

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Шебзухова Татьяна Александровна
Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета
Дата подписания: 24.08.2023 16:14:55
Уникальный программный ключ:
d74ce93cd40e39275c3ba2f58486412a1c8ef96f

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ

Методические указания по выполнению лабораторных работ
по дисциплине «Искусственный интеллект в профессиональной сфере»
для направления подготовки 38.03.01 Экономика
направленность (профиль) Корпоративная экономика и финансы

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 12000002A633E3D113AD425FB50002000002A6

Владелец: Шебзухова Татьяна Александровна

Действителен: с 20.08.2021 по 20.08.2022

Пятигорск
2022

СОДЕРЖАНИЕ

<u>ВВЕДЕНИЕ</u>	3
<u>СТРУКТУРА И СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ ЗАНЯТИЙ</u>	5
<u>Лабораторная работа 1. Анализ современных программных средств с применением ИИ....</u>	5
<u>Лабораторная работа 2 Формализация знаний Использование семантических сетей для представления знаний</u>	6
<u>Лабораторные работы 3Создание онтологии в системе PROTÉGÉ.</u>	
<u>Основы RDF и OWL</u>	11
<u>Список литературы</u>	70

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 12000002A633E3D113AD425FB50002000002A6

Владелец: Шебзухова Татьяна Александровна

Действителен: с 20.08.2021 по 20.08.2022

ВВЕДЕНИЕ

Цели и задачи освоения дисциплины

Целью освоения дисциплины является ознакомление студентов с кругом задач в области искусственного интеллекта, а также знакомство с основными понятиями и методами машинного обучения и их применением к задачам, относящимся к профессиональной области.

Задачи освоения дисциплины – ознакомление студентов с возможностями использования технологий искусственного интеллекта в профессиональной сфере.

В результате освоения дисциплины студенты должны:

- использовать принципы работы современных информационных технологий для решения задач профессиональной деятельности;
- выбирать цифровые технологии, программные средства для решения профессиональных задач;
- применять при решении задач профессиональной деятельности специализированное программное обеспечение и методы искусственного интеллекта;
- разрабатывать основные модули интеллектуальных систем, владеть приемами решения лабораторных задач в предметной области.

Перечень планируемых результатов обучения по дисциплине (модулю), соотнесённых с планируемыми результатами освоения образовательной программы

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-2 Способен осуществлять сбор и обработку информации бизнес-анализа для обоснования управленческих решений, в том числе с применением методов искусственного интеллекта и алгоритмов машинного обучения	ИД-4 ПК-2 Применяет при решении задач профессиональной деятельности специализированное программное обеспечение, методы искусственного интеллекта и машинного обучения ИД-5 ПК-2 Способен обосновать применение того или иного алгоритма машинного обучения для решения конкретной задачи ИД-6 ПК-2 Разрабатывает оригинальные модели и алгоритмы искусственного интеллекта и машинного обучения в профессиональной деятельности	Готовность применять при решении задач профессиональной деятельности специализированное программное обеспечение и методы искусственного интеллекта. Готовность разрабатывать основные модули интеллектуальных систем, владеть приемами решения практических задач в предметной области.

ДОКУМЕНТ ПОДПИСАН 1. Наименование дисциплины ЭЛЕКТРОННОЙ ПОДПИСЬЮ занятий Сертификат: 12000002A633E3D113AD425FB50002000002A6 Владелец: Шебзухова Татьяна Александровна Действителен: с 20.08.2021 по 20.08.2022		Наименование дисциплины, их краткое содержание	Объем часов
--	--	--	-------------

5 семестр		
1.	Анализ современных программных средств с применением ИИ	1,5
2.	Формализация знаний Использование семантических сетей для представления знаний	1,5
3.	Создание онтологии в системе Protégé	3
	Итого за 5 семестр	6
	Итого	6

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 12000002A633E3D113AD425FB50002000002A6

Владелец: Шебзухова Татьяна Александровна

Действителен: с 20.08.2021 по 20.08.2022

СТРУКТУРА И СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Лабораторная работа 1.

Анализ современных программных средств с применением ИИ.

Вопросы (компетенции, навыки) для освоения:

1. Изучить современные программные продукты, основанные на ИИ.
2. Научиться анализировать возможности программных средств с применением ИИ.

Задания для выполнения и методические рекомендации:

Задание 1

Используя приведенный ниже список интернет-ссылок, а также другие интернет-источники, изучите интеллектуальные системы, предлагаемые на рынке.

1. <https://www.zeluslugi.ru/info-czentr/stati/primery-iskusstvennogo-intellekta-programmy>
2. <https://3dnews.ru/1058498/intellektualnoe-prevoshodstvo-10-zaslugivayushchih-vnimanija-aiservisov-i-prilogeniy>
3. <https://soware.ru/categories/artificial-intelligence-platforms>

Определите их направление:

- экспертные системы (ЭС);
- системы с базами знаний;
- интеллектуальное обучение;
- нейронные сети.

Задание 2

Проанализируйте возможности, области применения, достоинства и недостатки изученных интеллектуальных систем. На основе анализа составьте отчет в табличной форме:

Название ПО или сервиса, версия	Возможности программы	Область применения	Стоимость	Источник информации (www.)

Сделайте вывод о рынке ПО. Рассчитайте удельный вес по области применения ПО. Изобразите результат графически (постройте диаграмму).

Лабораторная работа 2

Формализация знаний

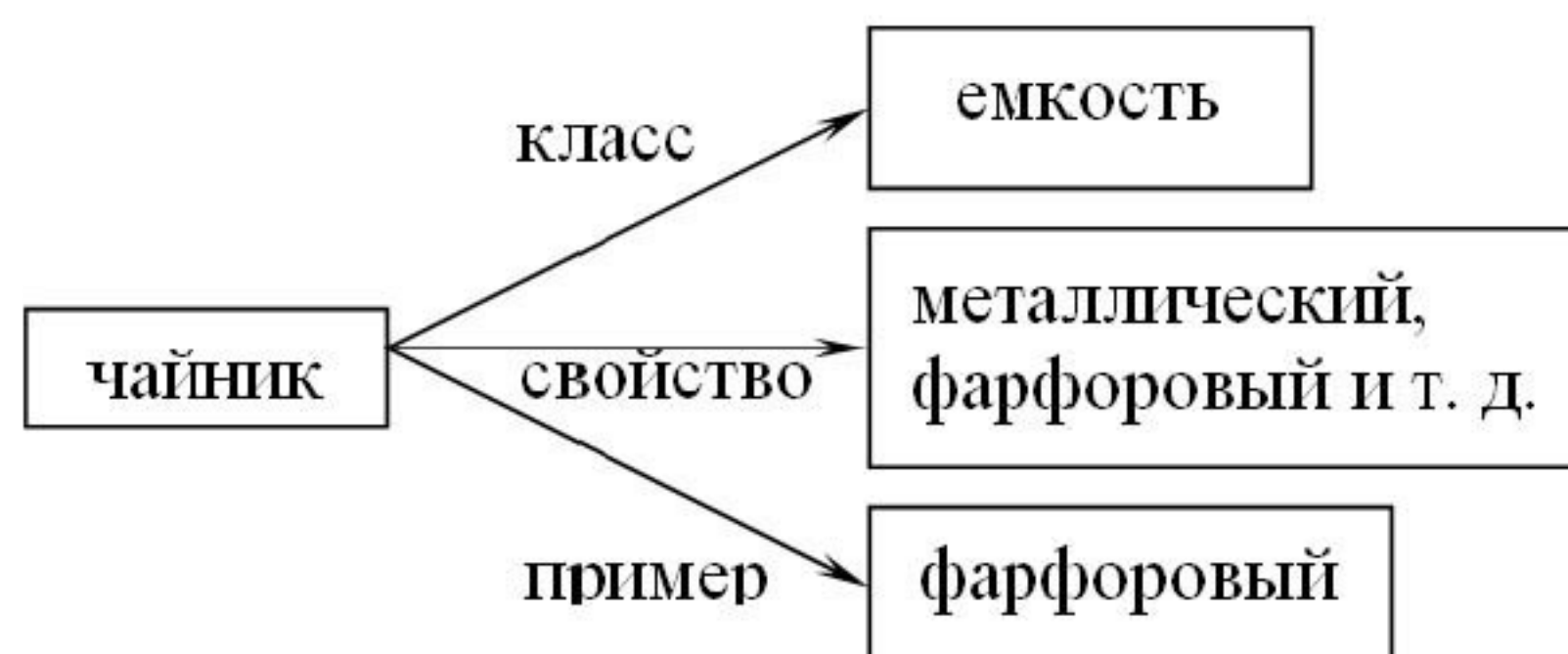
ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ	
Сертификат:	12000002A633E3D113AD425FB50002000002A6
Владелец:	Шебзухова Татьяна Александровна
Действителен: с 20.08.2021 по 20.08.2022	

1. Изучить общее представление о семантической сети и фреймах для представления знаний.
2. Научиться строить семантическую сеть и используя фреймы реализовать структуру отношений.

Задания для выполнения и методические рекомендации:

Семантическая сеть – это один из способов представления знаний. Изначально семантическая сеть была задумана как модель представления долговременной памяти в психологии, но впоследствии стала одним из способов представления знаний в ЭС.

Семантика – означает общие отношения между символами и объектами из этих символов.



Простейший образец семантической сети

Вершины – это объекты, дуги – это отношения. Семантическая модель не раскрывает сама по себе каким образом осуществляется представление знаний. Поэтому семантическая сеть рассматривается как метод представления знаний и структурирования знаний. При расширении семантической сети в ней возникают другие отношения:

IS –A (принадлежит) и PART OF (является частью) отношение:

целое → часть.

