

ООО «ТриниДата»

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ
редактора онтологий Onto.pro

г. Екатеринбург, 2013-2018

Оглавление

1.	Основные понятия онтологического моделирования.....	3
1.1.	Уникальные идентификаторы объектов	4
1.2.	Свойства	4
1.3.	Пример онтологии.....	5
2.	Работа с элементами онтологии.....	7
2.1.	Создание и редактирование классов	7
2.2.	Сортировка и поиск классов	10
2.3.	Создание свойств-литералов.....	11
2.4.	Создание свойств-указателей на объекты	15
2.5.	Создание индивидуальных объектов. История значений атрибутов .	16
2.6.	Прикрепление файлов и комментариев	19
2.7.	Управление списком объектов и свойств	20
2.8.	Групповые операции с элементами онтологии.....	21

Работа в редакторе Onto.pro состоит в создании онтологических информационных моделей, которые могут использоваться для широкого круга применений. Для работы с ним необходимо понимание концепций онтологического моделирования.

Данное руководство дополняет методическое пособие по онтологическому моделированию, расположенное по адресу: <http://trinidata.ru/files/SemanticIntro.pdf>. Рекомендуем обратиться к этому пособию для получения подробной информации о методике и технологиях создания онтологических моделей.

Операции, обычно требующие наличия административных прав – импорт/экспорт данных через Excel, настройка прав доступа – описаны в Руководстве администратора Onto.pro.

Результатом работы в Onto.pro является онтология, которая затем может быть использована в системе управления мастер-данными АрхиГраф.MDM, системе управления знаниями АрхиГраф.СУЗ, экспортирована в файл формата RDF/XML, или использоваться непосредственно через точку доступа SPARQL, в которую сохранялась информация.

1. Основные понятия онтологического моделирования

В онтологическом моделировании все сущности условно делятся на несколько типов, основные среди которых – *индивидуальные объекты* и *классы* (совокупности индивидуальных объектов).

Отношение между классами и индивидами можно проиллюстрировать на примере таким образом:

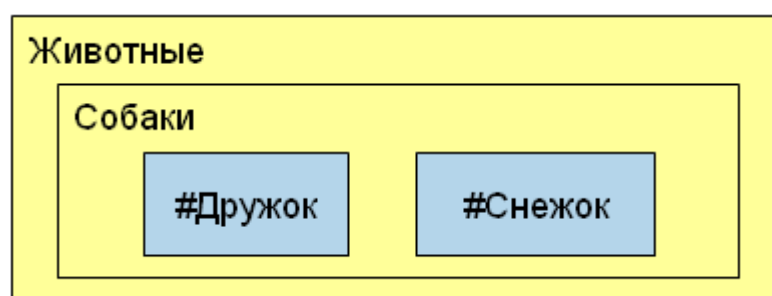


Рис. 1. Классы и индивидуальные объекты

На этой диаграмме «Животные» и «Собаки» – классы, причем «Собаки» является подклассом класса «Животные». «#Дружок» и «#Снежок» – два индивидуальных объекта, принадлежащие классу «Собаки».

Важно подчеркнуть, что класс является произвольным объединением сущностей. Определение класса не содержит формализованной информации о том,

на каком основании выделен именно такой класс, по какому принципу сущности сгруппированы в нем – все это остается на усмотрение инженера-онтолога.

Индивидуальные объекты могут принадлежать классам (один индивид – любому числу классов одновременно). Классы, в свою очередь, могут находиться в различных отношениях друг с другом. Например, на рисунке выше показано, что класс «Собаки» полностью входит в класс «Животные», то есть является его подклассом.

1.1. Уникальные идентификаторы объектов

У каждого объекта (как класса, так и индивида) есть свой уникальный идентификатор – URI, который выглядит таким образом:

`http://some.address/MyOntology#ObjectId`

или

`http://some.address/MyOntology/ObjectId`

Уникальный идентификатор состоит из двух частей: первая часть, до символа решетки или последнего символа /, называется префиксом, и обычно обозначает ту онтологию (модель, совокупность объектов), к которой он относится. После решетки или последнего символа слэша (/) следует собственно уникальный идентификатор объекта. Префикс при записи или отображении часто пропускается, или заменяется на сокращение. Например, в указанном выше примере может быть введено сокращение

`http://some.address/MyOntology# = MyOntology`

Тогда URI указанного выше объекта будет записываться как `MyOntology:ObjectId`.

1.2. Свойства

Индивидуальные объекты могут иметь свойства – точнее, обладать значениями свойств. Само по себе свойство (его определение) – это сущность онтологии, существующая отдельно от классов (в отличие от ООП, где свойство принадлежит классу).

В онтологических моделях определение каждого свойства имеет две важнейших характеристики: диапазон значений (*range*), определяющий, какие типы величин могут становиться значением этого свойства (фиксированное значение – литерал, или индивидуальный объект какого-либо класса), и область применения (*domain*) – список классов, к индивидам которых применимо данное свойство. У одного свойства может быть любое число как диапазонов значений, так и классов,

к которым оно применимо. Например, свойство «имя» применимо как к объектам класса «Люди», так и к объектам класса «Собаки». Это принципиальный момент, отличающий онтологическое моделирование от создания реляционных схем данных.

На диаграмме это можно проиллюстрировать так:

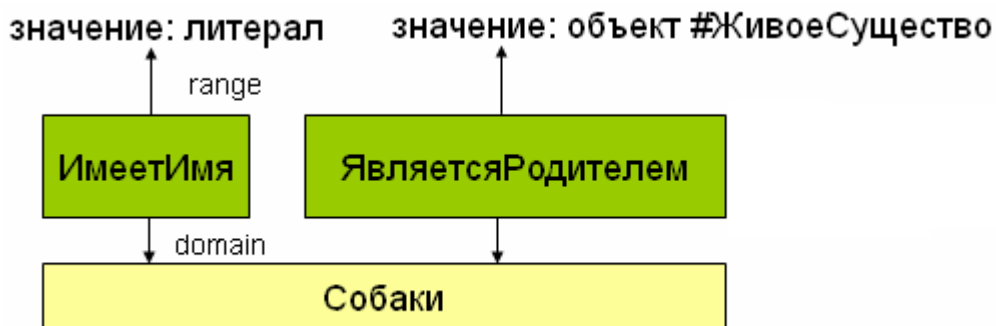


Рис. 2. Диапазоны значений и применимость свойств

Редактор Onto.pro позволяет создавать сущности всех этих типов – классы, индивидуальные объекты, свойства-литералы, свойства-указатели на объекты.

