

УДК 004.318

Петренко Н.Г.^а, Зеленцов Д.Г.^б

О ПРАКТИЧЕСКОМ ИСПОЛЬЗОВАНИИ ОНТОЛОГИЧЕСКИХ МОДЕЛЕЙ ПРЕДМЕТНЫХ ОБЛАСТЕЙ

^а Институт кибернетики имени В.М. Глушкова НАН Украины, г. Киев, Украина^б ГВУЗ «Украинский государственный химико-технологический университет», г. Днепр, Украина

В статье рассмотрены наиболее известные инструменты формального описания онтологий предметных областей, проведен их сравнительный анализ и выбор формального языка и онторедатора для практического использования. В работе приводятся два варианта трактовки термина «онтология». Предлагается его рассмотреть с точки зрения онтологического инжиниринга. Подробно рассмотрены три составляющие определения термина «онтология» — концептуализация, формальная, явная спецификация. Выделены проблемы выразительности языков описания онтологий. Приведены различные подходы к выбору формального языка для обозначения элементов концептуализации. В качестве примера системы выбрана некоторая производственная компания с ее производственными объектами, сотрудниками и производственными взаимоотношениями между ними. Исследование проводится для совокупности задач, связанных с человеческими ресурсами компании. Приводится CRF-модель (concept–relation–interpretation function model) онтологии предметной области. Для расширенного примера строительной компании представлены сам онтограф и индивиды классов. Проведено сравнение наиболее известных логических теорий описания онтологий. Рассмотрены особенности различных моделей построения онтологий предметных областей, таких как CRF, Frame и OWL. Отмечены их преимущества и недостатки. Сделан вывод, что при разработке больших онтологий следует применять все три указанные модели: на начальном этапе — CRF-модель для автоматизации построения структуры концептов предметной области и их семантической согласованности; на этапе ввода и формализации — модель Protégé-фреймов; на этапе использования онтологии — OWL-модель с механизмами запросов и рассуждений. При этом совместное использование онтологии предполагает ее размещение в Semantic Web. Сравнительный анализ показал, что перспективным направлением является построение веб-онтологий в среде Protégé. Для предметной области «Строительная компания» в статье приведен пример построения и использования OWL-онтологии в интегрированной среде онторедатора Protégé, механизма рассуждений Pellet и SPARQL-процессора. с соответствующими к ней запросами.

Ключевые слова: дескрипционная логика, онтология, модель онтологии, концептуализация, онтограф, Protégé.

DOI: 10.32434/2521-6406-2019-6-2-58-73

Постановка проблемы

Для термина «онтология» распространены являются два смысла: 1) в философском смысле онтология изучает природу и структуру вещей как таковых, независимо от их фактического существования; 2) второй смысл наиболее распространен в информатике как осо-

бый вид информационного объекта. Важность действительного существования сущностей (рассматриваемых в этой онтологии) прагматична [1–3]. Онтологии в онтологическом инжиниринге являются средством формального моделирования структуры системы, т.е. соответствующих объектов или субъектов и отношений

между ними, принадлежащих соответствующей предметной области (ПдО) и используются для решения практических задач [4].

Далее термин «онтология» будет рассматриваться во втором смысле, с точки зрения онтологического инжиниринга.

Примером такой системы выбрано некоторую производственную компанию с ее производственными объектами, сотрудниками и производственными взаимоотношениями между ними. Инженер по знаниям анализирует соответствующие сущности и классифицирует их на понятия и отношения, представленные соответственно унарными и бинарными (в общем случае n -арными) предикатами. Базовая структура онтологии основана на процедурах обобщения/специализации иерархии понятий, т.е. таксономии. Предположим, что нас интересует совокупность задач, связанных с человеческими ресурсами компании. Тогда сущности Сотрудник, Мастер и Рабочий являются релевантными для решения совокупности практических задач, связанных с человеческими ресурсами компании. Первое понятие является суперпонятием (обобщающим понятием, родовым понятием) для остальных двух понятий. Отношение «работает-с» является релевантным отношением между сотрудниками (не только рабочими). Тогда конкретный человек, работающий в компании, будет экземпляром (индивидом) соответствующего понятия.

Цель статьи

Целью данной работы является рассмотрение наиболее известных инструментов формального описания онтологий ПдО, их сравнительный анализ и выбор формального языка и онто редактора для практического использования.

Анализ последних исследований и публикаций

Изначальное определение термина «онтология» [1] претерпело ряд изменений, и в конечном итоге в сообществе онтологов пришли к следующему: «Онтология – это формальная, явная спецификация общей (коллективно используемой) концептуализации» [2,3,5,6]. Далее кратко рассмотрим три составляющие приведенного определения (концептуализация, формальная, явная спецификация и почему общая (коллективно используемая) концептуализация так важна) с учетом того, что они подробно обсуждались в [2,3,5–8].

Изложение основного материала

Концептуализация

В [2,3] дано следующее определение понятия «концептуализация», в котором утвержда-

ется: «Совокупность формально представленных знаний ПдО основана на концептуализации: объектах, понятиях и других сущностях, которые предположительно существуют в некоторой рассматриваемой ПдО, и отношений между ними. Концептуализация – это абстрактное, упрощенное представление о мире, которое рассматривается с некоторой целью. Каждая база знаний, знание-ориентированная система или знание-ориентированный агент придерживаются определенной концептуализации, явно или неявно». Несмотря на сложную ментальную природу понятия «концептуализация», в [8] его определение основано на простом математическом представлении: экстенциональной реляционной структуре.

Экстенциональная реляционная структура (или концептуализация согласно [8]) является кортежем (D,R) , где D – множество понятий X предметной области, названное областью определения; R – множество отношений на D .

Заметим, что в этом определении элементами множества R являются обычные математические отношения на D , т.е. множества упорядоченных кортежей элементов из D . Таким образом, каждый элемент R является экстенциональным отношением, отражающим конкретное состояние мира с элементами из D .

Несмотря на простоту определения экстенционального понятия «концептуализация», в действительности она слабо подходит для решения практических задач, главным образом потому, что сильно зависима от конкретного состояния ПдО.

Суть в том, что экстенциональные отношения, принадлежащие R , отражают конкретное состояние фрагмента ПдО. Следовательно, необходимо сосредоточиться на смысловом значении основополагающих понятий, не зависящих от конкретного состояния ПдО (или согласно Крипке, возможного мира): например, значение отношения «работает-с» заключается в том, как взаимодействуют два конкретных рабочих. На практике понимание такого значения подразумевает наличие правил для принятия решений, соблюдение различных моделей поведения, независимо от того, работают ли эти два человека или нет.

Допустим, в примере для двух рабочих отношение «работать-с» означает, что:

- оба заявляют, что имеют одну и ту же цель;
- оба делают что-то для достижения этой цели.

Тогда значение «работать-с» можно определить как функцию, которая для каждого глобального поведенческого контекста, включающего всю область определения, дает список пар рабочих, которые фактически работают вместе в этом контексте. Обратное действие этой функции обосновывает значение понятия в конкретном состоянии ПдО. Обобщая этот подход и абстрагируясь от времени ради простоты, допустим, что интенциональное отношение (или концептуальное отношение согласно [9]) является функцией из множества максимальных состояний ПдО (глобальные поведенческие контексты в нашем случае) в экстенциональные отношения. Это общий способ выражения содержания понятий, который восходит к Карнапу [10], принят и расширен в семантике Монтегю [11].

В [7] представлено следующее определение интенционального или концептуального отношения. Пусть S – произвольная система, D – произвольное множество различных элементов в S , W – множество состояний ПдО для S . Кортеж $\langle D, W \rangle$ называется областью определения для S , поскольку он интуитивно фиксирует пространство изменений области рассуждения в D относительно возможных состояний S . Интенциональное отношение (или концептуальное отношение) ρ^n арности n на $\langle D, W \rangle$ – это полная функция $\rho^n: W \rightarrow 2^D$ из множества W во множество всех n -арных (экстенциональных) отношений на D .

