

РАЗРАБОТКА OWL-ОНТОЛОГИИ ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ СГТУ

Н. Б. Фролова

Саратовский государственный технический университет им. Гагарина Ю. А.

Поступила в редакцию 02.07.2016 г.

Аннотация. Разработана OWL-онтология образовательных ресурсов Саратовского государственного технического университета имени Гагарина Ю. А., позволяющая представить информацию в глобальной семантической сети об учебных программах и объеме обеспечивающих их материалов по учебным курсам ВУЗа, и спроектировано приложение автоматического сбора данных и наполнения онтологии.

Ключевые слова: онтология, RDF-хранилище образовательных ресурсов, приложение автоматизации сбора данных.

Annotation. The aim of this research is development of RDF-store on the basis of the designed ontology of Yuri Gagarin State Technical University of Saratov educational resources allowing to provide information in a global semantic network on training programs and volume of the materials providing them on training courses of university and an application programming of automatic data collection and filling of ontology.

Keywords: ontology, TripleStore, RDF-store of educational resources, software application automation data collection.

ВВЕДЕНИЕ

Проектирование семантической сети на данный момент времени является задачей актуальной и мало реализованной в сфере представления учебных заведений высшего образования Российской Федерации. Достоинством онтологического подхода моделирования данных является возможность дополнения и расширения онтологии путем присоединения других разработок с явным указанием соответствия классов в каждой из них и сопоставление их с понятиями, представленными в семантическом вебе. Такая возможность расширения онтологий основана на главном принципе – открытости данных и знаний, объединении описаний понятий различных предметных областей с целью

формирования общего терминологического подхода в той или иной сфере деятельности человечества.

Саратовский государственный технический университет имени Гагарина Ю. А. (СГТУ) представляет собой один из ведущих ВУЗов Поволжья, предоставляет хорошую информационную поддержку процесса обучения средствами информационно-образовательной среды (ИОС), позволяющей получить доступ к полному пакету учебно-методических материалов по каждой дисциплине для авторизованных пользователей, т.е. студентов и преподавателей. В связи с тем, что семантическое представление данных в глобальной сети на данный момент времени является неотъемлемой частью высоко-технологического подхода к представлению в том числе и высших учебных заведений, была спроектирована OWL-онтология, позволя-

ющая представить уровень наполненности УМКД и, соответственно, информационную поддержку студентов в данном ВУЗе. Кроме того, современный рейтинговый подход к работе преподавателей в нашем ВУЗе требует контролировать объем предоставляемых материалов обучающимся по каждой дисциплине, что также возможно осуществить с помощью разработанной онтологии.

Прежде чем приступить непосредственно к созданию онтологии, в первую очередь необходимо определить перечень задач, которые она должна решать и на какие вопросы давать ответы [1].

ОСНОВНЫЕ ВОПРОСЫ К ОНТОЛОГИИ И БАЗОВАЯ СТРУКТУРА УЧЕБНОГО ПРОЦЕССА

Согласно поставленной цели для онтологии были сформулированы следующие вопросы:

1. По каким направлениям проводится обучение в Саратовском государственном техническом университете имени Гагарина Ю. А.
2. На каком факультете и на какой кафедре проводится подготовка по заданным направлениям обучения.
3. Каким учебным планом поддерживается обучения по направлению.
4. Какие дисциплины входят в учебный план конкретного направления.
5. Какие информационные материалы и в каком объеме обеспечивают информационное сопровождение и поддержку дисциплины (количество доступных лекций, презентаций, методических указаний по выполнению практических заданий и самостоятельной работы студентов, входящих в учебно-методический комплекс дисциплины (УМКД)).
6. Какова рабочая программа (РП) дисциплины.
7. Каковы основные параметры дисциплины: курс и семестр, на котором дисциплина преподается, форма отчетности, наличие курсового проекта, курсовой работы или расчетно-графической работы, краткое описание дисциплины.

Организация учебного процесса требует предварительного анализа его структуры для конкретного ВУЗа.

В основу онтологии учебных курсов ВУЗа были положены основные принципы структуризации учебного процесса в СГТУ, за основу взята структура ИОСа. В соответствии с этими принципами была сформирована структура и выделены основные компоненты учебных курсов [2].

На основе сформулированных вопросов был сформирован словарь онтологии, из которого согласно разработанной структуре была построена таксономия. Разработанная база знаний с выделенными классами, объектными свойствами, свойствами типа данных и допустимыми значениями выделенных перечислимых классов представлена схематично на рис. 1.

Разработанная таксономия была реализована в виде OWL-онтологии средствами редактора Protégé v.5.0 [3].

ПРОЕКТИРОВАНИЕ ОНТОЛОГИИ, СОЗДАНИЕ КЛАССОВ

В представленной работе в качестве языка представления онтологии был выбран синтаксис RDF/XML для максимальной совместимости с другими приложениями.

