

Принципы разработки распределенных систем сбора информации на основе онтологий¹

Аннотация. В работе исследуются вопросы автоматизированного построения распределенных систем сбора информации с использованием формализованного описания предметной области. Выполнен обзор методик создания онтологических описаний. Предложена оригинальная методика построения онтологии предметной области, на основе которой формируются объектная, реляционная и иерархическая программные модели. Описана технология отделения функциональности системы сбора данных от предметного описания.

Ключевые слова: онтология, система сбора информации, owl, xml.

Введение

Исследование направлено на решение проблемы автоматизированного построения распределенных систем сбора слабоструктурированной информации, основанное на формализованном описании предметной области (ПрО). Распределенные системы сбора информации характеризуются иерархической структурой узлов сбора, на которых функционируют программные комплексы, выполняющие задачи извлечения данных, их верификации, хранения, отображения, преобразования, приема и передачи другим узлам. В перечисленных задачах можно выделить два аспекта – функциональный и предметный. Например, для задачи хранения данных функциональный аспект выражается в процедурах управления данными, их размещения во внешней памяти. Предметный аспект состоит в определении структур и связей между элементами данных, соответствующими заданной ПрО. Для задачи хранения данных смешивание этих двух аспектов считается плохой практикой. Вместо этого используют, например, СУБД как функциональную основу, которая не зависит от конкретной структуры данных. При этом структура данных является некоторой проекцией ПрО,

выражая предметный аспект. Расширение этого подхода для остальных задач сбора информации позволит не только создать формальную методику построения систем, но и сократить время на их разработку и сопровождение, особенно с учетом факторов сложности и изменчивости ПрО.

При использовании традиционных подходов к построению информационных систем структура ПрО оказывается неотъемлемой частью системы. Поэтому ее выделение, например, для модификации имеющейся системы или для повторного использования в других системах становится трудной задачей [1]. Формализованное описание ПрО представляет собой набор утверждений, выраженных с помощью специального языка, и часто называется онтологией. В данной работе под термином «онтология» мы будем подразумевать «спецификацию концептуализации» [13].

Данная работа выполнена в рамках подхода к построению информационных систем на основе онтологий (Ontology-Driven Information Systems Engineering [14]), при котором в основу системы ложится онтология предметной области. Хотя исследования в этой области начались совсем недавно, можно утверждать, что онтологии обладают достаточным потенциалом для

¹ Работа поддержана РФФИ проект № 12-07-31018.

успешного использования на всех этапах разработки программных систем. При этом ключевым этапом является создание онтологии, который требует экспертных знаний и плохо поддается формализации. Также важны вопросы автоматизации технологического процесса создания системы за счет применения формального онтологического описания.

Таким образом, целью этой работы является разработка принципов создания распределенных систем сбора информации на основе онтологий. Для достижения поставленной цели необходимо решение следующих задач:

- разработка методологии построения онтологии, включающей в себя обзор методов онтологического инжиниринга и применение полученной методики;
- построение на основе онтологии необходимых программных моделей, выделение ограничений и требований к исходной онтологии (предметный аспект);
- разработка функциональных модулей распределенной системы сбора информации, управляющих программными моделями (функциональный аспект).

Полученные результаты были опробованы на примере задачи ведения Государственного лесного реестра, решаемой в рамках научно-исследовательских работ по заказу Рослесхоза.

1. Методика построения онтологий

Подходы к построению онтологий можно условно разделить на классический и современный. Под классическим подходом понимается построение онтологий в соответствии со стандартом онтологического анализа IDEF5 [3]. Современный подход к онтологическому анализу подразумевает, в основном, построение web-онтологий – онтологий в контексте семантической паутины (web 3.0) и сводится к разработкам консорциума W3C (The World Wide Web Consortium) [4]. Одной из таких разработок является Web Ontology Language (OWL) – язык для представления онтологий и связанной информации в виде семантической сети [2].

Среди методологий и методов построения онтологий можно выделить следующие:

- подход Сус [5];
- метод Усколда и Кинга [6];
- методология Грюнингера и Фокса [7];

- метод KACTUS [11];
- методология METHONTOLOGY [8];
- метод SENSUS [10];
- методология On-To-Knowledge [9].

