

От спецификаций требований к концептуальной схеме *

© А.Е. Вовченко¹, В.Н. Захаров¹, Л.А. Калиниченко¹, Д.Ю. Ковалёв²,
О.В. Рябухин¹, Н.А. Скворцов¹, С.А. Ступников¹

¹Институт проблем информатики РАН, г. Москва

²Московский государственный университет им. М. В. Ломоносова

leonidk@synth.ipi.ac.ru, dm.kovalev@gmail.com, nskv@ipi.ac.ru

Аннотация

Статья представляет исследование методов проектирования концептуальных схем предметных областей для решения задач. При проектировании используется семантически акцентированный подход к формированию спецификаций предметной области. Процесс проектирования включает формулирование модели требований к предметной области, исходя из вербальных спецификаций требований к задаче, разработку онтологии предметной области и преобразование онтологии к концептуальной схеме. Затронуты вопросы реализации концептуальной схемы предметной области в среде предметных посредников.

1 Введение

В лаборатории Композиционных методов проектирования информационных систем Института проблем информатики РАН разработаны методы и средства решения задач над неоднородными информационными ресурсами в среде предметных посредников. Они включают средства: спецификации концептуальной модели предметной области задачи, унификации неоднородных информационных моделей, регистрации неоднородных информационных ресурсов в посреднике и исполнительную среду, позволяющую описывать задачи в терминах предметной области посредника и решать их с использованием зарегистрированных информационных ресурсов [1]. Акцент в проекте предметных посредников сделан на семантических подходах к представлению неоднородных информационных ресурсов в спецификации предметной области посредника и возможности формального обоснования корректности такого представления.

Однако вопросы обоснованного формирования самих спецификаций предметной области для решения задач в среде предметных посредников до сих пор не рассматривались. Целью настоящей работы является разработка семантически акцентированного подхода к конструированию концептуальных схем предметной области для решения задач в среде предметных посредников, исходя из сформулированных на естественном языке спецификаций требований к задаче, таких, как техническое задание.

Данная проблема соприкасается с областями, имеющими большую историю развития [2]. Концептуальные схемы баз данных используются как логическое представление информационных структур, понятное человеку и независимое от способа реализации в информационных системах. С развитием моделей данных развивались и представления о концептуальных схемах, которые помимо структур включали спецификации поведения. Технологии объектно-ориентированного анализа и дизайна используют результаты, полученные в области концептуальных схем баз данных, для разработки объектных спецификаций информационных систем, независимых от реализации. Исследования в области онтологии продиктованы проблемами семантической интероперабельности систем и специализируются на формальном описании понятий предметных областей, понятных как человеку, так и машине.

В разделе 2 представлен краткий анализ исследований, касающихся упомянутых технологий. Раздел 3 описывает семантически акцентированный процесс разработки концептуальных схем в целом. В разделах 4 – 6 подробнее описаны этапы анализа предметной области задачи, включающие разработку модели требований задачи, разработку онтологии предметной области задачи и трансформацию онтологии в концептуальную схему предметной области задачи.

Труды 12^й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL'2010, Казань, Россия, 2010

2 Исследования, связанные с настоящей работой

Среди исследований, которые необходимо упомянуть, говоря о проектировании концептуальных схем баз данных и информационных систем и исходя из спецификаций требований, наиболее известны работы, связанные с языком спецификации UML [uml]. При разработке схем информационных систем на UML вначале рассматриваются участвующие агенты и их деятельности. Затем описывается последовательность их взаимодействия при помощи сигнатур функций и, наконец, определяется объектная схема, специфицирующая информационную структуру классов и их поведение в соответствии с деятельностями агентов. Спецификации функций задаются на языке ограничений OCL. Таким образом, процесс разработки в целом ориентирован, в первую очередь, на сигнатуры функций, определяющих компоненты процесса решения задачи.

Такой подход, рассматривающий деятельности агентов как чёрные ящики, критикуется в работах Дж. Милопулоса и др. [3], и в качестве решения проблемы предлагается технология выделения понятий предметной области, основанная на целях [4]. Спецификация информационной системы начинается с построения дерева целей, сформулированных на естественном языке, которые должна выполнять система, их конкретизации, выбора пути их достижения, после чего проводится формализация этих спецификаций в виде информационных структур. Данный подход позволяет более обоснованно и структурированно разрабатывать поведенческие спецификации, а также учитывать нефункциональные требования, предъявляемые к разрабатываемым системам.