Ласточка IS – A птица, «нос» PART OF «тело». Например:



отношением:

период → «весна – лето».

Получается иерархическая структура понятия ЮКО. Можно разбить на подсхемы. Большой проблемой для семантических сетей является то, что результат вывода не гарантирует достоверности, так как вывод есть просто наследование свойств ветви is-a.

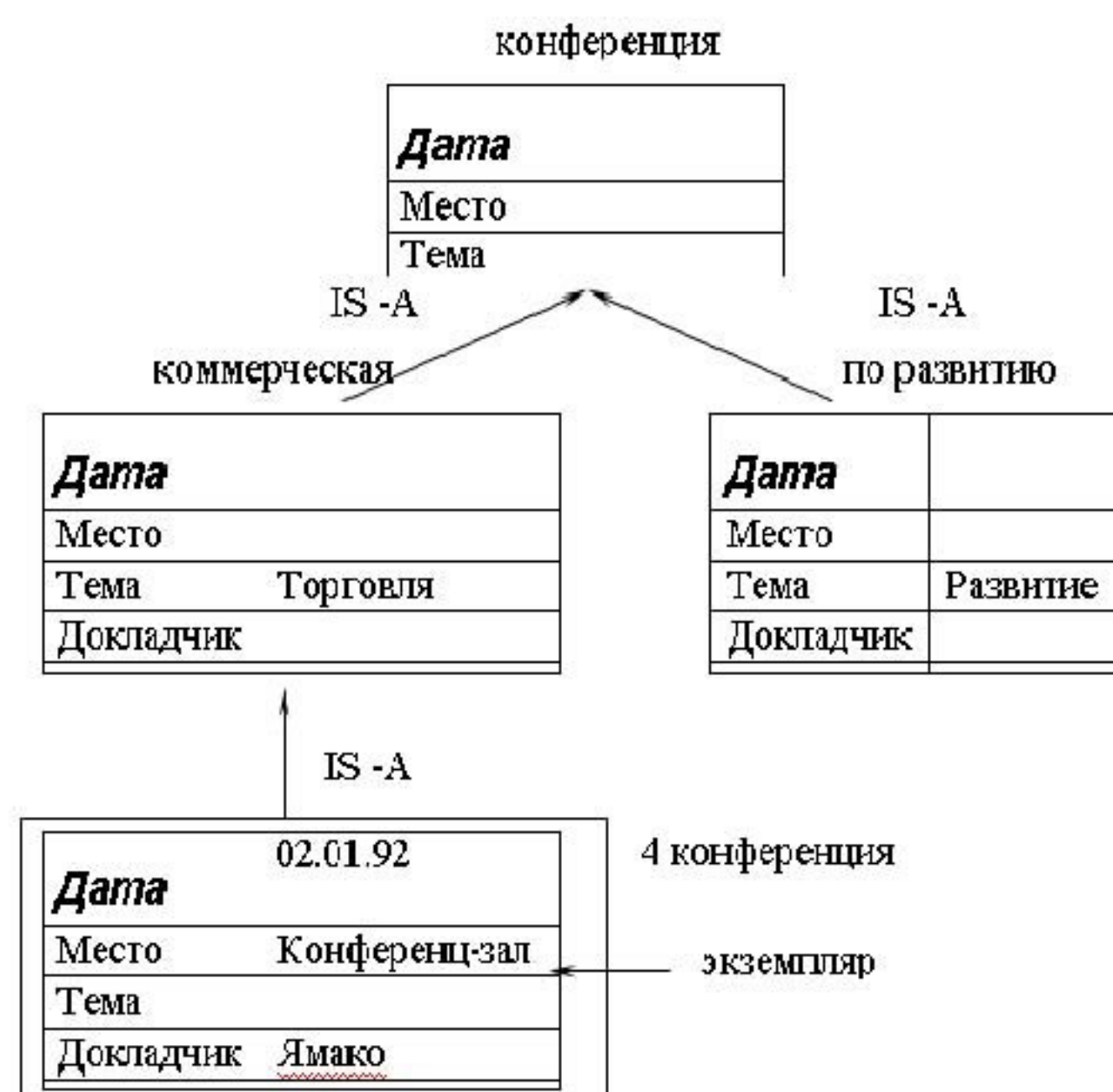
Для отображения иерархических отношений между объектами и введения единой семантики в семантические сети было предложено использовать процедурные сети. Сеть строится на основе класса (понятия); вершины, дуги и процедуры представлены как объекты.

Задание 1

Используя соответствующие дуги построить семантическую сеть, касающуюся:

имя во фрейме, к которому он принадлежит. Обычно имя слота не несет никакой смысловой нагрузки и является лишь идентификатором данного слота.

3. Указатели наследования. Эти указатели касаются только фреймовых систем иерархического типа, основанные на отношениях “абстрактное-конкретное”, они показывают, какую информацию об атрибутах слотов во фрейме верхнего уровня наследуют слоты с такими же именами во фрейме нижнего уровня. Типичные указатели наследования Unique (U: - уникальный), Same (S: такой же), Range (R: установление границ), Override (O: игнорировать) и т.п. U показывает, что фрейм может иметь слоты с разными значениями: S - все слоты должны иметь одинаковые значения, R - значение слотов фрейма нижнего уровня должны находиться в пределах, указанных значениями слотов фрейма верхнего уровня, O - при отсутствии указания значение слота фрейма верхнего уровня становится значением слота фрейма нижнего уровня, но в случае определения нового значения слотов фреймов нижних уровней указываются в качестве значений слотов.



Пример фреймовой модели

Задание 2.

Используя фреймовую модель представления знаний реализовать структуру отношений, описывающие следующие ситуации:

1. экзамен по дисциплине за семестр у преподавателя при составляющих: семестр, экзамен, преподаватель, оценка, студент, получать.
2. ведомость при составляющих: дисциплина, студент, экзамен, семестр, преподаватель, оценка.
3. конференция по коммерческим вопросам при составляющих: дата, место проведения, тема, цель выступающие.
4. получение оценки при составляющих: преподаватель, студент, оценка, получать.
5. использования изделия при составляющих: организация, разработка технологического решения, исследование «физического эффекта», методы создания изделия.
6. информационная структура БД в машиностроении при составляющих: физические эффекты, технические решения, изделия, объект поставки изделия, приборы и стенды, нормативы.
7. классификация продукта при составляющих: название, область применения, способ хранения, способ транспортировки.
8. аудитория (описание) при составляющих: вместимость, назначение, составляющие, местонахождение.
9. животный мир при составляющих: вид, тип, среда обитания, особенности поведения.

Лабораторные работы 3

Создание онтологии в системе PROTÉGÉ. Основы RDF и OWL

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ	
Сертификат:	12000002A633E3D113AD425FB50002000002A6
Владелец:	

3. Определить предметную область, создать классы, отношения с использованием системы Protégé, а также на примере созданной БЗ рассмотреть модель RDF, язык OWL и синтаксис Turtle.
4. Научиться реализовывать понятия Defined Classes, DL Query в системе Protégé
5. Изучить возможности языка запросов SPARQL
6. Научиться анализировать создавать правила для проверки информации о товарах с помощью языков SWRL и SQWRL .

Задания для выполнения и методические рекомендации:

В качестве примера будет рассмотрена информация о конкретном товаре (конкретном смартфоне¹) и на ее основании составлена первая версия БЗ.

Перед началом работы с системой Protégé [6] необходимо составить хотя бы некоторую упрощенную концептуальную модель предметной области. И, для того чтобы потом не было сложностей с ее форм

предиката.

Скалярные ресурсы указывают некоторые конкретные характеристики и являются просто значением определенного типа. Соответственно, такие ресурсы могут находиться только на месте объекта. Но, при этом для них может определяться тип и для строковых значений может определяться язык. Например, для нашей предметной области такими сущностями могут являться: 6.2 (конкретная диагональ экрана; float), «Samsung Galaxy S9+» (оригинальное название; string); «черный»@ru (цвет; string; русский).

Также необходимо отметить, что «предикат» также может быть «субъектом» или «объектом» в рамках других троек. Это необходимо чтобы, например, описать ограничения данного отношения.