Определив концептуальное отношение, далее представлено определение общей концептуализации (структуры интенциональных отношений).

Структура интенциональных отношений (или общая концептуализация, далее просто концептуализация) представляет собой тройку $C = (D, W, \mathfrak{R})$, где D – область дискурса; W – множество возможных миров; \mathfrak{R} – множество концептуальных отношений на доменном пространстве $\langle D, W \rangle$.

Формальные, явные спецификации

В практических приложениях необходимо использовать язык для обозначения элементов концептуализации. В этом случае говорят, что язык L фиксирует концептуализацию. Далее необходимо в явном виде определить концептуализацию и здесь проявляется роль онтологий как «явных спецификаций концептуализации». В принципе, такую концептуализацию можно четко определить двумя способами: экстенционально и интенционально. В данном примере экстенциональное определение концеп-

туализации потребует перечисления расширений каждого (концептуального) отношения для всех возможных миров. В общем случае, более эффективный способ определения концептуализации состоит в том, чтобы зафиксировать язык и ограничить интерпретации такого языка интенционально, с помощью подходящих аксиом (в [10] названных смысловыми постулатами). Например, можно принять простые аксиомы, утверждающие, что отношение подчиняется – асимметричное и нетранзитивное, а отношение «работает-с» симметричное, неререфлексивное и нетранзитивное. И, онтология – это набор таких аксиом, то есть логическая теория, разработанная для выбора предполагаемых моделей, которые соответствуют определенной концептуализации, и исключить непредполагаемые модели. Результатом будет приблизительная спецификация концептуализации: будут отобраны лучшие предполагаемые модели и исключены непредполагаемые модели.

Аксиомы для интенционального и экстенционального специфицирования концептуализации могут быть представлены на неформальном или формальном языке L . Как требуется в определении онтологии (и согласно [2,5,6]), явная спецификация должна быть формальной. Термин «формальный» предполагает, что выражения на этом языке должны быть машиночитаемыми, поэтому естественный язык исключается. Далее следует представить все вышеизложенные понятия более формальным способом.

Пусть язык L является (вариантом) логического языка первого порядка, причем словарь V состоит из множества констант и предикатных символов (функциональные символы не рассматриваются). Введем понятие онтологического связывания (англ. ontological commitment), расширив стандартное понятие (экстенциональной) структуры первого порядка до интенциональной структуры первого порядка.

Онтологическое связывание для L представляется кортежем $K = (C, \hat{I})$, где \hat{I} (интенциональная функция интерпретации) – полная функция $\hat{I}: V \rightarrow D \cup \mathfrak{R}$, которая отображает каждый словарный символ из V либо в элемент D , либо в интенциональное отношение, принадлежащее множеству \mathfrak{R} .

Теперь видно, что определение онтологического связывания расширяет обычное (экстенциональное) определение «смысла» для словарных символов в интенциональном случае, заменяя понятие модели понятием концептуализации.

Как было показано, понятие онтологическое связывание является расширением стандартного понятия модели. Последнее понятие – экстенциональная оценка смысла, первое – интенциональная оценка смысла. Следует выяснить, какая взаимосвязь между ними? Конечно, как только будет определен интенциональный смысл словаря через его онтологическое связывание, также будут некоторым образом ограничены и его модели. Введем понятие предполагаемой модели в отношении определенно-онтологического связывания.

Пусть $C=(D,W,\mathfrak{R})$ – концептуализация, L – логический язык первого порядка со словарем V и онтологическим связыванием $K=(C,\hat{I})$. Модель $M=(S,I)$, где $S=(D,R)$, называется предполагаемой моделью L на K , если:

1. Для всех константных символов $c \in V$ имеем $I(c)=\hat{I}(c)$.

2. Существует такой мир $w \in W$, что для каждого предикатного символа $v \in V$ существует такое интенциональное отношение $\rho \in \mathfrak{R}$, что $\hat{I}(v)=\rho$ и $I(v)=\rho(w)$. Множество $I_K(L)$ всех моделей L , совместимых с K , называется множеством предполагаемых моделей L на K .

Вышеприведенное условие 1 требует, чтобы отображение константных символов в элементы области дискурса было идентичным. Условие 2 утверждает, что должен существовать такой мир, что каждый предикатный символ отображается в интенциональное отношение, значение которого для этого мира совпадает с экстенциональной интерпретацией такого символа. Это означает, что наша предполагаемая модель будет описанием этого мира. Определив понятие предполагаемых моделей, можно уточнить роль онтологии, рассматриваемой как логическая теория и предназначенной для учета предполагаемого значения словаря, использующего логический язык.

Онтология

Пусть C – концептуализация, L – логический язык со словарем V и онтологическим связыванием K . Онтология O_K для C со словарем V и онтологическим связыванием K является логической теорией, состоящей из набора формул из L , построенных таким образом, что множество его моделей приближается как можно точнее к множеству предполагаемых моделей L по K .

Исходя из изложенного выше, можно сделать вывод, что «идеальная» онтология – это такая онтология, чьи модели точно совпадают (по модулю изоморфизмов) с предполагаемыми

моделями. Однако не все так просто: даже «прекрасная» онтология может не точно определить свою целевую концептуализацию, если ее словарь и область дискурса не подобраны соответствующим образом. Причина заключается в различии между логическим понятием модели и онтологическим понятием возможного мира. Первое понятие представляет собой комбинацию присваиваний из абстрактных реляционных структур (построенных на области дискурса) с элементами словаря; второе понятие представляет собой комбинацию действительных (наблюдаемых) состояний ПдО конкретной системы. Конечно, количество возможных моделей зависит как от размера словаря, так и от расширения области дискурса, которые выбираются более или менее произвольно, исходя из того, что, по-видимому, имеет отношение к целевому назначению. Напротив, число состояний ПдО зависит от наблюдаемых переменных, даже тех, которые на первый взгляд считаются не имеющими отношения к целевому назначению. Следует подчеркнуть, что степень, с которой онтология определяет концептуализацию, зависит: (1) от объема области дискурса; (2) от представительного множества терминов, вошедших в словарь; (3) от аксиоматизации. В свою очередь, аксиоматизация зависит от языковой выразительности.

Совместное (коллективное) использование онтологий

Первые определения «онтологий» не рассматривали аспект совместного использования онтологий [3,12] и только позднее он был введен в [5]. Действительно, можно с уверенностью утверждать, что невозможно сделать совместно используемыми концептуализации, являющиеся частными случаями для каждого индивида.

Что можно сделать совместно используемым для приближений концептуализаций, основанных на ограниченном множестве примеров, и показывающих фактические состояния, при которых имеют место определенные концептуальные отношения (например, фактические ситуации, показывающие случаи, когда возникает отношение «работает-с»). Помимо простых примеров также можно сделать совместно используемыми смысловые постулаты, т.е. явные формальные ограничения. Однако такие определения предполагают взаимное согласие на примитивные термины, используемые в этих определениях. Поскольку смысловые постулаты не могут в полной мере характеризовать он-

тологическое связывание с помощью примитивных терминов, следует учесть, что обобщение концептуализаций в лучшем случае является частичным. Для практического использования онтологий понятно, что без минимального общего онтологического связывания (совместного использования, например, веб-онтологий) от заинтересованных в онтологии сторон, преимущества онтологии ограничены. Причина в том, что онтология формально определяет доменную структуру при ограничении, что разработчик правильно понимает примитивные термины. Другими словами, онтология может оказаться бесполезной, если она используется таким образом, что противоречит общему онтологическому связыванию. Любая онтология всегда будет менее полной и менее формальной, чем возможно теоретически. Вот почему важно, чтобы онтологии, предназначенные для поддержки крупных проектов, были обоснованными, в том смысле, что основные примитивы, на которых они построены, были бы достаточно хорошо выбраны, аксиоматизированы и понятны для понимания.

