

Моделирование онтологий в onto.pro

Моделирование онтологий в onto.pro.....	1
Регистрация в onto.pro.....	1
Основы семантического моделирования.....	2
Классы и индивиды.....	2
URI.....	2
Свойства.....	3
Общая картина.....	3
Работа с элементами онтологии.....	4
Создание классов.....	5
Создание свойств-литералов.....	6
Создание свойств-указателей на объекты.....	8
Создание индивидуальных объектов.....	9
Просмотр информации о классах в wiki-представлении.....	10

Регистрация в onto.pro

Для работы с onto.pro необходимо создать в нем учетную запись. Для этого нужно нажать на одноименную ссылку на главной странице, и заполнить форму следующего вида:

СОЗДАТЬ УЧЕТНУЮ ЗАПИСЬ	
Логин	Tester
Пароль	•••
E-mail	some@mail.ru
Ваше имя	Иван Тестеров
Префикс онтологии (англ. слово)	Testerov
ISO 15926	<input type="checkbox"/>
Все поля обязательны для заполнения	
OK	Назад

Логин и пароль идентифицируют вас, как пользователя onto.pro.
E-mail необходим для подтверждения регистрации.
Имя нужно для того, чтобы система могла обратиться к вам.

Последние два поля формы, «Префикс» и «ISO 15926», имеют особое значение. Редактор onto.pro позволяет моделировать как онтологии, соответствующие стандарту ISO 15926, так и произвольные онтологии. Если установить переключатель «ISO 15926» при регистрации, onto.pro создаст для вас собственную точку доступа («песочницу»), в которой вы сможете смоделировать свой собственный RDL – каталог справочных данных. Описанию работы с onto.pro в соответствии с ISO 15926 посвящен отдельный документ.

Если же вы собираетесь работать с произвольными онтологиями (чему посвящен данный документ), то переключатель устанавливать не нужно. Зато, в этом случае, нужно заполнить поле «Префикс». Более детально концепция префиксов рассмотрена ниже, в разделе «URI»; пока скажем только, что элементы вашей онтологии будут иметь уникальные идентификаторы вида <http://www.onto.pro/Префикс/ИдЭлемента>, где Префикс – это как раз то значение, которое вы введете при регистрации (изменить его потом будет нельзя!).

Префикс должен представлять собой английское слово, без специальных символов и пробелов.

После нажатия кнопки ОК, на ваш e-mail будет отправлено письмо со ссылкой для подтверждения адреса. После того, как вы перейдете по ссылке, вы сможете войти в систему. На главной странице вы увидите меню выбора точки доступа. Наряду со стандартными точками, содержащими онтологии, связанные с ISO 15926, в меню будет присутствовать точка под названием «[Ваше имя] Sandbox» («песочница»). Ее-то и нужно выбрать: в этой точке вы сможете моделировать свою онтологию.

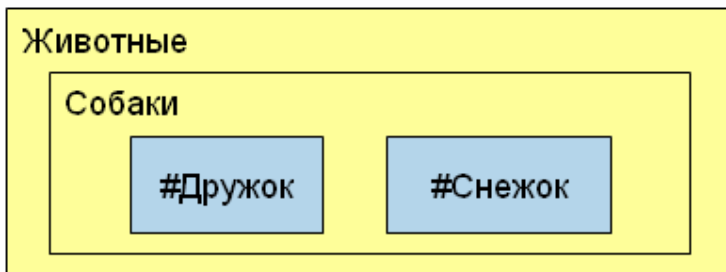
Входить в Onto.pro можно как с использованием имени/пароля, заданных при регистрации, так и с использованием аккаунта Google или Facebook, привязанного к тому же e-mail адресу, который был использован при регистрации. Если учетной записи с таким адресом еще нет в Onto.pro, она будет автоматически создана при первом входе в систему; однако, мы рекомендуем сначала создать учетную запись при помощи описанной выше процедуры, т.к. она предоставляет больше возможностей управления создаваемым аккаунтом.

Основы семантического моделирования

Классы и индивиды

Сначала разберемся с подходами к моделированию. В семантике все объекты (термины, понятия) условно делятся на два основных типа: *индивидуальные объекты*, и *классы* (совокупности индивидуальных объектов).