Рассмотренные методики имеют ряд общих характеристик – выбор основных понятий, сущностей и связей между ними, а затем их кодирование посредством различных инструментов. Некоторые методологии содержат этап, предшествующий выборке понятий – определение специфики предметной области, целей и сфер применения построенной онтологии. Немаловажной частью методик являются алгоритмы и инструкции, по которым происходит выделение второстепенных сущностей и интеграции их в уже имеющуюся онтологию, а также метод построения основания будущей онтологии, используемой для последующего наращивания. Заключительным этапом является определение компетентности онтологии, проверка соответствия поставленным требованиям и решаемым задачам, а также сфер применения и использование построенной онтологии. Рассмотренные методики можно разделить по способам построения на несколько типов:

- переиспользование уже имеющихся онтологий, т.е. модификация и расширение некоторой онтологии взятой за основу (метод KACTUS);
- слияние нескольких онтологий и источников знаний в один объект (метод SENSUS);
- построение онтологии исходя из бизнес-процессов предметной области, т. е. ориентация на предметную область моделирования бизнес-процессов (метод Усколда и Кинга, методология Грюнингера и Фокса);
- построение онтологий исходя из знаний о сущностях и их свойствах (метод Сус).

Построение онтологии отраслевой системы сбора данных предполагает использования как знаний об объектах предметной области, так и знания о бизнес-процессах рассматриваемой сферы деятельности (тип 3). Метод Усколда и Кинга включает в себя этап фиксации онтологии. Узким моментом является согласование всех знаний, полученных в процессе фиксации разрабатываемой онтологии, а также этап построения оценки полученной онтологии. Узким моментом методологии Грюнингера и Фокса является то, что процесс построения онтологии ведется от неформальных вопросов проверки

компетенции онтологии. Т.е. из множества неформальных вопросов проверки компетенции онтологии выделяется множество терминов, которые должны быть базисом.

После анализа рассмотренных методик была сформулирована собственная методика, которая учитывает специфику предметной области, определяет ресурсы и аспекты. Их использование позволяет определить ключевые понятия и сущности, а также дополнительные знания для наращивания и расширения скелета или построенного прототипа онтологии. Методика построения онтологии включает в себя следующие этапы:

1. Обследование предметной области и выявление бизнес-процессов и документов.
2. Определение каркаса онтологии.
3. Спецификация бизнес-процессов и определение форм документов.
4. Построение блоков онтологии и их компоновка.
5. Наполнение справочных массивов данных.

Задачами обследования ПрО являются описание текущего состояния организаций и пользователей, осуществляющих деятельность в рамках ПрО, сбор используемых документов, нормативно-справочной информации, а также информации о существующих информационных системах и программных продуктах предметными пользователями. Выявление бизнес-процессов и документов осуществляется на основе полученных в ходе обследования документов и сведениях. Для каждого бизнес-процесса на данном этапе определяются информационные потоки, формы используемых документов, задействованные роли.

Определение каркаса онтологии состоит в выявлении ее ключевых сущностей, вокруг которых на последующих этапах будет сформирована остальная ее часть. Ключевые сущности характеризуются тем, что участвуют в большинстве полученных бизнес-процессов, а удаление ключевой сущности приводит к невозможности их реализовать.

Спецификация бизнес-процессов и определение форм документов являются результатом анализа полученных ранее материалов, их обобщения и формализации. Специфицирование бизнес-процессов представляет собой процесс построения формальной модели организационных и информационных потоков ПрО. Для построения онтологии на данном этапе сущест-

венным является точное определение форм документов, которые участвуют в бизнес-процессах, и правила их заполнения. Для описания бизнес-процессов рекомендуется использовать общепринятые методологии и нотации, например, IDEF, EPC, BPMN.

Построение блоков онтологии и их компоновка являются основным этапом и выполняются на основании спецификаций бизнес-процессов, включающих формы используемых документов и нормативно-справочной информации. Как правило, каждый блок онтологии строится вокруг одной ключевой сущности. При необходимости возможно разбиение одного блока на несколько более мелких. Отличительной особенностью каждого блока является его информационная замкнутость вокруг ключевой сущности и относительно небольшое количество связей с другими блоками, которые формируются в результате компоновки блоков в единое целое. Рекомендуется отдельным блоком выделить сущности, которые описывают нормативно-справочную информацию; это позволит определить классификаторы ПрО.