1.3. Пример онтологии

Продемонстрируем пример онтологии, содержащей определения классов, индивидуальных объектов, свойств и их значений.

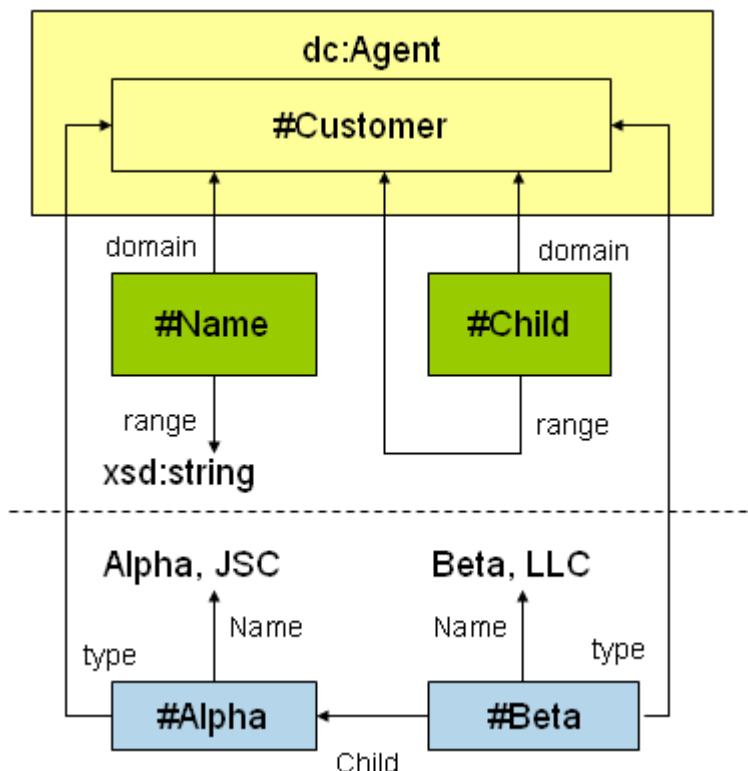


Рис. 3. Пример онтологии

Все, что показано выше пунктирной линии, является определениями (классов и свойств); ниже линии показаны индивидуальные объекты и значения свойств. С технической точки зрения, сущности этих типов выражаются различными элементами OWL: классы являются сущностями типа Class, индивидуальные объекты – NamedIndividual, обычные свойства – DatatypeProperty, свойства-указатели на другие объекты – ObjectProperty.

Выше линии показан класс Customer, являющийся подклассом класса Agent, определенного в стандартной онтологии Dublin Core (о чем говорит префикс dc:). Два свойства, Name и Child (дочерняя организация), применимы к индивидам этого класса. Значения свойства Name имеют тип «строка», значения свойства Child являются другими объектами того же класса Customer.

Ниже линии показаны два индивидуальных объекта, относящихся к классу Customer – Alpha и Beta. Они имеют имена (литералы), плюс свойство Child объекта Beta указывает на объект Alpha.

2. Работа с элементами онтологии

Все примеры в этом разделе будем рассматривать на следующей онтологии:



Рис. 4. Пример онтологии

В верхней части диаграммы мы видим определения трех классов: «Персона», «Организация», и подкласс организаций – «Компания». Организации и персоны имеют свойство-литерал «Адрес», которое может принимать любое количество значений. У персоны есть также свойство «Дата рождения», которое имеет только одно значение для каждой персоны. Каждая компания может иметь несколько персон-учредителей – эти отношения выражаются свойством «имеет учредителя», типом значений для которого являются объекты класса «Персона».

В нижней части диаграммы показаны экземпляры и значения свойств. Организация ООО «Альфа» расположена по адресу ул. Вишневая, 1. Ее учредителями являются граждане Иванов и Петров. Известна дата рождения Иванова – 01.01.1970.

2.1. Создание и редактирование классов

Для удобства визуального восприятия, классы в Onto.pro представлены в виде иерархии, построенной на отношениях «является подклассом». На самом деле, настоящей иерархии у классов может и не быть; иерархия является лишь одним,

частным случаем организации взаимоотношений классов – так, один класс может быть подклассом сразу нескольких классов.

После входа в только что созданную точку доступа (процесс управления точками доступа описан в Руководстве администратора), в дереве слева присутствует только один автоматически созданный класс-пример:

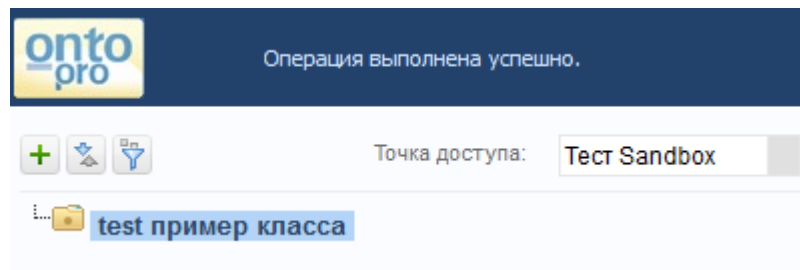


Рис. 5. Начальный вид дерева классов

Чтобы создать новый класс, нажмем кнопку «+» на панели инструментов (кнопка показана на рисунке выше). Справа появится форма создания класса:

The screenshot shows the 'Создать класс' (Create Class) form. At the top, it displays the prefix 'Префикс: http://www.onto.pro/test/' and a '+ СОЗДАТЬ' button. The form is divided into two main sections: 'Сущность' (Entity) and 'Надклассы' (Superclasses). The 'Сущность' section contains four fields: 'Идентификатор' (Identifier), 'Название' (Name), 'Определение' (Definition), and 'Определен в' (Defined in). The 'Идентификатор' field is empty. The 'Название' and 'Определение' fields are empty and have a small grid icon in the bottom right corner. The 'Определен в' field has a '+-' icon and an 'X' icon. Below the 'Сущность' section, there is a checkbox labeled 'Удален' (Deleted) which is currently unchecked. The 'Надклассы' section is currently collapsed and has a '+ СОЗДАТЬ' button at the bottom. A note below the 'Надклассы' section reads 'поля, отмеченные *, обязательны для заполнения' (fields marked with * are mandatory for filling).