При исследовании семантически акцентированного подхода к проектированию концептуальных схем следует опираться на онтологическое моделирование и методологии разработки онтологий [5]. Общий принцип разработки онтологий включает анализ предметной области, выделение в ней наиболее существенных понятий и исследование их связей, включение в онтологию связанных понятий по принципу структурной индукции, оценку взаимных ограничений понятий и контроль непротиворечивости. Возникают также задачи обратного проектирования, когда онтология предметной области должна быть восстановлена, исходя, например, из существующих концептуальных схем конкретных систем.

Различные проекты, связанные с проектированием концептуальных схем на основе знаний, почерпнутых из онтологий, используют в качестве тех и других существенно разные представления. Например, в [7] представлен принцип разработки ER-моделей на основе семантических отношений между терминами, присущих тезаурусам, а в [6] описан метод выборки понятий, отношений и ограничений из онтологии широкой предметной области и построения на их основе концептуальной схемы для

решения конкретной задачи. Термины «онтология» и «концептуальная схема» при этом часто используются непоследовательно. Так, множество проектов манипулирует термином «онтология», имея в виду схемы баз данных, либо базы знаний, основанные на информационных моделях, используемых для спецификации онтологий. Появляются также работы по проектированию концептуальных схем на основе онтологий, фактически же разрабатывающие отображение спецификаций схем из одной информационной модели в другую.

Необходимо отметить предшествующие результаты, связанные с технологией предметных посредников, в контексте которой проводится настоящее исследование. Для спецификации концептуальной схемы предметной области посредника в проекте используется развитый объектно-фреймовый язык СИНТЕЗ [8] с логической ориентацией. Для него разработано отображение [9] в язык AMN, что позволяет формально обосновывать корректность уточнения типов при отображении спецификаций, и в частности, проверять корректность регистрации схем неоднородных информационных ресурсов в посреднике.

Концептуальные схемы в проекте используются для спецификации типов и поведения объектов, используемых при решении задач в предметной области посредника. Для спецификации понятий предметной области в проекте используются онтологии, описываемые на том же языке спецификации СИНТЕЗ. Посредством аннотирования элементов концептуальных схем онтологическими понятиями специфицируется их связь с семантикой предметной области.

3 Подход к разработке спецификаций предметной области

Предлагаемый подход к проектированию концептуальных схем в среде предметных посредников является комбинированным, объединяющим подходы к разработке моделей требований, созданию онтологий и построению концептуальных схем. Процесс проектирования включает следующие этапы:

- разработка модели требований к задаче;
- построение глоссария предметной области;
- построение онтологии предметной области на основе глоссария;
- построение концептуальной схемы по онтологии и спецификации требований.

Такой подход позволяет последовательно уточнять и формализовать семантику предметной области задачи в разрабатываемых спецификациях. При этом в процессе проектирования будут формироваться все необходимые спецификации, составляющие определение предметного посредника.

В следующих разделах подробнее описан каждый из этапов проектирования спецификаций предметной области.

4 Построение модели требований

В процессе обсуждения решаемой задачи со специалистами предметной области либо в процессе анализа текстов технического задания фиксируются требования к спецификациям задачи. Требованиям будем называть условие или способность, которым должна удовлетворять система для решения поставленной задачи или для достижения цели [10]. Спецификации выявленных требований уточняются путём декомпозиции на более конкретные требования, которые необходимо выполнить, чтобы достичь выполнения рассматриваемых требований. При этом естественным образом наиболее общая спецификация требований конкретизируется деревом требований, которые уточняются на каждом следующем уровне. Декомпозиция выполняется до тех пор, пока требование может быть сформулировано без спецификации способа его выполнения с помощью использования конкретных информационных ресурсов или агентов. Такой принцип преобразования декомпозиции требований позволяет абстрагироваться от аспектов реализации, которые, в частности, могут присутствовать в текстах технического задания, и, в дальнейшем, выделить понятия предметной области задачи независимо от предметных областей повторно используемых информационных ресурсов.