CRF-модель онтологии ПдО

На практике важным моментом представления и верификации онтологий является визуализация множеств (D, V) в виде онтологического графа (онтографа). Последний, в самом простом случае, представляется графом в виде дерева и описывает иерархическую таксономию понятий и отношений между ними. Онтограф имеет корневую вершину, которой обычно присваивается имя и значение исследуемой ПдО и «нулевой» уровень иерархии. Нижележащие уровни понятий формируются в соответствии с принятой таксономией. Более подробные рекомендации по структурированию онтографов приведены в [13].

В этом случае онтология представляется известной моделью в виде тройки:

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

где X – множество понятий (терминов) на D ; R – множество концептуальных отношений, связывающих между собой концепты X ; F – множество функций интерпретации на X и/или R (аксиоматизация ПдО и словаря на языке некоторой теории первого порядка). По сути, множества (X, R) определяют структуру онтографа. Назовем модель (1) CRF (concept–relation–interpretation function) моделью.

Для расширенного примера строительной

компании онтограф и индивиды классов представлены на рис. 1 (сам онтограф показан выше утолщенной линии на уровнях 0–4). Понятия X описаны на естественном языке, расшифровка отношений R показана на рисунке вверху справа, а функции интерпретации F для понятий и отношений представляются интуитивно понятными. Корневой вершиной в онтографе является концепт «Строительная компания». Ей присвоен «нулевой» уровень в иерархии концептов. Человеческие ресурсы компании обозначены концептами «Дирекция», учета сотрудников «Отдел кадров» и начисления заработной платы «Бухгалтерия». Основная производственная деятельность компании обозначена концептом «Производство». Им присвоен «первый» уровень в иерархии. И так далее. Подчеркнем, что в CRF-модели вершины онтографа с именами «Директор», «Прораб», «Мастер» и «Рабочий» являются понятиями, а конкретные люди на этих должностях уже относятся к базе данных компании.

Следует отметить, что в CRF-модели отношение в онтографе (например, R_1) может связывать вершину-родителя с несколькими вершинами-потомками. Предполагается по умолчанию, что вершина-родитель является целым для частей вершин-потомков (для отношения R_1 – целое-часть). Такое утверждение верно для рассматриваемого примера, пока не определен конкретный (формальный) язык описания онтологий. В OWL-языке отношения (Object Property) являются бинарными, связывающими между собой индивидов двух классов, поэтому в OWL-описаниях и онторедакторе Protégé их все необходимо указывать в явном виде, что и продемонстрировано на рис. 3.

Ниже онтографа («утолщенной» линии) указаны конкретные сотрудники компании на текущий момент времени, что в CRF-модели не рассматривается и соответствует базе данных, а в OWL-модели – значениям данных индивидов.

Сформулируем определение онтологии с учетом модели (1). Компьютерная онтология ПдО – это:

- иерархическая структура конечного множества понятий, описывающих заданную ПдО;
- структура представляет собой онтограф, вершинами которого являются понятия, а дугами – семантические (концептуальные) отношения между ними;
- понятия и отношения интерпретируются в соответствии с общезначимыми функциями интерпретации, взятыми из электронных

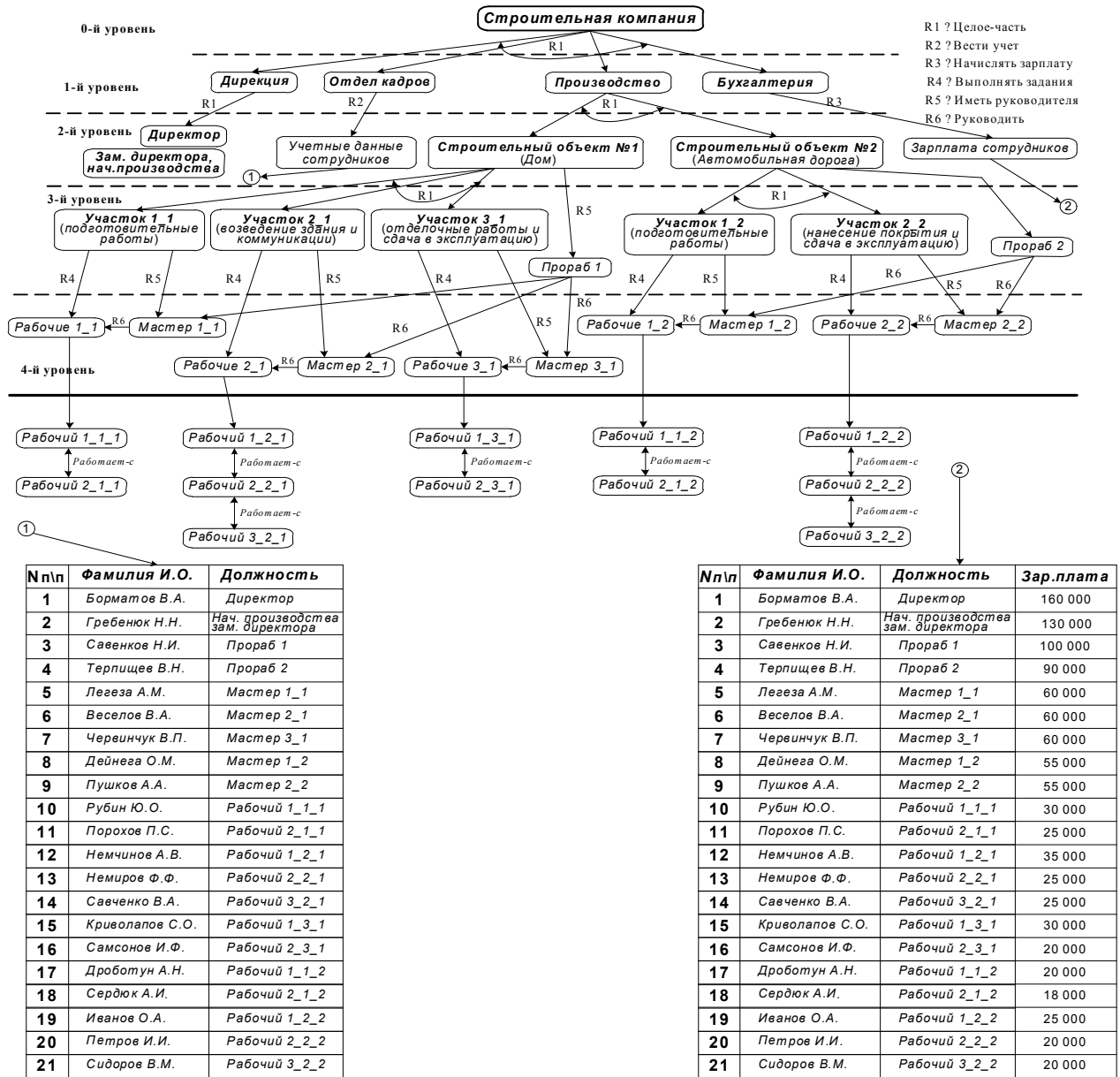


Рис. 1 Онтограф с индивидами классов «Строительная компания»