Рис. 6. Форма создания класса

Введем в поле «Название» слово «Организация». После нажатия кнопки «Создать», оно будет автоматически скопировано в идентификатор (при этом будут удалены пробелы и спецсимволы, если таковые присутствовали во введенном тексте). Идентификатор – это завершающая часть уникального идентификатора объекта

(URI). Он не должна содержать пробелов и спецсимволов, поэтому при сохранении записи они будут отфильтрованы. Название – это читаемое название сущности (label), которое будет использоваться при ее выборе в редакторе.

После нажатия кнопки «Создать» класс «Организация» появится в дереве слева. Теперь нам нужно создать его подкласс – «Компания». Для этого нажмем кнопку «+» на панели инструментов еще раз. Форма в середине страницы снова очистится, и мы сможем ввести название и ярлык для нового класса – «Компания». Обратим внимание на нижнюю часть формы:

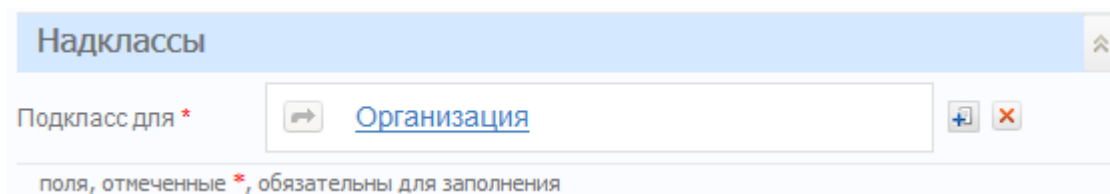
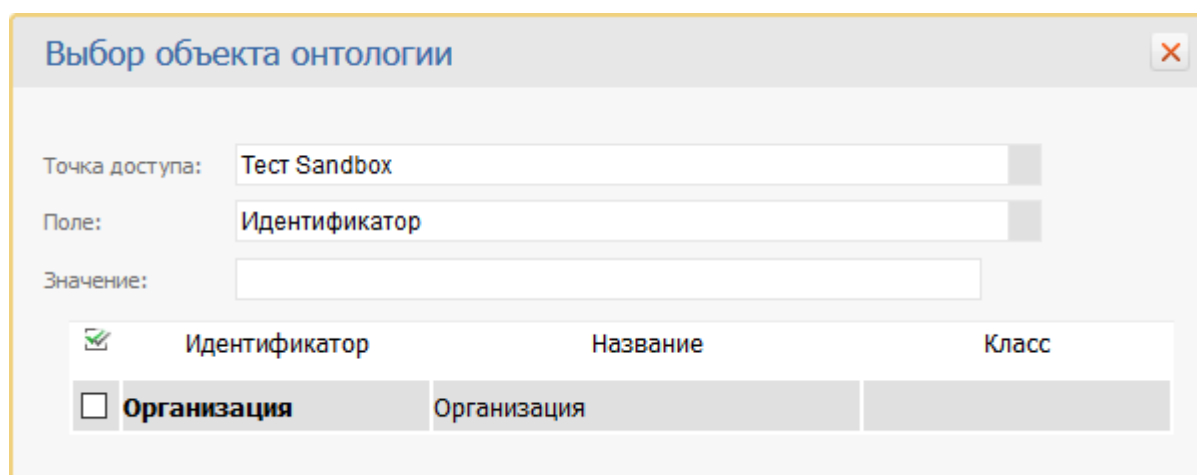


Рис. 7. Выбор надкласса в форме создания класса

В области «Надклассы» указывается родительский класс для того класса, который мы создаем. По умолчанию здесь выбирается последний класс, который мы создавали или просматривали. Нажатием на красную кнопку «X» можно очистить выбранный класс – эта возможность нам понадобится, когда мы будем создавать класс «Персона», находящийся на верхнем уровне иерархии. Нажатие на синюю кнопку «+» открывает диалоговое окно выбора объекта – здесь можно выбрать любой другой надкласс для создаваемого класса:



	Идентификатор	Название	Класс
<input checked="" type="checkbox"/>	Идентификатор		
<input type="checkbox"/>	Организация	Организация	

Рис. 8. Выбор объекта онтологии

В этом диалоговом окне можно быстро найти нужный элемент подбором по части названия. Щелчок на имени элемента выбирает его.

При помощи этого же инструмента можно перенести уже созданный класс из одной ветви онтологии в другую (изменить его надкласс), или добавить новый надкласс в дополнение к существующему.

После того, как класс создан, в форме «Надклассы» справа появится зеленая кнопка «+», при помощи которой можно будет добавить еще надклассы для данного класса. Мы можем убедиться в этом, создав класс «Компания», и посмотрев в область «Надклассы».

После создания трех классов из нашего примера, дерево классов примет следующий вид:

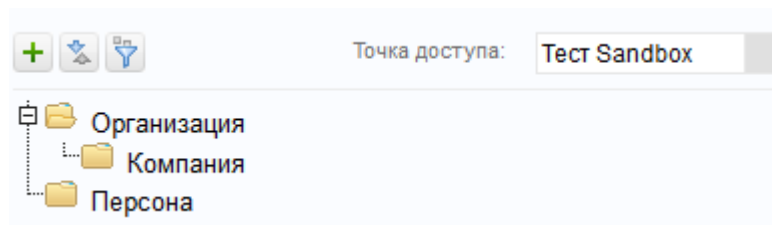


Рис. 9. Дерево классов

2.2. Сортировка и поиск классов

Над деревом расположены кнопки сортировки и фильтрации элементов. Нажатие на кнопку сортировки вызывает диалоговое окно:

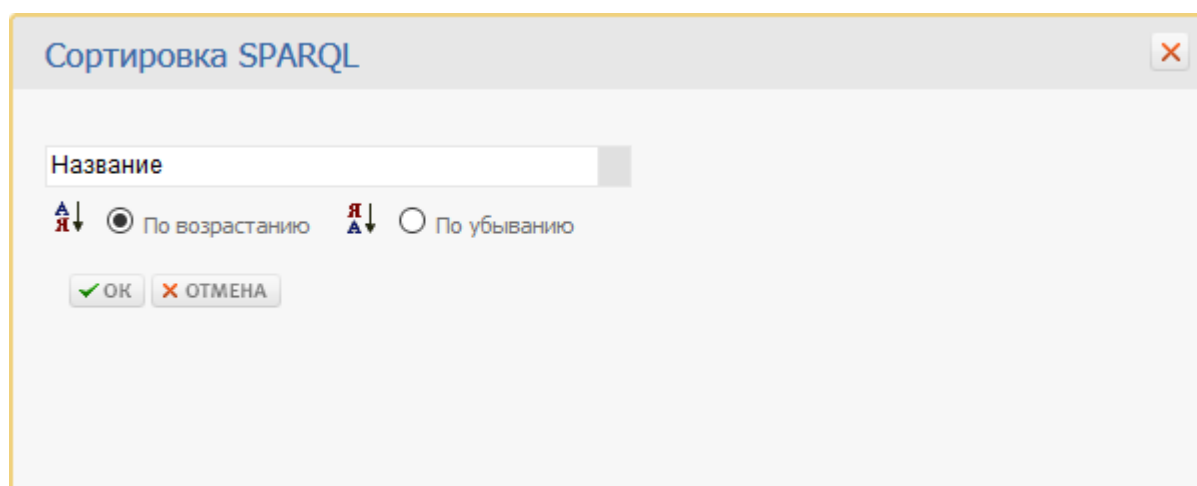


Рис. 10. Диалоговое окно сортировки

В этом окне можно выбрать свойство, по которому происходит сортировка (классы можно сортировать только по названию), и ее направление. Аналогично, нажатие на кнопку фильтра позволяет установить отбор классов – например, по содержанию какой-либо подстроки:

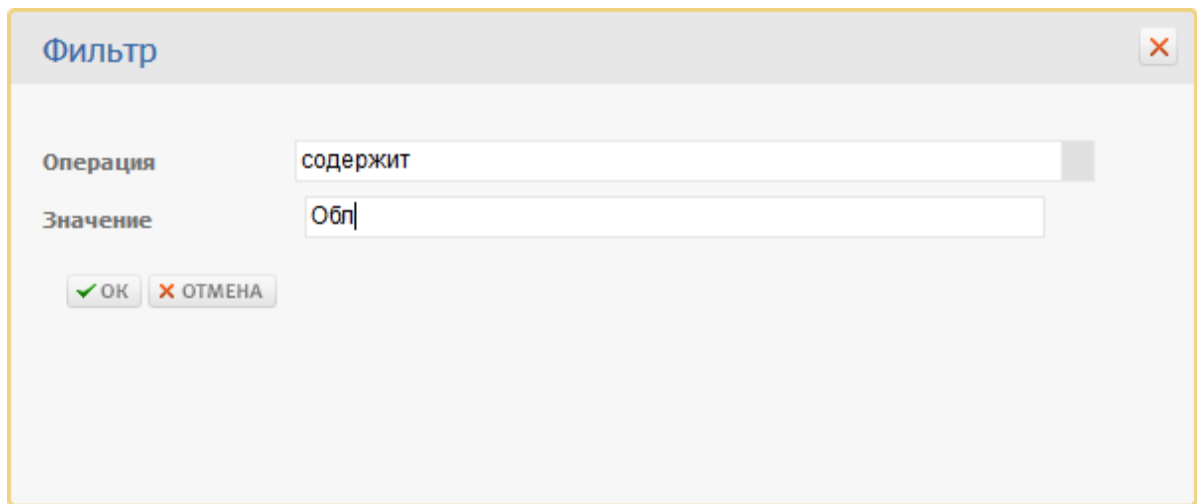


Рис. 11. Диалоговое окно установки фильтра

Информация об установленной сортировке и фильтрации отображается над деревом:

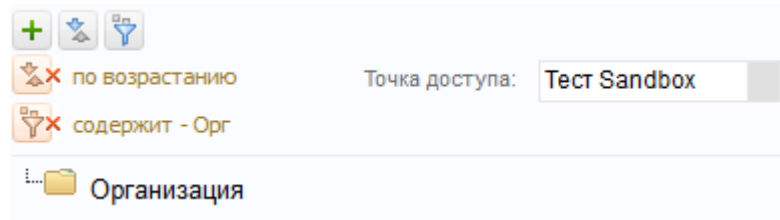


Рис. 12. Установленные фильтр и сортировка

Необходимо иметь в виду, что если класс, содержащий введенные символы, находится не на верхнем уровне дерева, то он не будет отображен, т.к. его вышележащие классы, скорее всего, не будут удовлетворять условию фильтра. Для поиска классов (и любых других сущностей) по названию лучше воспользоваться строкой поиска, расположенной в верхней части окна редактора. По мере ввода символов в нее, будут появляться подсказки. Нажатием на любую подсказку можно перейти непосредственно к данному объекту, а нажатием Enter или кнопки «Искать» – к странице результатов поиска, где будут отображены все найденные варианты.

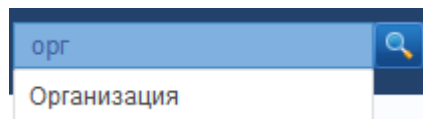


Рис. 13. Форма быстрого поиска

2.3. Создание свойств-литералов

Чтобы иметь возможность выразить содержательные сведения о наших объектах, необходимо создать для них наборы свойств. Как мы уже говорили,

свойства делятся на два типа: одни принимают значения-литералы (строки, числа и т.д.), другие предназначены для связывания разных объектов между собой.

Начнем с создания свойства-литерала. В дереве выберем класс, объекты которого будут обладать данным свойством. Переключимся на просмотр списка свойств-литералов, выбрав соответствующую кнопку над списком сущностей, связанных с этим классом (расположен справа от дерева):

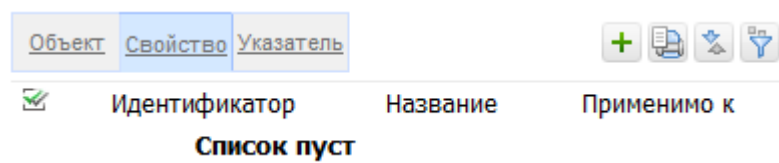


Рис. 14. Пустой список объектов/свойств

После нажатия кнопки «+» в панели инструментов этого списка, убедимся, что в форме создания сущности выбран ее тип – «Свойство». Нам нужно создать свойство «Адрес»; заполненная форма его создания будет выглядеть так:

Префикс: <http://www.onto.pro/test/>

Сущность

Идентификатор

Название

Определение

Определен в

Удален

Применимо к *

Диапазон значений *

Мин. кол-во

Макс. кол-во

поля, отмеченные *, обязательны для заполнения

Рис. 15. Форма создания свойства-литерала

Обратите внимание, что значение в поле «Диапазон значений» мы выбрали при помощи диалогового окна, в котором отображается список типов данных XSD, поддерживаемых Onto.pro.