Для разработки процесса решения задач создаются спецификации функциональных требований, определяющих функциональность системы, которую она должна нести для решения задачи [10]. Декомпозиция функционального требования производится с помощью его разбиения на набор требований, совместно необходимых для выполнения данного требования (AND-декомпозиция), либо на набор альтернативных требований, выполнение одного из которых обеспечивает выполнение декомпозируемого требования (OR-декомпозиция). Нефункциональные требования определяют качественные атрибуты системы, характеризующие её в разных аспектах, таких как надёжность, полнота информации, эффективность, применимость, мобильность и другие.

Элементы дерева функциональных требований оцениваются с точки зрения их соответствия нефункциональным требованиям. По этим данным при наличии альтернативных требований выбирается наилучший план решения задачи из листьев дерева требований. Алгоритм выбора плана выполнения функциональных требований для решения задачи при наилучших оценках выполнения нефункциональных требований описан в [11]. В требованиях, попавших в план решения задачи, описываются пред- и постусловия для их взаимодействия.

В качестве примера приведём фрагмент дерева требований (рис. 1) для решения задачи в области астрономии, а именно, задачи выбора звёзд-стандартов, необходимых для точной калибровки объекта в определённом участке неба. Решение дан-

ной задачи необходимо для исследования послесвечений гамма-всплесков.

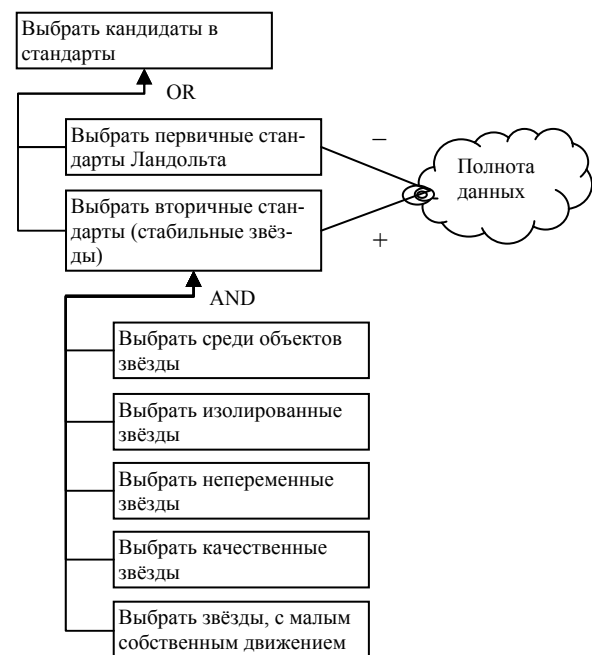


Рис. 1. Фрагмент дерева требований

Выбор кандидатов в стандарты может быть выполнен одним из двух альтернативных способов: выбрать известные стандарты из таблиц Ландольта или вторичные стандарты из звёзд со стабильными характеристиками. Первый способ проигрывает при оценке с точки зрения нефункционального требования полноты данных, так как в заданном участке неба может не оказаться достаточно известных табличных стандартов.

Функциональное требование выбора астрономических объектов со стабильными характеристиками декомпозировано на пять требований: стандартами могут быть только звёзды, эти звёзды должны иметь малое собственное движение, быть изолированными, переменными и качественными с точки зрения астрометрии и фотометрии.

После выбора плана решения задачи для листьев дерева производится операционализация требований, специфицируются их входные и выходные параметры, пред- и постусловия.

Требование: выбрать звёзды с малой величиной собственного движения.

Имя операции: getZeroMotionStars.

Вход: набор астрономических объектов.

Выход: набор астрономических объектов.

Предусловие: астрономические объекты из входного набора должны быть звёздами.

Постусловие: для любого объекта из выходного набора астрономических объектов величина собственного движения равна 0.

