



УДК 004.822

ЛОГИЧЕСКИЙ ВЫВОД НА ВИЗУАЛЬНЫХ ГРАФОАНАЛИТИЧЕСКИХ УФО-МОДЕЛЯХ ПУТЕМ ИНТЕГРАЦИИ СО СРЕДСТВАМИ ОНТОЛОГИЧЕСКОГО ИНЖИНИРИНГА

THE INFERENCE ON THE VISUAL UFO MODELS BY INTEGRATING WITH THE ONTOLOGICAL ENGINEERING TOOLS

А.А. Кондратенко, С.И. Маторин
A.A. Kondratenko, S.I. Matorin

*Белгородский государственный национальный исследовательский университет,
Россия, 308015, г. Белгород, ул. Победы, 85*

*Belgorod State National Research University,
85, Victory St., Belgorod, 308015, Russia*

E-mail: a.kondratenko-sl@yandex.ru

Аннотация. Статья посвящена методу логического вывода на визуальных графоаналитических моделях, построенных в терминах «Узел-Функция объект». Актуальность задач логического вывода на УФО-моделях обосновывается потребностью в средствах и инструментах машинной обработки визуальных моделей предметных областей при организации управления знаниями. Для решения данных задач разработан новый метод логического вывода на системно-объектных моделях, в основе которого лежит предложенный авторами ранее подход к построению онтологий на основе средств системно-объектного подхода (УФО-подхода). В рамках разработанного метода существующие инструменты логического вывода на онтологиях (специализированные языки запросов и решатели) применяются к формальному описанию онтологии, полученной путем преобразования УФО-моделей. Рассматриваются примеры решения прикладных задач логического вывода на УФО-моделях с использованием языка запросов SPARQL и SWRL-правил.

Resume. The article is devoted to the method of the inference on the visual domain models, which are built in the terms «Unit-function object». The topicality of the inference on UFO models tasks is substantiated by the need of the tools for the visual domain models computer processing in knowledge management. A new method of inference on the system-object models was developed for solve these tasks. This method is based on the authors previously proposed approach to building ontologies based on the system-object approach (UFO approach) tools. In this method tools for the inference on the ontologies (like specialized query languages and reasoners) apply to the formal description of ontology obtained by conversion from the UFO model. The examples of solving inference on the UFO models tasks using the SPARQL query language and SWRL-rules.

Ключевые слова: УФО-подход, онтология, онтологический инжиниринг, RDF, SPARQL, SWRL.
Keywords: UFO approach, ontology, ontology engineering, RDF, SPARQL, SWRL.

Введение

Одним из актуальных направлений современной инженерии знаний является машинная обработка визуальных моделей предметных областей. Методы и средства извлечения и обработки содержащихся в подобных моделях знаний позволяют использовать модели в качестве источников знаний в прикладных информационных системах, решающих задачи поддержки принятия решений в сложных сферах деятельности.

Исследователи предлагают ряд решений задач машинной обработки знаний, содержащихся в визуальных моделях предметных областей, и, в частности, в графоаналитических моделях в терминах «Узел-Функция-Объект» (УФО-моделях). Так, Жихаревым А.Г. [Жихарев, 2013] разработан формализованный метод и алгоритм вывода на графоаналитических УФО-моделях организационных знаний, представляемых средствами СОМПЗ (системно-объектного метода представления знаний). Данная работа ориентирована в основном на формализованное представление организационных знаний.

В то же время, необходимо отметить, что задача извлечения и обработки знаний из УФО-моделей несколько шире, поскольку в терминах системно-объектного подхода может осуществляться моделирование широкого круга предметных областей: от конкретного технологического процесса до сложных систем и целых областей знаний экспертов. Соответственно, актуальной задачей является

поиск универсального метода логического вывода на графоаналитических УФО-моделях, позволяющего осуществлять обработку аккумулированных в них знаний вне зависимости от предметной области и специфики модели.

Взаимосвязь задачи логического вывода на УФО-моделях и метода построения онтологии на их основе

Для решения указанной задачи могут применяться механизмы логического вывода на формальных представлениях онтологий. Это возможно на основании результатов исследований [Слободюк и др., 2013; Кондратенко, Маторин, 2016а; Кондратенко, Маторин, 2016б], в которых предложен метод построения онтологий на основе визуальных графоаналитических УФО-моделей предметных областей. В частности, представление УФО-моделей предметных областей с помощью формализованного языка представления онтологий RDF (Resource Description Framework), описанное в указанных работах, открывает широкие возможности в области машинной обработки УФО-моделей. Формируемое в данном случае формальное представление визуальной графоаналитической модели в виде кода на языке RDF, а также возникших на его основе модификаций, позволяет сделать исходную УФО-модель машиночитаемой. Наличие в языке RDF нескольких нотаций, конвертация между которыми возможна с помощью достаточно распространенных специальных программных средств, позволяет использовать наиболее удобную в том или ином случае. Так, например, нотация N-Triples является более простой для восприятия человеком, в то время как нотация RDF/XML приближена к формату XML, что позволяет использовать ее в информационных системах и специализированных программных средствах, в том числе, с использованием обработки существующими XML-парсерами.