Наполнение справочных массивов данных осуществляется по нормативно-справочным документам, в которых описаны регламенты, нормативные значения показателей и допустимые значения полей форм документов.

Корректность онтологии, построенной с помощью этой методики, определяется: полнотой обследования ПрО, корректностью формального описания бизнес-процессов и их взаимной согласованностью со структурой и элементами онтологии. Таким образом, формально проверить правильность онтологии весьма затруднительно, что отражает творческую суть процесса ее создания. Однако в рамках этой работы мы сможем установить, насколько точно созданная с помощью нашей онтологии система сбора информации выполняет свои функции, и считать это критерием ее корректности.

2. Построение онтологии лесного хозяйства

Изучение предметной области выполняется путем интервьюирования работников лесного хозяйства, параллельно выявляя основные процессы и документы, используемые в обращении, структуру организации управления и про-

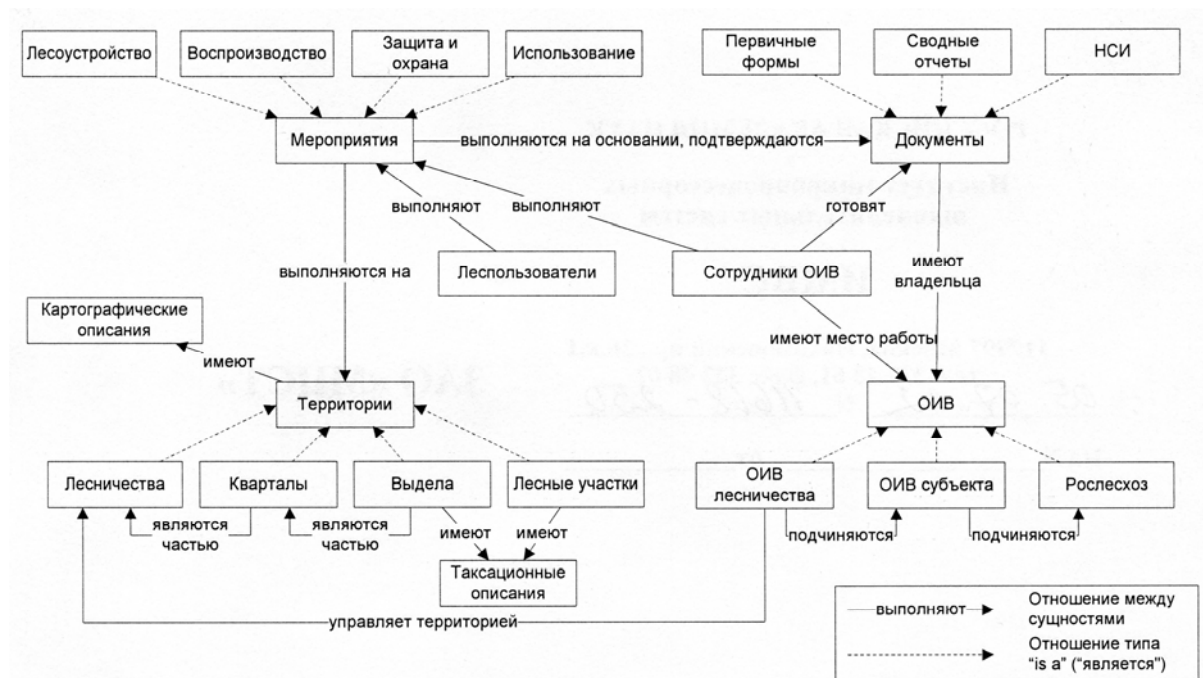


Рис. 1. Концептуальный каркас онтологии лесного хозяйства

чую информацию, востребованную в будущем для построения онтологии.