5 Построение онтологии предметной области задачи

Для описания предметной области, первым делом, необходимо произвести её концептуализацию. Этому поможет составление глоссария предметной области по спецификациям требований задачи:

- в дереве требований, а также в смысловых отсылках из него на другие документы, специфицирующие задачу, в частности, на техническое задание, выделяются существенные термины, которые попадут в словарь;
- для каждого термина выделяются фрагменты текстов в дереве требований и в других документах описания задачи, определяющие или ограничивающие его интерпретации в данной предметной области;
- каждый термин снабжается вербальным определением, их можно выявить среди выделенных фрагментов текста, либо во внешних источниках, таких как толковые словари, совместно используемые онтологии, публикации, специализированные информационные ресурсы в данной предметной области;
- из определений по индукции выбираются новые существенные для задачи термины, и для них продлевается та же процедура до насыщения глоссария.

В приведённом выше примере в дереве требований выявляются существенные термины для глоссария: *Astronomical Object*, *Star*, *Standard*, *Primary Standard*, *Secondary Standard*, *Morphology*, *Quality*, *Proper Motion*, *Isolation*. Они определяются вербально, например: «*Astronomical Object – significant physical entity on the sky having electro-magnetic emission in particular passbands*»; «*Proper Motion – changing of coordinates of astronomical objects on the sky caused by relational movement of stars and Solar System*». С терминами глоссария связываются утверждения или требования из вербальных спецификаций задачи, которые определяют их или ограничивают в предметной области задачи. Так, с термином *Astronomical Object* связывается утверждение технического задания: «желательно избегать объектов с большой величиной собственного движения». Также глоссарий дополняется значимыми терминами из вербальных определений уже присутствующих в нём терминов и связанных с ними утверждений по индукции: *Magnitude*, *Passband*, *Coordinate*, *Coordinate System*, *Equatorial Coordinate System*, *Right Ascension*, *Declination*, и другие.

Из вербальных определений либо из внешних источников выявляются семантические отношения терминов глоссария, такие, как синонимия, гипер/гипонимия, часть/целое, зависимость, ассоциативная связь. Определяются свойства терминов: являются ли они сущностями, ролями сущностей, свойствами сущностей, процессами, измеряемыми величинами, группами, и др. Фактически глоссарий превращается в тезаурус.

Дальнейший анализ вербальных определений служит преобразованию тезауруса в онтологию, которая состоит из понятий, их отношений и ограничений. Терминам предметной области ставятся в соответствие онтологические определенные понятия. На основе отношений гипер/гипонимии определяется их иерархия. На основе вербальных определений, семантических отношений и свойств терминов производится выявление бинарных (или более арных) отношений понятий, имеющих имена, области определения и значения. Производится формализация вербальных ограничений, относящихся к понятиям, в ограничения на языке спецификации онтологий.

В рассматриваемом примере в соответствие термину *Astronomical Object* определяется онтологическое понятие *AstronomicalObject*, с которым связаны бинарные отношения и свойства: *hasCoordinate*, *hasMorphology*, *hasMagnitude*, *hasProperMotion*, *hasColorIndex*, *isIsolated*, *isQualitative*. Одни из них обнаруживаются в вербальном определении данного понятия, другие следуют из вербальных определений связанных с ним понятий.

На языке СИНТЕЗ [8] спецификация бинарного отношения *hasCoordinate* выглядит следующим образом:

```
{ hasCoordinate;  
  in: association, metaclass;  
  instance_section:  
    { domain: astronimicalObject;  
      range: coordinate;  
      inverse: coordinateOf;  
    }  
};
```