Подобный метод обработки содержащихся в УФО-моделях знаний о предметной области позволяет осуществлять процесс логического вывода на онтологиях, основой которых являются указанные модели. На основе механизмов логического вывода и получения знаний из онтологий, созданных на базе УФО-моделей, могут быть построены интеллектуальные информационные системы, включающие в себя элементы экспертных систем и систем поддержки принятия решений.

Рассмотрим подробнее метод логического вывода на визуальных графоаналитических УФО-моделях за счет их интеграции с формальными средствами записи онтологии. В основе данного метода лежит идея извлечения знаний, содержащихся в любой УФО-модели, преобразовании их к виду, для которого становится возможным применение механизмов и программных средств логического вывода на онтологиях. Составляющими указанного метода являются перечисленные ниже шаги (этапы):

— Извлечение знаний о предметной области из компьютерной графоаналитической системно-объектной модели.

— Представление извлеченных знаний на формальном языке записи онтологий, то есть в виде, пригодном для машинной обработки,

— Применение инструментов логического вывода (языка запросов или решателя) к полученному формальному представлению онтологии, построенной на основе УФО-модели.

Действительно, УФО-модель содержит в себе совокупность знаний о предметной области, которые могут быть представлены в виде отдельных фактов, которые систематизируются с помощью классификации, предложенной в [Слободюк, 2014]. Возможность извлечения подобного набора фактов о предметной области обоснована с помощью соответствующей адаптации средств формализации УФО-подхода [Кондратенко, Маторин, 2016б]. Таким образом, компьютерную графоаналитическую УФО-модель можно считать источником знаний о предметной области.

Помимо знаний, содержащихся в такой модели в явном виде и извлекаемых из нее в процессе построения онтологии, из УФО-модели могут быть выведены новые знания о предметной области. Для их получения (вывода) необходимо применение соответствующих механизмов и алгоритмов. Как правило, подобные механизмы поддаются автоматизации, и на их основе разрабатываются специальные программные средства. Однако применение подобных алгоритмических и программных средств возможно только к формализованному представлению источника знаний, то есть УФО-модели. Поскольку для решения задач логического вывода на УФО-моделях предполагается использовать наработки, существующие для решения аналогичных задач логического вывода на онтологиях, в качестве базы знаний предлагается использовать онтологию, построенную на основе анализируемой УФО-модели предметной области по описанному ранее методу.

Иными словами, предлагаемый метод логического вывода знаний на графоаналитических УФО-моделях основан именно на переходе от анализируемой УФО-модели предметной области к онтологии и применении соответствующих механизмов к полученному формальному описанию построенной онтологии. Опыт мирового исследовательского сообщества позволяет воспользоваться в этом случае лучшими практиками применения онтологий для решения прикладных задач.



Классификация задач логического вывода на УФО-моделях

Определим круг основных задач логического вывода на построенных на основе УФО-моделей онтологиях, которые могут быть полезными при решении проблем в прикладных областях. Такие задачи можно условно сгруппировать по типу данных, на основе которых они строятся. В зависимости от предметной области и цели извлечения знаний формулировка задач может варьироваться.

Далее приведена классификация типовых задач логического вывода на онтологиях, построенных на основе УФО-моделей.

1. Задачи на основе данных о связях.
 - 1.1. Построение классификации связей.
 - 1.2. Вывод перечня подклассов заданного класса связей.
 - 1.3. Получение полной информации о заданном классе связей.
2. Задачи на основе данных об узлах.
 - 2.1. Получение подробной информации о заданном узле.
 - 2.2. Определение функции, балансирующей заданный узел.
 - 2.3. Получение информации об объекте, занимающем данный узел.
 - 2.4. Вывод перечня портов (всех, входящих, исходящих) заданного узла.
 - 2.5. Определение узлов, имеющих сходные порты с заданным.
 - 2.6. Формирование перечня пустых/занятых портов заданного узла.
3. Задачи на основе данных о функциях.
 - 3.1. Вывод подробной информации о заданной функции.
 - 3.2. Построение списка узлов, балансируемых заданной функцией.
 - 3.3. Получение списка объектов, реализующих заданную функцию.
4. Задачи на основе данных об объектах.
 - 4.1. Отображение подробной информации о заданном объекте.
 - 4.2. Вывод списка узлов, занимаемых заданным объектом.
 - 4.3. Получение функций, реализуемых конкретным объектом.
5. Задачи на основе данных об УФО-элементах в целом.
 - 5.1. Формирование перечня входов и выходов заданной функции (на основе данных об узле, его портах и балансирующей его функции).
 - 5.2. Получение информации о связях между двумя заданными УФО-элементами (на основе информации об узлах, их портах и существующих в модели связях).
 - 5.3. Построение цепочек взаимодействия между заданными УФО-элементами (на основе данных об узлах, их портах, существующих в модели связях).