После проведения обследования была систематизирована информация об основных процессах лесного хозяйства, его структуре, информационных потоках и маршрутах следования документов. Была выделена нормативно-справочная информация, которая регламентирует лесохозяйственную деятельность. Были построены модели бизнес-процессов в формате IDEF0, выявленные по итогам обследования:

- ведение государственного лесного реестра;
- воспроизводство лесов;
- выявление и устранение очагов вредителей и болезней леса;
- обнаружение и ликвидация пожаров;
- осуществление государственного лесного контроля;
- оформление и контроль выполнения договоров купли-продажи лесных насаждений;
- оформление и контроль выполнения договоров аренды;
- оформление и контроль выполнения государственных контрактов на выполнение работ по охране, защите и воспроизводству лесов;
- администрирование платежей;
- управление бюджетным финансированием.

Для построения каркаса онтологии следует использовать определение Государственного лесного реестра (ГЛР) – систематизированного

свода документированной информации о лесах, об их использовании, охране, защите, воспроизводстве, о лесничествах и лесопарках [12]. Из этого определения следует, что текущее состояние лесного фонда описывается материалами лесоустройства и подтверждающими документами, полученными в ходе мероприятий по использованию, воспроизведению, охране и защите лесов. При этом описание мероприятий неразрывно связано с описанием следующих сущностей: лесопользователи (выполняют лесохозяйственные работы), документы (фиксируют факты выполнения мероприятий), органы исполнительной власти (ОИВ) и их сотрудники (контроль за надлежащим выполнением мероприятий), территории (данные о различных пространственных объектах: лесничествах, кварталах, выделах и лесных участках). Рассмотрение описаний бизнес-процессов показало, что перечисленные сущности являются ключевыми и из них может быть сформирован каркас онтологии лесного хозяйства (Рис. 1).

На основе построенного каркаса были разработаны следующие блоки онтологии.

1. Блок описания лесного фонда, содержащий таксационные описания и другую информацию из материалов лесоустройства, а также изменения, которые в них произошли с момента его проведения в результате лесохозяйственной деятельности.

2. Блок документов. Сущность «Документ» является центральной, поскольку представляет собой основную единицу сбора данных. Выделены три основных класса документов: первичные, сводные и нормативно-справочные. Первичные документы формируются, как правило, на уровне лесничества и содержат первичные данные, подлежащие передаче на верхние уровни и учету в ГЛР. Сводные документы формируются путем фильтрации и агрегации первичных данных. В отличие от первичных, этот класс документов не требует детализированного описания, поскольку функция формирования отчетов находится за пределами онтологии. Нормативно-справочные документы определяют регламенты и нормативы выполнения различных работ. В онтологии нормативно-справочная информация представляется в отдельном блоке преимущественно в виде справочных сущностей.

3. Блок мероприятий содержит описания всей лесохозяйственной деятельности. Каждое мероприятие отражается в одном или нескольких документах, которые фиксируют сведения о мероприятии (кто, что, где и когда выполнял) и соответствующие изменения в лесном фонде.

4. Блок организационной структуры описывает организационную модель органов исполнительной власти в области лесных отношений, а также лесопользователей.

5. Блок пространственных данных. Картографическая информация формируется в результате проведения лесоустройства или иных специальных работ и позволяет визуализировать информацию с привязкой к местности и указанием административных границ и других зон. Картографическая информация делится на тематические слои, как правило, не зависящие друг от друга. Часть слоев не изменяется со временем или изменяется пренебрежимо мало по сравнению с жизненным циклом системы – это границы регионов, лесничеств, кварталов. Другие слои, например, слой лесохозяйственных мероприятий или слой границ выделов, изменяются по мере поступления в систему новых данных.

6. Блок справочников содержит элементы НСИ, которые используются при формировании первичных документов.

Компоновка онтологии представляет собой консолидацию блоков онтологии в единое целое и проверка ее согласованности и корректности. Логическая корректность построенной онтологии была проверена следующими методами:

- Проверка структуры первичных документов. Для любого первичного документа, участвующего в процессах ведения ГЛР, должна быть возможность установить взаимно однозначное соответствие между полями первичного документа и структурой его сущности в онтологии, включая связанные сущности.

- Проверка справочников. Справочным полям первичных документов должны однозначно соответствовать сущности из блока «Справочники».

- Проверка верифицируемости данных. Для любого элемента данных должна быть возможность определить следующую информацию: документ, на основании которого были внесены данные; дата регистрации данных в системе; авторы и организационная принадлежность документа; пространственная привязка данных; временной интервал актуальности данных.