Ниже приведён пример спецификации понятия *AstronomicalObject* и некоторых связанных с ним понятий.

```
{ AstronomicalObject;  
  metaframe in: metaproperty, entity; end  
  in: type, concept;  
  def: {significant physical entity on the sky having  
    electro-magnetic emission in particular  
    passbands};  
  coordinate: Coordinate;  
    metaslot in: hasCoordinate; end  
  morphology: Morphology;  
    metaslot in: hasMorphology; end  
  magnitudes: {set; type_of_element: Magnitude;};  
    metaslot in: hasMagnitude; end  
  properMotion: ProperMotion;  
    metaslot in: hasProperMotion; end  
  colorIndex: ColorIndex;  
    metaslot in: hasColorIndex; end  
  isIsolated: Isolation;  
    metaslot in: isIsolated; end  
  isQualitative: Quality;  
    metaslot in: isQualitative; end  
},  
{ Coordinate;
```

```

metaframe in: metaproperty, measurable; end
in: type, concept;
def: {a point on the sky defined in some
coordinate system};
coordinateOf: AstronomicalObject;
metaslot in: coordinateOf; end
coordinateSystem: CoordinateSystem;
epoch: Epoch;
},
{ CoordinateSystem;
in: type, concept;
def: {a method of point location definition};
},
{ EquatorialCoordinateSystem;
in: type, concept; def: {celestial coordinate system,
including declination and right ascension};
supertype: CoordinateSystem;
ra: RightAscension;
de: Declination;
},
{ RightAscension;
metaframe in: metaproperty, measurable; end
in: type, concept;
def: {one of the two coordinates of the equatorial
coordinate system comparable to longitude,
projected onto the celestial sphere};
},
{ Declination;
metaframe in: metaproperty, measurable; end
in: type, concept;
def: {one of the two coordinates of the equatorial
coordinate system comparable to latitude,
projected onto the celestial sphere};
}, ...

```

В метафреймах спецификаций понятий указываются свойства, говорящие о том, что данное понятие описывает сущность или измеримую величину. Слот `in` содержит метаклассы, в которые входят данные описания, и говорит о том, что спецификация является абстрактным типом данных и онтологическим понятием. Слот `def` содержит вербальное определение понятия. Атрибуты созданы для описания связи с другими понятиями, а вид отношения указывается в метаслотах атрибутов.

Онтология может быть подразделена на модули, определяющие тематически связанные понятия. Введение модульности облегчает интеграцию существующих разделяемых онтологий в онтологию задачи в случае, если в задаче понятия интерпретируются в соответствии с онтологическими обязательствами, принятыми для совместно используемой онтологии.

Задача автоматизации анализа вербальных определений для выявления связей и описания онтологических понятий может быть решена с привлечением методов извлечения знаний из текстов на основе неоднородных семантических сетей [12]. Для русского языка извлечение онтологических знаний из определений терминов может быть получено по методике, разрабатываемой Г.В. Лезиным [13].

Спецификации представительных ресурсов обычно описываются в их собственных терминах, а не в терминах задачи. Таким образом, могут определяться онтологические контексты информационных ресурсов. Если к ресурсу не применима онтология предметной области задачи, то онтология ресурса может быть создана в результате обратного проектирования из концептуальной схемы ресурса с использованием документации, спецификаций требований к ресурсу. Для дальнейшего использования ресурса в предметном посреднике необходимо произвести отображение онтологического контекста ресурса на онтологию предметной области задачи [15].

6 Построение концептуальной схемы задачи

Концептуальная схема определяет информационные структуры и правила преобразования информации в них при решении задачи. Спецификации концептуальной схемы включают определения абстрактных типов данных (АТД) с их структурой, ограничениями целостности и методами; классов как множеств однотипных объектов; функций и процессов, определяющих поведение системы.

Используя знания о предметной области из онтологии, можно построить концептуальную схему для решения задач данной предметной области. Данный процесс основан на решениях эксперта, которому могут предлагаться варианты предпочтительных трансформаций схемы. Общие принципы создания концептуальной схемы на основе онтологии следующие:

- на основе определений выбранных понятий онтологии создаются прототипы АТД, эксперту предлагается создать АТД для понятий, соответствующих сущностям;
- убираются лишние суперпонятия, их структура и ограничения перетекают в текущий АТД;
- убираются некоторые связанные понятия, их структурные описания могут перетекать в текущий АТД;
- вместо именованных отношений определяются типы атрибутов АТД, необходимые для представления информации (скалярные типы, типы перечисления, ассоциации);
- ограничения целостности определяются инвариантами понятий, а также формализуются на основе фрагментов текста, формулирующих требования к соответствующим сущностям;
- классы в соответствии с понятиями определяются существованием множества самостоятельных сущностей, соответствующих объектам типов; для одного типа могут создаваться несколько классов;
- производится операционализация функциональных требований, также функции могут быть определены в соответствии с понятиями процессов и зависимых значений;

- функции над единственным экземпляром АТД становятся методами данного АТД.

Рассмотрим преобразования онтологических понятий в типы концептуальной схемы на примере `AstronomicalObject` и связанных с ним типов.