В поле «Применимо к» можно указать не один, а несколько классов. Как именно будет трактоваться в этом случае область применимости свойства, зависит от значения константы «Сохранять множественные domain и range как объединение вместо пересечения»:

- значение константы «нет». В этом случае свойство применимо только к тем элементам, которые принадлежат одновременно всем перечисленным в поле «Применимо к» классам.

- значение константы «да» (предустановлено по умолчанию). В этом случае свойство относится к элементам, принадлежащим хотя бы к одному из перечисленных классов.

Управлять значением констант может администратор Onto.pro. Данная настройка обычно выполняется при разворачивании системы и в дальнейшем не изменяется. В последующих примерах будем полагать, что установлено значение «да» для этой константы.

После того, как свойство будет создано, укажем, что оно присуще не только объектам класса «Компания», но и персонам. Для этого нужно нажать зеленую кнопку «+» справа от первого значения поля «Применимо к». Появится вторая строка, для следующего экземпляра поля; нажмем на кнопку выбора значения, и в появившемся диалоговом окне выберем (или найдем по названию) класс «Персона». Результат будет выглядеть так:

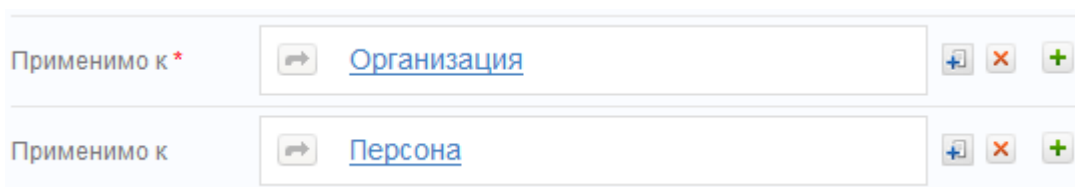


Рис. 16. Указание классов, к объектам которых применимо свойство

Теперь форму нужно сохранить; после этого можно будет добавить еще один экземпляр значения свойства, и так далее.

Убедимся, что мы создали свойство, которое присуще объектам двух разных классов. Выберем в дереве класс «Организация», и установим переключатель типов сущностей в расположенном справа списке в положение «Свойства». В списке появится наше свойство – «Адрес»:

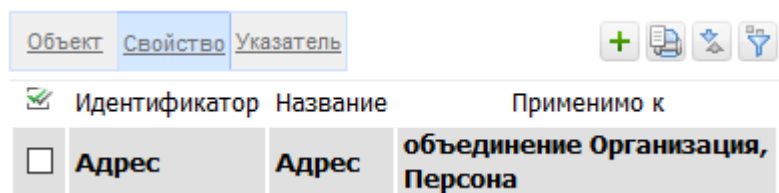


Рис. 17. Созданное свойство в списке

Выбрав в дереве класс «Персона», можно убедиться, что свойство «Адрес» действительно и для него. Теперь нужно создать второе свойство из нашего примера – «Дата рождения». Делается это по аналогии с созданием адреса, только в качестве типа значения надо выбрать «Дата», а в поле «Применимо к» оставить только одно значение – «Персона». Также, поскольку каждая персона имеет одну и только одну дату рождения, в поля «Мин. количество» и «Макс. количество» нужно ввести цифры 1. Эти поля указывают, какое количество значений может иметь данное свойство для каждого объекта. Возможность иметь более одного значения для каждого свойства – одно из отличий семантической информационной модели от реляционной.

Рис. 18. Создание свойства типа «дата», с ограничением числа значений

Свойства готовы. Рассмотрим создание свойств-указателей на объекты, а затем перейдем к созданию экземпляров, и заполнению значений свойств.

2.4. Создание свойств-указателей на объекты

Свойства-указатели на объекты создаются точно также, как и обычные свойства; нужно только устанавливать переключатель типа сущности в положение «Свойство-указатель на объект».

Поле «Диапазон значений» в этом случае приобретает тип «выбор объекта». Здесь мы должны выбрать класс, на экземпляры которого будет ссылаться данное свойство. Нам нужно создать свойство «Является учредителем». Для этого выбираем в дереве класс, объекты которого будут иметь это свойство – «Компания»; в списке справа выбираем переключатель «Свойства-указатели на объект», и нажимаем кнопку «+» («Добавить»). Заполняем форму создания свойства:

Сущность	
Идентификатор	<input type="text"/>
Название	Имеет учредителя
Определение	<input type="text"/>
Определен в	<input type="button" value="+"/> <input type="button" value="x"/>
Удален	<input type="checkbox"/>
Применимо к *	<input type="button" value="→"/> Компания <input type="button" value="+"/> <input type="button" value="x"/>
Диапазон значений *	<input type="button" value="→"/> Персона <input type="button" value="+"/> <input type="button" value="x"/>
Мин. кол-во	<input type="text"/>
Макс. кол-во	<input type="text"/>
поля, отмеченные *, обязательны для заполнения	
<input type="button" value="+ СОЗДАТЬ"/>	

Рис. 19. Создание свойства-указателя на объект

Так же, как для обычных свойств, в поле «Применимо к» можно указать несколько классов. Кроме того, для свойств-ссылок можно указать несколько классов для поля «Диапазон значений». Область применимости свойств и диапазон значения в случае задания нескольких классов в полях «Применимо к» и «Диапазон значений» определяются исходя из значения константы «Сохранять множественные

domain и range как объединение вместо пересечения» так же, как это было для свойств-литералов:

- значение «нет» означает, что область применения/диапазон значений состоят из объектов, принадлежащих одновременно всем указанным классам

- при заданном по умолчанию значении константы «да» область применения/диапазон значений включают в себя объекты, принадлежащие хотя бы одному из классов.

2.5. Создание индивидуальных объектов. История значений атрибутов

Перейдем к созданию индивидуальных объектов, относящихся к ранее созданным классам. Для этого выберем класс-родитель в дереве, а затем нажимаем кнопку «+» внизу, над списком сущностей, относящихся к данному классу. Для начала, создадим конкретный персон. В средней части страницы появится форма, которую надо заполнить таким образом:

Сущность

Идентификатор

Тип * [Персона](#)

Название Иванов И.И.

Определение

Определен в

Удален

поля, отмеченные *, обязательны для заполнения

+ СОЗДАТЬ

Рис. 20. Создание объекта

Обратите внимание, что над списком активен переключатель типа сущности «Объект».