Пустая вершина в основном необходима для «имитация» n-арных связей. Как видно из концепции модели, предикат может связывать только 2 ресурса. Но, это решается с помощью введения промежуточной вершины и разделением связи на несколько бинарных. При этом вершина не будет представлять понятие реального мира и ей нет смысла давать уникальный глобальный идентиф

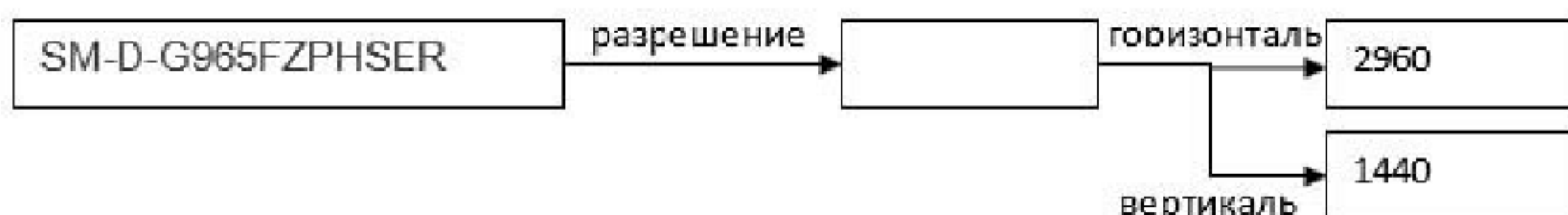


Рис. 1 Граф, описывающий разрешение дисплея

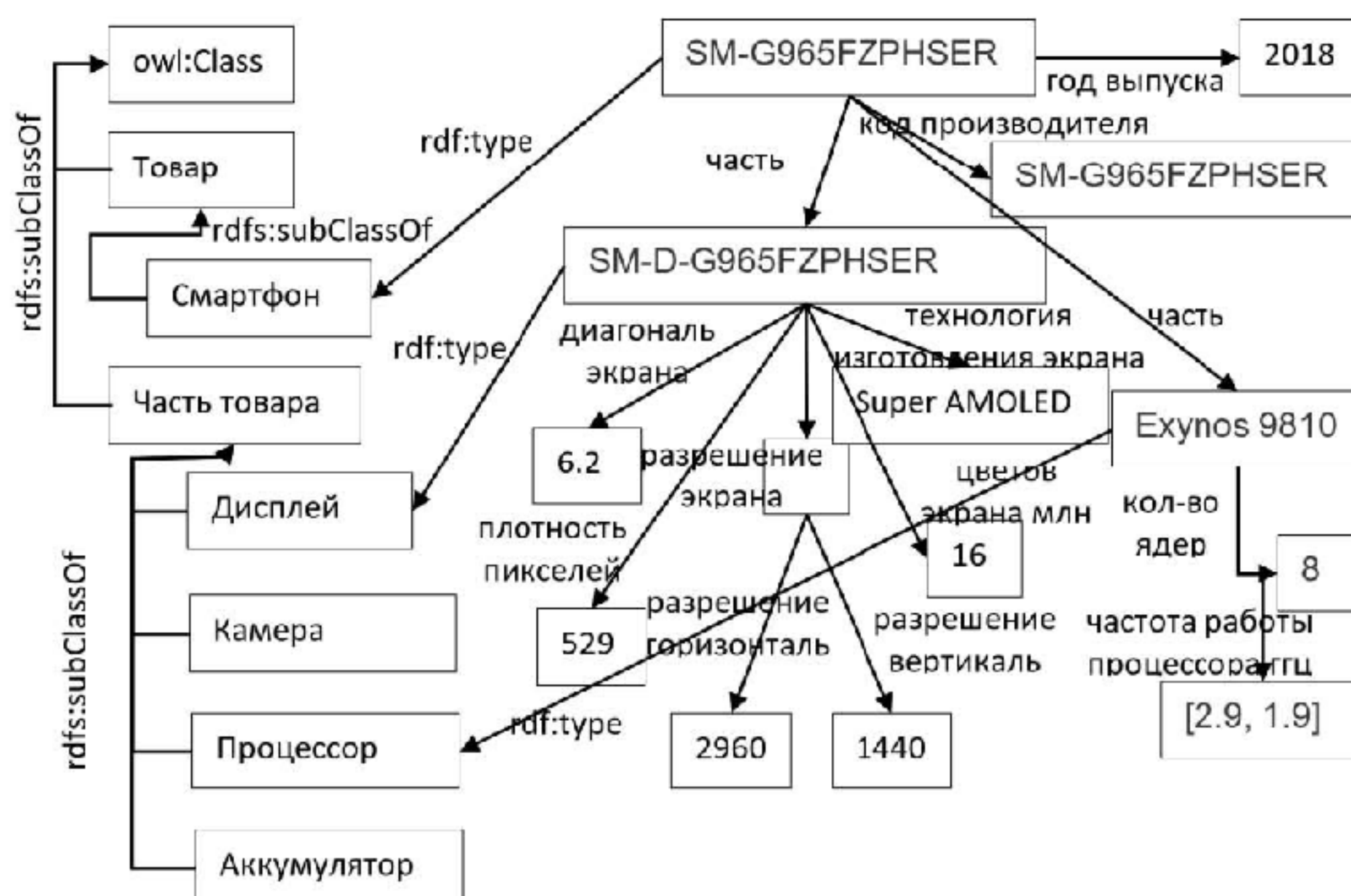
Стоит отметить, что данные понятия также можно было описать с использованием 2 связей «разрешение_гор», «разрешение_вер» и без промежуточной вершины. Что показывает, что создание онтологии остается вариативным процессом, зависящим от конкретных целей и предпочтений разработчика.

В связи с этим описание онтологии можно разделить на:

- 1 Сущности. Все понятия, описанные ранее в рамках RDF, а также некоторые новые из OWL;
- 2 Выражения. Описание понятия через комбинацию множества других сущностей. Одним из основных примеров применения является определение класса через выражение. Например, «Камерофон» – это телефон, который имеет не менее 2 основных камер с разрешением не менее 20 мегапикселей; новый тип «число» – это объединение множеств целых чисел и чисел с дробной частью.
- 3 Аксиомы. Точно известные положения и положения, которые всегда должны оставаться истинными. Например, на отношение «плотностьПикселей» могут быть наложены ограничения на область определения – экземпляр класса «Дисплей» и область значения – «целое число». При этом если в базе знаний будет тройка «G плотностьПикселей 529», то это может привести как к ошибке (если G точно не может являться экземпляром класса «Дисплей»), так и к выводу нового отношения «G rdf:type Дисплей».

Как уже было сказано выше, OWL добавляет новые понятия и аксиомы. Для данной

Теперь, когда сущности, которые могут быть использованы, обозначены, можно спланировать общее описание предметной области. Оно разделено на описание классов, экземпляров и отношений между ними (рис. 2) и описание самих отношений (рис. 3).



как подклассы класса «Часть товара»: «Дисплей», «Камера», «Процессор», «Аккумулятор». Сюда выделены только самые важные части, которые больше всего интересуют покупателей и наиболее ценны при обработке информации о товаре. Остальные характеристики будут задаваться отношениями между конкретным смартфоном и конкретными значениями (на изображении в качестве примера таких отношений можно видеть: «год выпуска», «код производителя»). Стоит отметить, что некоторые узлы (например, «[2.9, 1.9]») заданы в абстрактном виде и их конкретное задание будет зависеть от используемого синтаксиса и/или возможностей инструментальной системы.

В рамках приведенного фрагмента свойств не сделано разделение на ObjectProperty и DataProperty во избежание излишней конкретизации (при реализации разделение делается очевидным образом). В рамках прямых наследников «owl:Property» выделены типы возможных отношений: «часть» (для связи товара с его частью), «числовое/текстовое/свободное/разрешение» (данные свойства определяют группу свойств, объединенных общей областью определения. Разделение на текстовое и свободное сделано для упрощения дальнейшей обработки. Под текстовыми подразумеваются конкретные значения,

- 4.1. Проверить относится ли характеристика к описания какого-то из подклассов класса «Часть товара». Подклассы класса «Часть товара», а также отношения, связанные с ними заданы заранее и не изменяются в процессе сбора информации. Для каждого из таких классов задаются списки полей и правила их преобразования. Конкретные реализации этой части не будут рассматриваться.
- 4.2. Если относится, то получить экземпляр (а при отсутствии создать) соответствующей части (формирования локальной части IRI для таких частей также прописано в правилах из п. 4.1.). Для полученного экземпляра задать отношение, связывающего его с рассматриваемой характеристикой.
- 4.3. Если не относится, то найти отношение по имени характеристики (при отсутствии создать отношение с таким именем как дочернее свойство для свойства «unknown» или типизированного свойства). Связать экземпляр товара с рассматриваемым значением характеристики полученным отношением.

Здесь проявляется гибкость онтологического подхода. В отличие от реляционной модели, в любой момент легко могут быть добавлены новые отношения, произведен перенос их в другую группу, произведено задание им новых ограничений. Также