Следует отметить, что данная классификация описывает лишь основные, наиболее общие задачи, которые могут возникнуть при обработке УФО-моделей. В зависимости от конкретной прикладной задачи приведенный список может быть дополнен специфическими примерами задач логического вывода на УФО-моделях. Тем не менее, предложенная классификация позволяет систематизировать основные виды решаемых задач и определить наиболее подходящие для их решения средства и инструменты.

Обзор инструментов логического вывода на онтологиях

Использование онтологий в прикладных информационных системах на сегодняшний день фактически является отдельным исследовательским направлением, в рамках которого разработаны принципы, подходы, алгоритмы, а также основанные на них библиотеки и программные средства, позволяющие осуществлять логический вывод на онтологиях, многие из которых могут быть использованы для осуществления логического вывода на УФО-моделях.

В качестве одного из наиболее распространенных средств обработки знаний, содержащихся в онтологии, можно выделить языки запросов, такие как DQL, N3QL, RDFQ, RDQ, RDQL, SPARQL. Среди них наиболее изученным и широко применяемым является SPARQL (SPARQL Protocol and RDF Query Language) – язык запросов к данным, представленным по модели RDF, а также протокол для передачи этих запросов и ответов на них. Данный язык является рекомендацией консорциума W3C и регламентирован соответствующими документами [SPARQL 1.1 Overview, 2013]. Запросы, формируемые на языке SPARQL, имеют конструкцию, сходную с запросами к базам данных на языке SQL (Structured query language) – языке управления данными в реляционных базах данных. Результаты SPARQL-запросов к RDF-графам могут быть представлены в различных форматах, таких как XML, Json, а также CSV и TSV. Языки запросов к формальному представлению онтологий на RDF успешно применяются для решения задач получения фрагментов явных знаний, содержащихся в онтологии. Так, подобным образом могут быть решены задачи 1.2, 2.2-2.6, 3.2, 3.3, 4.2, 4.3 приведенной ранее классификации задач логического вывода на УФО-моделях.



Более сложные инструменты, предназначенные, в частности, для вывода новых знаний из онтологий, в литературе [Abburu, 2012; Dentler et al., 2011] также называются решателями (reasoner). Решатели для осуществления логического вывода на онтологиях можно условно разделить на следующие типы:

- На основе правил (O-DEVICE).
- На основе резолюций (поиска противоречий): KAON2.
- На основе семантического табло (semantic tableau, табло-алгоритма): FaCT++, Hermit, Pellet.

Среди решателей, базирующиеся на правилах, в свою очередь, выделяют подвиды в зависимости от используемого алгоритма: forward-chaining («прямые цепочки») и/или backward-chaining («обратные цепочки»). Решатели первого подвида основываются на исходных (уже известных) фактах, выводя из них новые. Подобные инструменты используются для выполнения конкретных запросов к имеющимся данным или получения особого, известного типа знаний. Решатели на основе «обратных цепочек» предназначаются для проверки заданного факта на соответствие исходным данным.

Большинство из существующих инструментальных средств – решателей предназначено для работы с онтологиями, представленными на языке OWL. Метод построения онтологий на основе УФО-моделей позволяет получить код в нотации RDF/XML языка RDF, который может быть преобразован в OWL-код. Это, в свою очередь, обеспечивает применение широкого круга инструментальных средств, в частности, предназначенных для работы с OWL решателей, для логического вывода на онтологиях, построенных на базе УФО-моделей.

Рассмотрим некоторые особенности инструментальных средств, предназначенных для осуществления вывода знаний на данных, представленных с помощью форматов OWL. Поскольку язык OWL основан на дескрипционных логиках, в литературе [Horrocks, 2007] приводится следующее соответствие понятий (см. таблицу).