Отношение между классами и индивидами можно проиллюстрировать таким образом:



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

URI

У каждого объекта (как класса, так и индивида) есть свой уникальный идентификатор – URI, который выглядит как <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 (совокупность объектов), к которой он относится. После решетки или последнего символа / 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

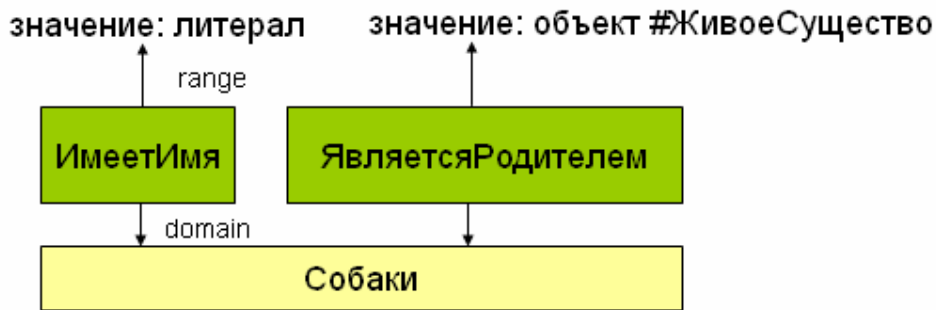
http://some.address/MyOntology# = MyOntology:

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

Свойства

Объекты могут иметь свойства (точнее, значения свойств). Само по себе свойство (его определение) – это некий объект, существующий отдельно от классов (в отличие от ООП, где свойство принадлежит классу). В семантических моделях определение каждого свойства имеет, в свою очередь, два обязательных свойства: *range*, определяющее тип возможного значения (литерал, или индивидуальный объект какого-либо класса), и *domain* – класс, к индивидам которых применимо данное свойство. Domain'ов может быть любое число, т.е. свойство «имя» применимо как к объектам класса «Люди», так и к объектам класса «Собаки».

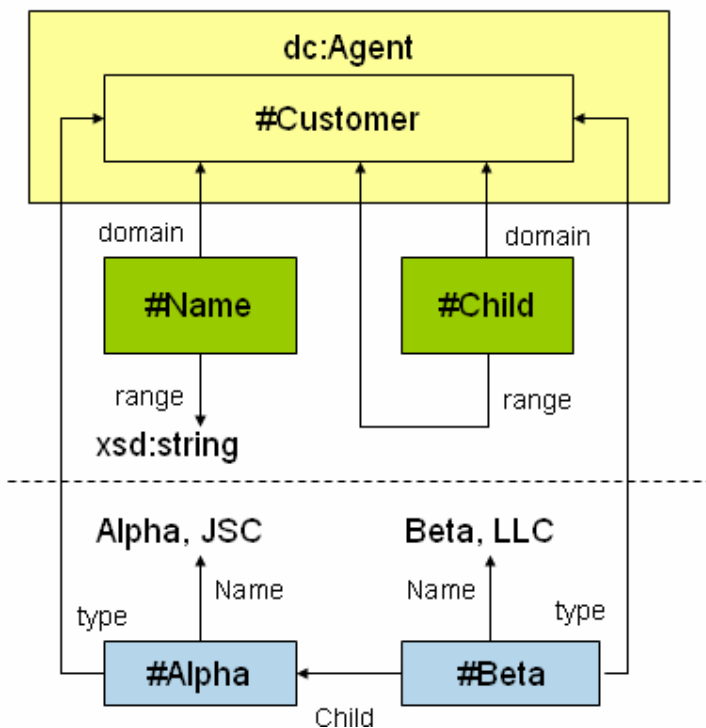
На диаграмме это можно проиллюстрировать так:



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

Общая картина

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



То, что показано выше пунктирной линией, 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.

Работа с элементами онтологии

Все примеры в этом разделе будем рассматривать на следующей онтологии:



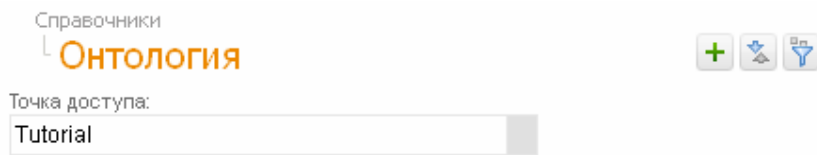
В верхней части диаграммы мы видим определения трех классов: «Персона», «Организация», и подкласс организаций – «Компания». Организации и персоны имеют свойство-литерал «Адрес», которое может принимать любое количество значений. У персоны есть также свойство «Дата рождения», которое имеет только одно 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.

Создание классов

Для удобства восприятия, классы в **Onto.pro** представлены в виде иерархии, основанной на отношениях «является подклассом» (хотя 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 (онтологию), дерево слева пусто – объектов в нем еще нет:



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

Обратите внимание на набор кнопок справа от заголовка формы «Сущность». С его помощью можно выбрать тип создаваемой сущности:

- Сущность без типа (точнее, с типом **Thing**, который применим к любым элементам **OWL**);
- Класс;
- Индивидуальный объект;
- Свойство-литерал;
- Свойство-указатель на объект.