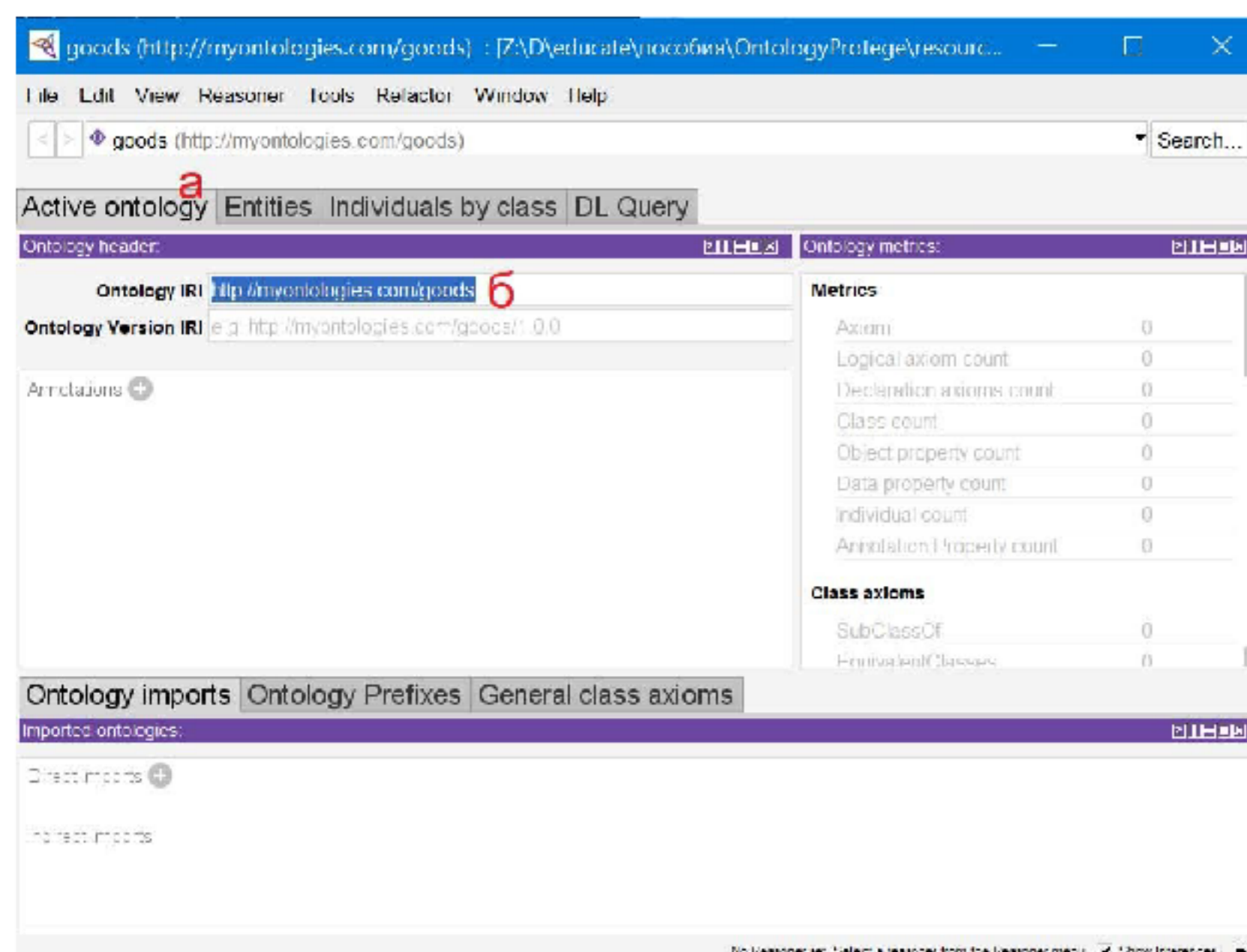
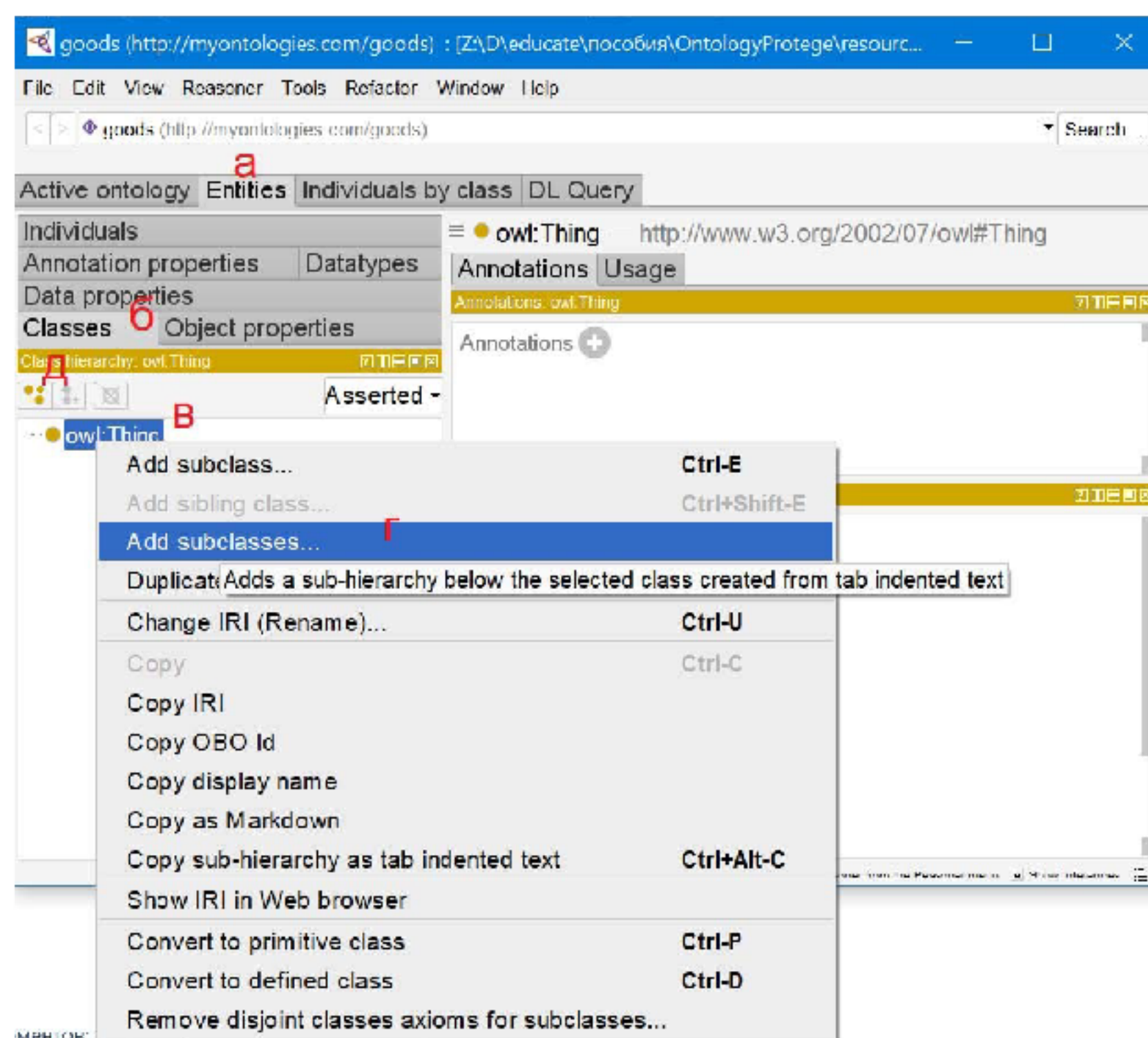


Рис. 4 Вкладка «Active ontology» в системе Protégé (а – панель в вкладок с активной вкладкой «Active ontology»; б – идентификатор онтологии)



ПКМ) и выбрать пункт «Add subclasses» (рис. 5г). Классы также могут добавляться по одному (рис. 5д).

При нажатии на кнопку «Add subclasses...» появляется окно добавления иерархии подклассов (рис. 6). Каждая введенная строка в данном окне создает отдельный класс (путем добавления в онтологию троек «`x rdf:type owl:Class . x rdfs:subClassOf y`»). Где `x` – IRI создаваемого класса, задаваемый как «`{IRI онтологии}#{обработанное_имя_класса}`» (в рамках обработки имени класса, происходит, например, замена пробела на «`_`»); `y` – IRI родительского класса, который задается как `owl:Thing` для классов, заданных без отступа и как IRI ближайшего верхнего класса с меньшим отступом для остальных (отступы задается как «`Tab`»).

После нажатия на «Continue» появляется окно, предлагающее создать для всех

инструментальная система создала на данный момент и лучше понять как она работает (в некоторых случаях поведение системы может быть настроено, здесь рассматривается поведение по умолчанию). Для этого необходимо сохранить онтологию (File -> Save As). При сохранении на данном этапе необходимо выбрать синтаксис «Turtle Syntax».