Основными компонентами онтологии являются классы – *owl:Class*. Главным является обязательный класс *owl:Thing*, представляющий множество всех возможных объектов любой онтологии, все остальные классы являются подклассами данного класса. Классы можно создавать в явном виде как суперклассы и подклассы, как анонимные классы через определение ограничений и списком индивидов.

В рассматриваемой предметной области согласно построенной таксономии выделено семь равноправных классов:

- Четыре класса являются примитивными (*primitive class*):
 - *Faculty* (Факультет),
 - *Department* (Кафедра),
 - *Program* (Направление подготовки),
 - *Discipline* (Дисциплины).

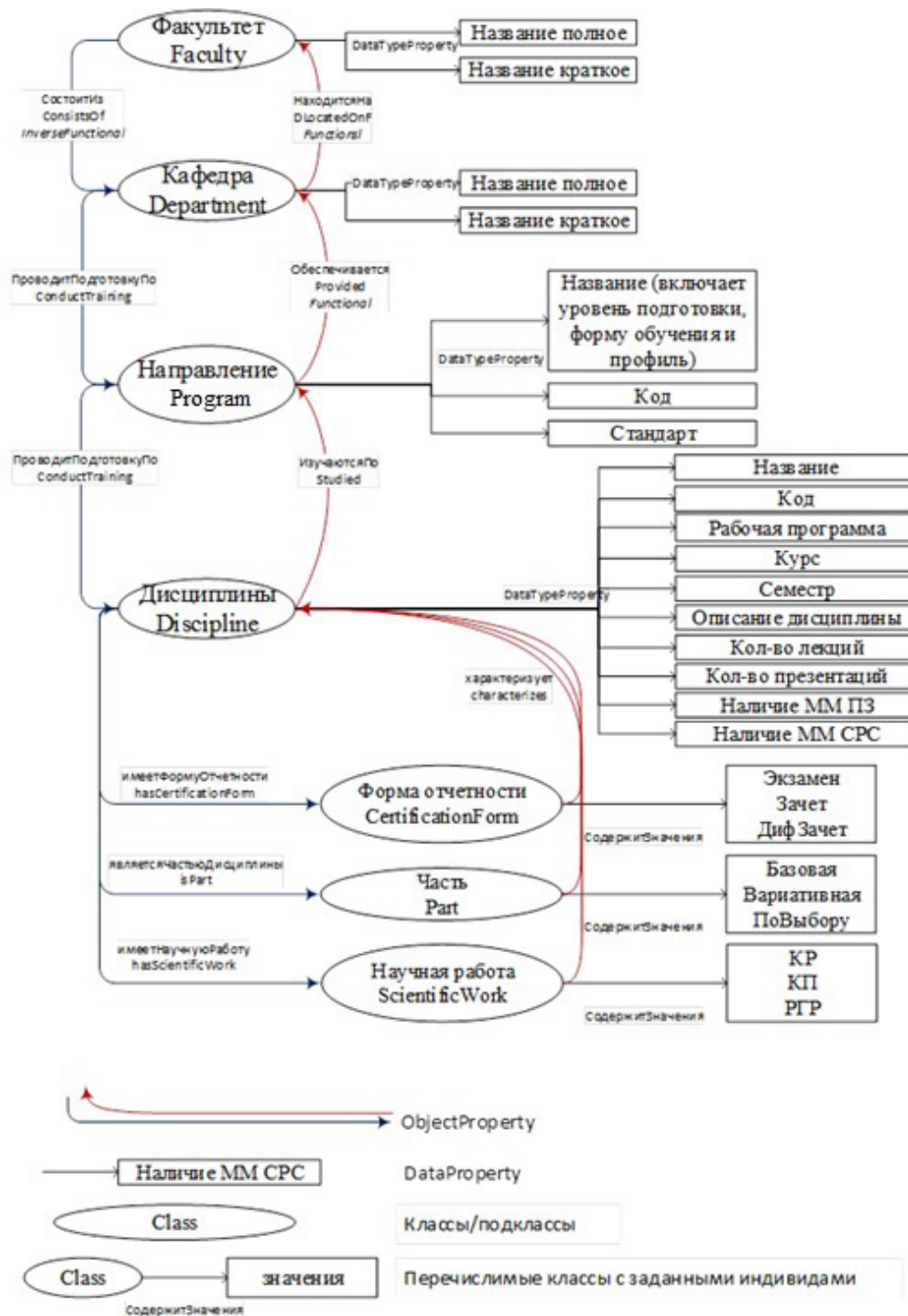


Рис. 1. Таксономия учебных курсов СГТУ с выделенными классами и свойствами

• Три класса, определенные списком индивидов, являются перечислимыми. В прикладной онтологии определено 3 определенных класса (*defined class*), заданных как перечислимые классы с набором допустимых значений, т.е. со следующими индивидами:

– *CertificationForm* (Форма Отчетности): {ДифЗачет, Зачет, Экзамен},

– *PartDis* (Часть): {Базовая, Вариативная, ПоВыбору},

– *ScientificWork* (Научная работа): {КП, КР, РГР}.

