название	Комментарии
1	2	3
Операционная система	Windows 2000	Возможен также Unix/Linux
База данных	Oracle 8.1.7.	Доступна для любой СУБД, совместимой с JDBC
Хранилище документов	Файловая система	
Web Server	Minerva Application Server+ Tomcat	Планируется переход на другие продукты
Применяемая онтология	Пять различных онтологий предметных областей	
Язык онтологий	WebODE Knowledge Model	Совместим с OKBC

Продолжение табл. 7.1

1	2	3
Система логического вывода	Основана на ОКВС	Используется при построении онтологии
Редактор онтологий	WebODE Ontology Editor, OntoDesigner, WAB	Возможно использовать другие с помощью средств импорта
Технологии пользовательского интерфейса UI	HTML, XML, JSP, Java апплеты	
Совместимость с браузерами	IE, Netscape 4.7 и выше	
Цена/лицензия	Свободная демонстрация	
Web-сайт продукта	http://www.esperonto.net	

Портал имеет трехуровневую классическую архитектуру (рис. 7.13). База данных Oracle и файловая система составляют уровень хранения данных. На среднем уровне Minerva Application Server отвечает за соединение с БД и WebODE Ontology репозитарием и за управление сервисами. Презентационный уровень реализуется стандартным браузером. Выполнение сервлетов осуществляется Web-сервером Apache Tomcat.

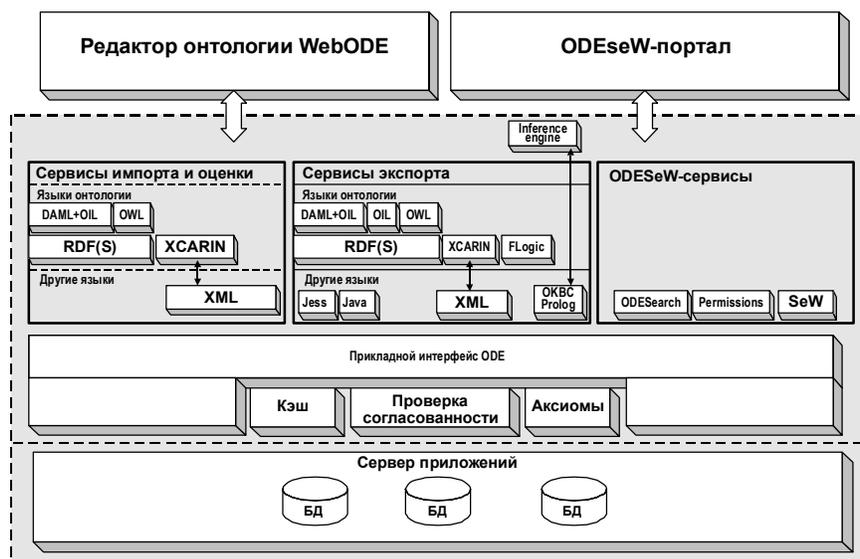


Рис. 7.13. Архитектура среды инженерии онтологий WebODE, используемой в SKW-портале Esperonto

Работа с онтологиями полностью выполняется нижележащей платформой WebODE, которая сама использует реляционную БД для хранения описаний. Для портала Esperanto было разработано пять различных онтологий предметных областей, а именно проектов, документации, людей, организаций и встреч. Эти онтологии описывают научно-исследовательские проекты, особенно IST-проекты и предназначены для многократного использования. Эти онтологии связаны несколькими отношениями. Однако для возможностей усовершенствованного поиска учитываются только непосредственные атрибуты одной онтологии.

Более детальный вариант архитектуры приведен на рис. 7.14 [Sheth A., Bertram C., Avant D., 2002].

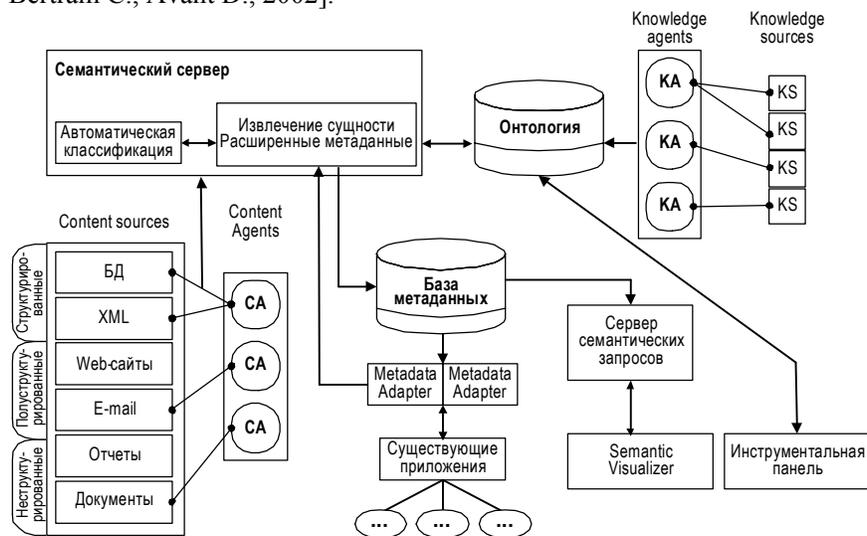


Рис. 7.14. Архитектура системы Semagix Freedom

Как следует из приведенных вариантов архитектуры, основными элементами СУЗ, развивающими функциональность современных ИС, являются репозитории метаописаний и онтологии, семантические серверы, контент-агенты, инструменты автоматической классификации, знаниевые ресурсы и агенты, а также другие, связанные с ними приложения. Будучи надстройкой над информационной системой, СУЗ имеет развитые интерфейсы к БД, Web-сайтам, отчетам, документам и традиционным сетевым сервисам Интернет и интранет.

На рис. 7.15 представлена архитектура СУЗ, реализованной в рамках программы IST Евросоюза по проекту On-To-Knowledge [Davies J., Fensel D., Frank van Harmelen, 2003].

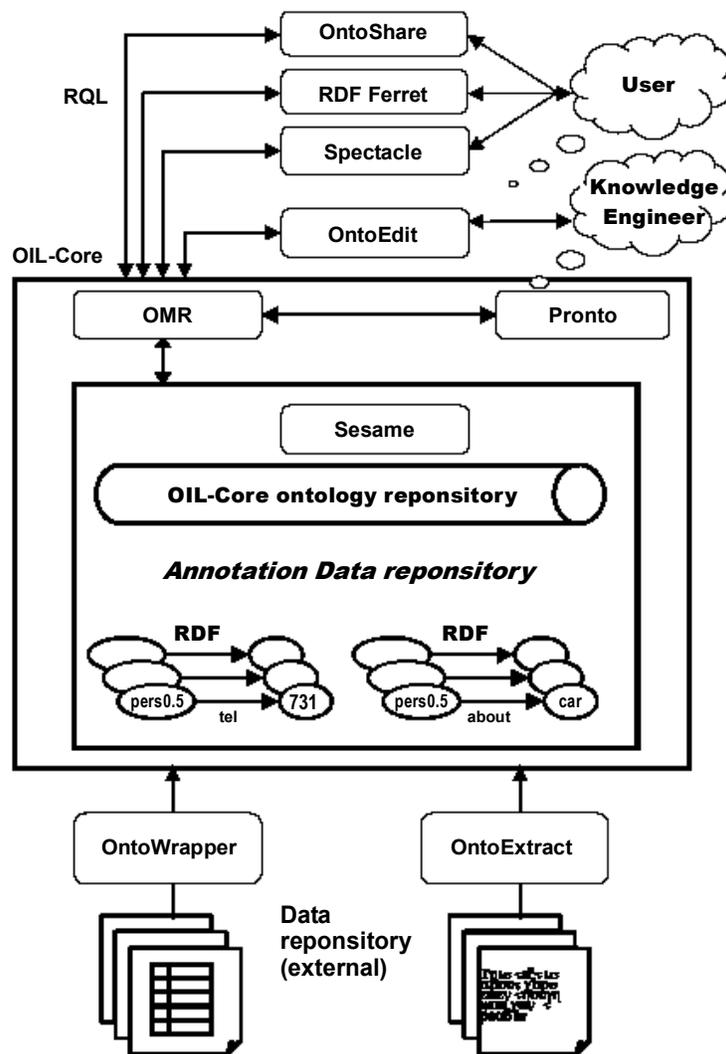


Рис. 7.15. Архитектура СУЗ в On-To-Knowledge

Ядром инструментальной части в приведенной СУЗ является система Sesame [Davies J., Fensel D., Frank van Harmelen, 2003], архитектура которой приведена на рис. 7.16.