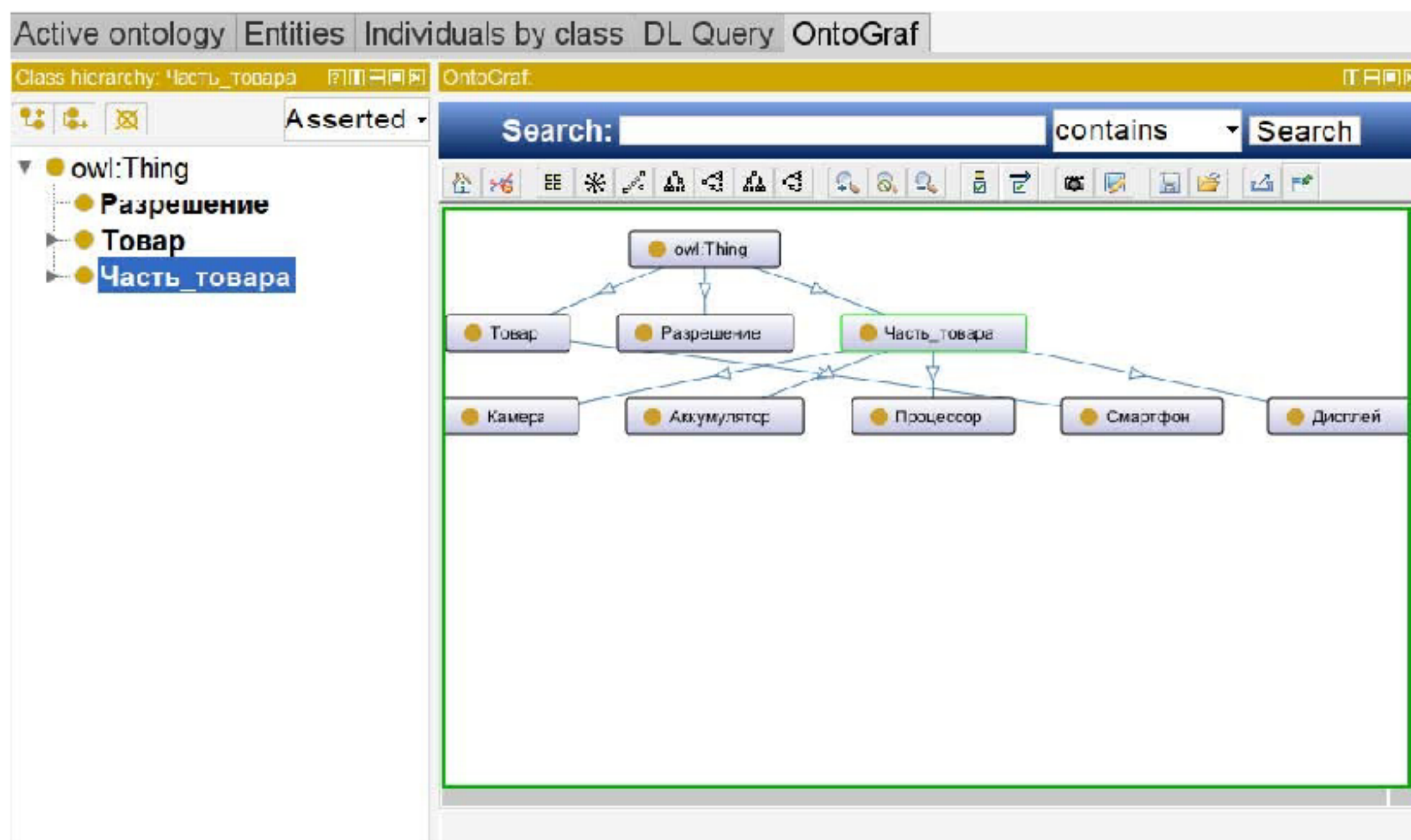


Рис. 7 Вкладка

:Аккумулятор rdf:type owl:Class ;

rdfs:subClassOf :Часть_товара .

#Т.е. это эквивалентно записи «:Аккумулятор rdf:type owl:Class . :Аккумулятор rdfs:subClassOf :Часть_товара»

#Для подклассов owl:Thing отношение rdfs:subClassOf может быть опущено при генерации файла

:Разрешение rdf:type owl:Class .

...

#Далее описываются общие аксиомы. В нашем случае это несколько условий на группы классов одного уровня, что никакие сущности не могут быть экземплярами одновременно нескольких классов из группы

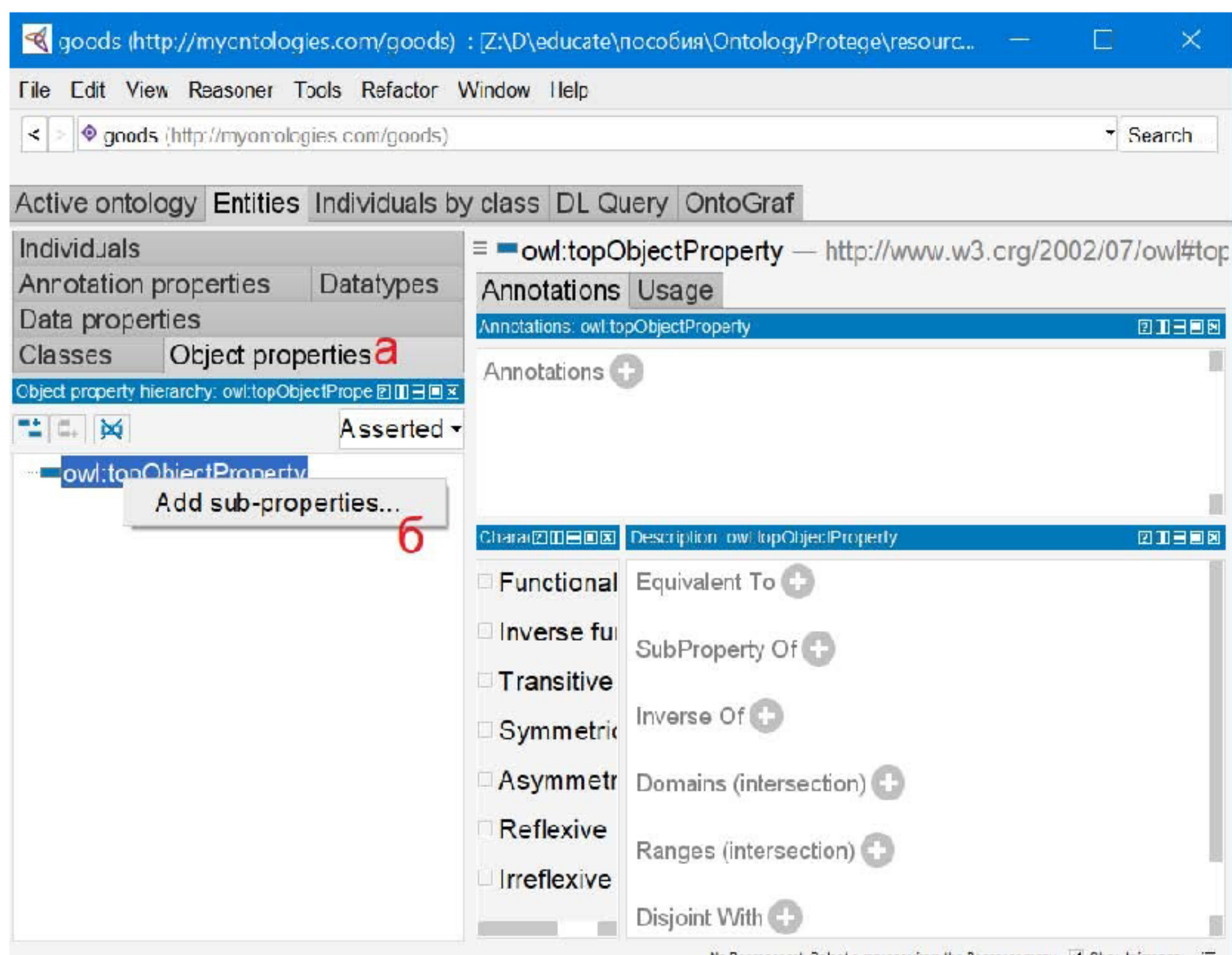


Рис. 8 Добавление Object Properties (а – активная вкладка Object Properties, б– кнопка добавления иерархии отношений)

На данном этапе необходимо задать свойства «часть» и «разрешение». Далее опишем эти свойства подробнее, а именно зададим область определения и область значения.

Для свойства «часть» область определения «Товар», область значения «Часть_товара». Для того, чтобы задать эту информацию, необходимо выбрать

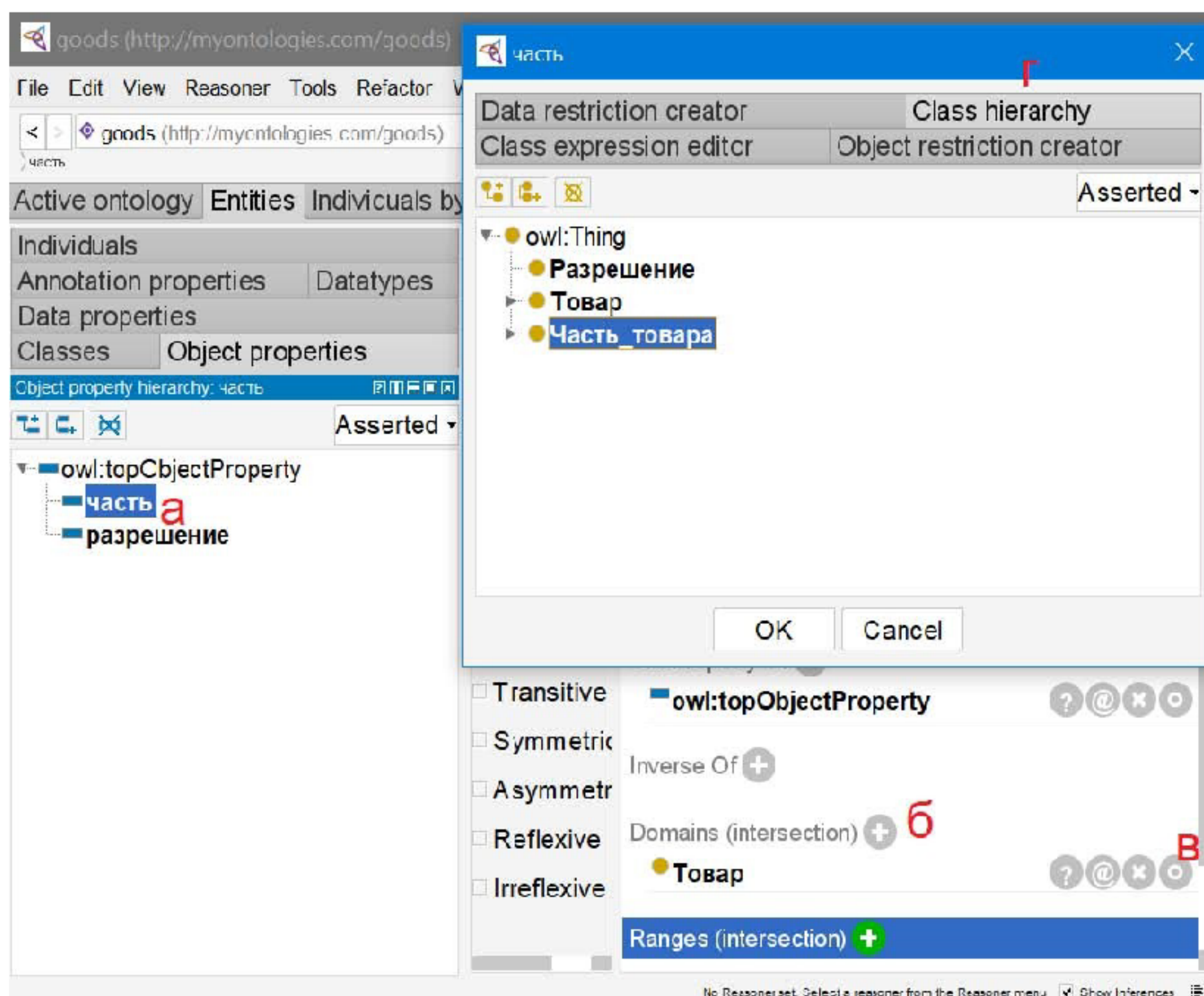


Рис. 9 Задание информации о Object Property в системе Protégé (а – выделенное свойство «часть», б – кнопка добавления новой области определения, в – кнопка редактирования существующей области определения, г – активная вкладка иерархии