Описание перечислимого класса представлено на рис. 2 на примере класса *CertificationForm* (Формы отчетности).

Как и при разработке любого программного продукта важным является текстовое

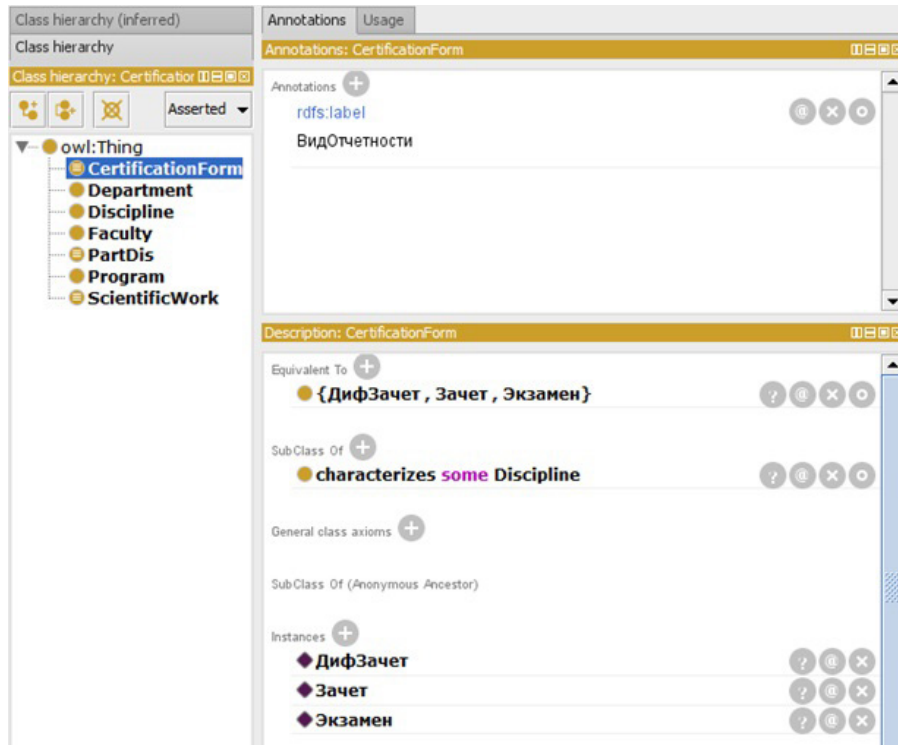


Рис. 2. Описание перечислимого класса *CertificationForm*

описание различных компонентов с целью повышения читабельности и удобства работы нескольких разработчиков. Язык OWL позволяет создавать аннотации с различной информацией (дата создания, комментариев, надпись, автор, ссылка на ресурс и пр.) и метаданные классов, свойств, индивидов и заголовка онтологии. Из допустимых в редакторе Protégé девяти свойств аннотации в данном проекте для всех компонентов, т.е. для классов, объектных свойств, свойств типа данных и индивидов указывались надписи (*rdfs:label*) и почти для всех комментарии (*rdfs:comment*), описывающие подробнее назначение и суть каждой составляющей.

РАЗРАБОТКА СТРУКТУРЫ СВОЙСТВ ОНТОЛОГИИ

После определения классов формируется иерархия свойств классов, определяющих отношения между ними. Свойства (*rdfs:Property*) в OWL представляют собой отношения, которые связывают между собой классы и/или индивиды. Здесь используется два типа свойств:

- Свойства объектов или объектные свойства (*owl:ObjectProperty*) определяют отноше-

ния между классами или индивидами двух классов.

- Свойства типа данных (*owl:DatatypeProperty*) связывают сущности классов с фиксированными значениями данных заданного типа (с литералами).

Объектные свойства онтологии

Любое свойство в онтологии имеет область значений – домен (*domain*), и область определений – диапазон (*range*). Домен представляет собой класс, индивиды которого будут связываться этим свойством, а диапазон – класс, с индивидами которого будет связываться указанное свойство или тип данных.

Причем, как и для классов, для свойств возможна иерархическая структуризация с выделением «свойств-подсвойств», вершиной в иерархии служит свойство *topObjectProperty*.

Формирование связей между классами осуществляется посредством указания объектных свойств.