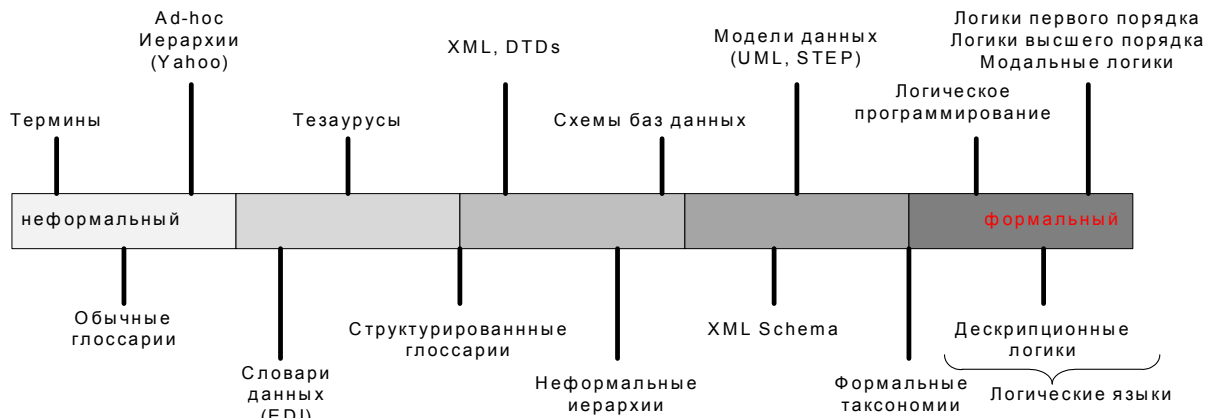


Рис. 2 Различные подходы к выбору языка L

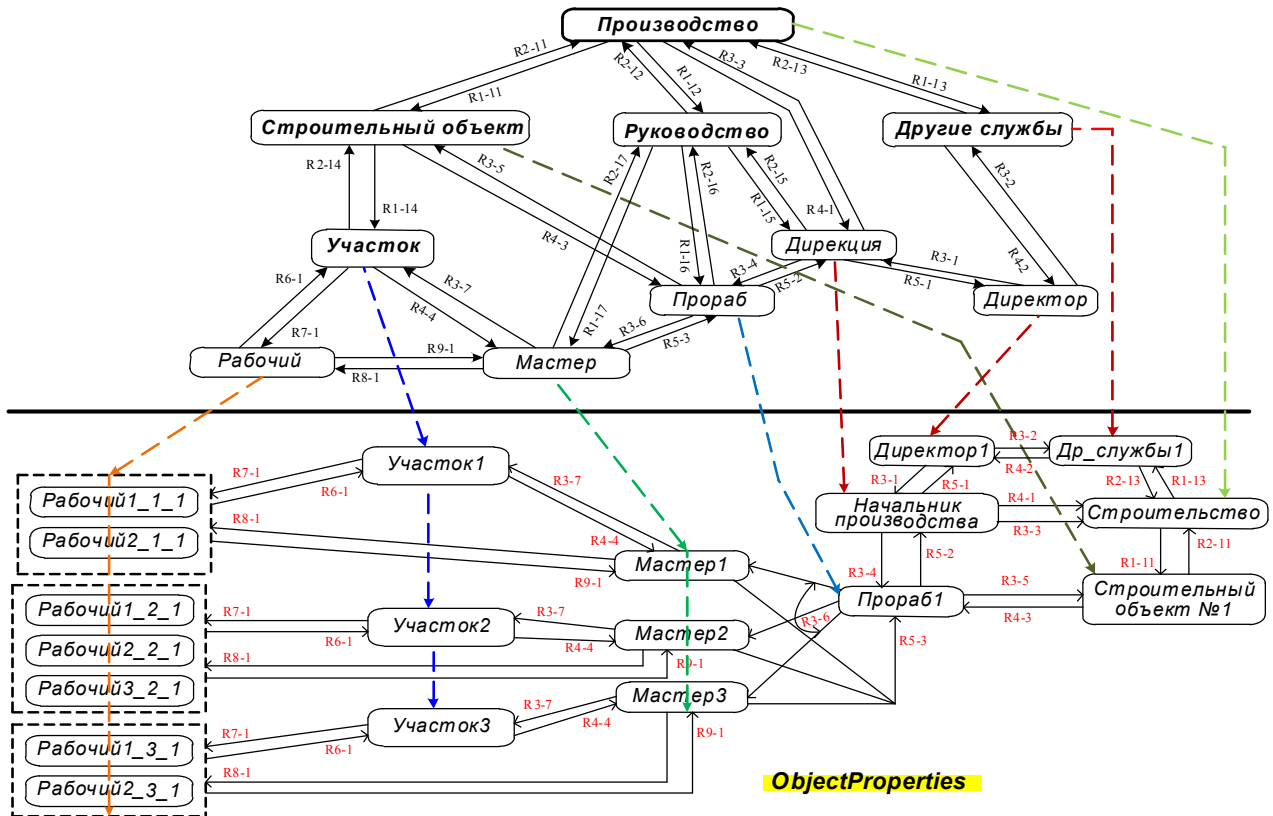
источников знаний заданной ПдО;

- формально онтограф описывается на одном из языков описания онтологий;
- функции интерпретации описаны в некоторой подходящей формальной теории.

Проблемы выразительности языков описания онтологий

С одной стороны, известны неформальные

подходы к языку L, которые допускают только определения терминов с их спецификацией. С другой стороны, известны формальные подходы, т.е. логические языки, позволяющие задавать строго формализованные логические теории. Эти противоположные подходы приводят к континууму, введенному в [14] и представленному на рис. 2. Четкой границы, где начинается



№п/п	Фамилия И.О.	Должность	Оклад
1	Бормотов В.О.	Директор	160 000
2	Гребенюк Н.Н.	Нач. производства	130 000
3	Савенков Н.И.	Прораб1	100 000
4	Легеза А.М.	Мастер1_1	60 000
5	Веселов В.А.	Мастер2_1	60 000
6	Червинчук В.П.	Мастер3_1	60 000
7	Рубин Ю.О.	Рабочий1_1_1	30 000
8	Порохов П.С.	Рабочий2_1_1	25 000
9	Немчинов А.В.	Рабочий1_2_1	35 000
10	Немиров Ф.Ф.	Рабочий2_2_1	25 000
11	Савченко В.А.	Рабочий3_2_1	25 000
12	Криволапов С.О.	Рабочий1_3_1	30 000
13	Самсонов И.Ф.	Рабочий2_3_1	20 000

ObjectProperties

- R1 ?Целое-часть**
 - R1_11 ?Целое_часть111
 - R1_12 ?Целое_часть112
 - R1_13 ?Целое_часть113
 - R1_14 ?Целое_часть114
 - R1_15 ?Целое_часть115
 - R1_16 ?Целое_часть116
 - R1_17 ?Целое_часть117
- R2 ?Часть-целое**
 - R2_11 ?Часть_целое211
 - R2_12 ?Часть_целое212
 - R2_13 ?Часть_целое213
 - R2_14 ?Часть_целое214
 - R2_15 ?Часть_целое215
 - R2_16 ?Часть_целое216
 - R2_17 ?Часть_целое217
- R3 ?Руководить**
 - R3_1 ?Руководить1
 - R3_2 ?Руководить2
 - R3_3 ?Руководить3
 - R3_4 ?Руководить4
 - R3_5 ?Руководить5
 - R3_6 ?Руководить6
 - R3_7 ?Руководить7
- R4 ?Иметь_руководителя**
 - R4_1 ?Иметь_руководство1
 - R4_2 ?Иметь_руководителя2
 - R4_3 ?Иметь_руководителя3
 - R4_4 ?Иметь_руководителя4
- R5 ?Подчиняться**
 - R5_1 ?Подчиняться1
 - R5_2 ?Подчиняться2
 - R5_3 ?Подчиняться3

DataProperties

- R1 ?Должность**
- R2 ?Оклад**
- R3 ?ФамилияИО**
- R6 ?Работать_на**
 - R6_1 ?Работать_на1
- R7 ?Иметь_рабочего**
 - R7_1 ?Иметь_рабочего1
- R8 ?Давать_задания**
 - R8_1 ?Давать_задания1
- R9 ?Выполнять_задания**
 - R9_1 ?Выполнять_задания1

Рис. 3. Protégé-совместимый онтограф

критерий формальности в этом континууме, не существует. На практике самая правая категория логических языков (рис. 2) обычно рассматривается как формальная. В этой категории при выборе языка L рассматривают компромисс между выразительностью и эффективностью вычислений. С правой стороны континуума расположены: логики высшего порядка, полная логика первого порядка и модальная логика. Они достаточно выразительны, но не всегда допускают полных логических выводов из-за своей неразрешимости. На другом конце континуума расположены менее строгие подмножества логики первого порядка, обычно допускающие разрешимые и более эффективные логические выводы. Логика с правой стороны можно разделить на две основные группы.