год выпуска; число sim карт; диагональ экрана; разрешение горизонталь; разрешение вертикаль; разрешение кадр/сек; плотность пикселей; количество цветов экрана млн; количество ядер; частота процессора ггц; объем оперативной памяти гб; объем встроенной памяти гб; максимальный объем карты памяти гб; количество основных камер; количество мегапикселей основной камеры; апертура основной камеры; количество мегапикселей фронтальной камеры; количество фронтальных камер; емкость аккумулятора мА*ч; время работы в режиме разговора; время работы в режиме ожидания; ширина; высота; толщина; вес

логическое

автофокусировка основной камеры; оптическая стабилизация основной камеры; наличие вспышки фронтальной камеры; наличие беспроводной зарядки

свободное

конфигурация процессора; датчики; особенности основной камеры; комплектация; биометрическая защита; особенности

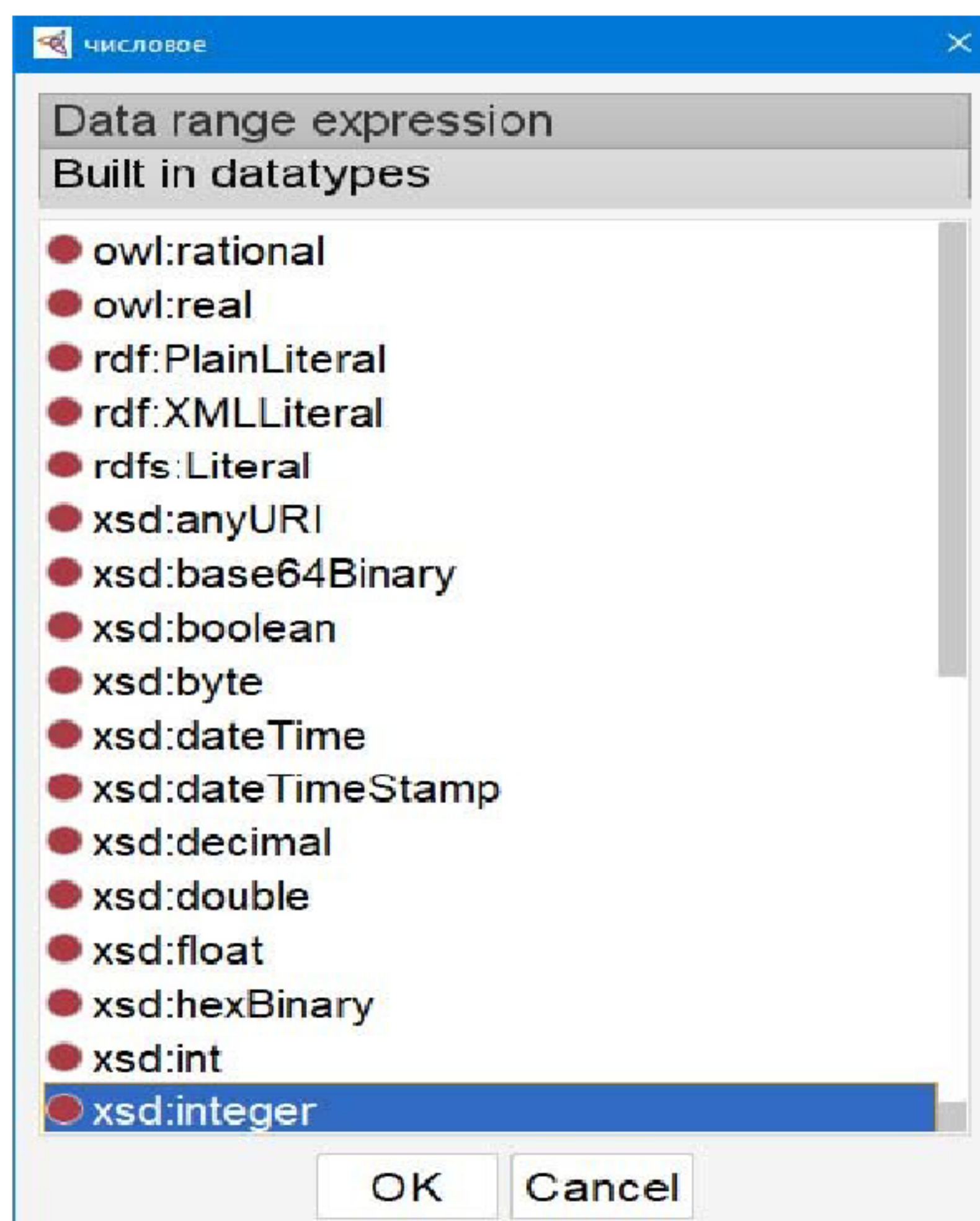


Рис. 10 Окно задания области значения для data property в системе Protégé

Для некоторых свойств необходимо задать дополнительные ограничения. Для этого используется вкладка «Data range expression» и Manchester Syntax. Разберем некоторые из свойств. Форматы сим карт стандартизированы: Полноразмерные (1FF), Mini-SIM (2FF), Micro-SIM (3FF), Nano-SIM (4FF), Встроенные

```
:формат_sim_карты rdf:type owl:DatatypeProperty ;  
    rdfs:subPropertyOf :текстовое ;  
    rdfs:range [ rdf:type rdfs:Datatype ; owl:oneOf ("1FF" "2FF" "3FF" "4FF"  
"E-SIM") ] .
```

Как видно, для описания области значений создается пустой узел, который является типом данных (экземпляром `rdfs:Datatype`) и его экземпляры могут принимать значения из заданного списка (за счет использования отношения `owl:oneOf`). При использовании ограничений на data property пустой узел `rdfs:Datatype` создается всегда, дальнейшие конструкции зависят от введенного разработчиком. Рассмотрим возможные конструкции для этого поля подробнее:

- 1 {} – задает список значений, в рамках которых могут быть только скалярные значения (возможно использовать задание типа данных через «[^]» и теги языков через «@»)
- 2 Любой тип данных

```
owl:unionOf ( [ rdf:type rdfs:Datatype ; owl:onDatatype xsd:integer ;  
    owl:withRestrictions ( [ xsd:minInclusive 0 ] [ xsd:maxInclusive 99 ] ) ]  
[ rdf:type rdfs:Datatype ; owl:onDatatype xsd:integer ;  
    owl:withRestrictions ( [ xsd:minExclusive 1973 ] ) ] ) ] .
```

Приведенное выражение несколько сложнее Manchester Syntax, но позволяет получить понимание как работают выражения в терминах RDF и как они, соответственно, потом обрабатываются.

Создаются 2 пустых узла с отношениями `xsd:minInclusive 0`, `xsd:maxInclusive 99`, которые объединяются в RDF список (`rdf:List` через сокращенное описание «()»). Далее создается новый пустой узел – тип данных, основанный на `xsd:integer` (`owl:onDatatype`), которые соединяется отношением `owl:withRestrictions` с созданным ранее списком.