Для каждого свойства можно в явном виде задать домен и диапазон, как, например, для свойства *hasScientificWork* (*имеетНаучнуюРаботу*), позволяющего в явном виде указать связь между классом *Discipline* (*Дисциплина*) и перечислимым классом *ScientificWork* (*НаучнаяРабота*) (рис. 3).


```
<!-- http://www.semanticweb.org/natalya/ontologies/2016/sstu#hasScientificWork -->

<owl:ObjectProperty rdf:about="http://www.semanticweb.org/natalya/ontologies/2016/sstu#hasScientificWork">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/natalya/ontologies/2016/sstu#Discipline"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/natalya/ontologies/2016/sstu#ScientificWork"/>
  <rdfs:label>имеетНаучнуюРаботу</rdfs:label>
</owl:ObjectProperty>
```

Рис. 3. *ObjectProperty* с явным указанием домена и диапазона на языке OWL

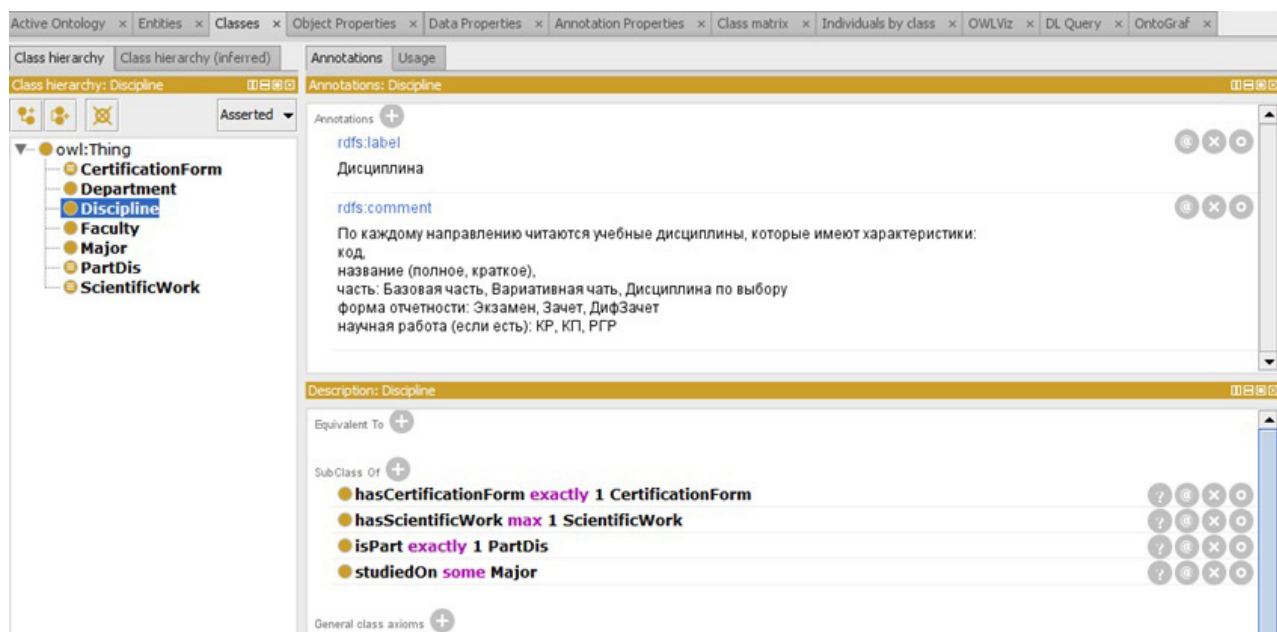


Рис. 4. Класс *Discipline* с явным определением ограничений

Такой подход считается не очень корректным, т.к. в больших онтологиях может приводить к ошибкам, поэтому допускается лишь для небольшой части однозначно определенных свойств или для малых онтологий.

Кроме того, OWL позволяет придавать свойствам различный смысл за счет использования характеристик свойств, называемых аксиомами или ограничениями. Различные ограничения могут быть наложены и на классы.

В разрабатываемой онтологии в явном виде заданы следующие характеристики свойств:

- *consistsOf* (состоитИз) – *owl:InverseFunctionalProperty*;
- *locatedOn* (находитсяНа) – *owl:FunctionalProperty*;
- *provided* (обеспечивается) – *owl:FunctionalProperty*.

Остальные свойства заданы в виде явных ограничений. Ограничения описывают класс на основе отношений, в которых участвуют индивиды этого класса. В OWL ограничения делятся на три категории:

- кванторные ограничения (*Quantifier Restrictions*),
- ограничения кардинальности (*Cardinality Restrictions*),
- ограничения на значения (*hasValue Restrictions*).

В разрабатываемой предметной области свойства в виде ограничений заданы для всех классов онтологии. Причем используются все три категории ограничений. В качестве примера (рис. 4) приведены ограничения, задающие свойства класса *Discipline* (Дисциплина).

Свойства типа данных

Второй тип свойств – свойства типа данных (*owl:DatatypeProperty*), которые связывают индивиды классов с некоторыми литеральными значениями. Также как и объектные свойства, они могут быть организованы в иерархии с общим верхним свойством *owl:topDatatypeProperty*.