Таблица
Table

Соответствие понятий в дескрипционных логиках и в языке OWL
The accordance between concepts in descriptive logics and OWL language

Понятие в дескрипционных логиках	Понятие в OWL
Концепт	Класс
Роль	Свойство
Индивид	Объект
База знаний	Онтология

Как следует из таблицы, в случае представления на языке OWL и осуществления операций логического вывода онтологию можно принимать за базу знаний. В такой базе знаний, согласно общим положениям дескрипционных логик, выделяют набор терминологических аксиом (TBox) и набор утверждений об индивидах (ABox). TBox содержит данные (аксиомы) следующих видов:

- Аксиома вложенности концептов - выражение вида $C \subseteq D$, где C и D – произвольные концепты.
- Аксиома эквивалентности концептов – выражение вида $C \equiv D$, где C и D – произвольные концепты.
- Аналогично, аксиомой вложенности ролей называется выражение вида $R \subseteq S$, где R и S – произвольные роли.
- Аксиомой эквивалентности ролей – выражение вида $R \equiv S$, где R и S – произвольные роли.

Немаловажным аспектом является поддержка существующими решателями логического вывода в области ABox, то есть вывода знаний об объектах (индивидах). Согласно обзору, приведенному в [Dentler, 2011], такой функционал реализован в инструментах FaCT++, Hermit, RP, TrOWL(REL).

Соответственно, при выборе существующего решателя для прикладных задач логического вывода на онтологиях необходимо учитывать ряд факторов:

- поддерживаемые алгоритмы;
- наличие поддержки решения задач логического вывода в области ABox;
- поддерживаемые дескрипционные логики;
- стоимость решения (бесплатное или коммерческое программное обеспечение);
- используемые и допустимые форматы данных;
- язык программирования, на котором реализован инструмент;
- возможности интеграции в собственное программное обеспечение.



Помимо этого, немаловажным критерием выбора инструментов логического вывода является возможность их модификации. В частности, лежащие в основе УФО-моделей, на базе которых строятся онтологии в предложенном методе, принципы и положения УФО-подхода, а также возможность применения указанного метода для построения онтологий в самых разнообразных предметных областях, обуславливают необходимость дополнительной настройки таких инструментов. Такая настройка заключается в задании дополнительных правил (алгоритмов) логического вывода, учитывающих особенности предметной области и принципы УФО-подхода. Поэтому в прикладных информационных системах требуется использование либо известных существующих, но модифицированных и дополненных инструментов логического вывода, либо собственных, разработанных с учетом указанных особенностей решателей.

С использованием существующих механизмов и инструментария логический вывод на онтологиях, построенных путем специальных преобразований УФО-моделей предметных областей, позволяет решить ряд существенных проблем, присущих эвристической обработке моделей предметных областей. В частности, такой механизм обработки и извлечения знаний из УФО-моделей обладает следующими преимуществами:

1. Отсутствие необходимости ручного формирования исходной базы знаний.

УФО-модели предметной области благодаря простоте УФО-подхода и удобству программных средств, поддерживающих его принципы при моделировании, могут быть построены как отдельным (внешним по отношению к моделируемой среде) специалистом-аналитиком, так и непосредственным экспертом в рассматриваемой области. В любом случае, получаемая в результате УФО-анализа модель предметной области уже содержит в себе знания и представления эксперта о данной предметной области. Поскольку логический вывод производится на онтологии, построенной путем строго определенных преобразований формального представления исходной УФО-модели, такая модель и является исходной базой знаний. В таком случае пропадает необходимость в отдельных процессах формирования первоначальной базы знаний, что позволяет сэкономить время и усилия экспертов.

2. Гибкость и масштабируемость базы знаний

Актуализация имеющихся данных о предметной области может быть выполнена в виде внесения изменений в исходную УФО-модель. При этом внесенные изменения транслируются в создаваемую на основе УФО-модели онтологию, по которой и осуществляется логический вывод необходимых знаний. Соответственно, подобная конструкция обладает простотой модификации и широкими возможностями дополнения исходных данных.

3. Применимость для решения сложных задач анализа моделей предметных областей

В случае анализа сложных предметных областей итоговые УФО-модели могут быть достаточно громоздкими и трудными для восприятия человеком. Зачастую визуальный анализ сложной графоаналитической модели не позволяет быстро и корректно получить необходимые сведения, содержащиеся в такой модели или выводимые на ее основе. Построение на базе УФО-модели онтологии и осуществление логического вывода на ее основе позволяют автоматизировать и существенно упростить анализ сложных объемных моделей предметных областей.

Примеры решения задач логического вывода на УФО-моделях

Рассмотрим способы решения основных видов задач логического вывода на онтологиях, построенных на базе визуальных графоаналитических УФО-моделях предметных областей. Как отмечалось ранее, задачи логического вывода на онтологиях, построенных на основе УФО-моделей, можно условно сгруппировать в соответствии с классификацией.