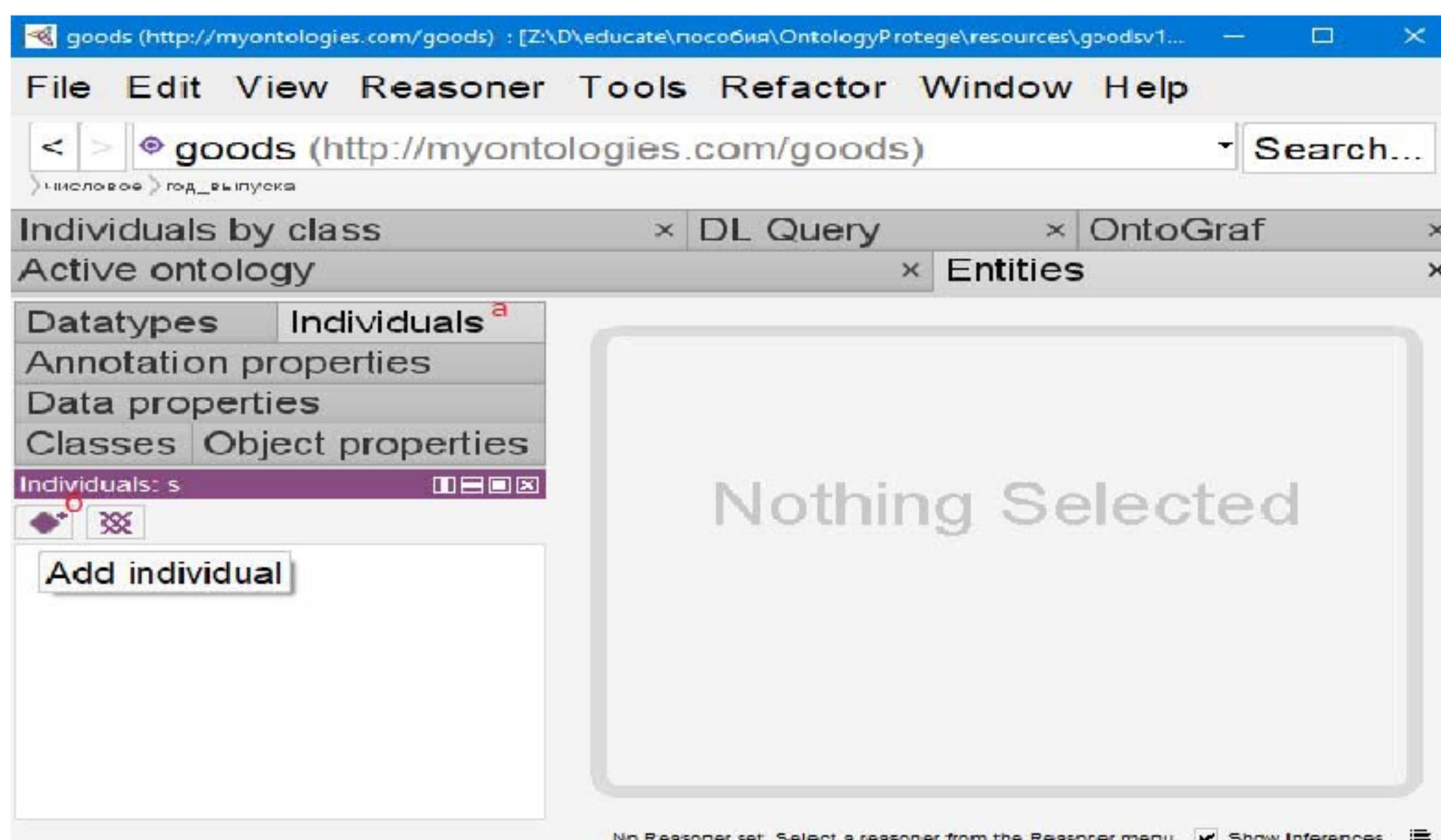
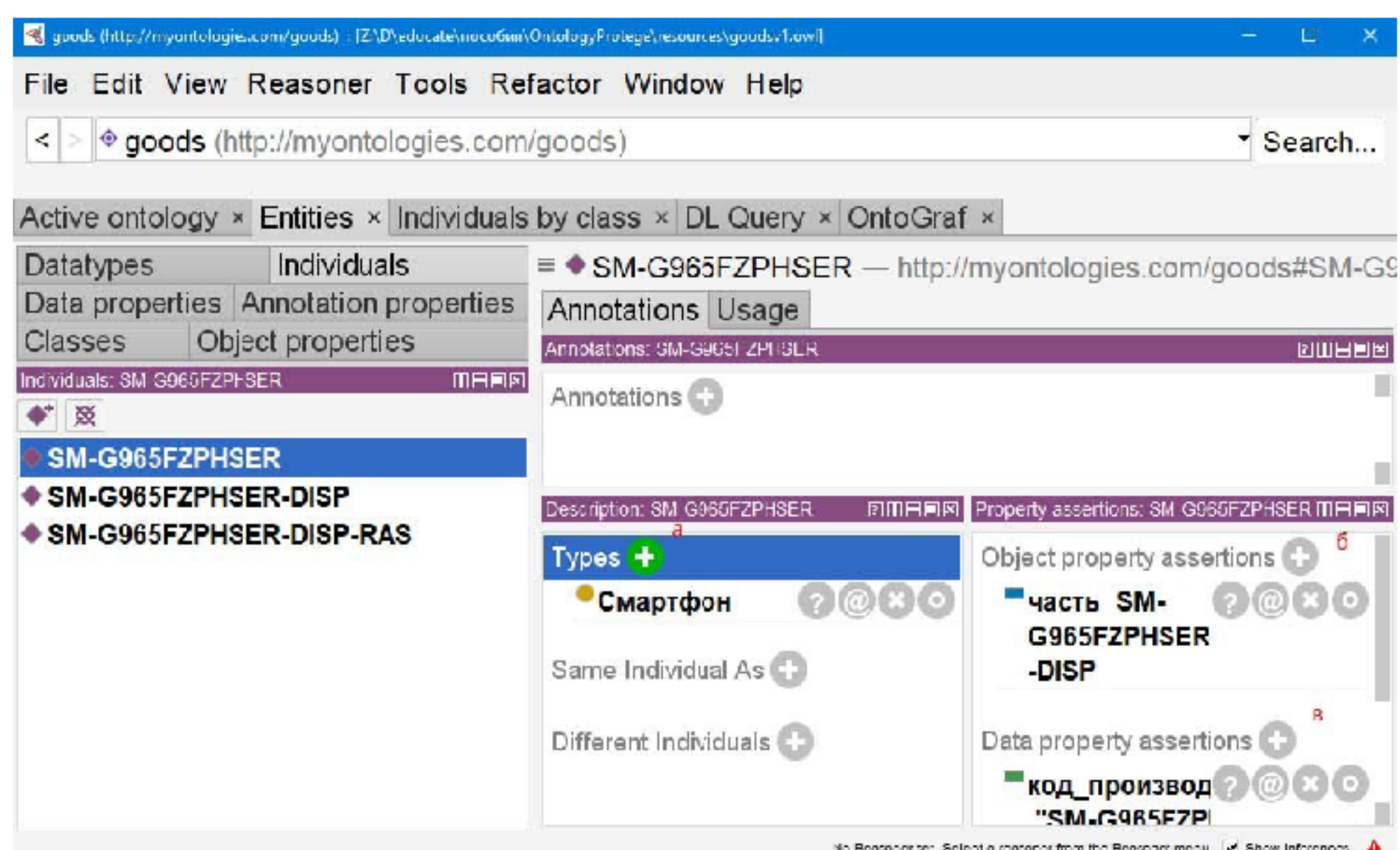


Рис. 12 Вкладка «Individuals» в системе Protégé (а – активная вкладка Individuals, б – кнопка создания нового экземпляра)



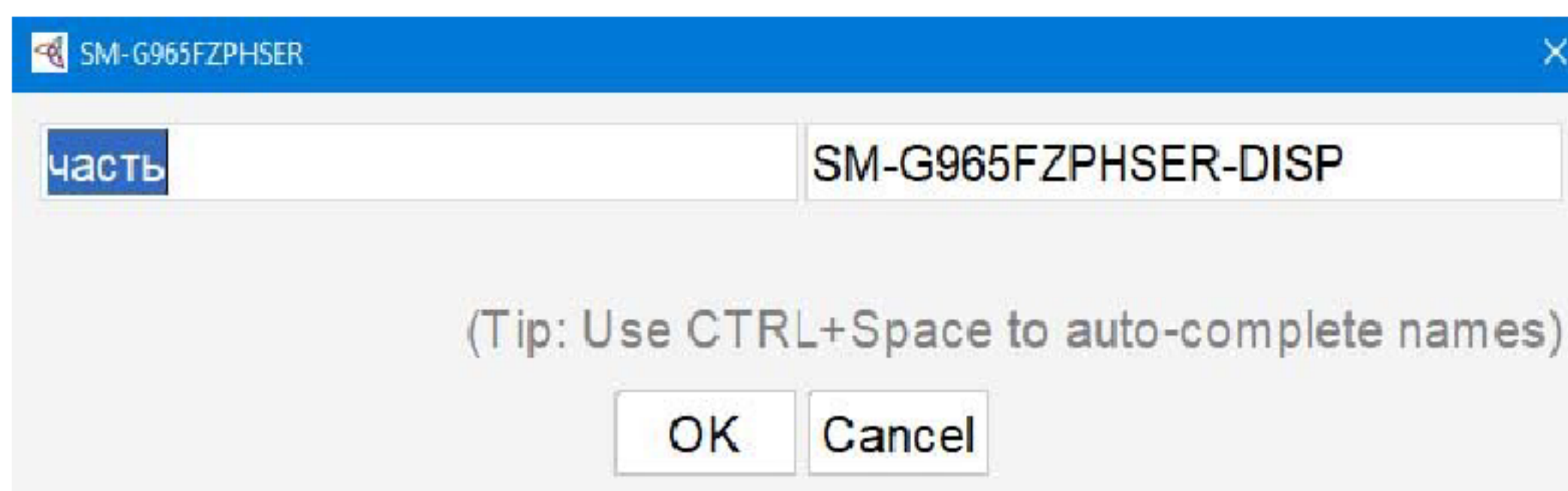
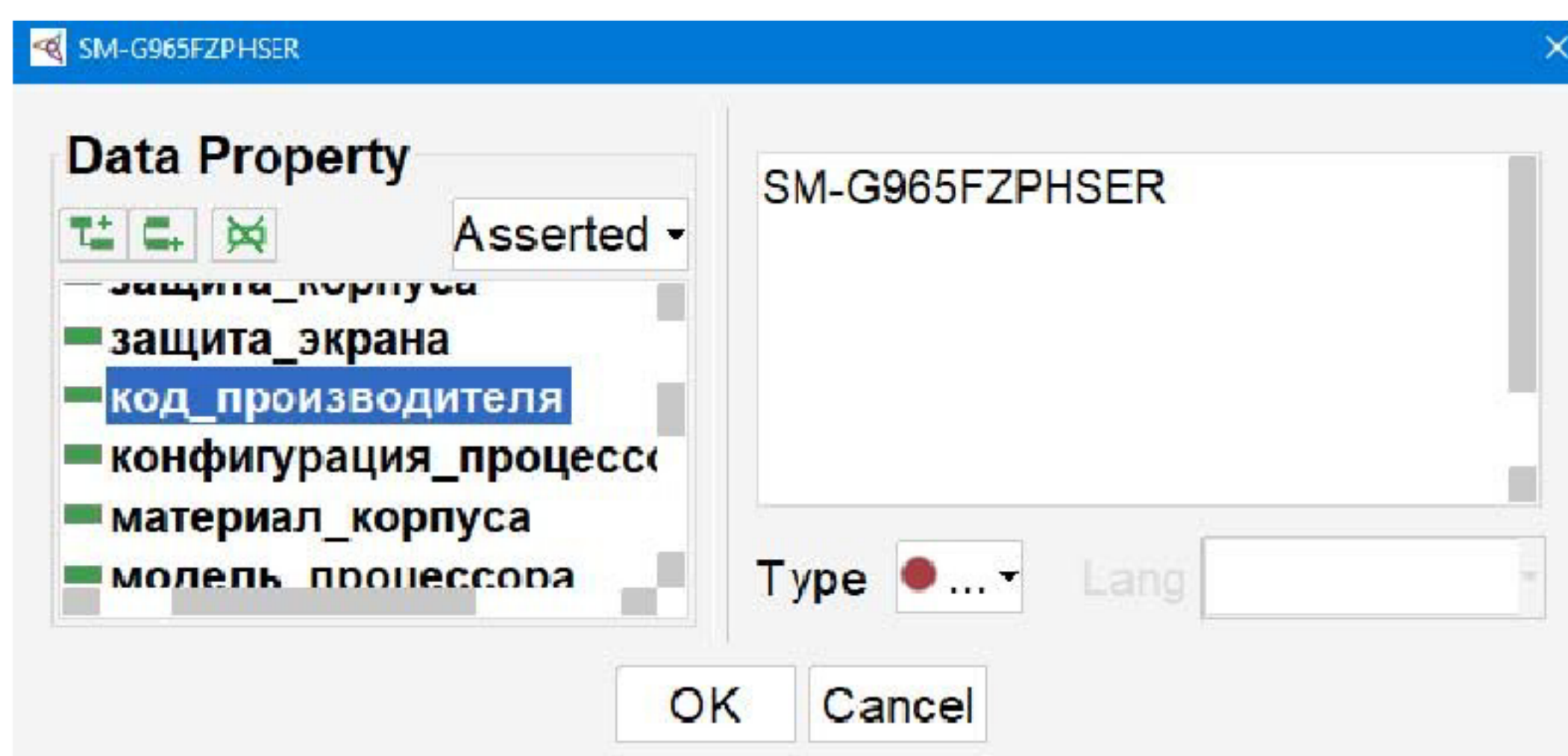