В разработанной онтологии учебных курсов выделено девять свойств типа данных первого уровня и у двух из них еще шесть

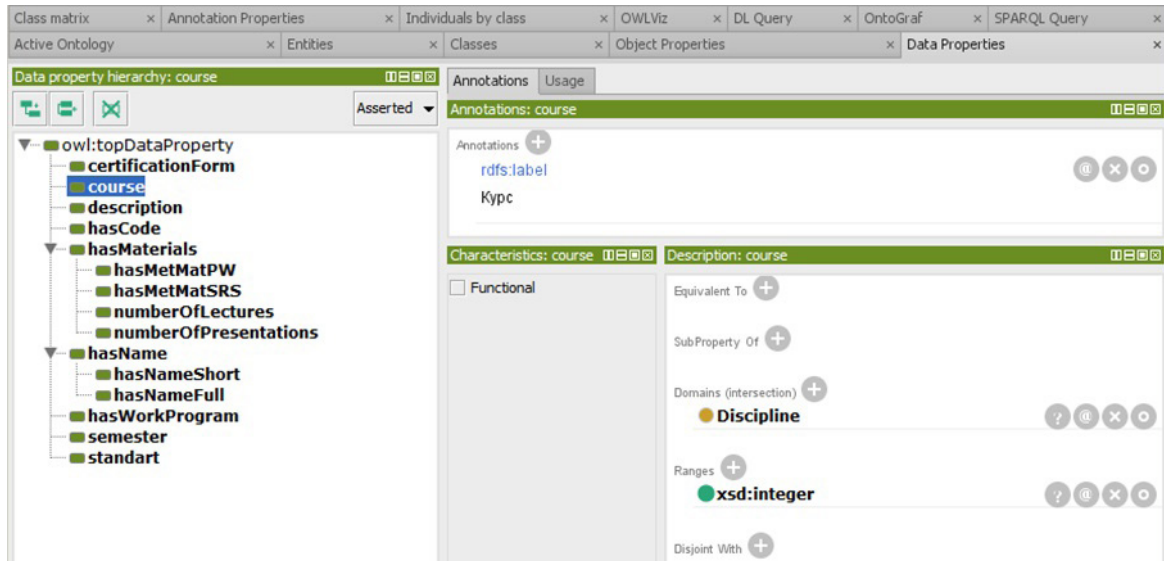


Рис. 5. Описание класса *Discipline* (Дисциплина) с явным определением ограничений