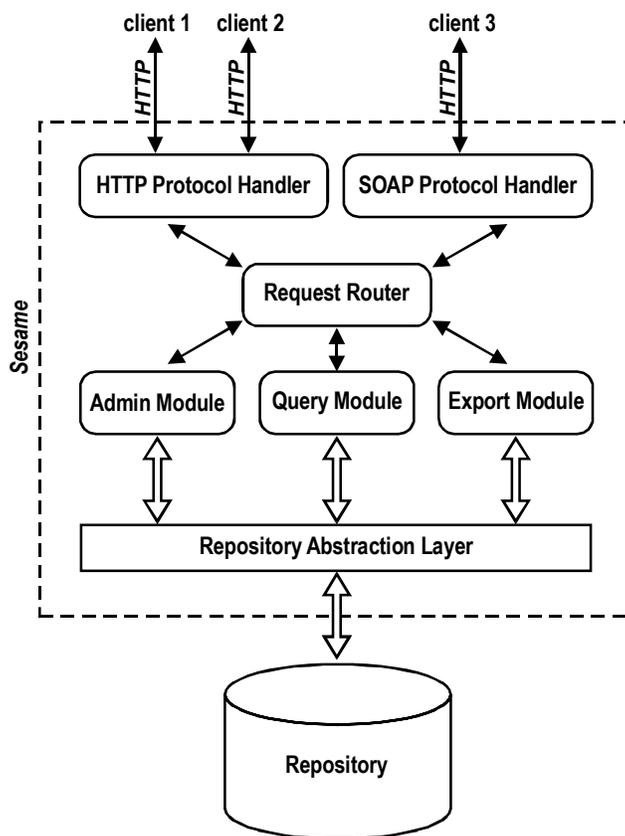


Рис. 7.16. Архитектура системы Sesame

К настоящему времени реализован целый ряд проектов, завершившихся созданием Web-порталов. К числу наиболее известных относятся Onto Web Portal, созданный в рамках проекта IST-2000-29243 в интересах промышленных партнеров, а также такие порталы, как: Empolis K42, Mondeca ITM, SWWS-портал, Mindswap, Onto Web Edu, Karlsruhe Portals.

Более детально состав блоков, элементов и связей между элементами SKW-портала показан на рис. 7.17, где приведена архитектура системы SCORE [Sheth A., Bertram C., Avant D., 2002].

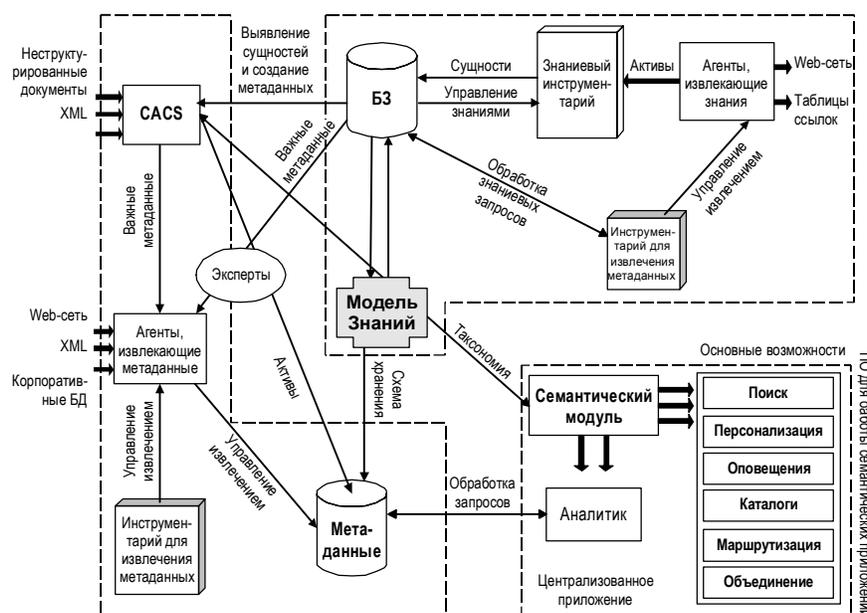


Рис. 7.17. Архитектура системы SCORE

В системе SCORE портал реализует все упомянутые выше функции и процессы поиска и обработки данных, метаописаний и знаний, используя широкий набор информационных и семантических инструментов, сервисов и агентов.

7.6. Состав и последовательность этапов по созданию СУЗ

Система управления знаниями – сложный и весьма дорогостоящий объект. Понятно, что решение о создании СУЗ не должно приниматься в отрыве от ряда существенных факторов, определяющих состояние деятельности конкретной компании, стратегию развития ее бизнеса, уровень имеющейся информационной системы и ряда других факторов

[Румизен М.К., 2004; Букович У., Уильямс Р., 2002]. По существу, ситуация в каждой компании в этом отношении уникальна, и потому решения о путях и способах создания СУЗ могут существенно различаться.

В ряде случаев *настраивание готовой системы* (Customised Off The Shelf), созданной некой компанией для себя либо для тиражирования на рынке, является достаточно популярным способом развертывания СУЗ. При этом, исходя из реальных потребностей, компанией могут приобретаться как система целиком, так и отдельные подсистемы и предоставляемые сервисы, которые затем настраиваются и встраиваются в информационную систему компании.

Несколько иной подход основан на *реинжиниринге решений* (Solution Re-engineering), когда с помощью консультантов по управлению знаниями и системотехников (своих либо внешних) осуществляется перепроектирование и приспособление неких общих решений по СУЗ к особенностям и потребностям конкретной компании. Хотя это и напоминает настраивание готовой системы, здесь есть и существенные отличия, и прежде всего в отсутствие претензий на создание конечного рыночного продукта.

Для многих компаний эффективной альтернативой является использование *сервисов знаний* (Knowledge Services). Как правило, это совокупность приложений, обеспечивающих возможность работы со знаниями, предоставляемыми внешними компаниями посредством Web-сети. Очевидным преимуществом является здесь отказ от платы за лицензию на программу и отсутствие необходимости ее поддержки. Однако многих не устраивают возникающие при этом проблемы безопасности и защиты прав интеллектуальной собственности.

В ряду возможных альтернатив находится обращение к *рынку знаний* (Knowledge Marketplace). На рынке знаний компания-потребитель владеет Web-сайтом, который абсорбирует большое количество поставщиков сервисов знаний. В качестве поставщиков могут быть: эксперты, сервисы, агентства и исследовательские организации, процедуры оценивания порталов и др.

Если же речь идет о мощной амбициозной компании, располагающей современной инфраструктурой информатизации и персоналом, в совершенстве владеющим новейшими информационными технологиями, и основывающей дальнейшую стратегию развития своего бизнеса на знаниях, то предпочтительной альтернативой для нее является *самостоятельная разработка СУЗ* (In-house Development).

Применительно к этой ситуации и излагается далее состав и последовательность этапов работ по созданию СУЗ.

Можно выделить следующие основные этапы, которые в дальнейшем будут детализированы до уровня состава работ [Tiwana A., 2000]:

1. Аудит компании, информационной системы и существующих активов знаний.

2. Проектирование СУЗ.

3. Развертывание и создание СУЗ.

4. Оценка эффективности и совершенствование СУЗ.

Аудит, осуществляемый на первом этапе, включает обследование, оценку и анализ:

1.1. Стратегии развития компании.

1.2. Основных направлений деятельности.

1.3. Организационной структуры компании.

1.4. Структуры и состава информационной системы (подсистемы, задачи, модели, алгоритмы).

1.5. Инфраструктуры информационной системы (каналы, серверы, ЛВС, интранет и экстранет).