- Бритва Оккама. В соответствии с принципом Бритвы Оккама, информационные сущности, атрибуты и связи присутствуют в онтологии в единственном числе.

3. Построение программных моделей

Для решения задач сбора информации необходимы, по крайней мере, три программные модели: объектная (обработка), реляционная (хранение) и иерархическая XML-модель (обмен). Исходное онтологическое описание основано на дескриптивной логике и является достаточно полным, поэтому может играть роль абстрактной метамодели, на основе которой будут строиться остальные модели. Кроме того, они имеют разную природу, и для функционирования системы необходимо также описать способы преобразования данных из одной модели в другую.

Прежде всего, необходимость наложить ряд ограничений на онтологическое описание (Табл. 1) для избегания ситуаций с неоднозначностью построения других моделей (например, в случае с множественным наследованием), а также на все возможности, присутствующие в онтологии, но не отраженные в других моделях.

Ограничения можно разделить на критичные («Ошибка»), невыполнение которых ведет к невозможности построения моделей, и не критичные («Предупреждение»), которые могут быть исправлены автоматически. Для каждого ограничения можно указать рекомендуемое решение при его нарушении.

Табл.1. Ограничения на исходную онтологию

№ п.п	Ограничение	Уровень	Рекомендуемое решение
Классы			
1	Класс не может быть построен путем явного перечисления входящих в него объектов	Ошибка	Класс игнорируется
2	Класс не может быть построен как объединение, пересечение или дополнение других (не анонимных) классов	Ошибка	Класс игнорируется
3	Класс не может наследоваться более чем от одного (не анонимного) класса	Ошибка	Класс игнорируется
4	Родительский класс должен удовлетворять всем ограничениям на классы	Ошибка	Класс игнорируется
Свойства			
5	Свойство должно быть приписано как минимум к одному классу	Ошибка	Свойство игнорируется
6	Класс, к которому приписано свойство, должен удовлетворять всем ограничениям на классы	Предупр.	Свойство не приписывается к неудовлетворяющему ограничениям классу
7	Одно свойство не может быть приписано и к родительскому классу, и к дочернему	Предупр.	Свойство приписывается только к родительскому классу
8	В качестве значения свойства могут выступать лишь не анонимные классы	Ошибка	Свойство игнорируется
9	В качестве значения свойства могут выступать лишь классы, удовлетворяющие всем ограничениям на классы	Ошибка	Свойство игнорируется
10	В качестве значений свойства могут выступать объекты только одного класса.	Ошибка	Свойство игнорируется
11	Должен быть указан тип значения свойства	Предупр.	Свойству приписывается строковый тип.
12	Если для свойства существует обратное, то оно не может быть приписано более чем к одному классу	Предупр.	Наличие обратного свойства игнорируется
13	Обратное свойство должно удовлетворять ограничениям на свойства (уровня "Ошибка")	Предупр.	Наличие обратного свойства игнорируется
14	Не допускается наложение на свойство условий транзитивности и симметричности	Предупр.	Симметричность и транзитивность игнорируются.
15	Не допускается ограничение на максимальное количество значений свойства, отличное от 1	Предупр.	Ограничение на свойство игнорируется
16	Не допускается ограничение на точное количество значений свойства, отличное от 1	Предупр.	Ограничение на свойство игнорируется
17	Не допускаются ограничения на свойства, отличные от указанных в пунктах 15 и 16	Предупр.	Ограничения на свойство игнорируются

В разработанной системе (Рис. 2) задаче анализа моделей соответствует компонент "Анализатор онтологии", в ходе работы которого исходная онтология проверяется на предмет соответствия определенным ограничениям, и строится корректная, с точки зрения указанных ограничений, семантическая модель.

На основе семантической модели строятся остальные предметные модели системы, используя методы порождающего программирования. Для каждого вида предметной модели реализован набор правил, преобразующих элементы онтологии в соответствующие конструкции языка реализации модели. Например, онтологические сущности преобразуются в объекты (объектная модель), таблицы (реляционная модель) и узлы DOM-модели (XML-модель). В целом, составление таких правил носит эмпирический характер и соответствует

общепринятыми практиками проектирования программного обеспечения. Однако существенным вопросом является взаимная согласованность полученных предметных моделей, поскольку в процессе работы системы они будут активно взаимодействовать между собой.