Типам `ProperMotion`, `Isolation`, `Quality` в онтологии соответствуют понятия со свойством измеримой величины, что позволяет заменить атрибуты, связывающие тип `AstronomicalObject` с ними, на скалярные типы, а сами типы в схеме становятся ненужными. Атрибуту `morphology` соответствует тип перечисления, так как он имеет определённый набор значений.

Тип `ColorIndex` экспертным решением в схеме преобразуется в функцию, вычисляющую показатель цвета по двум параметрам, соответствующим связям понятия `ColorIndex` с двумя полосами пропускания. Так как `colorIndex` работает с одним экземпляром типа `AstronomicalObject`, то функция `colorIndex` становится методом этого типа.

В задаче используется только экваториальная система координат в эпохе 2000-го года. В связи с этим посредством введения дополнительных ограничений на тип `Coordinate` производится его уточнение типом `CoordEQJ2000`: система координат может быть только экваториальная, эпоха только J2000. Таким образом, сами типы `Coordinate` и `CoordinateSystem` напрямую не используются. Они удаляются из схемы, а их спецификации копируются в подтипы `CoordEQJ2000`. У атрибутов `ra` и `de` типа `EquatorialCoordinateSystem` тип значений меняется на скалярный, так как им соответствуют понятия измеряемых величин, из схемы удаляются типы `RightAscension` и `Declination`. Эксперт ещё упрощает схему, удаляя тип `EquatorialCoordinateSystem` и перенося его атрибуты `ra` и `de` непосредственно в тип `CoordEQJ2000`.

```
{ AstronomicalObject;
  in: type;
  coordinate: CoordEQJ2000;
  morphology: {enum; enum_list:
    {point, fuse, elliptic}};
  magnitudes: {set; type_of_element: Magnitude};
  properMotion: real;
  quality: real;
  isVariable: {in: function;
    params: {-returns/boolean};
  };
  colorIndex: {in: function;
    params: {+firstPB/Passband,
      +secondPB/Passband,
      -returns/real};
  };
},
{ Star;
  in: type;
  supertype: AstronomicalObject;
},
{ Standard;
```

```
  in: type;
  supertype: Star;
},
{ CoordEQJ2000; in: type;
  ra: real;
  de: real;
};
```

Схема дополняется классами для типов, которым в онтологии соответствуют понятия с метасвойством `сущность` или `роль`.

```
{ astronomicalObject;
  in: class;
  instance_section: AstronomicalObject;
},
{ star;
  in: class;
  superclass: astronomicalObject;
  instance_section: Star;
},
{ standard;
  in: class;
  superclass: star;
  instance_section: Standard;
};
```

В результате операционализации функциональных требований схема дополняется функциями и методами, необходимыми при решении задачи. Например, функция `getZeroMotionStars`, возвращающая набор звёзд с малым собственным движением. Она станет функцией модуля, так как не связана с одним экземпляром какого-либо типа.

Следует отметить, что определённые в функциональных требованиях пред- и постусловия в процессе их операционализации используются для формирования пред- и постусловий, создаваемых в соответствии с ними функций. Ограничения понятий процессов также могут быть полезны для спецификации созданных функций. Так в концептуальной схеме могут быть определены полные спецификации функций.

При создании спецификаций концептуальных схем на основе онтологии есть возможность сохранять связь между онтологическими понятиями и элементами концептуальных схем с помощью онтологического аннотирования, как это принято в спецификациях предметных посредников [14]. При этом не теряется онтологическая информация, накопленная для данных элементов схем в процессе проектирования.

В частности, в типе `CoordEQJ2000` при замене типов значений атрибутов `ra` и `de`, соответствующих измеряемым величинам, на скалярный тип теряется информация о семантике этих данных с точки зрения онтологии, из которой получена концептуальная схема. Более того, не известно, что означает тип `CoordEQJ2000` после уточнения типа `Coordinate` в терминах онтологии.

Для того чтобы избежать потери семантики, необходимо, во-первых, при создании прототипа концептуальной схемы их онтологии аннотировать типы, атрибуты, функции, параметры соответствующими онтологическими понятиями и, во-вторых, при трансформациях схемы отслеживать изменение семантики онтологических аннотаций.