1.6. Информационных ресурсов информационной системы (файлы, БД, БЗ, порталы).

1.7. Программного обеспечения информационной системы (системного, общего назначения, прикладного).

Аудит может выполняться аутсорсинговой компанией или специальной группой экспертов, сформированной руководством компании.

Проектирование СУЗ предполагает:

2.1. Определение роли СУЗ в бизнес-стратегии и корпоративной культуре компании.

2.2. Разработка концепции СУЗ.

2.3. Формирование и/или привлечение команды для разработки проекта СУЗ.

2.4. Разработка проекта СУЗ (архитектура, состав подсистем, задач и сервисов).

2.5. Разработка инфраструктуры СУЗ (корпоративная память, среды и технологии поиска, сотрудничества).

2.6. Выбор программных систем и инструментов инфраструктурного и технологического характера.

2.7. Разработка технологии функционирования подсистем, сред и сервисов.

Поэтапное развертывание и создание СУЗ предполагает:

3.1. Приобретение и установку инфраструктурных и программных элементов.

3.2. Формирование баз данных, репозитория и баз знаний.

3.3. Внесение усовершенствований в организационные и технологические регламенты выполнения работ.

3.4. Управление изменениями, культурой отношений и вознаграждениями.

Оценка эффективности СУЗ предполагает:

4.1. Определение совокупности количественных и качественных критериев эффективности нововведений и методики их агрегации.

4.2. Оценка возврата инвестиций (ROI).

Совершенствование СУЗ по опыту ее развертывания и функционирования.

4.3. Ряд перечисленных этапов и работ носят общий характер в методологии проектирования и создания информационных и управляющих систем. СУЗ, как специфическая система, имеет ряд отличительных особенностей, которые проявляются в создании таких ее компонентов, как онтологии, репозитории знаний, поисковые системы, среды для взаимодействия экспертов и проектных групп. Далее приводится состав задач по специфическим для СУЗ компонентам.

Построение инженерной среды KMS:

a. Анализ «узких мест» в инфраструктуре ИС, исходя из потребностей СУЗ.

b. Установка и освоение дополнительных программных систем.

c. Тестирование их функциональных возможностей и адаптация к требованиям СУЗ.

d. Определение состава дополнительно разрабатываемых программных средств.

e. Определение специализации и расстановка серверов KMS в информационных системы компании.

Построение онтологий системы:

a. Определение состава онтологий верхнего уровня.

b. Определение программных средств для работы с онтологиями в распределенной среде (закупка или разработка).

c. Установка программных средств.

d. Формирование групп для создания онтологий по отдельным направлениям и организация их работы.

e. Определение языка метаописаний.

Создание репозитария знаний:

a. Разработка (или освоение, адаптация) программных средств работы пользователя с репозитарием.

b. Разработка (или освоение, адаптация) языковых и программных средств составления метаописаний объектов хранения в репозитарии (категоризация объектов).

c. Создание (адаптация) программных средств для работы с репозитарием (публикация, получение, просмотр).

Создание средств семантического поиска в репозитарии и компьютерных сетях:

a. Поиск знаний в отдельных документах на основе понятий онтологии.

b. Поиск в репозитарии явных знаний.

c. Поиск в базе знаний профилей компетентности специалистов.

d. Поиск в интранет и Интернет-сетях (периодический, по заявкам).

Создание среды для форумов и дискуссионных групп специалистов компании:

a. Разработка (адаптация) программных средств поддержки форумов и дискуссионных групп.

b. Организация базовых дискуссионных групп.

c. Проведение организационно-методической работы по включению дискуссионных групп в практику работы компании.

Создание сети экспертов:

a. Разработка (адаптация) программных средств для описания и поиска экспертов.

b. Разработка (адаптация) программных средств для проведения консультаций с экспертом.

c. Определение профилей компетентности специалистов.

d. Определение статуса эксперта.

e. Определение состава экспертов.

Создание среды для работы распределенных проектных групп:

a. Определение регламента работы проектных групп.

b. Разработка средств для работы участников и руководителей проектных групп.

c. Разработка средств ведения библиотеки проектной группы.

d. Разработка средств для on-line общения участников проектных групп.

e. Фиксация структуры и состав результатов работы проектных групп.

Перечисленные этапы и работы раскрывают, но, естественно, не исчерпывают всю совокупность того, что предстоит сделать в процессе создания СУЗ. Следует также иметь в виду взаимосвязанность решений

и результатов на каждом из этапов, наличие циклов возврата к ранее принятым решениям с целью их уточнения, оптимизации и т.п. Тем не менее их можно принять в качестве начального приближения для планирования и организации работ по созданию столь нового и сложного объекта, каким является СУЗ.

7.7. Проблематика КТ и KMS в зарубежных исследованиях и разработках

В предыдущих разделах монографии приводилось немало аргументов относительно важности методологии управления знаниями на данном уровне развития техники и технологии, на данном уровне глобализации экономики, на данном уровне развития человеческих ресурсов и культуры корпоративного управления. Отмечалось также, что управление знаниями как новое научное и прикладное направление стало развиваться немногим более десяти лет назад, однако темпы роста публикаций на эту тему кратно увеличиваются из года в год. В качестве иллюстрации в табл. 7.2 приводится статистика публикаций, почерпнутая из Интернет с помощью поисковой системы Google по запросам КМ (Knowledge Management) и KMS (Knowledge Management System).

Таблица 7.2

Разделы	2002 г.	2003 г.	2004 г.
КМ	3.410.000	4.880.000	10.300.000
KMS	2.610.000	3.660.000	7.730.000

Такой рост публикаций свидетельствует о растущем внимании к проблематике управления знаниями со стороны научного мира. Однако от этапа методологических и теоретических исследований во многих странах мира уже давно перешли к этапу ориентированных на практическое применение разработки технологий, элементов и систем управления знаниями.

В качестве подтверждения этого вывода в табл. 7.3 приводится выборка проектов, финансируемых в пятой рамочной программе Евросоюза, размещенных в Интернете на сайте КТ Web (Knowledge Technology Web – www.ktweb.org). В приведенных описаниях содержится наименование проекта, его описание, координатор, стоимость и время выполнения. С помощью поисковой системы CORDIS (www.cordis.lu/ist) возможно получение более подробной информации о существовании каждого из этих проектов.

Таблица 7.3

**Описания некоторых проектов разработки систем управления знаниями,
финансируемые в пятой рамочной программе ЕС**

Название	Описание	Участники	Финансирование, сроки	Web-сайт
1	2	3	4	5
OntoWeb (Thematic network)	Основанный на онтологиях обмен информацией для УЗ и электронной коммерции	Institute of Computer Science, University of Innsbruck (Project co-ordinator) 25 contractors и 146 организаций-участниц	1.87 million euro 2001 – 2004	http://ontoweb.ontoware.org/
Knowledge Web (Network of Excellence)	Развитие технологий Semantic Web для промышленности (eWork и eCommerce). Включает образовательные и исследовательские цели	University of Innsbruck, Austria, 18 contractors и 5 партнеров	6.7 million euro 2004 – 2008	http://knowledgeweb.semanticweb.org/index.html
ACCESS-MAINTS	Реализация платформы корпоративного управления знаниями для поддержки инженерных работ в географически распределенных подразделениях больших и малых компаний	Societa Italiana Avionica S.P.a., Italy, 10 организаций-участниц	2000 – 2002 1.6 million euro	http://www.metriqs.it/ACCESSMAINTS/index.htm
AcKnowNET	Разработка и проверка решений по «Активным сетям знаний» для фиксации, интеграции, управления и распространения знаний, поддержки сотрудничества, управления workflow и планирования	DFKI GMBH – Knowledge management department, German, 3 организации-участницы	349958.00 euro 2001 – 2002 (6 monthes)	http://www.tupaisystems.co.il/Ac-knownetSite/
ASSIST	Разработка справочных систем больших компаний, которые поддерживают удаленных работников средствами управления знаниями	Saios Sa, Luxembourg, 4 организации-участницы	1.40 million euro 2000 – 2002	www.assistknowledge.com