Часть задач, приведенных в классификации, предлагается решать с помощью специализированного языка запросов SPARQL к формальному представлению онтологии. Язык SPARQL для поиска необходимых триплетов позволяет сформировать запрос, синтаксис которого близок к синтаксису языка SQL.

Так, например, типичным случаем, требующим применения языка SPARQL, является поиск информации об объекте, занимающем заданный узел. Подобную задачу можно представить следующим образом: в коде онтологии на языке RDF требуется найти триплеты вида: «Объект занимает узел», где в качестве объекта выступает конкретный узел исходной УФО-модели. Применительно к описанной задаче поиска объекта, занимающего конкретный узел исходной УФО-модели, строится SPARQL-запрос вида:

```
PREFIX ufo: < http://www.ufo-toolkit.ru/>
SELECT ?o
WHERE
{
  ?o ufo:represents ?u.
}
```

В результате выполнения такого запроса формируется список субъектов предикатов, удовлетворяющих заданному в теле условия запроса «шаблону». Полученные концепты и являются искомыми объектами, занимающими заданный узел UFO-элемента.

Более сложные задачи логического вывода требуют применения гибких инструментов, позволяющих использовать сложные правила (алгоритмы), в том числе специализированные, задаваемые пользователем или включенные в онтологию. В частности, таким инструментом является язык правил семантического веба SWRL (Semantic Web Rule Language). Правила, написанные на SWRL, состоят из консеквента и антецедента, то есть заголовка и тела соответственно. Консеквент является истинным тогда, когда антецедент пуст или все его составляющие истинны. Рассмотрим ряд практических задач, которые могут быть решены с помощью языка правил вывода SWRL, который применяется к формальному представлению онтологии на языке OWL или синтаксически совместимых нотаций языка RDF. Для обеспечения простоты восприятия человеком используем так называемый «человекочитаемый» синтаксис языка SWRL (Human Readable Syntax), который позволяет представлять SWRL-правила в виде: антецедент (тело, условие)⇒консеквент (заголовок, следствие).

— Включение в онтологию фактов вида «Функция преобразует вход...».

В данном случае для получения факта требуется его логический вывод на основе имеющихся данных. В частности, используются сведения о балансировании функцией определенного узла, а также о наличии у узла конкретной входящей связи. Поскольку первый факт может быть включен в онтологию в одном из двух видов («Функция балансирует узел» или «Узел балансируется функцией»), необходимо использовать два SWRL-правила, представленных ниже:

$$\begin{aligned} & \text{balances}(? f, ? u) \wedge \text{hasInput Relation}(? u, ? Li) \Rightarrow \text{translateInput}(? f, ? Li), \\ & \text{isBalancedBy}(? u, ? f) \wedge \text{hasInput Relation}(? u, ? Li) \Rightarrow \text{translateInput}(? f, ? Li). \end{aligned}$$

— Включение в онтологию фактов вида «Функция выдает выход...».

Данная задача аналогична описанной ранее, с тем отличием, что во второй части правила используются факты наличия у узла исходящей связи. SWRL-правила для вывода факта «Функция выдает выход...» показаны ниже.

$$\begin{aligned} & \text{balances}(? f, ? u) \wedge \text{hasOutput Relation}(? u, ? Lj) \Rightarrow \text{giveOutput}(? f, ? Lj), \\ & \text{isBalancedBy}(? u, ? f) \wedge \text{hasOutput Relation}(? u, ? Lj) \Rightarrow \text{giveOutput}(? f, ? Lj). \end{aligned}$$

— Включение в онтологию фактов вида «Объект занимает узел».

В XML-представлении UFO-моделей, формируемом наиболее распространенным программным продуктом для моделирования предметных областей с использованием средств UFO-подхода «UFO Toolkit», наличие объекта как составляющей UFO-элемента описывается в виде «дочерней» по отношению к Функции характеристикой (вложенным XML-узлом). Поэтому при автоматизации преобразования UFO-модели в конструкции языка описания онтологий факт вида «Объект занимает Узел» может выводиться из других, ранее сформированных триплетов онтологии, характеризующих Узел и Функцию. Так, для вывода факта об отношении Объекта и Узла требуется иметь в онтологии факт балансирования Функцией данного Узла и факт реализации Объектом данной Функции. С учетом возможных вариаций записи указанных фактов получим четыре SWRL-правила для вывода описанного факта:

$$\begin{aligned} & \text{realizes}(? o, ? f) \wedge \text{balances}(? f, ? u) \Rightarrow \text{represents}(? o, ? u), \\ & \text{realizes}(? o, ? f) \wedge \text{isBalancedBy}(? u, ? f) \Rightarrow \text{represents}(? o, ? u), \\ & \text{is RealizedBy}(? f, ? o) \wedge \text{balances}(? f, ? u) \Rightarrow \text{represents}(? o, ? u), \\ & \text{is RealizedBy}(? f, ? o) \wedge \text{isBalancedBy}(? u, ? f) \Rightarrow \text{represents}(? o, ? u). \end{aligned}$$