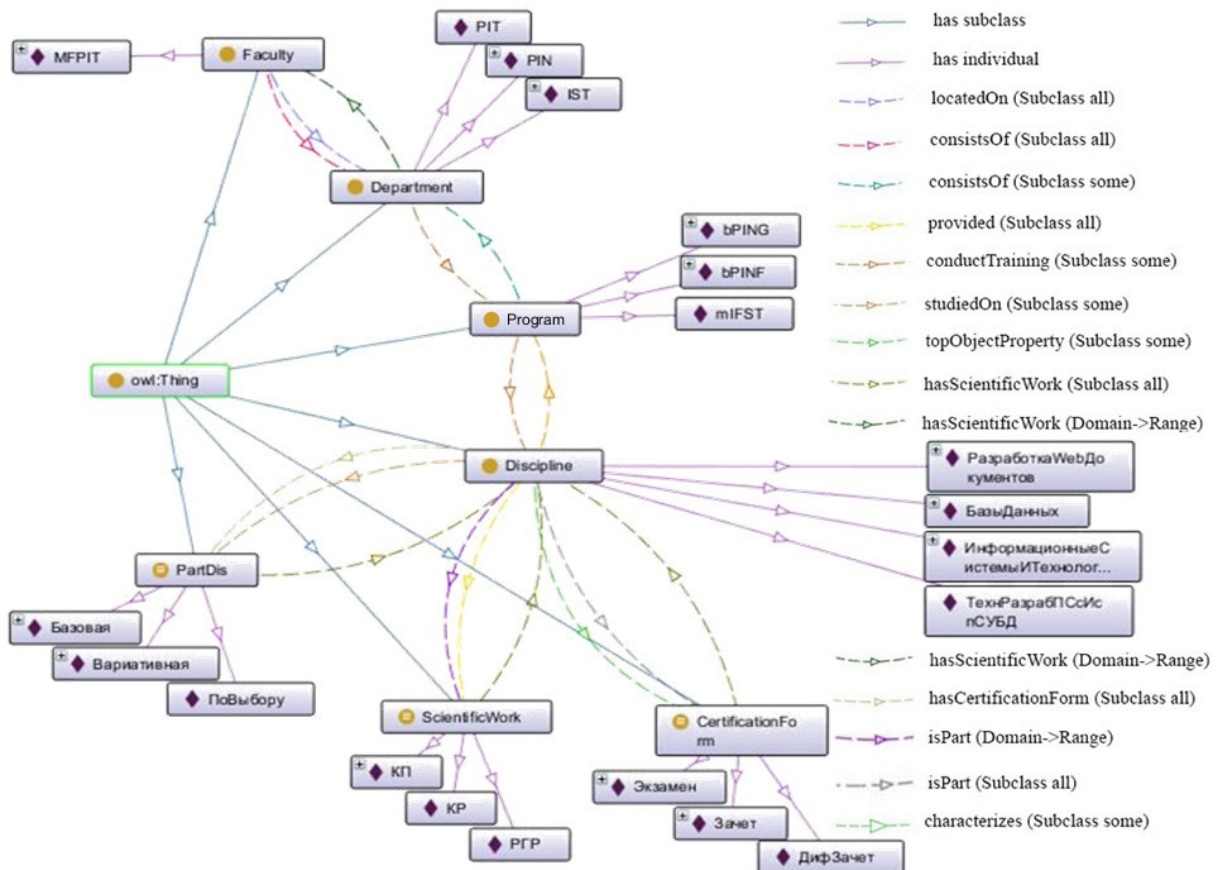


Рис. 6. Онтограф предметной области «Учебные курсы СГТУ»

подсвойств. Структура свойств типа данных представлена на рис. 5.

Для каждого свойства типа данных в отличие от объектных свойств можно дополнительно ввести только одну характеристику – функциональное свойство.

Четыре свойства: *certificationForm* (форма отчетности), *hasCode* (имеет код), *hasWorkProgram* (имеет рабочую программу) и *standart* (стандарт) определены в работе как функциональные.

Доменом для свойств типа данных служат классы, индивиды которых должны быть заданы определенными литеральными значениями. В качестве диапазона для этого типа свойств может быть указано либо конкретное выражение, либо один из допустимых в заданной нотации типов данных [4]. В работе в качестве диапазона заданы типы данных либо *string*, либо *integer*, либо данная характеристика в явном виде не указана и определяется автоматически согласно введенным данным.

ПОСТРОЕНИЕ СЕМАНТИЧЕСКОЙ СЕТИ ПРЕДМЕТНОЙ ОБЛАСТИ

Спроектированная онтология представляет собой некоторую разновидность семантической сети, имеющей вид ориентированного графа, вершинами которого являются классы, дугами – свойства, представляющие направленные отношения или связи между ними. В результате онтологического моделирования рассматриваемой предметной области средствами редактора Protégé 5.0 [4] был построен граф семантической сети (рис. 6). В данном примере приведен граф, включающий классы, индивиды и объектные свойства [5].

ПРОВЕРКА ОНТОЛОГИИ НА СОГЛАСОВАННОСТЬ

При проектировании любой онтологии важную роль играет вопрос непротиворечивости вводимых объектов и индивидов. Для проверки OWL-онтологии на согласованность используется так называемая логическая машина вывода (reasoner). Основываясь на описании класса, резонер заключает, может ли данный класс содержать какие-нибудь индивиды. В том случае, если это невозможно, класс считается несогласованным или противоречивым. Подобной проверке подвергаются и индивиды классов, указывая противоречивость последних описанию класса или переопределению того или иного индивида к какому-либо классу, основываясь на описании этих индивидов и классов.

Кроме того, машина вывода используется для автоматизации процесса построения ие-

рархий классов [6, 7], т. е. она умеет достраивать иерархию классов, выявляя неописанные отношения типа «класс-подкласс», что обычно называют классификационным тестом. В результате работы такого теста строится так называемая выведенная (inferred) иерархия классов.

Таким образом, резонеры могут:

- проводить классификацию и показывать иерархию выводимых классов,
- проверять согласованность и показывать несоответствия,
- определять типы индивидов, их принадлежность к определенному классу.

Использовать логическую машину вывода рекомендуется регулярно, как при добавлении любых новых аксиом, так и при наполнении онтологии индивидами.

В OWL резонеры работают на базе концепции Open World Reasoning – принципа открытости мира, в противовес концепции баз данных, основанных на подходе замкнутости мира.

На данный момент времени в Protégé 5.0 доступно три резонера: Fact++, HermiT 1.3.8.413 и Pellet. Данные типы резонеров различаются алгоритмом выстраивания связей в иерархических структурах (например, табличные алгоритмы, резолюции и т.п.); поддерживаемыми форматами данных; языком, на котором написано приложение; поддерживаемыми профилями языка онтологического моделирования и другими параметрами.

Основные характеристики резонеров имеют следующий вид:

- Fact++ поддерживает алгоритм Tableau-based, реализован на C++;
- HermiT поддерживает алгоритм Hypertableau-based, реализован на Java;
- Pellet поддерживает алгоритм Tableau-based, реализован на Java.

Все представленные резонеры относятся к классу Open source.