В первую группу входят языки из семейства дескрипционных логик (DL), являющиеся строгими подмножествами логики первого порядка. Вторая группа исходит из традиции логического программирования (LP) [15], известным представителем которого является F-логика. Несмотря на то, что логическое программирование часто использует синтаксис, сопоставимый с логикой первого порядка, оно предпо-

лагает другую интерпретацию формул. В отличие от теории моделей Тарского [16], основанной на логиках первого порядка и дескрипционных логиках, логическое программирование описывает только подмножество моделей для суждений о семантическом вхождении формул. Существуют разные способы выбора подмножеств моделей, приводящие к различной семантике, – все они направлены на более эффективную работу с большими массивами данных, чем подходы, основанные на логике первого порядка. Одним из наиболее заметных различий, возникающих в результате этого разного стиля логических моделей, является то, что теории выразительного логического программирования становятся немонотонными.

Сравнение известных логических теорий описания онтологий

Ниже представлена таблица сравнительных характеристик наиболее известных логических теорий описания онтологий.

Из таблицы видно, что:

1. CRF-модель онтологии описывает концептуальную структуру знаний ПдО, включающую иерархию понятий (X,R) и определения на естественном языке. Методология разработки

Наименование компонент	Модели онтологии ПдО			DL (дескрипционная логика)	
	CRF	Frame (Protégé)	OWL(OWL2)		
Объект/субъект (о/с)	Понятие/термин	Класс (концепт)	Класс (концепт)	Концепт (одноместный предикат)	
Связь между о/с	Семантическое (интенциональное) отношение	Слот	Свойство (бинарное отношение)	Роль (двухместный предикат), RBox	
Аксиомы описания о/с	Функции интерпретации	Аннотации	Аннотации	Набор терминологических аксиом	
Данные	–	Экземпляр	Индивид	Индивид	
Связи данных	–	Свойство типа экземпляра	Свойство типа данных/ характеристики	Набор утверждений об индивидах	
Значения данных	–	Данные	Данные	Данные	
Машина логического вывода	–	+	+	+	
Инструментарий для построения и визуализации онтографов	–	+	–	–	
Конструкторы	–	–	Пересечение (\cap) Объединение (\cup) Дополнение (\neg) Ограничения (\forall, \exists)	Пересечение (\cap) Объединение (\cup) Дополнение (\neg) Ограничения (\forall, \exists) и другие	
База знаний	–	+	+	+	

этой модели онтологии подробно описана в [13]. Следует отметить, что CRF-модель знаний ПДО не нашла широкого применения в приложениях пользователей, прежде всего из-за отсутствия средств интеграции с базами данных и апробированной формальной логической теории описания модели. Однако для CRF-модели хорошо проработан подготовительный этап ее построения. Известен инструментарий автоматизированного построения множеств понятий X , отношений R и функций интерпретации F .

2. Frame-модель онтологии ПДО и ее Protégé-реализация наиболее широко применяется в сообществе онтологического инжиниринга. Начиная с четвертой версии Protégé включает (и может подключить дополнительно в качестве plugin) самый разнообразный инструментарий манипулирования не только компонентами модели онтологии, но и проверки (ризонером) согласованности разработанной онтологии, ее визуализации (OntoGraph, OntoViz), размещения как на локальном компьютере (файл с расширениями owl или xml), так и в Semantic Web для совместного использования (Shared using), причем в полном соответствии со стандартами и рекомендациями консорциума W3C. Инструментарий Protégé распространяется с открытым программным кодом. Известен и другой аналогичный инструмент TopQuadrant's commercial TopBraid Composer разработки онтологических приложений функционально более эффективный, чем Protégé. Недостатком методологии создания как этой, так и последующей модели онтологии является отсутствие какой-либо поддержки подготовительного этапа формирования онтологий. Как следствие, инженер по знаниям субъективно формирует компоненты модели, а также вручную создает их структуру и вводит описания. В лучшем случае – при наличии онторедатора, в худшем – созданием вручную текстового файла с большим количеством RDF-троек.

3. Язык OWL позволяет описывать классы (концепты) и предоставляет дополнительные возможности по сравнению с Protégé. У него более богатый набор логических операторов и он основан на несколько другой логической модели. Кроме того, последняя позволяет использовать механизм рассуждений – ризонер, позволяющий автоматически выводить иерархию классов и поддерживать ее в корректно построенном состоянии (проверка на совместимость и непротиворечивость). Начиная с четвертой версии, в Protégé интегрированы в каче-

стве plugin многие функциональные возможности языка OWL и расширены в пятой версии. В настоящее время актуальным является язык OWL2 DL, позволяющий описывать онтологии как веб-документы и размещать их в WWW для совместного использования с другими owl-онтологиями [17].

4. Последний столбик в таблице дает общее представление о дескрипционной логике (целом подмножестве логик) и сравнение ее компонент с компонентами онтологических моделей.

Пример построения OWL-онтологии «Строительная компания»

Формальная онтология разрабатывается для примера, представленного на рис. 1, в среде онторедатора Protégé 5.5.0 со встроенными plugin – ризонером Pellet, визуализацией онтографов OntoViz и механизмом SPARQL-запросов. По утверждению авторитетных источников [17–19] этот онторедатор является наиболее широко используемым в сообществе онтологического инжиниринга, в том числе из-за возможности представления текстовых описаний в мультязычном формате.

Предполагается, что на компьютере создана локальная конечная точка, в которой хранятся файлы разработанной онтологии с расширениями *.owl и *.ttl. Последний файл используется SPARQL-процессором для обработки запросов (в синтаксисе Turtle). Отметим, что в примере онтограф и индивиды (рис. 1) несколько сокращены с целью сокращения объема OWL-описания модели онтологии.

Перед вводом информации в Protégé рекомендуем построить отображение сущностей CRF-модели (рис. 1) в Protégé-совместимый онтограф. Для примера такое отображение представлено на рис. 3.

Открываем главное меню онторедатора и присваиваем онтологии URL http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-1.

Руководства по созданию OWL-онтологий в Protégé и SPARQL-запросов представлены в общедоступных источниках, например в [17–19]. На рис. 4 представлено окно онторедатора, в котором отображены некоторые виды вкладок онторедатора, в которых показаны результаты в процессе разработки OWL-онтологии.

На рис. 5 представлен онтограф OWL-онтологии «Строительная компания», разработанный в приложении «OntoGraf», а на рис. 6 – метрика этой онтологии. Как видно из после-