— Учет нескольких уровней декомпозиции (UFO-элемент является частью другого элемента на несколько уровней выше).

В случае, когда сведения о том, что UFO-элемент является частью декомпозиции другого UFO-элемента, находящегося более чем на одном уровне выше, необходимо включить непосредственно в состав онтологии, применяется SWRL-правило вида:

$$\text{isPartOf}(? u_i^{n-2}, ? u_i^{n-1}) \wedge \text{isPartOf}(? u_i^{n-1}, ? u_i^n) \Rightarrow \text{isPartOf}(? u_i^{n-2}, ? u_i^n).$$

— Определение узлов, имеющих сходные порты с заданным.

Такая задача решается с помощью циклической обработки каждого факта наличия входящего или исходящего порта у первого сравниваемого узла. В ходе такой проверки используются правила (для входящих и исходящих портов соответственно):

$$\begin{aligned} & \text{hasInputPort}(? u1, ? L_i) \wedge \text{hasInputPort}(? u2, ? L_i) \Rightarrow \text{hasSamePort}(? u1, ? u2), \\ & \text{hasOutputPort}(? u1, ? L_i) \wedge \text{hasOutputPort}(? u2, ? L_i) \Rightarrow \text{hasSamePort}(? u1, ? u2). \end{aligned}$$

Для поиска узлов, один из которых имеет входящий, а другой – исходящий порт одного типа, применяется следующая SWRL-конструкция:



$$hasInputPort(?u1, ?L_i) \wedge hasOutputPort(?u2, ?L_i) \Rightarrow hasSamePort(?u1, ?u2),$$

$$hasOutputPort(?u1, ?L_i) \wedge hasInputPort(?u2, ?L_i) \Rightarrow hasSamePort(?u1, ?u2).$$

SWRL-правила обеспечивают возможность гибкой настройки логического вывода новых сведений о предметной области на основе уже имеющихся фактов (триплетов). Приведенный перечень практических задач не является исчерпывающим и может быть расширен для каждого конкретного случая. Следовательно, эксперты в процессе развития онтологии предметной области могут дополнять ее новыми фактами, включая их непосредственно в само описание онтологии на формальном языке, что обеспечивает легкость масштабирования и возможность оперативного дополнения онтологий необходимыми «стандартными» знаниями.

Описанные средства могут применяться как по отдельности, так и комплексно. Примером второго варианта может служить логический вывод на знаниях, содержащихся в УФО-модели настройки сложного программного продукта. Данный продукт представляет собой специализированную платформу, предоставляющую инструменты для быстрого создания прикладных web-приложений с учетом специфики предметной области. Специалисты, осуществляющие разработку приложений с помощью данной платформы, сталкиваются с большим объемом слабоструктурированной и, зачастую, практически не задокументированной информации о практическом опыте использования платформы, полезных советах, порядке работы и используемых инструментах платформы. Для упрощения поиска, а иногда и вывода необходимых знаний используется визуальная графоаналитическая УФО-модель, фрагмент которой показан на рис. 1.