Продолжение табл. 7.3

1	2	3	4	5
AUDIOTAIN	Создание решений по УЗ, которые дадут возможность провайдерам аудио контента проверять и применять цифровые развлекательные аудио-сервисы	Sender Freies Berlin, GER-MANY, 5 организаций-участниц	1.70 million euro 2001 – 2003	www.audiotain.com
BIZON	Объединение исследований по онтологиям и знаниевым технологиям	Ksolutions S.P.a., Italy, 4 организации-участницы	989999.00 euro 2002 – 2003	http://www.ist-bizon.org/
CIKM	Исследование и анализ влияния УЗ на инновации Creation of innovation through knowledge management	Fraunhofer institut fur arbeitswirtschaft und organisation, Germany, 8 организаций-участниц	616000.00 euro 2002 – 2003	www.cikm-project.net
COCONET	Подготовка стратегии и плана по исследованиям и разработкам технологий, которые окажут большое влияние на учитывающие контекст системы совместной работы, в следующем поколении бизнеса	Stichting telematica instituut, Netherlands, 8 организаций-участниц	299778.00 euro 2002 – 2003	http://coconet.telin.nl
COMMA	Реализация и проверка систем управления корпоративной памятью на основе агентной технологи	Atos Integration S.A., FRANCE, 8 организаций-участниц	997989.00 euro 2000 – 2002	http://www.ii.atos-group.com/sphia/comma/homepage.htm
CORMA	Разработка среды УЗ для поддержки процессов разработки новой продукции	Nokia Networks Oy, FINLAND, 5 организаций-участниц	1.40 million euro 2000 – 2002	www.corma.net
DÉCOR	Разработка интегрированных методов и инструментов для автоматизации выполнения бизнес-задачи, активно использующих знания	Deutsches Forschungszentrum fuer kuenstliche intelligenz gmbh, German, 6 организаций-участниц	1.40 million euro 2000 – 2002	http://www.dfki.uni-kl.de/decor/index.html
E-COGNOS	Разработка основанных на онтологиях и моделях адаптивных систем, которые на основе понимания семантики разнородных документов в Европейской	DERBI, France, 6 организаций-участниц	1.99 million euro 2001 – 2003	http://e-cognos.cstb.fr/default.htm

Продолжение табл. 7.3

1	2	3	4	5
	строительной промышленности смогут организовать документы в соответствии с их контентом и взаимозависимостями			
ECOLNET	Создание бизнес-модели построения сетей сотрудничества и информационных систем для поддержки обмена знаниями между средними и малыми предприятиями	IBERMATICA S.A., Spain, 7 организаций-участниц	900000.00 euro 2000 – 2001	www.ecolnet.com
ENKE	Совершенствование внедрения УЗ на предприятиях с помощью двух специализированных программных систем	EUROCOPTER RESEARCH PROGRAMME DIRECTORATE, FRANCE, 6 организаций-участниц	1.09 million euro 2001 – 2003	http://www.ist-enke.com
H-TECHSIGHT	Разработка платформы, способствующей обнаружению технологических трендов, требуемого опыта, новых направлений и рынков	The University of Surrey Department of Chemical and Process Engineering, UNITED KINGDOM, 9 организаций- участниц	2.00 million euro 2002 – 2004	http://prise-serv.cpe.surrey.ac.uk/techsight/index.asp
IBROW	Разработка интеллектуальных брокеров, которые способны распределенно конфигурировать многократно используемые компоненты в знаниевых системах через WWW	UNIVERSITEIT VAN AM- STERDAM SOCIAL SCI- ENCE INFORMATICS, NETHERLANDS, 7 организаций-участниц	1.10 million euro 2000 – 2003	http://www.swi.psy.uva.nl/projects/ibrow
ICONS	Разработка и использование основанных на Web платформ по управлению знаниями и мультимедиа контентом на основе использования результатов, полученных в области ИИ и СУБД	Rodan Systems Spolka Akcyjna, Poland, 7 организаций-участниц	1.90 million euro 2002 – 2003	http://www.icons.rodan.pl
INKASS	Разработка основанного на технологиях сети Интернет рынка активами	Planet Ernst & Young S.A. Management Consulting	1.70 million euro 2002 – 2004	http://www.inkass.com

Окончание табл. 7.3

1	2	3	4	5
	знаний, который предназначен для средних и малых предприятий и их потребностей в знаниях	Services, Greece, 7 организаций-участниц		
KNOWCOAT	Разработка платформы для разработки, управления и распределения знаниями в индустрии покрытий	VALTION TEKNILLINEN TUTKIMUSKESKUS (VTT) TECHNICAL RESEARCH CENTER OF FINLAND, 6 организаций-участниц	2.00 million euro 2002 – 2004	http://www.zoorobotics.com
LEVER	Разработка интегрированной технологии УЗ и системы управления людскими ресурсами для использования ценности капитала знаний и капитала людских ресурсов	GRUPPO FORMULA S.P.A., Italy, 5 организаций-участниц	1.20 million euro 2000 – 2002	http://lore.cezannesw.com
LORE	Приспособление и проверка общих решений по УЗ для индустрии разработки программного обеспечения	Planet Sa, Greece, 7 организаций-участниц	650000.00 euro 2000 – 2001	http://www.kmlever.com и http://www.kmlever.org
On-To-Knowledge	Разработка инструментов и методов для поддержки УЗ, основанного на совместно и многократно используемых онтологиях	Vereniging voor Christelijk Wetenschappelijk Onderwijs Faculty of Sciences Division Of Mathematics, NETHER- LANDS, 7 организаций-участниц	1.34 million euro 2000 – 2002	www.ontoknowledge.org
VISION – Next Generation Knowledge Management (Thematic network contracts)	Разработка стратегического плана перехода к организационному УЗ нового поколения	ORSCHUNGSZENTRUM INFORMATIK AN DER UNIVERSITAET KARLSRUHE FOR- SCHUNGSBEREICH WIS- SENSMANAGEMENT, Ger- many, 7 организаций-участниц	400000.00 euro 2002 – 2003	http://www.km-vision.org

Как следует из названий приведенной выборки проектов, основное внимание в них уделяется онтологическим системам, семантическим технологиям, Web-сервисам, поисковым системам, мультимедийному, контентному анализу и другим технологиям и сервисам для работы со знаниями.

Разнообразны и объекты приложения КТ и КМС. Здесь и корпоративное управление, и управление качеством продукции, электронная коммерция и взаимодействие сообществ, принятие решений в процессе проектирования и образование.

Характерно также, что тематика проектов посвящена в основном созданию инструментальных средств и технологий реализации определенного класса функций и сервисов в операциях со знаниями. Локализованы и сферы приложения этих инструментов. Проекты, охватывающие создание целостной КМС с полным набором структурных, инфраструктурных и технологических элементов, вовлекающие в бизнес-процессы компании не только явные, но и неявные знания сотрудников, интеллектуальный капитал компании, в проектах стран Евросоюза пока отсутствуют.

Отмеченное обстоятельство послужило для нас дополнительным аргументом и побудительным мотивом как к написанию монографии, так и к определению ее содержания.

ЗАКЛЮЧЕНИЕ

Корпоративные системы управления знаниями являются в последние годы объектом пристального внимания, исследования и разработок ученых и практиков многих стран мира. Столь пристальное внимание к данному направлению определяется не только новизной и сложностью проблемы. Не в меньшей мере оно является следствием того влияния, которое системы управления знаниями оказывают на конкурентные преимущества компании в современных условиях глобализации экономики в таких важнейших направлениях, как активизация человеческого потенциала, интеллектуализация бизнес-процессов, повышение корпоративной культуры, оптимизация принимаемых решений. Компании, создающие и эффективно использующие знания, отличаются более высокой компетентностью, оперативностью, производительностью, эффективностью.

Управление знаниями – это новый этап в развитии корпоративных информационных систем, призванный существенно повысить эффективность вложений в инфраструктуру информатизации. Управление знаниями – это и новая методология совершенствования управления компанией как системой.