В первоначальном прототипе концептуальной схемы, созданной для решения задачи поиска звёзд-стандартов, создаются абстрактные типы данных схемы, однозначно соответствующие понятиям, из которых они созданы. Так, тип *Coordinate* аннотируется онтологическим понятием *Coordinate*, атрибуты *ra* и *de* в типе *EquatorialCoordinateSystem* аннотируются понятиями, соответствующими их типам значений, соответственно *RightAscension* и *Declination*.

В процессе трансформации типа *Coordinate* в тип *CoordEQJ2000* и передвижения в него атрибутов *ra* и *de* отслеживаются и изменения их семантики с точки зрения онтологии. Они будут принадлежать классам понятий, выраженных формулами.

Тип *CoordEQJ2000* соответствует подпонятию понятия *Coordinate*, у которого эпоха равна 2000-му году и система координат экваториальная:

```
{x/Coordinate | in(x, coordinate) & x.epoch=J2000 &
in(x.coordinateSystem, equatorialCoordinateSystem)}
```

Атрибуты *ra* и *de* при перемещении из типа в тип и изменении типов значений так и остаются принадлежащими понятиям *RightAscension* и *Declination* соответственно. Однако путь их принадлежности связанным понятиям также можно проследить, уточнив понятия, которым они соответствуют. Понятие для атрибута *ra*:

```
{x/RightAscension | in(x, rightAscension) &
ex y/Coordinate, z/EquatorialCoordinateSystem
( in(y, coordinate) &
in(z, equatorialCoordinateSystem) &
y.coordinateSystem = z & z.ra = x) }
```

Аналогично, понятие для атрибута *de* будет следующим:

```
{x/Declination | in(x, declination) &
ex y/Coordinate, z/EquatorialCoordinateSystem
( in(y, coordinate) &
in(z, equatorialCoordinateSystem) &
y.coordinateSystem = z & z.de = x)) }
```

На основании отображения понятий из разных онтологических контекстов и аннотированных ими элементов концептуальных схем можно найти релевантные элементы схем ресурса и задачи. На этом этапе могут быть разработаны функции разрешения конфликтов, в дальнейшем используемые при регистрации ресурсов в посреднике.

Процесс построения концептуальной схемы не зависит от того, каким образом она будет реализо-

вываться, соответственно, и от требования реализации в среде посредников. Однако реализация схемы в среде посредников имеет свои особенности. Сами спецификации схемы посредника в комплексе с исполнительной средой посредника и зарегистрированными ресурсами уже являются реализацией схемы, так как дополнительных действий по реализации со стороны посредника производить не нужно. Те типы и классы концептуальной схемы предметной области, которые могут использоваться для регистрации представительных ресурсов, попадут в схему посредника. Также туда попадают спецификации, которые могут быть реализованы в виде специального информационного ресурса, регистрируемого в посреднике как обычные ресурсы. Остальная часть концептуальной схемы предметной области должна быть реализована традиционным способом в виде программы над предметным посредником.

7 Заключение

В статье кратко описан процесс проектирования спецификаций предметной области, необходимых для определения предметного посредника и решения задач предметной области в его среде с повторным использованием внешних информационных ресурсов. Данный подход отличает направленность на выявление и сохранение в концептуальной схеме семантики предметной области, независимость от участвующих в решении задачи конкретных информационных ресурсов. Указаны особенности реализации концептуальных схем предметных областей в среде посредников.

Литература

- [1] Брюхов Д.О., Вовченко А.Е., Захаров В.Н., Желенкова О.П., Калиниченко Л.А., Мартынов Д.О., Скворцов Н.А., Ступников С.А. Архитектура промежуточного слоя предметных посредников для решения задач над множеством интегрируемых неоднородных распределённых информационных ресурсов в гибридной грид-инфраструктуре виртуальных обсерваторий// Информатика и её применения. – 2008. – Т. 2, Вып. 1. – С. 2-34.
- [2] Коголовский М.Р., Калиниченко Л.А. Концептуальное моделирование в технологиях баз данных и онтологические модели// Труды Симпозиума «Онтологическое моделирование», г. Звенигород, 19 – 20 мая 2008 г., ред. Калиниченко Л.А. – М: ИПИ РАН, 2008. – 303 с. – ISBN 978-5-902030-54-6. – С. 114-148.
- [3] Jiang L., Topaloglou T., Borgida A., Mylopoulos J.. Goal-oriented conceptual database design// Conf. on Requirements Engineering (RE'07), Delhi, 2007.
- [4] Lapouchnian Y.Yu., Mylopoulos J. Requirements-driven design and configuration management of business processes//Proc. 5th Int. Conf. on Business Process Management (BPM 2007), Brisbane, Aus-