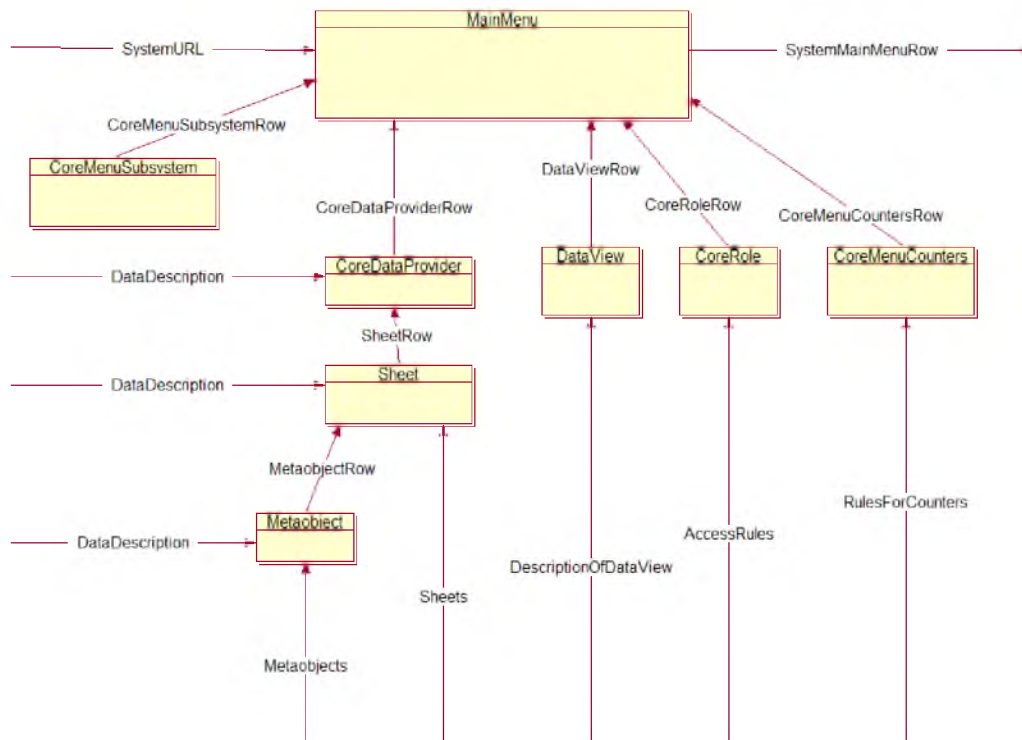


Рис. 1. Фрагмент УФО-модели создания web-приложения средствами специализированной платформы
Fig. 1. Fragment of the UFO model about web application creating within special platform

Зачастую при создании и настройке нового приложения специалисту требуется перечень объектов системы, которые необходимо настроить для реализации желаемого функционала (или перечень объектов, которых коснутся вносимые изменения). Ручной поиск ответа на данный вопрос представляет собой поочередный просмотр настроек ряда потенциально связанных объектов системы и поиск в них информации о значимой в данном контексте связи с другими объектами. Такая процедура требует длительного времени и внимания со стороны специалиста. В то же время к разработанной УФО-модели может быть применен метод преобразования в формализованное описание онтологии на языке RDF, что позволит использовать комплекс инструментов логического вывода на онтологиях. В частности, для решения описанной задачи предполагается выполнение ряда SPARQL-запросов, а также использование SWRL-правил, описанных выше, в комплексе представляющих собой следующий алгоритм (рис. 2).

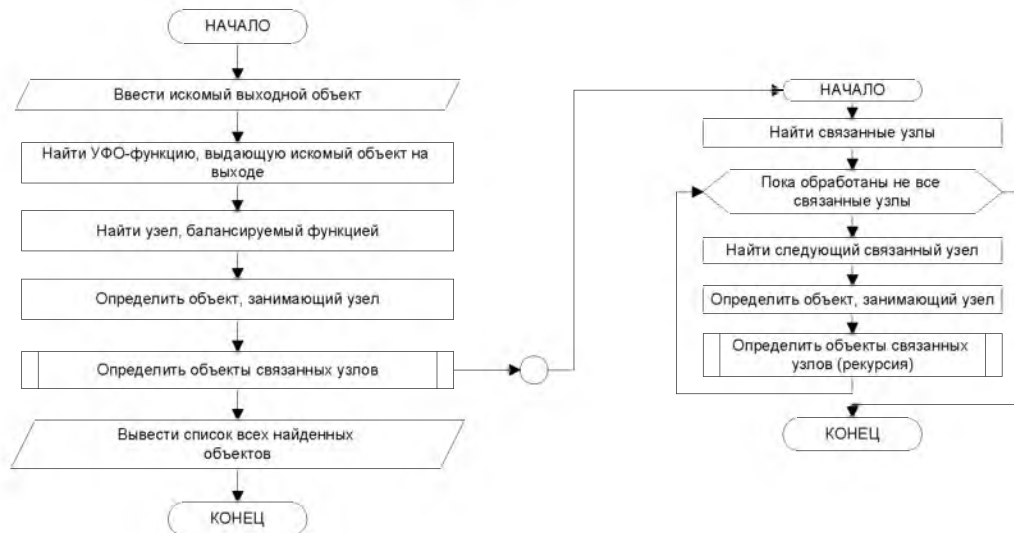


Рис. 2. Блок-схема алгоритма поиска объектов платформы, используемых при реализации функциональной возможности приложения
 Fig. 2. The block diagram of the algorithm for searching platform objects, which are used for application feature realization

Заключение

Таким образом, метод построения онтологий с использованием в качестве источника знаний о предметной области визуальных графоаналитических системно-объектных моделей позволяет также решать задачи логического вывода на подобных моделях. В частности, преобразование УФО-модели в формализованное представление онтологии позволяет применить к полученному описанию существующие методы и инструменты логического вывода на онтологиях, такие как язык запросов SPARQL и язык правил SWRL. Данные средства, а также поддерживающие их программные средства могут успешно применяться для решения прикладных задач логического вывода на УФО-моделях, что подтверждают приведенные примеры решения задач обработки и вывода знаний на основе визуальной графоаналитической модели настройки сложного специализированного программного продукта.