Алгоритмы этих систем основываются на различных вариациях табличного метода, который имеет хорошую производительность и расширяемость. Данный метод предлагает проверку существования модели выполнения класса путем построения структуры

```

----- Running Reasoner -----
FaCT++ Kernel: Reasoner for the SROIQ(D) Description Logic, 64-bit
Copyright (C) Dmitry Tsarkov, 2002-2015. Version 1.6.4 (23 August 2015)
Pre-computing inferences:
- class hierarchy
- object property hierarchy
- data property hierarchy
- class assertions
- object property assertions
- data property assertions
- same individuals
Fact++
Ontologies processed in 102 ms by FaCT++

----- Running Reasoner -----
Pre-computing inferences:
- class hierarchy
- object property hierarchy
- data property hierarchy
- class assertions
- object property assertions
- data property assertions
- same individuals
HermiT 1.3.8.413
Ontologies processed in 1188 ms by null

----- Running Reasoner -----
Pre-computing inferences:
- class hierarchy
- object property hierarchy
- data property hierarchy
- class assertions
- object property assertions
- data property assertions
- same individuals
Pellet
Ontologies processed in 778 ms by Pellet

```

Рис. 7. Результат работы трех основных резонеров для разработанной онтологии учебных курсов СГТУ

специального вида – таблицы, которая представляет собой размеченный граф определенного вида, удовлетворяющий набору ограничений. Ограничения описывают семантику конкретной логики. При этом модель существует только в том случае, если существует таблица, удовлетворяющая ограничениям.

В представленной работе, онтология является достаточно простой, и логическая машина вывода используется только с целью проверки ее на согласованность, поэтому использовались для сравнения все три машины логического вывода, однако Fact++ имеет большую скорость выполнения при проверке на онтологии, заполненной минимальным набором основных индивидов (рис. 7). При этом важной чертой резонера является поддержка объяснения ошибок, что доступно для всех рассматриваемых вариантов.

НАПОЛНЕНИЕ ОНТОЛОГИИ ИНДИВИДАМИ

Построенная онтология является по сути каркасом, представлением некоторой предметной области. Наполнение онтологии индивидами позволит сформировать RDF-хранилище данных.

Специфика разработанной онтологии требует ручной выборки данных из различных источников, причем осуществляется она по сложным критериям с многоуровневым переходом по страницам сайта СГТУ и ИОСа, что дополнительно усложняется необходимостью авторизации пользователя при сборе данных.

Поэтому вопрос ручного заполнения разработанной онтологии представляется крайне сложным и длительным. В результате было разработано приложение, позволяющее автоматизировать процесс сбора данных, конвертации их в RDF-триплеты, наполнения хранилища и получения ответов на поставленные перед онтологией вопросы.

Спроектированное приложение имеет веб-интерфейс (рис. 8) и формально может быть разделено на две части:

1 часть представляет собой программу сбора требуемых данных с сайта СГТУ и ИОС с подключением библиотеки Selenium ChromeDriver [8], представление собранных данных в виде RDF-триплетов с передачей их в хранилище онтологии (Triple Store) средствами библиотеки BrightstarDB [9].

2 часть предназначена для визуализации выборки данных для конвертации и отобра-

СГТУ

Факультет
Международный факультет

Кафедра
Прикладная информатика и программная инженерия (ПИИ)

Направление
Программная инженерия (бакалавры)

Дисциплина
Базы данных

ПОИСК

Факультет	Международный факультет прикладных информационных технологий	
Кафедра	Прикладная информатика и программная инженерия (ПИИ)	
Направление обучения	Название	Программная инженерия (бакалавры)
	Сокращенное название	ПИИ-Ж
	Код	09.03.04
	Стандарт	http://fgosvo.ru/uploadfiles/fgosvob/090304.pdf
Дисциплина	Название	Базы данных
	Код	Б.3.1.6
	Форма отчетности	Экзамен
	Часть	Базовая
	Научная работа	Нет
	Рабочая программа	http://www.sstu.ru/files/umu/op_sym/MFPIT/b-PINZh/Metod_38_b-PINZh_29.09.2015.pdf?t=1464630315
	Семестр	3
	Курс	2
	Количество лекций	9
	Количество презентаций	9
	Количество мет. материалов ПЗ	13
	Количество мет. материалов СРС	4
	Описание	Студент должен знать: основные модели данных, их организацию, архитектуру современных СУБД, принципы построения запросов и манипулирования данными средствами специальных языков запросов, описания данных и манипулирования данными, методы и средства проектирования баз данных, особенности построения информационных систем в архитектуре «клиент-сервер», принципы физической организации и защиты баз данных. Студент должен уметь: разрабатывать инфологические и реляционные модели данных, спроектировать БД и реализовать проект средствами конкретной СУБД, формировать запросы к существующей базе данных средствами языка манипулирования данными. Студент должен владеть: практическими навыками проектирования и физической реализации реляционных баз данных, языком обращения к базам данных SQL.