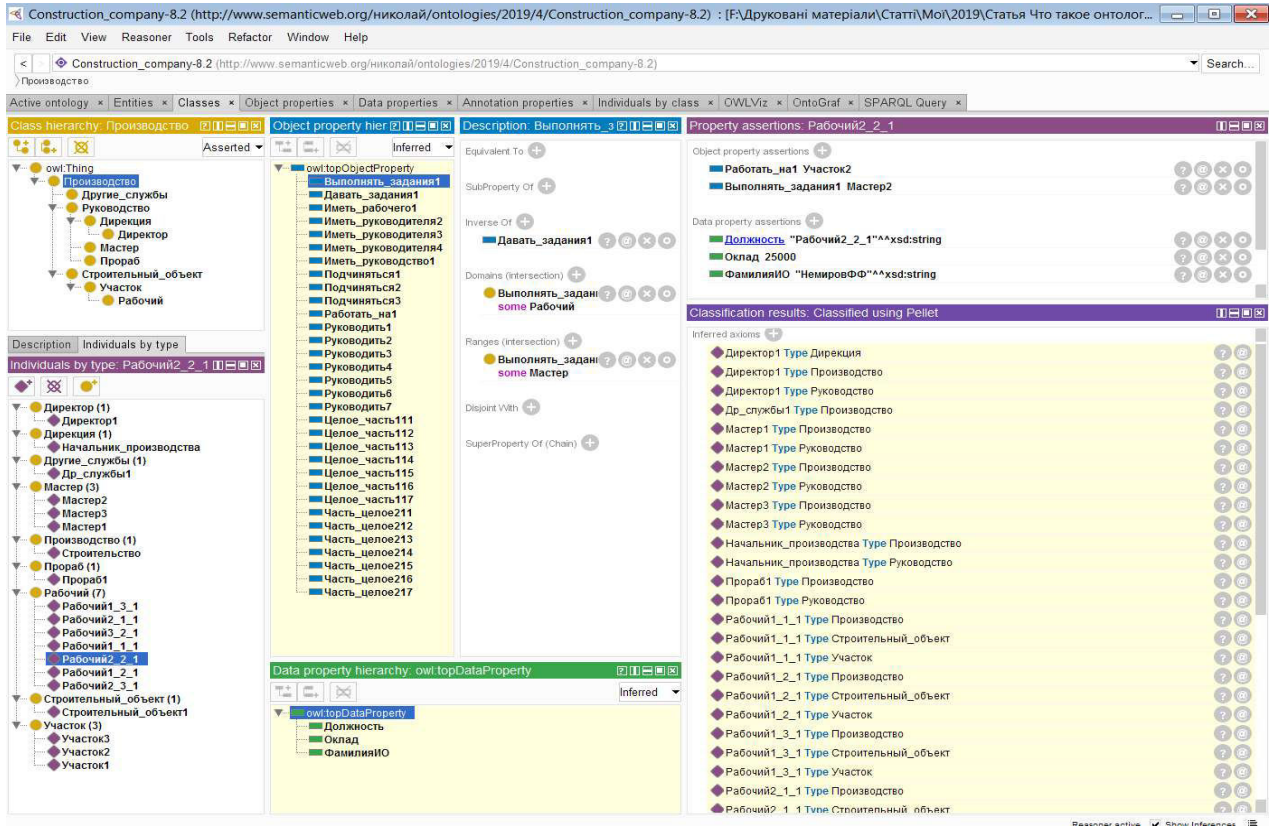


Рис. 4. Пример представления основных результатов разработки OWL-онтологии

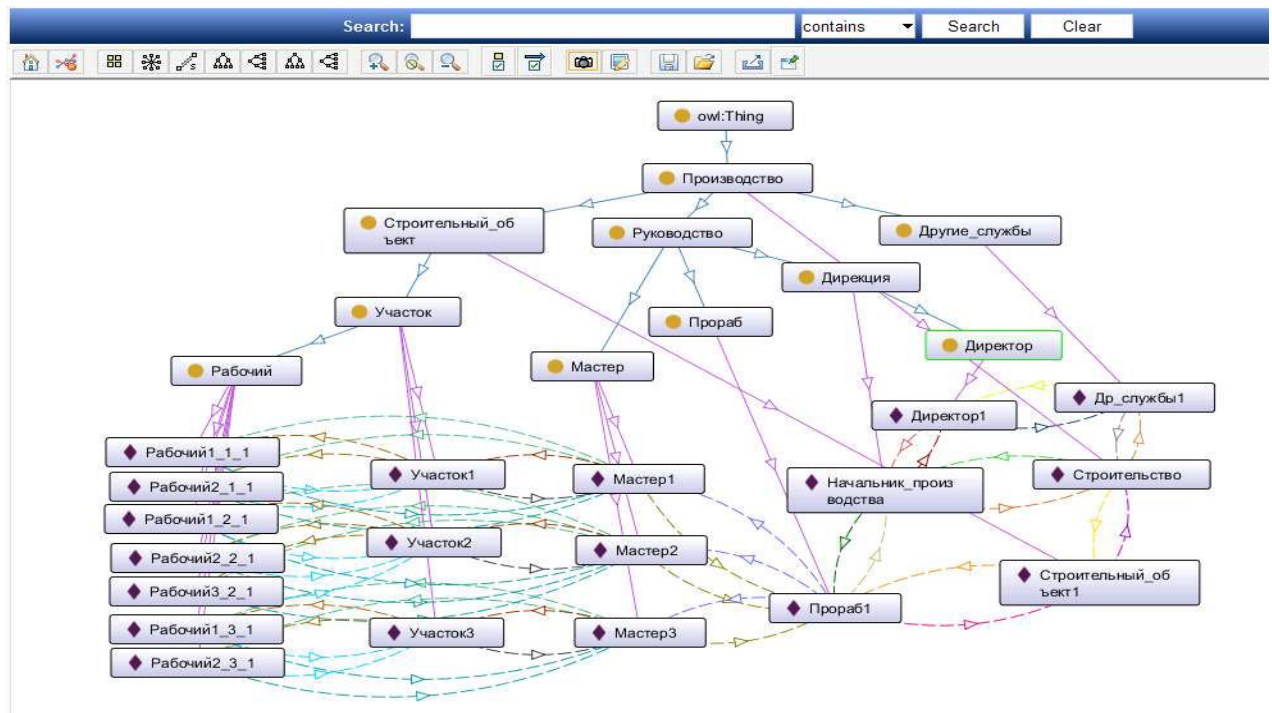


Рис. 5. Онтограф OWL-онтологии «Строительная компания»

Ontology metrics:

Metrics	
Axiom	258
Logical axiom count	194
Declaration axioms count	64
Class count	10
Object property count	32
Data property count	3
Individual count	19
Annotation Property count	0
Class axioms	
SubClassOf	9
EquivalentClasses	0
DisjointClasses	0
GCI count	0
Hidden GCI Count	0
Object property axioms	
SubObjectPropertyOf	0
EquivalentObjectProperties	0
InverseObjectProperties	9
DisjointObjectProperties	0
FunctionalObjectProperty	0
InverseFunctionalObjectProperty	0

Ontology metrics:

SymmetricObjectProperty	0
AsymmetricObjectProperty	0
ReflexiveObjectProperty	0
IrreflexiveObjectProperty	0
ObjectPropertyDomain	32
ObjectPropertyRange	32
SubPropertyChainOf	0
Data property axioms	
SubDataPropertyOf	0
EquivalentDataProperties	0
DisjointDataProperties	0
FunctionalDataProperty	0
DataPropertyDomain	0
DataPropertyRange	0
Individual axioms	
ClassAssertion	19
ObjectPropertyAssertion	54
DataPropertyAssertion	39
NegativeObjectPropertyAssertion	0
NegativeDataPropertyAssertion	0
SameIndividual	0
DifferentIndividuals	0

Рис. 6. Метрика OWL-онтологии «Строительная компания»

дней, создание вручную OWL-описания сущностей онтологии и его отладка представляется довольно трудоемкой работой. Protégé, кроме всего прочего, создает log-файл, в котором указаны ошибки проектирования, вычисленные ризонером.

Ниже представлены некоторые SPARQL-запросы к OWL-онтологии «Строительная компания» и ответы на них.

Запрос № 1

Кто подчиняется Прораб1?

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX фирма: <http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-8.2#>

SELECT ?кто

WHERE

```
{
?кто фирма:Подчиняться2 фирма:Прораб1
}
```

Ответ

Мастер2

Мастер1

Мастер3

Запрос № 2

Какая фамилия у Рабочий2_2_1?

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX фирма: <http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-8.2#>

SELECT ?фамилия

WHERE

```
{
фирма:Рабочий2_2_1 фирма:ФамилияИО ?фамилия
}
```

Ответ

«Немиров Ф.Ф.»

Запрос № 3

Кто работает на том же участке, что и Немиров?

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX фирма: <http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-8.2#>

SELECT ?рабочий ?фамилия ?участок

WHERE

```
{
?кто фирма:ФамилияИО «Немиров Ф.Ф.» .
?кто фирма:Работать_на1 ?участок .
?рабочий фирма:Работать_на1 ?участок .
?рабочий фирма:ФамилияИО ?фамилия .
}
```

Ответ

Рабочий1_2_1 «Немчинов А.В.»

Рабочий3_2_1 «Савченко В.А.»

Участок2

Запрос № 4

Сколько получает Рубин?