- tralia, Sept. 24–28, 2007. G. Alonso, P. Dadam, and M. Rosemann (Eds.), LNCS. – Berlin – Heidelberg: Springer-Verlag, 2007. – V. 4714. – P. 246-261.
- [5] Загоруйко Ю.А. Методы и методологии разработки, сопровождения и реинжиниринга онтологий// Труды Симпозиума «Онтологическое моделирование», г. Звенигород, 19–20 мая 2008 г., ред. Л. А. Калиниченко – М: ИПИ РАН, 2008. – 303 с. – ISBN 978-5-902030-54-6. – С. 167-200.
- [6] Conesa J., Olivé A. A method for pruning ontologies in the development of conceptual schemas of information systems//ER2004, Shangai, China: Springer, 2004.
- [7] Sugumaran V., Storey V. The role of domain ontologies in database design – an ontology management and conceptual modeling environment// ACM Transactions on Database Systems. – V. 31, No 3. – P. 1064-1094.
- [8] Kalinichenko L.A., Stupnikov S.A., Martynov D.O. SYNTHESIS: a language for canonical information modeling and mediator definition for problem solving in heterogeneous information resource environments. – Moscow: IPI RAN, 2007. – 171 p.
- [9] Ступников С.А. Отображение спецификаций ядра канонической модели в нотацию абстрактных машин// Формальные методы и модели для композиционных инфраструктур распределённых информационных систем: Системы и средства информатики, специальный выпуск. – М: ИПИ РАН, 2005. – С. 69-95.
- [10] IEEE Standard Glossary of Software Engineering Terminology. IEEE Std 610.12-1990.
- [11] Sebastiani R., Giorgini P., Mylopoulos J. Simple and minimum-cost satisfiability for goal models// Proc. Int. Conf. on Advanced Information Systems Engineering (CAiSE 2004), Riga, Latvia, 2004.
- [12] Helbig H., Gnorlich C. Multilayered extended semantic networks as a language for meaning representation in NLP systems// Computational Linguistics and Intelligent Text Processing (CICLing 2002). Ed. by A. Gelbukh. – Berlin: Springer, 2002. – V. 2276 of LNCS. – P. 69-85.
- [13] Лезин Г.В. Онтологическая семантика текста: форматирование лексики в семантическом словаре// RCDL'2009, Петрозаводск, 2009.
- [14] Скворцов Н.А. Связывание объектных спецификаций по семантике онтологического уровня// RCDL'2006, Суздаль. – Ярославль: Ярославский государственный университет им. П.Г. Демидова, 2006. – С. 70-74.
- [15] Скворцов Н.А. Специфика подходов к отображению онтологий// Вопросы искусственного интеллекта. SCMAI Transactions. – М.: 2010. – № 2.

From specifications of requirements to conceptual schema

A.E. Vovchenko, V.N. Zakharov,
L.A. Kalinichenko, D.Yu. Kovalyov,
O.V. Pyabukhin, N.A. Skvortsov, S.A. Stupnikov

The paper presents an investigation of subject domain conceptual schema modeling methods for problem solving using a semantic-based approach to subject domain specifications development. The development process includes analysis of requirements to the subject domain starting from verbal specifications of the problem, development of subject domain ontology and transformation of the ontology to a conceptual schema. Issues of conceptual schema implementation in a subject mediator environment are discussed.

* Работа выполнена при частичной финансовой поддержке РФФИ (проекты 08-07-00157, 10-07-00342) и Программы фундаментальных исследований Президиума РАН № 15П «Фундаментальные проблемы системного программирования», раздел 3