Принципы взаимодействия объектно-ориентированной и реляционной модели данных рассматриваются в подходе ORM [15] (Object-relational mapping – Объектно-реляционное отображение). Однако, в силу специфики, детализация этих принципов может осуществляться только в рамках конкретной технологической платформы. Авторами исследования была выбрана технологическая платформа Java, а в качестве средства ORM – библиотека Hibernate. Ее использование позволило автоматически аннотировать java-объекты описаниями связанных реляционных конструкций,

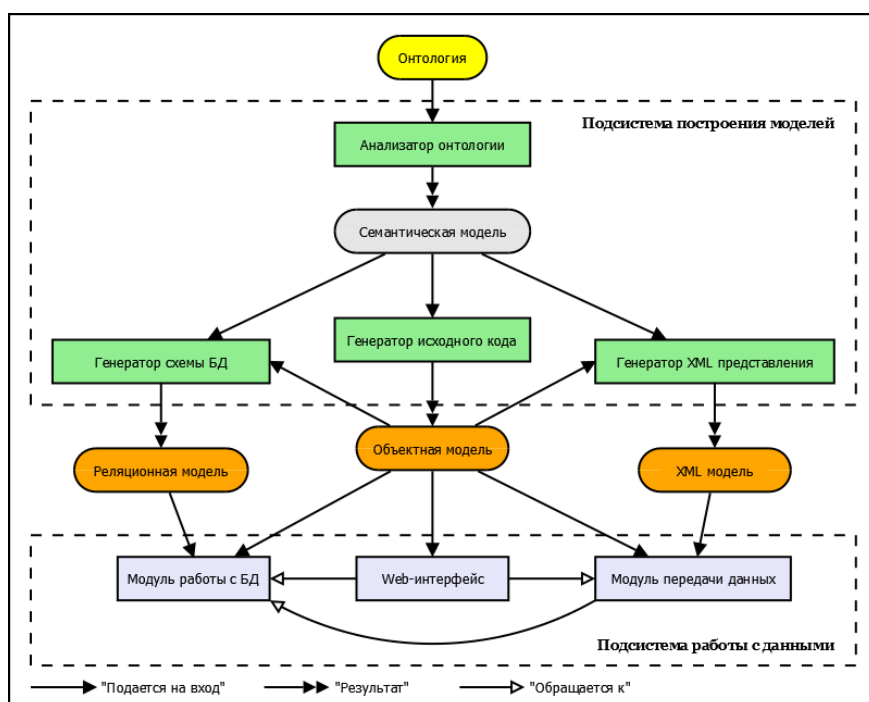


Рис. 2. Общий принцип работы технологии

верхние прямоугольники – компоненты предметного анализа, овалы – предметные модели, нижние прямоугольники – функциональные компоненты

на основе которых была сгенерирована база данных третьей нормальной формы и программный модуль взаимодействия с ней.

Основная задача иерархической модели заключается в представлении данных в виде последовательности байтов в читаемом виде для передачи между узлами системы. Для этой задачи в качестве основы был выбран язык разметки XML, а для его генерации – технология JAXB (Java Architecture for XML Binding). Это технология сводит построение иерархической модели к выделению в каждом предметном классе списка полей и определению правил их кодирования. Опираясь на объектную модель, JAXB обеспечивает кодирование (для передачи) и декодирование (при приеме) объектов предметной области в соответствии с заданной XML схемой.

4. Прототип распределенной системы сбора данных

Одним из применений построенной онтологии предметной области является разработка на ее основе распределенной системы сбора данных, которая содержит в себе решение следующих задач (Рис.2):

- построение объектной модели предметной области;
- построение реляционной модели предметной области и осуществление связи с объектной моделью;
- построение иерархической XML- модели предметной области;
- разработка модуля взаимодействия с базой данных, осуществляющего операции добавления, поиска, обновления и удаления объектов;
- разработка web-интерфейса для ввода и отображения данных предметной области;
- разработка модуля приема-передачи данных предметной области.

Каждый компонент системы отвечает за решение отдельной задачи, при этом можно выделить две подсистемы: подсистема построения моделей (предметные аспекты) и подсистема работы с данными (функциональные аспекты). Разработанные компоненты позволяют выполнять процедуры ввода, хранения, приема и передачи данных (Рис. 3).