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX фирма: <http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-8.2#>

SELECT ?фамилия ?получает

WHERE

```
{
?рабочий фирма:ФамилияИО «Рубин Ю.О.» .
?рабочий фирма:Оклад ?получает .
?рабочий фирма:ФамилияИО ?фамилия .
}
```

Ответ

«Рубин Ю.О.» «30000»^^<http://www.w3.org/2001/XMLSchema#integer>

Запрос № 5

Кто получает больше, чем Рубин?

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX фирма: <http://www.semanticweb.org/николай/ontologies/2019/4/Construction_company-8.2#>

SELECT ?фамилия ?получает

WHERE

```
{
?рабочий фирма:ФамилияИО «Рубин Ю.О.» .
?рабочий фирма:Оклад ?окладРубина .
?сотрудник фирма:Оклад ?получает .
?сотрудник фирма:ФамилияИО ?фамилия .
FILTER (?получает > ?окладРубина )
}
```

Ответ

«Легеза А.М.» «60000»^^http://www.w3.org/2001/XMLSchema#integer

«Боромотов В.А.» «160000»^^<http://www.w3.org/2001/XMLSchema#integer>

«Гребенюк Н.Н.» «130000»^^<http://www.w3.org/2001/XMLSchema#integer>

«Савенков Н.И.» «100000»^^<http://www.w3.org/2001/XMLSchema#integer>

«Немчинов А.В.» «35000»^^<http://www.w3.org/2001/XMLSchema#integer>

«Червинчук В.П.» «60000»^^<http://www.w3.org/2001/XMLSchema#integer>

«Веселов В.А.» «60000»^^<http://www.w3.org/2001/XMLSchema#integer>

Отметим еще одну полезность CRF-модели онтологии ПдО.

Согласно [13], она проектируется на основе лингвистического корпуса текстов, описывающего знания заданной ПдО. Из него извлекаются максимально полные (по кардинальности) множества понятий X , отношений между понятиями R , а функции интерпретации F согласовываются с определениями соответствующих понятий, приведенными в энциклопедиях, толковых словарях и других научных публикациях. Семантика описаний понятий и отношений между ними максимально согласована с представлениями соответствующего научного сообщества.

Protégé не проверяет семантику введенных имен классов, свойств и индивидов. Другими словами, в Protégé семантика разрабатываемой онтологии субъективна, что может вызвать проблемы при совместном использовании онтологий.

Следовательно, при разработке больших по объему онтологий, возможно, имеет смысл предварительно разработать CRF-модель, что обеспечит семантическую согласованность такой онтологии.

Выводы

Кратко рассмотрена формализация определения онтологии ПдО, выполнен сравнительный анализ известных моделей онтологий, таких как CRF, Frame и OWL. Отмечены их преимущества и недостатки, при этом сделан вывод, что при разработке больших онтологий следует применять все три указанные модели. На начальном этапе – CRF-модель для автоматизации построения структуры концептов ПдО и их семантической согласованности. На этапе ввода и формализации – модель Protégé-фреймов. И на этапе использования онтологий – OWL-модель с механизмами запросов и рассуждений. При этом совместное использование онтологий предполагает ее размещение в Semantic Web. В статье приведен пример построения и использования OWL-онтологии для некоторой строительной компании.

СПИСОК ЛИТЕРАТУРЫ

1. *Ohrstrom P., Andersen J., Schärfe H.* What has happened to ontology // In F. Dau, M.L. Mugnier, and G. Stumme, editors, *Conceptual Structures: Common Semantics for Sharing Knowledge, 13th International Conference on Conceptual Structures, ICCS 2005, Kassel, Germany, July 17–22, 2005, Proceedings*, volume 3596 of *Lecture Notes in Computer Science*, 2005. – P.425-438. https://doi.org/10.1007/11524564_29
2. *Gruber T.R.* A Translation Approach to Portable Ontologies. *Knowledge Acquisition*. – 1993. – P.199-220. <https://doi.org/10.1006/knac.1993.1008>
3. *Gruber T.R.* Toward Principles for the Design of Ontologies Used for Knowledge Sharing // *International Journal of Human Computer Studies*. – Vol.43(5–6). – 1995. – P.907-928.
4. *Палагин А.В., Петренко Н.Г.* Развитие и становление трансдисциплинарных и междисциплинарных исследований и роль информатики // Комп'ютерне моделювання: аналіз, управління, оптимізація. – Дніпро: ДВНЗ УДХТУ, 2018. – №1 (3). – С.46-70. URL: <http://kmauo.org/wp-content/uploads/2018/1/Palagin.pdf>. DOI випуску: <https://doi.org/10.32434/2521-6406-2018-3-1>
5. *Borst W.* *Construction of Engineering Ontologies*. PhD thesis, Institute for Telematica and Information Technology, University of Twente, Enschede, The Netherlands. – 1997.
6. *Studer R., Benjamins R., Fense D.I.* Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*. – Vol.25(1–2). – 1998. – P.161-198. [https://doi.org/10.1016/S0169-023X\(97\)00056-6](https://doi.org/10.1016/S0169-023X(97)00056-6)
7. *Guarino N., Giaretta P.* Ontologies and Knowledge Bases: Towards a Terminological Clarification. In N. Mars, editor, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, IOS Press, Amsterdam, – 1995. – P.25-32.
8. *Genesereth M.R., Nilsson N.J.* *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, Los Altos, CA 94305, 1987. – 26 p.
9. *Guarino N.* Formal Ontology in Information Systems. In N. Guarino, editor, *Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, June 6-8, 1998*, IOS Press, Amsterdam, 1998. – P.3-15.
10. *Carnap R.* *Meaning and Necessity – A Study in Semantics and Modal Logic*. The University of Chicago Press, second edition, 1956. – 266 p.
11. *Dowty D.R., Wall R., Peters S.* *Introduction to Montague Semantics*, volume 11 of *Studies in Linguistics and Philosophy*. Springer, Heidelberg, 1980. – P.11-62. <https://doi.org/10.1007/978-94-009-9065-4>
12. *Gruber T.R.* Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, Deventer, The Netherlands, 1993. – P.1-23.
13. *Palagin A.V., Kryvyy S.L., Petrenko N.G.* Ontological methods and means of processing subject knowledge. Lugansk: V.I. Dal East Ukr. Nac. University. – [Электронный ресурс]: Retrieved from <http://www.aduis.com.ua/Monography.pdf>.
14. *Uschold M.* Ontologies and Semantics for Seamless Connectivity. *SIGMOD Record*, – 33(4). – 2004. – P.58-64. <https://doi.org/10.1145/1041410.1041420>
15. *Das S.K.* *Deductive Databases and Logic Programming*.

Addison Wesley, 1992. – 256 p.

16. *Vaught R.L.* Alfred Tarski's Work in Model Theory // *The Journal of Symbolic Logic*, Vol.51(4). – 1986. – P.869-882. <https://doi.org/10.2307/2273900>

17. OWL 2 Web Ontology Language Primer (second edition). [Электронный ресурс]: <http://www.w3.org/TR/2012/REC-owl2-primer-20121211/>

18. *Matthew Horridge.* *A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools.* Edition 1.3. – Copyright The University Of Manchester. – March 24, 2011. – 107 p.

19. *Bob DuCharme.* *Learning SPARQL. Querying and Updating with SPARQL 1.1* (Second edition). – O'Reilly Media. All rights reserved. August 2013: ISBN: 978-1-449-37143-2. – 367 p.

Поступила в редакцію 20.05.2019

ПРО ПРАКТИЧНЕ ВИКОРИСТАННЯ ОНТОЛОГІЧНИХ МОДЕЛЕЙ ПРЕДМЕТНИХ ОБЛАСТЕЙ

Петренко М.Г., Зеленцов Д.Г.