Идентификатор для индивидуальных объектов также можно не указывать при создании – он будет сгенерирован автоматически из имени класса и уникального случайного числа.

После нажатия кнопки «Создать», будет создан индивидуальный объект, который отобразится в списке объектов:

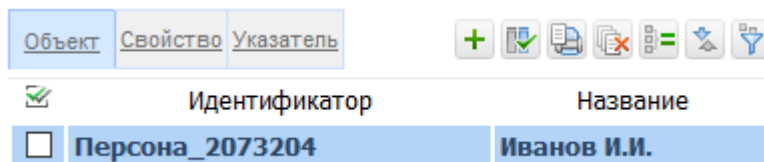


Рис. 21. Созданный объект в списке

Выбрав какой-либо объект, при помощи форм в правой части страницы мы сможем отредактировать все его свойства, написать к нему комментарии, прикрепить файлы.

Обратим внимание на то, что объект имеет свойства, определенные для того класса, к которому он относится, а также унаследованные от родительских классов. Так, форма свойств объекта «Иванов» имеет следующий вид:

Сущность - Индивидуальный объект (экземпляр)

Идентификатор

Тип *

Название

Определение

Определен в

Удален

Дата рождения

Адрес

Адрес

поля, отмеченные *, обязательны для заполнения

Рис. 22. Форма редактирования объекта

Обратите внимание, что поле «Адрес» в этом примере имеет несколько значений. Добавить новые значения можно нажатием на зеленую кнопку «+», расположенную справа от поля.

В случае, если значение какого-либо атрибута изменялось, редактор позволяет просмотреть историю изменения. Если значение атрибута имеет историю, справа от поля для его редактирования появляется иконка просмотра истории:

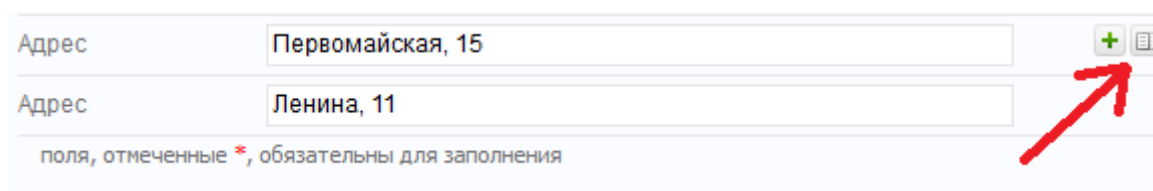


Рис. 23. Кнопка просмотра истории значений

При нажатии на нее, появляется диалоговое окно, выглядящее таким образом:

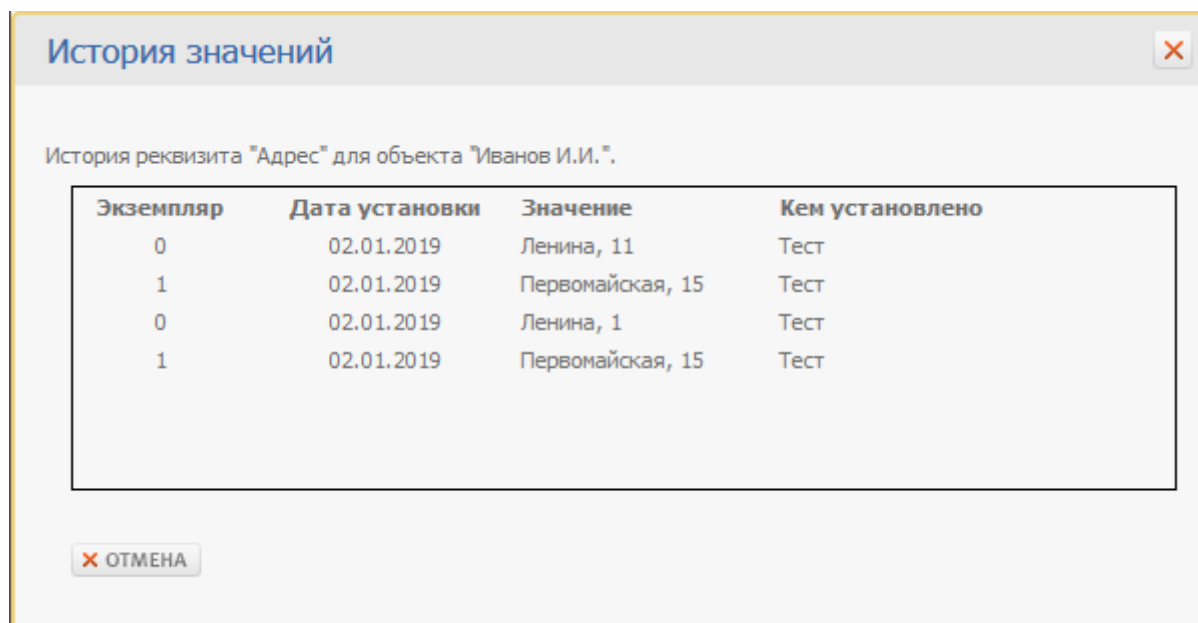
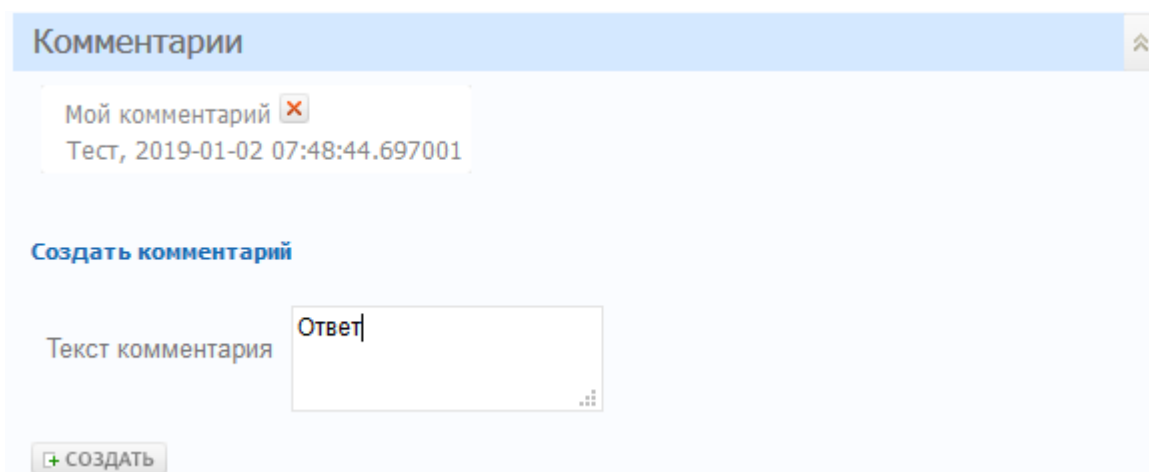