Уровень развития науки, техники, технологии существенно отличается в разных странах мира. Россия, все еще обладающая высоким научным и инженерным потенциалом, существенно отстает от лидеров мировой экономики (Европы, Азии, Америки) в технологическом отношении, в том числе в сфере IT-технологий. Этим, а также слабой вовлеченностью России в мировую экономику объясняется и то значительное запаздывание, с которым российское научное и бизнес-сообщество обратило свое внимание на проблематику KMS.

Изложенные обстоятельства имеют отношение к прошлому и настоящему, однако вовсе не препятствуют и скорее даже побуждают к более энергичным действиям в будущем. Крупные, средние и даже малые инновационные предприятия, все, кто сегодня по праву именуется «точками роста» новой экономики, могут и должны эффективно взаи-

модействовать с социальной бизнес-средой, опираться на корпоративные знания и знания персонала для оптимизации бизнес-процессов, качества и конкурентоспособности продукции для выработки стратегии своего развития.

В этих стратегических направлениях деятельности современных компаний им могут существенно помочь системы управления знаниями.

СПИСОК ЛИТЕРАТУРЫ

1. *Большая Советская Энциклопедия* / Под ред. А.М. Прохорова. – М.: Советская Энциклопедия, 1972. – Т. 9. – 555 с.
2. *Борохович Л., Монастырская А., Трохова М.* Ваша интеллектуальная собственность. – СПб.: Питер, 2001. – 416 с.
3. *Броуэн Т.С., Скэннел Э.* Это таинственное управление знаниями // Computerworld. – 1999. – № 9.
4. *Брукинг Э.* Интеллектуальный капитал: Пер. с англ. – СПб.: Питер, 2001. – 288 с.
5. *Букович У., Уильямс Р.* Управление знаниями: руководство к действию (Wendi R. Bukowitz, Ruth L. Williams The Knowledge Management Field-book). – М.: ИНФРА-М, 2002. – 504 с.
6. *Быстрицкий Е.К.* Практическое знание в мире человека // Заблуждающийся разум?: Многообразие вненаучного знания. – М.: Политиздат, 1990. – С.210–238.
7. *Вебер А.В., Данилов А.Д., Шифрин С.И.* Knowledge-технологии в консалтинге и управлении предприятием. – М.: Наука и техника, 2002. – 176 с.
8. *Гаврилова Т.А., Хорошевский Ф.В.* Базы знаний интеллектуальных систем. – СПб.: Питер, 2001. – 384с.
9. *Гаврилова Т.А.* Онтологический инжиниринг // Труды 8 национальной конф. по искусственному интеллекту. – М.: Физматлит, 2002. – С. 846 – 853.
10. *Данилин А.* О проблематике управления знаниями // Computerworld. – 1999. – № 31.
11. *Дейтл Х.М., Дейтл П.Дж. и др.* Как программировать на XML: Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2001. – 944 с.
12. *Климов С.М.* Интеллектуальные ресурсы организации. – М.: Знание; ИВЭСЭП, 2000. – 168 с.
13. *Люггер Д.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем. 4-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 864 с.
14. *Калятин В.О.* Интеллектуальная собственность (исключительные права): Учебник для вузов. – М.: НОРМА, 2000. – 480 с.

15. *Клецев А.С., Артемьева И.Л.* Математические модели онтологий предметных областей // Информационные процессы и системы, НТИ, серия 2. – 2001. – № 2. – С. 20 – 27.
16. *Лачинов В.М., Поляков А.О.* Информодинамика или путь к Открытому миру. – СПб.: Изд-во СПбГТУ, 1999. – 432 с.
17. *Нонака И., Такеучи Х.* Компания – создатель знания. Зарождение и развитие инноваций в японских фирмах (The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation). – М.: Олимп-Бизнес, 2003. – 384 с.
18. *Перегудов Ф.И., Ямпольский В.З. и др.* Основы системного подхода. – Томск: Изд-во Том. ун-та, 1976.
19. *Полани М.* Личностное знание. – М.: Прогресс, 1985. – 344 с.
20. *Портал как инструмент государственного управления, Решения для «Электронной России» на платформе Microsoft // Информационный бюллетень Microsoft: Спец. вып. – 2002.*
21. *Похолков Ю.П., Чучалин А.И.* (ред.) Менеджмент качества в вузе. – Томск: Изд-во ТПУ, 2004. – 251 с.
22. *Рубцов С.В.* Целевое управление в корпорациях. Управление изменениями. – М., 2001.
23. *Румизен М.К.* Управление знаниями: Как изменить вашу корпоративную культуру, чтобы люди не скрывали свои знания, а делились ими: Пер. с англ. (The Complete Idiots Guide to Knowledge Management Серия). – М.: АСТ, Астрель, 2004. – 318 с.
24. *Солсо Р.* Когнитивная психология. – СПб.: Питер, 2002. – 592 с.
25. *Спенсер П.* XML. Проектирование и реализация. – М.: Лори, 1999. – 509 с.
26. *Степин В.С.* Теоретическое знание. – М.: Прогресс-Традиция, 2003. – 744 с.
27. *Гоффлер Э.* Третья волна. – М.: Изд-во АСТ, 2002. – 768 с.
28. *Тузовский А.Ф., Ямпольский В.З.* Системы управления знаниями в образовании // Современные средства и системы автоматизации. – Томск: Изд-во Том. ун-та, 2002. – С. 295 – 299.
29. *Tuzovsky A.F., Yampolsky V.Z.* Knowledge management system, as new maturity level of company information system // Proceedings KORUS'2002. – Новосибирск, 2002. – P. 145 – 147.
30. *Tuzovsky A.F., Yampolsky V.Z.* The system approach to knowledge management systems designing and development // Proceedings KORUS'2003. – Ulsan, 2003. – P. 213 – 216.
31. *Тузовский А.Ф., Ямпольский В.З.* Основные принципы создания системы управления знаниями компании // Вычислительные технологии: Сб. научн. трудов. Т. 8. Спец. вып. – Новосибирск, 2003. – С. 26 – 34.

32. *Тузовский А.Ф., Васильев И.А.* Структура системы управления знаниями // Труды Междунар. симп. «Информационные и системные технологии в индустрии, образовании и науке». – Караганда, 2003. – С. 286 – 289.
33. *Тузовский А.Ф., Усов М.В.* Определение семантики произвольных документов // Труды Междунар. симп. «Информационные и системные технологии в индустрии, образовании и науке». – Караганда, 2003. – С. 289 – 292.
34. *Тузовский А.Ф., Ямпольский В.З., Васильев И.А.* Система управления результатами научно-технической деятельности предприятия // Кибернетика и вуз. – Томск, 2003, – Вып. 30. – С. 5 – 12.
35. *Тузовский А.Ф., Ямпольский В.З.* Информационные технологии в управлении знаниями // Кибернетика и вуз. – Томск, 2003. – Вып. 30. – С. 13 – 21.
36. *Тузовский А.Ф., Ямпольский В.З.* Интеллектуальное пространство в системах управления знаниями // Изв. вузов. Физика. – 2004. – № 7. – С. 23 – 29.
37. *Тузовский А.Ф., Ямпольский В.З.* От баз данных к базам знаний // Информационные технологии территориального управления, Томская модель информатизации (1972 – 2004 годы). – 2004. – № 42. – С. 49 – 56.
38. *Тузовский А.Ф., Ямпольский В.З.* Анализ концептуальных моделей работы со знаниями, как этап обоснования архитектуры системы управления знаниями // Изв. ТПУ. – 2004. – № 7. – С. 111 – 116.
39. *Тузовский А.Ф., Васильев И.А., Усов М.В.* Программная реализация основных компонент информационно-программного обеспечения системы управления знаниями // Изв. ТПУ. – 2004. – № 7. – С. 116 – 122.
40. *Турчин В.Ф.* Феномен науки: Кибернетический подход к эволюции. 2-е изд. – М.: ЭТС, 2000. – 368 с. (доступно по адресу <http://www.ets.ru/turchin/>).
41. *Управление корпоративными знаниями: обзор проблематики и технологий Lotus*, <http://www.it.ru/docs/encyclopedia/knowledge.pdf?737>
42. *Холман К.* XML и связанные с ним стандарты: Краткий справочник для руководителя, информационный бюллетень Jet Info. – 2000. – № 7 (86). – 24 с. (доступен на <http://www.jetinfo.ru/2000/7/2000.7.pdf>).
43. *Черняк Л.* Управление знаниями и информационные технологии // Открытые системы. – 2000. – № 10.
44. *Экк К.Д.* Знание как новая парадигма управления // Проблемы теории и практики управления. – 1998. – № 2. – С. 2 – 14.
45. *Alavi M., Leidner D.* Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research issues // Management Information Systems Quarterly. – V. 25. – No. 1. – P. 107 – 136.