У статті розглянуті найбільш відомі інструменти формального опису онтологій предметних областей, виконано їх порівняльний аналіз і вибір формальної мови і онторедактора для практичного використання. У роботі наводяться два варіанти трактування терміну «онтологія». Пропонується його розглянути з точки зору онтологічного інжиніринга. Детально розглянуті три складові визначення терміна «онтологія» – концептуалізація, формальна, явна специфікація. Виділено проблеми виразності мов опису онтологій. Наведено різні підходи до вибору формальної мови для позначення елементів концептуалізації. Як приклад системи обрана деяка виробнича компанія з її виробничими об'єктами, співробітниками і виробничими відносинами між ними. Дослідження здійснюється для сукупності завдань, пов'язаних з людськими ресурсами компанії. Наводиться CRF-модель (concept-relation-interpretation function model) онтології предметної області. Для розширеного прикладу будівельної компанії надані сам онтограф та індивіди класів. Здійснено порівняння найбільш відомих логічних теорій опису онтологій. Розглянуто особливості різних моделей побудови онтологій предметних областей, таких як CRF, Frame і OWL. Відзначено їх переваги та недоліки. Зроблено висновок, що при розробці великих онтологій слід застосовувати всі три зазначені моделі: на початковому етапі – CRF-модель для автоматизації побудови структури концептів предметної області та їх семантичної узгодженості; на етапі введення і формалізації – модель Protégé-фреймів; на етапі використання онтології – OWL-модель з механізмами запитів і міркувань. При цьому спільне використання онтології передбачає її розміщення в Semantic Web. Порівняльний аналіз показав, що перспективним напрямом є побудова веб-онтологій в середовищі Protégé. Для предметної області «Будівельна компанія» в статті наведено приклад побудови і використання OWL-онтології в інтегрованому середовищі онторедактора Protégé, механізму міркувань Pellet і SPARQL-процесора з відповідними до неї запитами.

Ключові слова: дескрипційна логіка, онтологія, модель онтології, концептуалізація, онтограф, Protégé.

ON THE PRACTICAL USE OF ONTOLOGICAL MODELS OF SUBJECT AREAS

Petrenko N.G.^a, Zelentsov D.G.^b

^a V.M. Glushkov Institute of Cybernetics, Kiev, Ukraine

^b Ukrainian State University of Chemical Technology, Dnipro, Ukraine

The article considers the most well-known tools for the formal description of ontologies of subject areas, their comparative analysis and the choice of a formal language and ontology editor for practical use. The paper provides two interpretations of the term "ontology". It is proposed to consider it from the point of view of ontological engineering. Three component definitions of the term "ontology" are considered in detail - conceptualization, formal and explicit specification. The problems of expressiveness of ontology description languages are highlighted. Various approaches to choosing a formal language for designating elements of conceptualization are presented. As an example of a system, a certain production company with its production facilities, employees and production relationships between them was selected. The study is conducted for a set of tasks related to human resources of the company. The CRF-model (concept – relation – interpretation function model) of a subject area ontology is given. For an extended example of the construction company, the ontograph itself and class individuals are represented. A comparison of the most well-known logical theories of description of ontologies is carried out. Features of various models for constructing ontologies of subject areas, such as CRF, Frame, and OWL, are considered. Their advantages and disadvantages are noted. It is concluded that when developing large ontologies, all three of the indicated models should be applied: at the initial stage, a CRF model to automate the construction of a structure of concepts of a subject area and their semantic consistency; at the stage of input and formalization - a model of Protégé-frames; at the stage of using ontology, an OWL model with query and reasoning mechanisms. At the same time, the ontology sharing implies its placement on the Semantic Web. A comparative analysis showed that the development of web ontologies in the Protégé environment is a promising direction. For the subject field «Construction Company», the article gives an example of building and using OWL ontology in the integrated environment of the Protégé ontology editor, the Pellet reasoning mechanism, and the SPARQL processor. with relevant requests.

Keywords: descriptive logic, ontology, ontology model, conceptualization, ontograph, Protégé.

REFERENCES

1. Ohrstrom P., Andersen J., Schärfe H. What has happened to ontology. In F. Dau, M.L. Mugnier, G. Stumme, editors, *Conceptual Structures: Common Semantics for Sharing Knowledge, 13th International Conference on Conceptual Structures, ICCS 2005, Kassel, Germany, July 17–22, 2005, Proceedings*, vol. 3596 of *Lecture Notes in Computer Science*, 2005. pp.425-438. https://doi.org/10.1007/11524564_29
2. Gruber T.R. A Translation Approach to Portable Ontologies. *Knowledge Acquisition*, 1993, pp.199-220. <https://doi.org/10.1006/knac.1993.1008>
3. Gruber T.R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human Computer Studies*, vol. 43(5–6), 1995, pp.907-928. <https://doi.org/10.1006/ijhc.1995.1081>
4. Palagin A.V., Petrenko N.G. *Razvytye y stanovlenye transdystyplynarnikh y mezhdystryplynarnikh yssledovanyi y rol ynformatyky* [Development and formation of transdisciplinary and interdisciplinary research and the role of computer science]. *Komp'uterne modelivannâ: analiz, upravlinnâ, optimizaciâ* [Computer Modeling: Analysis, Control, Optimization], 2019. No 1 (3), pp. 46-70. URL: <http://kmauo.org/wp-content/uploads/2018/1/Palagin.pdf>. DOI issue: <https://doi.org/10.32434/2521-6406-2018-3-1>
5. Borst W. Construction of Engineering Ontologies. PhD thesis, Institute for Telematica and Information Technology, University of Twente, Enschede, The Netherlands, 1997.
6. Studer R., Benjamins R., Fensel D. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 1998, pp.161-198. [https://doi.org/10.1016/S0169-023X\(97\)00056-6](https://doi.org/10.1016/S0169-023X(97)00056-6)
7. Guarino N., Giaretta P. Ontologies and Knowledge Bases: Towards a Terminological Clarification. In N. Mars, editor, *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing*, IOS Press, Amsterdam, 1995, pp.25-32.
8. Genesereth M.R., Nilsson N.J. Logical Foundations of Artificial Intelligence. Morgan Kaufmann, Los Altos, CA 94305, 1987, 26 p.
9. Guarino N. Formal Ontology in Information Systems. In N. Guarino, editor, *Formal Ontology in Information Systems. Proceedings of FOIS '98, Trento, Italy, June 6-8, 1998*, IOS Press, Amsterdam, 1998, pp.3-15.
10. Carnap R.. Meaning and Necessity – A Study in Semantics and Modal Logic. The University of Chicago Press, second edition, 1956, 266 p.
11. Dowty D.R., Wall R., Peters S. Introduction to Montague Semantics, vol. 11 of *Studies in Linguistics and Philosophy*. Springer, Heidelberg, 1980, pp.11-62. <https://doi.org/10.1007/978-94-009-9065-4>
12. Gruber T.R. Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, Deventer, The Netherlands, 1993, pp.1-23.
13. Palagin A.V., Kryvyy S.L., Petrenko N.G. Ontological methods and means of processing subject knowledge. Lugansk: V.I. Dal East Ukr. Nac. University. Retrieved from <http://www.aduis.com.ua/Monography.pdf>.
14. Uschold M. Ontologies and Semantics for Seamless Connectivity. *SIGMOD Record*, vol. 33(4), 2004, pp.58-64. <https://doi.org/10.1145/1041410.1041420>
15. Das S.K. *Deductive Databases and Logic Programming*. Addison Wesley, 1992, 256 p.
16. Vaught R.L. Alfred Tarski's Work in Model Theory. *The Journal of Symbolic Logic*, vol. 51(4), 1986, pp.869-882. <https://doi.org/10.2307/2273900>
17. OWL 2 Web Ontology Language Primer (second edition). Retrieved from: <http://www.w3.org/TR/2012/REC-owl2-primer-20121211/>.
18. Matthew Horridge. A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools. Edition 1.3. Copyright The University Of Manchester, March 24, 2011, 107 p.
19. Bob DuCharme. Learning SPARQL. Querying and Updating with SPARQL 1.1 (Second edition). O'Reilly Media. All rights reserved. August 2013: ISBN: 978-1-449-37143-2, 367 p.