Рис. 24. Диалоговое окно истории значений

В таблице в этом окне отображаются все значения, которые имел данный атрибут текущего объекта. Каждая строка таблицы соответствует одному изменению. В колонке «Экземпляр» отображается номер экземпляра атрибута, который был изменен – если атрибут имеет единственное значение, в ней всегда будет содержаться 0. В следующих колонках указаны, соответственно, дата изменения или установки значения атрибута, присвоенное значение, и имя пользователя, который выполнил это действие. Удалить историю, или каким-либо образом управлять ей – нельзя, даже с правами администратора. История значений хранится не в самой модели, а в метаданных о ней, размещенных в реляционной базе данных.

2.6. Прикрепление файлов и комментариев

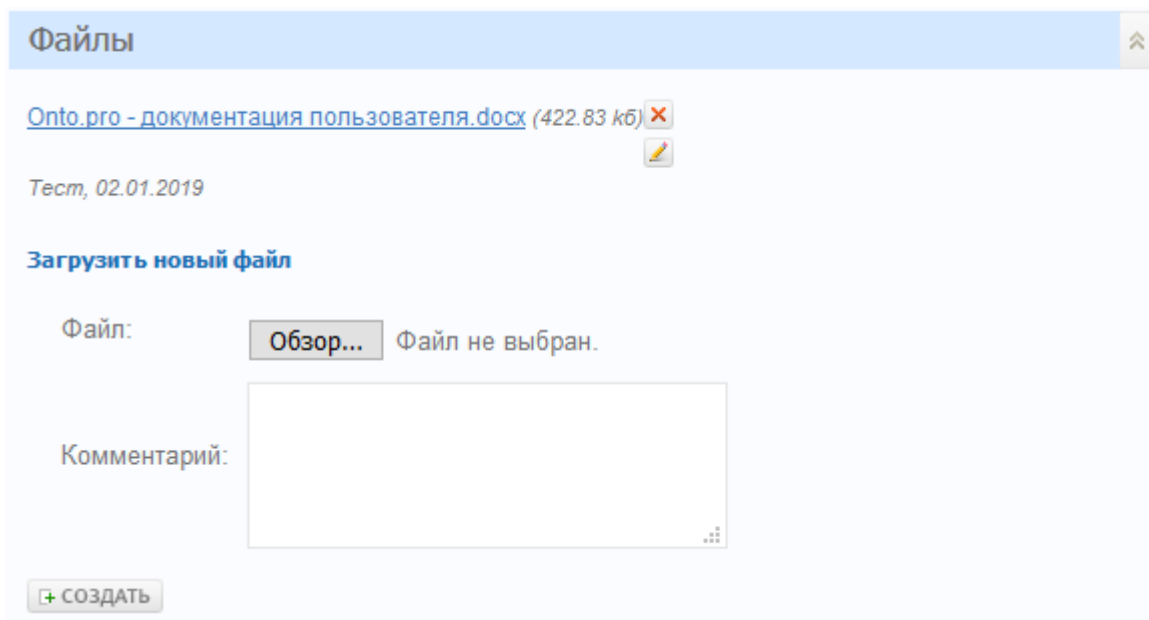
В нижней части страницы редактирования элемента модели находятся формы создания комментариев и прикрепления файлов. Любой пользователь, имеющий доступ к объекту онтологии, может оставить свой комментарий, или прикрепить файл. Форма создания комментариев выглядит так:



The screenshot shows a light blue header with the title 'Комментарии' and an upward arrow icon. Below the header, there is a white box containing the text 'Мой комментарий' with a red 'X' icon to its right, and 'Тест, 2019-01-02 07:48:44.697001' below it. Underneath is a blue link 'Создать комментарий'. A text input field labeled 'Текст комментария' contains the word 'Ответ'. At the bottom left, there is a button with a plus sign and the text 'СОЗДАТЬ'.

Рис. 25. Форма создания комментариев

Форма работы с прикрепленными файлами выглядит таким образом:



The screenshot shows a light blue header with the title 'Файлы' and an upward arrow icon. Below the header, there is a blue link 'Onto.pro - документация пользователя.docx (422.83 кб)' with a red 'X' icon to its right and a pencil icon below it. Underneath is the text 'Тест, 02.01.2019'. Below that is a blue link 'Загрузить новый файл'. A label 'Файл:' is followed by a button 'Обзор...' and the text 'Файл не выбран.'. Below this is a text input field labeled 'Комментарий:'. At the bottom left, there is a button with a plus sign and the text 'СОЗДАТЬ'.

Рис. 26. Форма работы с прикрепленными файлами

Нажатие на кнопку с красным крестиком позволяет удалить комментарий или файл, на кнопку с карандашом – отредактировать комментарий к файлу. Щелчок на имени файла позволяет его скачать.

2.7. Управление списком объектов и свойств

Также, как и для классов, для объектов и свойств доступны фильтрация и сортировка. Здесь диалоговые окна имеют больший функционал, поскольку сортировать и фильтровать можно по любому свойству, доступному для объектов выбранного класса. Кроме того, можно выбрать столбцы для отображения в списке – для этого нужно нажать на кнопку «Выбрать столбцы» над ним. Откроется следующее диалоговое окно:

Порядок	Столбец	Ширина	Л	Ц	П	-
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Идентификатор	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
<input type="button" value="▲"/>	<input checked="" type="checkbox"/> Название	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
<input type="button" value="▼"/>	<input type="checkbox"/> Тип	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
<input type="button" value="▲"/>	<input type="checkbox"/> Определение	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
<input type="button" value="▼"/>	<input type="checkbox"/> Работает в компании	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>
<input type="button" value="▲"/>	<input type="checkbox"/> Дата рождения	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>

Рис. 27. Управление столбцами в списке

При помощи этого окна можно выполнить следующие настройки:

- Выбрать столбцы для отображения, отметив их переключателями-«галочками»;
- Установить порядок столбцов, перемещая их кнопками «вверх»-«вниз»;
- Задать ширину отдельных столбцов в процентах;
- Указать выравнивание значений внутри столбца – по левому краю, центру или по правому – при помощи переключателей Л, Ц, П;
- Запретить перенос строк в отображаемых значениях при помощи переключателя в последней колонке.