Модуль работы с БД фактически представляет собой функцию двустороннего преобразования объектной и реляционной модели на основе ORM-подхода, реализованного в библиотеке Hibernate. Для проектирования двух

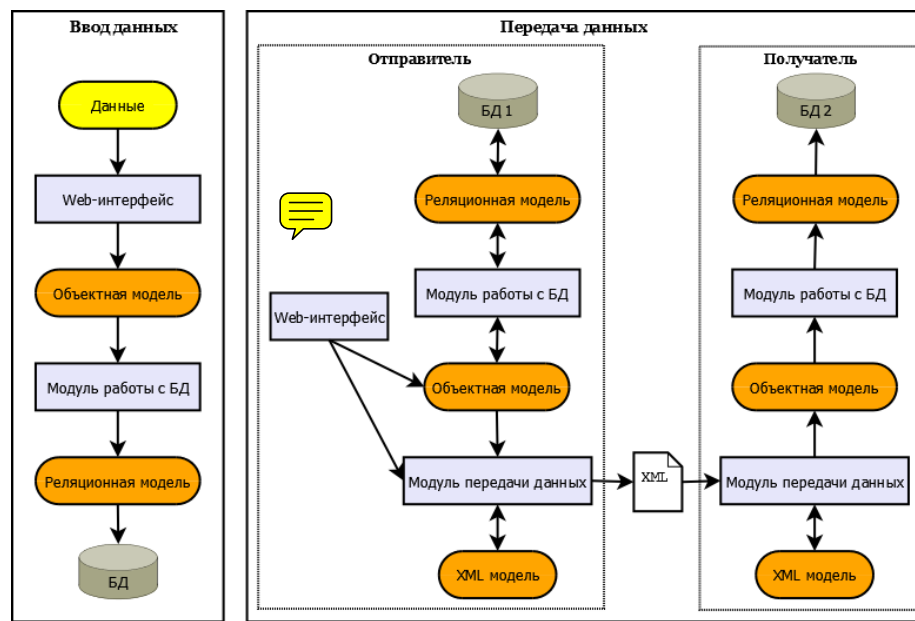


Рис. 3. Схема взаимодействия компонентов системы при вводе и передаче данных

других функциональных модулей потребовалось введение дополнительного онтологического метакласса «Документ», наследники которого являются единицами информационного обмена и группируют в себе остальные предметные сущности. Например, класс документов «Лесная декларация» содержит сущности «Лесопользователь», «Лесной участок», «Отчетный период» и «Лесохозяйственные мероприятия». Модуль пользовательского интерфейса использует классы документов для генерации форм ввода информации и ее последующего преобразования в объектную модель. Модуль передачи данных использует классы документов в качестве корня XML-документов, содержащих соответствующие предметные данные.

В рамках данной работы был реализован прототип распределенной системы сбора данных, предоставляющий следующие возможности:

- просмотр списка доступных классов документов;
- просмотр списка документов заданного класса;
- создание, просмотр, редактирование и удаление документа;
- отправка выбранного документа в другой узел системы.

Отличительной особенностью созданного прототипа является возможность работать с разными предметными онтологиями без изменения исходного кода. Таким образом, проце-

дура создания прототипа системы сбора данных может быть сведена к разработке онтологии предметной области.

Разработанная система сбора данных может быть также использована для верификации онтологии и программных моделей. Верификация осуществляется путем эмуляции документооборота по схеме, изображенной на Рис. 3. Возьмем произвольный документ и введем все данные в форму ввода, предназначенную для выбранного типа документа. Введенные данные будут преобразованы в объектную и реляционную модели, после чего сохранятся в БД. Затем выполняется процесс приема-передачи данных, в течение которого данные из реляционной модели через объектную модель преобразуются в иерархическую, и наоборот. Эти действия повторяются для репрезентативного множества документов из разных классов. Если в каждом случае исходный и конечный документы будут идентичны, то можно утверждать, что онтология и соответствующие программные модели корректны.