46. *Applehans W., Globe A., Laugero G.* Managing knowledge: a practical web-based approach. – Addison Wesley Professional, 1999. – 115 p.
47. *Antoniou G., Frank van Harmelen.* Web Ontology Language: OWL. (in Handbook on Ontologies). – Springer Verlag, 2004. – P. 67 – 92.
48. *Baader F. (editor), et al.* The Description Logic Handbook. – Cambridge University Press, 2003. – 574 p.
49. *Baader F., Horrocks I., Sattler U.* Description logics as ontology languages for the semantic web // Dieter Hutter and Werner Stephan, editors, Festschrift in honor of Jörg Siekmann, Lecture Notes in Artificial Intelligence. – Springer, 2003.
50. *Berners-Lee T., Hendler J., Lassila O.* The Semantic Web // Scientific American. – May 2001.
51. *Collins H.* Enterprise knowledge portals: next generation portal solutions for dynamic information access, better decision making and maximum results. – New York: AMACOM, 2003.
52. *Corcho O., Fernández-López M., Gómez-Pérez A.* Methodologies, tools and languages for building ontologies. Where is their meeting point? // Data & Knowledge Engineering. – 2003. – V. 46(1). – P. 41–64.
53. *Daconta M.C., McDonald Bradley* Declaration of Data Independence. – 2003 (<http://www.daconta.net/briefs/Declaration-of-data-flyer.pdf>)
54. *Davenport T., Prusak L.* Working Knowledge: how organizations manage what they know. – Boston: Harvard Business School Press, 1998.
55. *Davies J., Fensel D., Frank van Harmelen (eds).* Towards the semantic Web: Ontology-driven Knowledge Management. – John Wiley & Sons Ltd, 2003.
56. *Davis R., Shrobe H., Szolovits P.* What is a Knowledge Representation? // AI Magazine. – 1993. – V. 14(1). – P. 17 – 33.
57. *Despres C., Chauvel D.* The Present and the Promise of Knowledge Management. – Butterworth-Heinemann, 2000. – 352 p.
58. *Drucker P.F.* Post-Capitalist Society. – Oxford University Press, 1993.
59. *Drucker P.F.* The age of social transformation // The Atlantic Monthly. – 1994. – V. 274. – No. 5. – P. 53 – 80.
60. *Erdmann M., Maedche A., Schnurr H.-P., Staab S.* From manual to semi-automatic semantic annotation: About ontology-based text annotation tools // P. Buitelaar & K. Hasida (eds). Proceedings of the COLING 2000 Workshop on Semantic Annotation and Intelligent Content, Luxembourg, August 2000.
61. *European Guide to good Practice in Knowledge Management – Part 1: Knowledge Management Framework*, <ftp://cenftp1.cenorm.be/PUBLIC/CWAs/e-Europe/KM/CWA14924-01-2004-Mar.pdf>

62. *European Guide to good Practice in Knowledge Management – Part 2: Organizational Culture*, <ftp://cenftp1.cenorm.be/PUBLIC/CWAs/e-Europe/KM/CWA14924-02-2004-Mar.pdf>
63. *European Guide to good Practice in Knowledge Management – Part 3: SME Implementation*, <ftp://cenftp1.cenorm.be/PUBLIC/CWAs/e-Europe/KM/CWA14924-03-2004-Mar.pdf>
64. *European Guide to good Practice in Knowledge Management – Part 4: Guidelines for Measuring KM*, <ftp://cenftp1.cenorm.be/PUBLIC/CWAs/e-Europe/KM/CWA14924-04-2004-Mar.pdf>
65. *European Guide to good Practice in Knowledge Management – Part 5: KM Terminology*, <ftp://cenftp1.cenorm.be/PUBLIC/CWAs/e-Europe/KM/CWA14924-05-2004-Mar.pdf>
66. *Fellbaum C.* WordNet: An Electronic Lexical Database (Language, Speech, and Communication). – The MIT Press, 1998. – 423 p.
67. *Fensel D.* Ontologies: a silver bullet for Knowledge Management and Electronic Commerce. – Berlin: Springer Verlag, 2001.
68. *Firestone J.M.* Enterprise Knowledge Portals: What They Are and What They Do // Knowledge and Innovation: Journal of the KMCI. – 2000. – V. 1. – No. 1. – October 15.
69. *Firestone J.M.* Enterprise information portals and Knowledge Management. – Butterworth-Heinemann, 2003.
70. *Fuller S.* Knowledge management foundations. – Butterworth-Heinemann, 2002.
71. *Gartner Group:* Knowledge Management Scenario: Trends and Directions for 1998 – 2003, 1999.
72. *Gennari J., Musen M.A., Fergerson R.W., et al.* The Evolution of Protege: An Environment for Knowledge-Based Systems Development, 2002.
73. *Gómez-Pérez A., Fernández-López M., Corcho O.* Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and Semantic Web (Advanced Information and Knowledge Processing). – Springer Verlag, 2003. – 415 p.
74. *Gruber T.R.* A Translational Approach to Portable Ontologies // Knowledge Acquisition. – 1993. – V. 5. – No. 2. – P. 199 – 220.
75. *Gruber T.R.* Towards principles for the design of ontologies used for knowledge sharing // International Journal of Human-Computer Studies. – 1995. – V. 43. – No. 5/6. – November-December. – P. 907 – 928.
76. *Guarino N.* Formal Ontology and Information Systems // Proceedings of International Conference of Formal Ontology and Information Systems (FOIS'98). N. Guarino (ed), Trenton, Italy, June 6 – 8, 1998. – Amsterdam: IOS Press, 1998. – P. 3 – 15.

77. *Haarslev V., Moller R.* Description of the RACER System and its Applications // Proceedings of International Workshop on Description Logics (DL-2001). Stanford, USA, 1 – 3 August 2001.
78. *Haarslev V., Moller R.* Racer: A Core Inference Engine for the Semantic Web // Proceedings of the 2nd International Workshop on Evaluation of Ontology-based Tools (EON2003), located at the 2nd International Semantic Web Conference ISWC 2003. Sanibel Island, Florida, USA, October 20, 2003. – P. 27 – 36.
79. *Handschuh S., Staab S. (eds.)*. Annotation for the Semantic Web. Volume 96 Frontiers in Artificial Intelligence and Applications. – IOS Press, 2003. – 240 p.
80. *Harvard Business Review on Knowledge Management.* – Harvard Business School Press, 2003.
81. *Hefke M.* State-of-the-Art Report on Research Projects and Market Situation/Deliverable D1.1, Project VISION, 2002, accessed at <http://www.aifb.uni-karlsruhe.de/apollo20080/fzi/vision/vision/docs/vision/docs/D11.zip>
82. *Heflin J., Hendler J., Luke S.* SHOE: A Blueprint for the Semantic Web // Fensel D., Hendler J., Lieberman H., and Wahlster W. (Eds.). Spinning the Semantic Web. – Cambridge, MA: MIT Press, 2003.
83. *Holsapple C.W., Joshi K.D.* Description and analysis of existing knowledge management frameworks // Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences, HICSS-32. Los Alamitos, CA, USA, 1999. IEEE Computer Society.
84. *Horrocks.* Optimising Tableaux Decision Procedures for Description Logics. PhD thesis. – University of Manchester, 1997.
85. *Horrocks I.* DAML+OIL: a Description Logic for the Semantic Web // Bull. of the IEEE Computer Society Technical Committee on Data Engineering. – 2002. – V. 25(1). – March. – P. 4 – 9.
86. *Horrocks I., Patel-Schneider P.* Reducing OWL entailment to description logic satisfiability // J. of Web Semantics. – 2004. – V. 1(4). – P. 345 – 357.
87. *Jos de Bruijn* Using Ontologies: Enabling Knowledge Sharing and Reuse on the Semantic Web, accessed at <http://deri.semanticweb.org/>.
88. *Gabel T., Sure Y., Voelker J.* KAON – Ontology Management Infrastructure (Deliverable D3.1.1.a, IST SEKT Project). – 2004. – 58 p.
89. *Kifer M., Lausen G.* F-logic: a higher-order language for reasoning about objects, inheritance, and scheme // Proceedings of the 1989 ACM SIGMOD international conference on Management of data table of contents. – Portland, Oregon, USA, 1989. – P. 134 – 146.
90. *Krogh G., Ichijo K., Nonaka I.* Enabling knowledge creation. – Oxford University Press, 2000.