Сделанные настройки сохраняются между сеансами работы в редакторе. Эти же настройки, как будет показано далее, используются при экспорте содержимого списка в Excel. Список с настроенными столбцами может выглядеть так:

Объект	Свойство	Указатель		
<input checked="" type="checkbox"/>	Идентификатор	Название	Дата рождения	
<input type="checkbox"/>	Персона_2073204	Иванов И.И.	05.11.1981	
<input type="checkbox"/>	Персона_28817794	Петров П.П.		

Рис. 28. Список с настроенным отображением

При большом количестве объектов в списке, он отображается с разбивкой на страницы. Навигация выглядит таким образом:

Объект	Свойство	Указатель		
<input checked="" type="checkbox"/>	Идентификатор	Название	Дата рождения	
все страницы				
			1 2	
<input type="checkbox"/>	Персона_2073204	Иванов И.И.	05.11.1981	
<input type="checkbox"/>	Персона_28817794	Петров П.П.		

Рис. 29. Постраничная навигация в списке объектов

2.8. Групповые операции с элементами онтологии

Для индивидуальных объектов доступен ряд групповых операций:

- Удаление нескольких объектов;
- Присвоение определенного значения какому-либо свойству группы объектов;
- Экспорт объектов в Excel (доступен и для списков свойств).

Вызов этих действий выполняется при помощи кнопок, расположенных над списком объектов. Сначала нужно отметить переключателями-«галочками» те объекты, над которыми требуется выполнить операцию (это можно сделать для всех объектов разом, нажав кнопку с галочкой над списком; правый щелчок по ней инвертирует выделение):

Объект	Свойство	Указатель		
<input checked="" type="checkbox"/>	Идентификатор	Название	Дата рождения	
<input checked="" type="checkbox"/>	Персона_2073204	Иванов И.И.	05.11.1981	
<input checked="" type="checkbox"/>	Персона_28817794	Петров П.П.		

Рис. 30. Работа с выделенными записями в списке

При нажатии на кнопку «Удалить все» система запрашивает подтверждение, а затем удаляет все отмеченные записи. **ВНИМАНИЕ!** Отменить эту операцию нельзя!

При нажатии на кнопку «Присвоить значение», отображается следующее диалоговое окно:

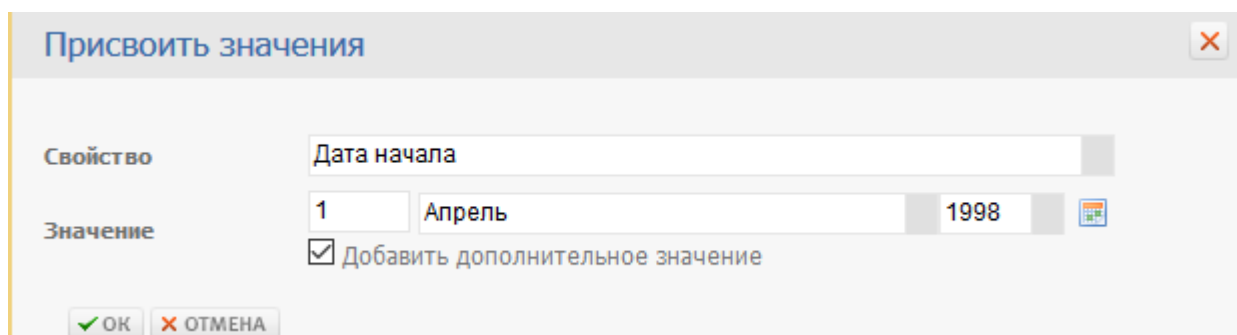


Рис. 31. Окно присвоения значения группе записей

В нем необходимо выбрать свойство, и ввести или выбрать его значение. После нажатия кнопки «ОК», указанное значение будет присвоено всем отмеченным в списке записям. Переключатель «Добавить дополнительное значение» позволяет добавить новое значение свойства, не стирая уже имеющихся.

Обратим внимание на то, что при помощи этой функции можно перенести несколько объектов из одного класса в другой. Для этого нужно выбрать в окне свойство «Тип», и выбрать класс-приемник:

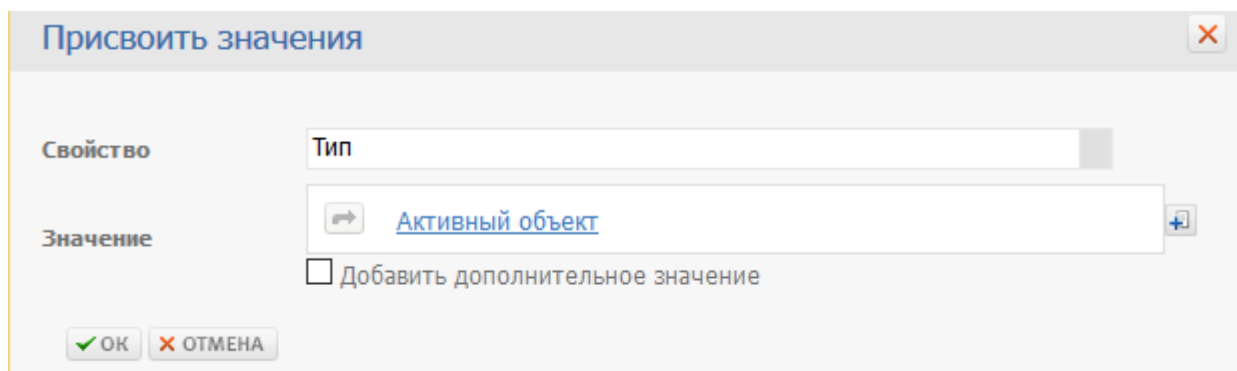


Рис. 32. Перенос объектов из одного класса в другой

Наконец, нажатие на кнопку «Выгрузить в Excel» приводит к открытию файла в формате MS Excel 2003, содержащего таблицу, аналогичную по составу столбцов и прочим настройкам списку, отображаемому в Onto.pro:

	A	B
1	Список записей раздела:Онтология	
2	Дата экспорта:24.11.2015	
3		
4	Идентификатор	Название
5	Иванов	Иванов И.И.
6	Петров	Петров П.П.

Рис. 33. Образец выгруженного Excel-файла