Заключение

В статье рассмотрены методы и технологии для автоматизированного создания распределенных систем сбора информации на основе онтологического описания предметной области. Выполнен обзор методов онтологического

инжиниринга и сформулирован обобщенный подход для построения онтологии. Проведено исследование возможности генерации программных моделей на основе онтологии (а именно: объектной, реляционной и иерархической), сформулированы ограничения на исходную онтологию.

В качестве подтверждения работоспособности данного подхода были реализованы компоненты построения необходимых моделей, которые вместе с онтологией образуют прототип распределенной системы сбора данных. Сформулированная методика построения онтологического описания была применена к конкретной предметной области «Лесное хозяйство».

Литература

1. Yildiz B., Miksch S. Ontology-Driven Information Systems: Challenges and Requirements – Proceedings of the International Conference on Semantic Web and Digital Libraries, 2007. – с.11.
2. OWL Web Ontology Language. Overview. – W3C Recommendation. – February 10. – 2004. – <http://www.w3.org/TR/owl-features/>.
3. Information Integration for Concurrent Engineering (IICE). IDEF5 Method Report. – Knowledge Based Systems, Inc., 1408 University Drive East College Station, Texas, USA. – September 21. – 1994.
4. Андреева Н. В., Любимов А. В. Выбор методов и средств онтологического анализа стандартов информационной безопасности // VI Всероссийская межвузовская конференция молодых ученых. — СПб: СПбГУ ИТМО, 2009. Вып. 6. - С. 29-33.
5. Lenat D., Guha R.V. Building Large Knowledge Based Systems: Representation and Inference in the Cyc Project, Addison-Wesley, 1989.
6. Uschold M., King M., Moralee S., Zorgios Y. The Enterprise Ontology The Knowledge Engineering Review. Vol. 13, Special Issue on Putting Ontologies to Use / M. Uschold, A. Tate (eds.), 1998.
7. TOVE Ontology Project. <http://www.eil.utoronto.ca/enterprise-modelling/tove/>.
8. Fernandez M., Gomez-Perez A., Juristo N. METHONTOLOGY: From Ontological Art Towards Ontological Engineering. AAAI Technical Report SS-97-06, 2006.
9. Staab S., Schnurr H.P., Studer R., Sure Y. Knowledge processes and ontologies. // IEEE Intelligent Systems - 2001. - V. 16. - № 1. - P. 26-34.
10. Swartout B., Ramesh P., Knight K., Russ T. Toward Distributed Use of Large-Scale Ontologies. Symposium on Ontological Engineering of AAAI. Stanford (California). Mars, 1997.
11. Schreibe, G., Wielinga B., Jansweijer W. The KACTUS View on the 'O' World. In Proceedings of the National Dutch AI Conference. NAIC'95. 1995.
12. Лесной кодекс Российской Федерации: Федеральный закон от 4 декабря 2006 г. №220-ФЗ.
13. Gruber T. R. Toward principles for the design of ontologies used for knowledge sharing. Presented at the Padua workshop on Formal Ontology, March 1993, published in International Journal of Human-Computer Studies, Vol. 43, Issues 4-5, November 1995, pp. 907-928.
14. Guarino N. Formal Ontology and Information Systems – Proceedings of the First International Conference on Formal Ontologies in Information Systems (FOIS), 1998. – с.3-15.
15. Scott W. Ambler Mapping Objects to Relational Databases: O/R Mapping In Detail. <http://www.agiledata.org/essays/mappingObjects.html>.

Гуськов Андрей Евгеньевич. Заместитель директора по научной работе Института вычислительных технологий СО РАН (г. Новосибирск). Окончил Новосибирский государственный университет в 2002 году. Кандидат технических наук. Автор 5 статей и одной монографии. Область научных интересов: автоматизированные системы, информационные системы, облачные технологии. E-mail: guskov@ict.sbras.ru.

Шабельников Сергей Андреевич. Аспирант Института вычислительных технологий СО РАН (г. Новосибирск). Окончил Новосибирский государственный университет в 2012 году. Область научных интересов: информационные системы, онтологии. E-mail: brown_men89@rambler.ru.

Васильков Алексей Витальевич. Инженер-исследователь Института вычислительных технологий СО РАН (Новосибирск). Окончил Новосибирский государственный университет в 2011 году. Область научных интересов: информационные системы. E-mail: alexvasilkov@gmail.com