Список литературы References

Жихарев А.Г. 2013. Формализованное графоаналитическое представление организационных знаний. Автореферат дис. ... кандидата технических наук. Белгород, 22 с.
 Zhiharev A.G. 2013. Formalizovannoe grafoanaliticheskoe predstavlenie organizacionnyh znaniy. Abstract. dis...cand.tech.sciences. Belgorod, 22. (in Russian)

Кондратенко А.А., Маторин С.И. 2016. Построение онтологий на основе моделей «Узел-Функция-Объект». Искусственный интеллект и принятие решений. (1): 47-56.
 Kondratenko A.A., Matorin S.I. 2016. Postroenie ontologij na osnove modelej «Uzel-Funkcija-Ob'ekt». Iskusstvennyj intellekt i prinjatie reshenij. (vol.1): 47-56. (in Russian)

Кондратенко А.А., Маторин С.И. 2016. Формальные аспекты взаимосвязи УФО-подхода и языка представления онтологий RDF. Научные ведомости БелГУ. Серия Экономика. Информатика. 2 (223) 2016, Выпуск 37: 119-127.
 Kondratenko A.A., Matorin S.I. 2016. Formal'nye aspekty vzaimosvjazi UFO-podhoda i jazyka predstavlenija ontologij RDF. Nauchnye vedomosti BelGU. Ser. Jekonomika. Informatika [Belgorod State University Scientific Bulletin. Economics Information technologies]. 2 (223) 2016, vol. 37: 119-127. (in Russian)

Слободюк А.А., Маторин С.И., Четвериков С.Н. 2013. О подходе к созданию онтологий на основе системно-объектных моделей предметной области. Научные ведомости БелГУ. Серия История. Политология. Экономика. Информатика. 22 (165) 2013, Выпуск 28/1: 186-195.
 Slobodyuk A.A., Matorin S.I., Chetverikov S.N. 2013. About approach for building ontologies based on UFO domain models. Nauchnye vedomosti BelGU. Ser. Istorija. Politologija. Jekonomika. Informatika [Belgorod State University Scientific Bulletin. History Political science Economics Information technologies]. 22 (165) 2013, vol. 28/1: 186-195. (in Russian)

Слободюк, А.А. 2014. Расширенная классификация фактов, извлекаемых из УФО-модели в целях построения онтологии предметной области. В кн.: Шестая международная научно-техническая конференция «Инфокоммуникационные технологии в науке, производстве и образовании» (Инфоком-6): сборник научных



трудов, Ставрополь, 21-27 апреля 2014 г.: часть II. Ставрополь, Северо-Кавказский федеральный университет: 404-406.

Slobodyuk A.A. 2014. Detailed classification of facts which are extracted from UFO domain model for building ontology. In: VI mezhdunarodnaja nauchno-tehnicheskaja konferencija «Infokommunikacionnye tehnologii v nauke, proizvodstve i obrazovanii» (Infokom-6): sbornik nauchnyh trudov, Stavropol', 21-27 of April 2014: p. II. Stavropol', Severo-Kavkazskij federal'nyj universitet: 404-406. (in Russian)

Abburu Sunitha. 2012. A Survey on Ontology Reasoners and Comparison. International Journal of Computer Applications. 57(17): 33-39.

Dentler Kathrin, Cornet Ronald, ten Teije Annette, de Keizer Nicolette. 2011. Comparison of Reasoners for Large Ontologies in the OWL 2 EL Profile. Semantic Web Journal. (2011): 1-5.

Horrocks Ian, Patel-Schneider Peter F., McGuinness Deborah L., Welty Christopher A. 2007. OWL: a Description Logic Based Ontology Language for the Semantic Web. In: Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors, The Description Logic Handbook: Theory, Implementation, and Applications (2nd Edition). Cambridge, Cambridge University Press: 14.

SPARQL 1.1 Overview. W3C Recommendation Recommendation 21 March 2013. Available at: <https://www.w3.org/TR/sparql11-overview/> (accessed 11 April 2016).