91. *Lausen H., Stollberg M., et al.* Semantic Web Portals – State of the Art Survey. – 2004. – 52 p., <http://www.deri.at/publications/techpapers/documents/DERI-TR-2004-04-03.pdf>
92. *Levenstein V.I.* Binary Codes capable of correcting deletion, insertion and reversals // *Cybernetics and Control Theory*. – 1966. – V. 10. – No. 8. – P. 707 – 710.
93. *Maedche A., Staab S.* Measuring Similarity between Ontology // *Proceedings EKAW-2002*. – Madrid, 2002. – P. 251 – 263.
94. *Maedche A., Staab S.* KAON: The Karlsruhe Ontology and Semantic Web Meta Project // *Künstliche Intelligenz. Special Issue on Semantic Web*. – 2003. – No. 3. – P. 27 – 30.
95. *Maedche A., Staab S.* Measuring Similarity between Ontology // *Proceedings of the European Conference on Knowledge Acquisition and Management – EKAW-2002*. Madrid, Spain, October 1 – 4. – P. 251 – 263.
96. *Maedche A., Zacharias V.* Clustering Ontology – based Metadata in the Semantic Web // *Proceedings PKKD-2002*. – Helsinki, 2002. – P. 342 – 360.
97. *Maier R.* Knowledge management systems: Information and communication technologies for knowledge management. – Berlin Heidelberg: Springer Verlag, 2004. – 635 p.
98. *Marwick A.D.* Knowledge Management Technology // *IBM System Journal*. – 2001. – V. 40. – No. 4.
99. *McElroy M.W.* The new knowledge management: complexity, learning, and sustainable innovation. – Butterworth-Heinemann, 2003.
100. *McGuinness D.L.* Ontologies Come of Age / Fensel D., Hendler J. et al. (eds.), *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. – MIT Press, 2003.
101. *Mertins K., Heisig P., Vorbeck J. (eds.)* Knowledge Management: concepts and best practices (2nd ed.). – Berlin: Springer Verlag, 2003. – P. 383.
102. *Mizoguchi R., Kozaki K., Sano T., Kitamura Y.* Construction and Deployment of a Plant Ontology // *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management*. – 2000. – P. 113 – 128.
103. *Morrison D., Northam P., Rueckert M., Tabel L.* Inside the Lotus Discovery Server. IBM Corporation, Redbooks. – 2001. – P. 286.
104. *Nonaka I., Takeuchi H.* The knowledge-creating company. – New York: Oxford University Press, 1995. – P. 284.
105. *Oberle D., Spyns P.* The Knowledge Portal OntoWeb In Steffen Staab and Rudi Studer, *Handbook on Ontologies*, chapter IV. – Springer, 2004. – P. 499 – 517.

106. *O'Leary D.E.* Enterprise Knowledge Management // IEEE Computer. – 1998. – 3, March. – P. 54 – 61.
107. *OntoWeb*. D1.1. Technical Roadmap, document of a research project funded by the IST Programme of the Commission of the European Communities as project number IST-2000-29243, http://www.ontoweb.org/download/deliverables/D11_v1_0.pdf
108. *Oxford Advanced Learners Dictionary of Current English*. – Oxford, 1982.
109. *Piatetsky-Shapiro G., Frawley W.* Knowledge Discovery in Databases. – AAAI Press, Menlo Park, Calif., 1991.
110. *Patel-Schneider P. F., Horrocks I.* DLP and FaCT // N. V. Murray, editor, Automated Reasoning with Analytic Tableaux and Related Methods: International Conference Tableaux'99, number 1617 in Lecture Notes in Artificial Intelligence, pages 19-23. Springer Verlag, June 1999.
111. *Phifer G., Valdes R., Gootzit D., et al.* Magic Quadrant for Horizontal Portal Products, 2004. Gartner Group, <http://mediaproducts.gartner.com/reprints/ibm/120327.html>
112. *Pohs W., Pinder G., Dougherty C., White M.* The Lotus Knowledge Discovery System – Tools and experiences // IBM System Journal. – 2001. – V. 40. – No 4.
113. *Polanyi M.* Personal Knowledge. Corrected edition. – London: Routledge, 1958/1962. (Пер. на рус.: *Полани М.* Личностные знания. – М.: Прогресс, 1985.)
114. *Polanyi M.* The Tacit Dimension. – London: Routledge and Kegan Paul, 1996.
115. *Polikoff I.* Ontology Tool Support: Ontology Development Lifecycle and Tools. – 2003. – 14 p., http://www.topquadrant.com/documents/TQ1202_Ontolog Tool Survey.pdf
116. *Popper K.R.* Objective knowledge: An Evolutionary Approach. – London: Oxford University Press, 1972.
117. *Popper K.R., Eccles J.C.* The Self and Its Brain. – Berlin: Springer, 1977.
118. *Romer P.M.* Beyond the Knowledge Worker // Word Link. – 1995, January-February. – P. 55 – 60.
119. *Ruggles R.L.E.* Knowledge Management Tools (Resources for the Knowledge-Based Economy). – Boston: Butterworth-Heinemann, 1997.
120. *Senge P.M.* The fifth discipline. – New York: Currency Doubleday, 1990.
121. *Schein E.H.* Organizational Culture and Leadership. – San Francisco: Jossey-Bass, 1985.
122. *Sheth A., Bertram C., Avant D.* Managing Semantic Content for the Web // IEEE Internet Computing. – 2002. – V. 6(4). – P. 80 – 87.

123. *Spek R. and Spijkervet A.* Knowledge Management: Dealing Intelligently with Knowledge. – CIBIT, Utrecht, 1997.
124. *Staab S., Maedche A.* Knowledge portals – ontologies at work // AI Magazine. – V. 21(2). – Summer 2001.
125. *Staab S., Studer R., Sure Y., Volz R.* SEAL – a SEmantic portAL with content management functionality // Gaining Insight from Research Information. CRIS 2002 – Proceedings of the 6th Conference on Current Research Information Systems. August 29 – 31, 2002, Kassel, Germany.
126. *Staab S., Studer R. (eds.)* Handbook on Ontologies (International Handbooks on Information Systems). – Springer Verlag, 2004. – 660 p.
127. *Stewart T.A.* Intellectual Capital: The New Wealth of Organizations. Currency. – 1998. – 320 p.
128. *Sure Y., Angele J., Staab S.* OntoEdit: Multifaceted Inferencing for Ontology Engineering // Journal on Data Semantics. – Springer, 2003. – V. 1(1). – P. 128 – 152.
129. *Sure Y., Angele J., Staab S.* OntoEdit: Multifaceted Inferencing for Ontology Engineering // Journal on Data Semantics, LNCS 2800. – Springer, 2003. – P. 128 – 152.
130. *Sveiby K.E.* The New Organizational Wealth: Managing & Measuring Knowledge-Based Assets. – Berrett-Koehler Publishers, 1997.
131. *Tarski A.* Logic, Semantics, Metamathematics. – Oxford University Press, 1956.
132. *The American Heritage Dictionary of the English Language.* – 1992.
133. *The Description Logic Handbook / Ed. by Baader F., Calvanese D., McGuinness D., Nardi D., Patel-Schneider P.* – Cambridge University Press, 2003. – 574 p.
134. *The Dublin Core Metadata Initiative, "Dublin Core Metadata Initiative – Home Page,"* <http://purl.org/dc/index.htm>.
135. *Tiwana A.* The knowledge management toolkit: practical techniques for building a knowledge management system. Prentice Hall PTR. – 2000.
136. *TopQuadrant Technology Briefing, Semantic Technology, Version 1.1,* September 2003, (http://www.topquadrant.com/documents/TQ04_Semantic_Technology_Briefing.pdf)
137. *TOVE Ontology Project,* <http://www.eil.utoronto.ca/enterprise-modelling/tove/index.html>
138. *Uschold M., King M., Moralee S., Zorgios, Y.* The Enterprise Ontology // The Knowledge Engineering Review / Eds. M. Uschold and A. Tate. – 1998. – V. 13.
139. *Vasconcelos J., Kimble C., Gouveia F., Kudenko D.* Knowledge Management, competence management, group memory, corporate memory, distrib-