© 2016 - University Ontology

Рис. 8. Клиентская часть приложения с выводом полной информации по дисциплине

жения основной информации средствами браузера. Для отображения страницы используется технология ASP.NET MVC. Для передачи данных между сервером и клиентом – технология ASP.NET Web API. На стороне клиента работа осуществляется посредством JavaScript фреймворка AngularJS [10].

Пример результата работы приложения с выборкой данных по заданному факультету, кафедре, направлению с отображением параметров конкретной дисциплины представлен на рис. 9 на примере курса «Базы данных», читаемого для направления подготовки 09.03.04 «Программная инженерия».

ЗАКЛЮЧЕНИЕ

На основе проведенного исследования была построена OWL-онтология учебных курсов Саратовского государственного технического университета имени Гагарина Ю. А., содержащая данные об информационном обеспечении учебных дисциплин по различным направлениям подготовки, реализуемым в рамках ВУЗа. В качестве информационного обеспечения выступает описание дисциплины, рабочая программа, количество лекционного и мультимедийного материала, методические указания по проведению практических

занятий и самостоятельной работы студентов, что позволит оценить объем и доступность учебных материалов обучающимся.

Для работы с построенной онтологией было спроектировано приложение, позволяющее автоматизировать процесс сбора данных из различных источников, наполнения RDF-хранилища и получения ответов на основные вопросы, сформулированные к разработанной онтологии. Особенностью представленного программного продукта является реализация многоуровневого поиска данных, как с сайта ВУЗа, так и возможность выборки данных из закрытой части информационно-образовательной среды с предварительной авторизацией зарегистрированного пользователя, преобразование полученных данных в RDF-триплеты с последующим заполнением онтологии и выводом ключевой информации пользователю с поддержкой веб-интерфейса.

Результаты работы планируется опубликовать в облаке открытых данных с целью представления информации о направлениях подготовки в СГТУ в семантическом вебе и использовать в научно-технической библиотеке СГТУ с целью контроля качества заполнения информационно-образовательной среды ВУЗа.

СПИСОК ЛИТЕРАТУРЫ

1. Wood D., Zaidman M., Ruth L. Linked Data. Structured data on the Web. Shelter Island: Manning Publications, 2014.
 2. Фролова Н. Б., Семченко Т. В. Проектирование онтологии учебных курсов вуза // Проблемы управления в социально-экономических и технических системах: сборник научных статей. Международная научно-практическая конференция (8 апреля 2016 г.) – Саратов : Издательский центр «Наука». – 2016. – С. 215–218.
 3. Protégé. Официальный сайт редактора. – Режим доступа: <http://protege.stanford.edu/>.
 4. Основы построения OWL-онтологий с использованием редактора PROTEGE DESKTOP 5.0 beta : учеб. пособие / Сытник А. А., Шульга, Т. Э.; Вагарина, Н. С.; Мельникова, Н. И.; Саратовский гос. техн. ун-т. – Электрон. текстовые дан. – Саратов : СГТУ, 2015. – Режим доступа: http://lib.sstu.ru/books/zak_160_15.pdf.
 5. Фролова Н. Б. Разработка RDF-хранилища учебных курсов СГТУ средствами редактора Protégé 5.0. // Проблемы управления в социально-экономических и технических системах: сборник научных статей. Международная научно-практическая конференция (8 апреля 2016 г.) – Саратов: Издательский центр «Наука». – 2016. – С. 226–229.
 6. Антониоу Г., Грос П., Хармелен ван Ф., Хоекстра Р. Семантический веб / пер с англ. Т. Шульга. – М. : ДМК Пресс, 2016. – 240с.
 7. Sunitha Abburu. A Survey on Ontology Reasoners and Comparison. // International Journal of Computer Applications (0975 – 8887). – 2012. – Vol. 57. – №17. – С. 33–39.
 8. Библиотека для получения данных с веб-страниц Selenium Chrome Driver. Официальный сайт приложения. – Режим доступа: <https://sites.google.com/a/chromium.org/chromedriver/getting-started>.
 9. BrightstarDB. Официальный сайт фрэймворка. – Режим доступа: <http://brightstardb.com/>.
 10. AngularJS. Официальный сайт проекта. – Режим доступа: <https://angularjs.org/>.
- Фролова Н. Б.** – канд. физ.-мат. наук, доцент кафедры прикладной информатики и программной инженерии, Международный факультет прикладных информационных технологий, Саратовский государственный технический университет им. Гагарина Ю. А. Тел.: +7-937-226-18-11
E-mail: natalya-fr@yandex.ru
- Frolova N. B.** – Cand. Sc. (Physics and Mathematics), associate professor, Department of Applied Computer Science and Software Engineering, International Faculty of Applied Information Technology, Yuri Gagarin State Technical University of Saratov. Tel.: +7-937-226-18-11
E-mail: natalya-fr@yandex.ru