Рис. 14 Окно задания object property для экземпляра

После этого зададим отношение в разделе data properties на примере «код производителя» (рис. 15).



Однако, на данном этапе, при удалении этого отношения, в системе Protégé не будет видна разница. Дальнейшее описание достаточно большого количества экземпляров и их отношений проще сделать напрямую в файле онтологии с использованием любого текстового редактора. При этом можно было бы заменить некоторые экземпляры на пустые узлы:

```
:SM-G965FZPHSER-DISP rdf:type
    owl:NamedIndividual , :Дисплей ; :разрешение
    [rdf:type :Разрешение ; :разрешение_вертикаль
    1920 ; :разрешение_горизонталь 1080] .
```

Система будет понимать такое описание, однако в интерфейсе не будет указываться сам экземпляр класса «Разрешение» и, как следствие, не будет возможности редактирования его описания. При этом все возможности обработки, в том числе вывод в рамках системы Protégé остаются возможными. Тем не менее, для большей наглядности при открытии онтологии в системе, далее будет приведен текст экземпляров и их отношений без использования таких пустых узлов.

:комплектация "комплект амбушюр, комплект переходников, документация, сетевой адаптер, скрепка для извлечения слота SIM-карты, наушники";

:биометрическая_защита "сканер лица, сканер радужной оболочки глаза, сканер отпечатков пальцев";

:особенности "ANT+, технология Dolby Atmos, камера Vixby, камера-переводчик, селфимоджи, стереодинамики от AKG, несъемный аккумулятор" .

:SM-G965FZPHSER-DISP rdf:type owl:NamedIndividual , :Дисплей ;

:разрешение :SM-G965FZPHSER-DISP-RAS;

:технология_

```

:разрешение_вертикаль 1920 ;
:разрешение_горизонталь 1080 ;
#полный IRI необходим из-за спец символа "/"
<http://myontologies.com/goods#разрешение_кадр/сек> 240 .
:SM-G965FZPHSER-CAM-FRONT rdf:type
    owl:NamedIndividual , :Камера ; :количество_мегапикселей_фронтальной_камеры 8;
:количество_фронтальных_камер 1 ; :разрешение :SM-G965FZPHSER-DISP-CAM-
FRONT-RAS-1 ; :наличие_вспышки_фронтальной_камеры 0 .
:SM-G965FZPHSER-DISP-CAM-FRONT-RAS-1 rdf:type owl:NamedIndividual ,
:Разрешение ;
:разрешение_вертикаль 2560 ;
:разрешение_горизонталь 1440 ;
#полный IRI необходим из-за спец символа "/"
<http://myontologies.com/goods#разрешение_кадр/сек> 30 .
:SM-G965FZPHSER-BAT rdf:type owl:NamedIndividual , :Аккумулятор ;
```


классы в системе Protégé называются «Defined Classes» («обычные» классы называются «Primitive Classes»). Для примера будут реализованы следующие классы: «Новый смартфон», «Камерофон», «Игровой смартфон», «Смартфон для фильмов», «Смартфон для походов». Это позволит образовать в онтологии группы экземпляров в соответствии с заранее заданными критериями, которые будут доступны для дальнейшей обработки.

Первоначально сформируем критерии, по которым экземпляры будут относиться к заявленным классам:

- 1 Новый смартфон. Выпущен после 18 года.
- 2 Камерофон. Минимум 2 основные камеры. Минимум 12 мегапикселей. Есть оптическая стабилизация. Съемка видео минимум Ultra HD 4K 60кадр./сек.
- 3 Игровой смартфон. Процессор Snapdragon или Exynos. Минимум 6 ядер. Частота работы процессора минимум 2.9ГГц. Объем оперативной памяти минимум 6Гб. Графический ускоритель Mali или Adreno.
- 4 Смартфон для фильмов. Игровой смартфон. Соотношение сторон

отсутствуют заданные отношения или они все удовлетворяют выражению. Например: «год_выпуска only xsd:integer[>= 18].

4. <локальное_имя_свойства> exactly <количество>. Все экземпляры, для которых задано указанное количество указанных отношений. Например: «год_выпуска exactly 1»

5. <локальное_имя_свойства> exactly <количество> <выражение>. Все экземпляры, для которых задано указанное количество указанных отношений, удовлетворяющих выражению. При этом может быть сколько угодно таких же отношений, не удовлетворяющих выражению. Например: «год_выпуска exactly 1 xsd:integer[>= 18]»

6. <локальное_имя_свойства> min/max <количество> [<выражение>]. Аналогично exactly, но должно быть не менее/не более, чем указанное количество указанных отношений.


```

(разрешение some (Разрешение and
  ((разрешение_горизонталь some xsd:integer[ >= 3840 ])
  or (разрешение_вертикаль some xsd:integer[ >= 3840 ]))
  and (разрешение_кадр/сек some xsd:integer[ >= 60 ])
)))
)

```

Задания

- 1 Составить запросы для оставшихся групп: «Смартфон для фильмов», «Игровой смартфон», «Смартфон для походов». Создать defined class для каждой.
- 2 Сформировать минимум 5 групп (на основании имен подборок на ресурсе источнике) для категорий «Ноутбуки» и «Холодильники». Для каждой из групп сформировать критерии выбора экземпляров и создать defined class.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ

Методические указания
для обучающихся по организации и проведению самостоятельной работы
по дисциплине «Искусственный интеллект в профессиональной сфере»
для направления подготовки 38.03.01 Экономика
направленность (профиль) Корпоративная экономика и финансы

**Пятигорск
2022**

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 12000002A633E3D1

СОДЕРЖАНИЕ

1. Общие положения.....	73
2. Цель и задачи самостоятельной работы.....	74
3. Технологическая карта самостоятельной работы студента.....	74
4. Порядок выполнения самостоятельной работы студентом.....	75
4.1. Методические рекомендации по работе с учебной литературой.....	75
4.2. Методические рекомендации по подготовке к практическим и лабораторным занятиям.....	76
4.3. Методические рекомендации по самопроверке знаний.....	77
4.4. Методические рекомендации по написанию научных текстов (докладов, рефератов, эссе, научных статей и т.д.).....	77
4.5. Методические рекомендации по	

1. Методические указания к самостоятельной работе (электронный вариант)

Интернет-ресурсы:

1. Воройский Ф. С. Информатика. Энциклопедический словарь-справочник: введение в современные информационные и телекоммуникационные технологии в терминах и фактах. - М.: ФИЗМАТЛИТ, 2006. - 768 с. – Доступно: <http://physics-for-students.ru/bookpc/informatika/slovar.zip>
2. Иванов В. Основы искусственного интеллекта – <https://libtime.ru/expertsystems/osnovy-iskusstvennogo-intellekta.html>
3. Романов П.С. Основы искусственного интеллекта; Учебно-метод. пособие. – <http://www.studfiles.ru/preview/2264160/></