- uted knowledge // Proceedings First European Conference on Knowledge Management. – 2000. – P. 91 – 99.
140. *Vasconcelos J., Kimble C., Gouveia F.R.* A design for a group memory system using ontologies // Proceedings of 5th UKAIS Conference. McGraw Hill. Apr. 2000. – P. 246 – 255.
 141. *Volz R., Oberle D., Staab S., Motik B.* KAON SERVER – A Semantic Web Management System // Proceedings of the WWW-2003 Alternate Track on Practice and Experience. Budapest, Hungary, May 20 – 24, 2003. – Published at <http://www2003.org>.
 142. W3C, "Extensible Markup Language (XML)," <http://www.w3.org/XML>.
 143. W3C, "Resource Description Framework (RDF)," <http://www.w3c.org/RDF>.
 144. W3C, "RDF/XML Syntax Specification (Revised)," <http://www.w3.org/TR/rdf-syntax-grammar/>
 145. W3C, "RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation 10 February 2004," <http://www.w3.org/TR/rdf-schema/>
 146. W3C, "The Semantic Web," <http://www.w3.org/2001/sw>.
 147. W3C, "OWL Web Ontology Language Overview, W3C Recommendation 10 February 2004," <http://www.w3.org/TR/owl-features/>
 148. W3C, "XML Schema Part 0: Primer Second Edition," <http://www.w3.org/TR/xmlschema-0/>
 149. W3C, "XML Schema Part 2: Datatypes Second Edition," <http://www.w3.org/TR/xmlschema-2/>
 150. *Wiig K.M.* Knowledge Management Foundations: Thinking About Thinking – How People and Organizations Create, Represent, and Use Knowledge. – Arlington, TX: Schema Press, 1993.
 151. *Wiig K.M.* Knowledge Management Methods. – Schema Press, 1995. – P. 489.
 152. *Wiig K.M.* Introducing knowledge management into the enterprise // Knowledge management handbook / Ed. by J. Liebowitz. – NY: CRC Press, 1999. – P. 3.1 – 3.41.
 153. *Wooldridge M.* An Introduction to Multiagent Systems. – John Wiley & Sons Ltd., 2002. – P. 368.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
Глава 1. Знание как понятие	8
1.1. Данные	9
1.2. Информация.....	10
1.3. Знание.....	11
1.4. Классификация знаний	15
1.4.1. Неявные знания	16
1.4.2. Явные знания	17
1.4.3. Связь между явными и неявными знаниями.....	19
1.5. Свойства знаний	22
1.6. Знания как движущая сила новой экономики.....	23
Глава 2. Жизненный цикл знаний.....	28
2.1. Основные этапы жизненного цикла знаний.....	29
2.2. Базовые виды деятельности, связанные со знанием	32
2.2.1. Идентификация (выявление) знаний	33
2.2.2. Создание новых знаний	33
2.2.3. Хранение знаний.....	34
2.2.4. Распространение знаний.....	35
2.2.5. Использование знаний	36
2.3. Рынок знаний.....	37
Глава 3. Управление процессом работы со знаниями.....	41
3.1. Подходы к управлению знаниями.....	41
3.2. Модель процесса создания нового знания организацией	46
3.3. Команда, создающая знания.....	51
3.4. Сообщества по интересам	53
3.5. Эксперты, экспертиза, консультации	57
3.6. Система обучения персонала	61
3.7. Корпоративная культура	65

Глава 4. Онтологические модели представления знаний	75
4.1. Модель интеллектуального пространства	77
4.2. Онтологии предметных областей	78
4.3. Формальная модель онтологии	82
4.4. Типы онтологий	85
4.5. Метаописания	92
4.6. Измерение близости объектов в интеллектуальном пространстве	95
4.7. Дескриптивные логики как формальные модели онтологий	101
4.7.1. Базовые формализмы ДЛ	102
4.7.2. Дескриптивный язык	104
4.7.3. Базовый дескриптивный язык AL	106
4.7.4. Терминологии	107
4.7.5. Обозначение дескриптивных логик	108
4.7.6. Логический вывод	109
4.7.7. Задачи логического вывода	110
Глава 5. Языки описания знаний	112
5.1. Требования к онтологическим языкам	112
5.2. Язык описания документов XML	115
5.2.1. Document Type Definitions (DTD)	122
5.2.2. Язык XML Schema	123
5.3. Язык описания метаинформации RDF	124
5.4. Язык описания онтологий RDFS	130
5.5. Онтологический язык OWL	134
5.5.1. Синтаксис языка	135
5.5.2. Элементы, определяющие классы	136
5.5.3. Элементы, определяющие свойства (отношения)	137
5.5.4. Ограничение свойств	138
5.5.5. Специальные свойства	141
5.5.6. Булевы комбинации	142
5.5.7. Перечисления (Enumerations)	144
5.5.8. Экземпляры (Instances)	144
5.5.9. Типы данных (Datatypes)	146
5.5.10. Разновидности языка OWL	147
5.5.11. Связь языка OWL с дескриптивной логикой	149

Глава 6. Инструментальные средства для работы со знаниями	152
6.1. Эволюция развития инструментальных средств	153
6.2. Программное обеспечение информационных систем	155
6.3. Программные инструменты для построения и работы с онтологиями.....	160
6.4. Программные инструменты для обеспечения логического вывода	179
6.4.1. Система FaCT (Fast Classification of Terminologies)	179
6.4.2. Система логического вывода RACER.....	181
6.4.3. Программная среда для обеспечения логического вывода.....	182
Глава 7. Системы управления знаниями	189
7.1. Назначение и архитектура систем управления знаниями.....	189
7.2. Корпоративная память	195
7.3. Подсистема поиска знаний	202
7.3.1. Инструментальные средства поиска	204
7.3.2. Средства интеллектуального поиска.....	206
7.3.3. Визуальные модели поиска.....	208
7.4. IT-среда для совместной интеллектуальной деятельности	211
7.5. Web-порталы.....	223
7.6. Состав и последовательность этапов по созданию СУЗ.....	235
7.7. Проблематика КТ и KMS в зарубежных исследованиях и разработках	240
ЗАКЛЮЧЕНИЕ	246
СПИСОК ЛИТЕРАТУРЫ.....	248

ДЛЯ ЗАМЕТОК

*Анатолий Федорович Тузовский
Сергей Владимирович Чириков
Владимир Захарович Ямпольский*

**СИСТЕМЫ УПРАВЛЕНИЯ ЗНАНИЯМИ
(методы и технологии)**

Под общей редакцией
В.З. Ямпольского

Редактор *Н.И. Шидловская*
Верстка *Д.В. Фортес*

К-ОКП ОК-005-93, код продукции 953380

Изд. лиц. ИД № 04000 от 12.02.2001. Подписано к печати 20.03.2005.
Формат 60 × 84 ¹/₁₆. Бумага офсетная. Печать офсетная. Гарнитура «Таймс».
Усл. п. л. 15,11. Уч.-изд. л. 16,93. Тираж 500 экз. Заказ № 4.

ООО «Издательство научно-технической литературы»
634050, Томск, пр. Ленина, 34а, тел. (382-2) 53-33-35

Отпечатано в типографии ЗАО «М-Принт», г. Томск, ул. Пролетарская, 38/1