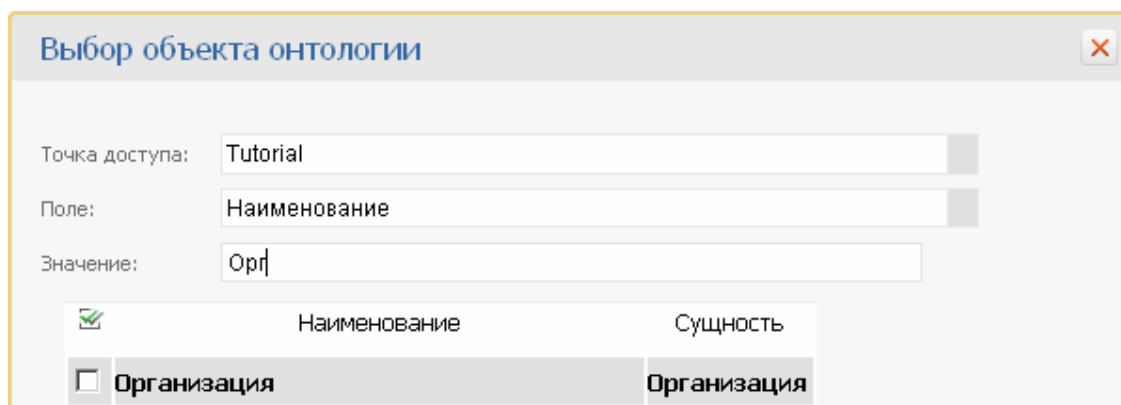
Нам необходимо создать класс, поэтому активна вторая кнопка переключателя.

Введем в поля «Название» и «Ярлык» значение «Организация». Название – это завершающая часть уникального идентификатора объекта (**URI**). Она не должна содержать пробелов и спецсимволов, поэтому при сохранении записи они будут отфильтрованы. Ярлык – 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:



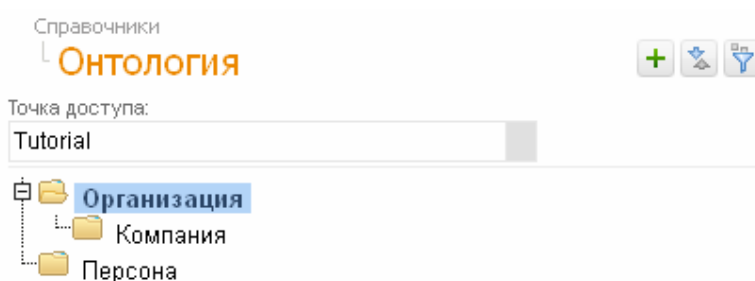
В области «Надклассы» указывается родительский класс для того класса, который мы создаем. По умолчанию здесь выбирается последний класс, который мы создавали или просматривали. Нажатием на красную кнопку «X» можно очистить выбранный класс – эта возможность нам понадобится, когда мы будем создавать класс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



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

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».

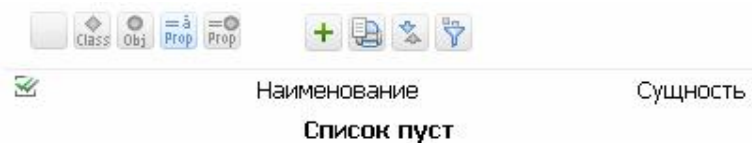
После создания трех классов из нашего примера, дерево классов примет следующий вид:



Создание свойств-литералов

Чтобы иметь возможность выражать 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 ниже дерева):



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/tutorial/>

Сущность Class Obj Prop Prop

Название *

Ярлык

Применимо к *

Диапазон значений *

Мин. кол-во

Макс. кол-во

поля, отмеченные *, обязательны для заполнения

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

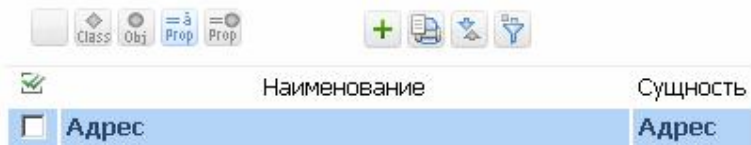
После того, как свойство будет создано, укажем, что оно присуще не только объектам класса «Компания», но и персонам. Для этого нужно нажать зеленую кнопку «+» справа от первого значения поля «Применимо к». Появится вторая строка, для следующего экземпляра поля; нажмем на кнопку выбора значения, и в появившемся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

Применимо к *

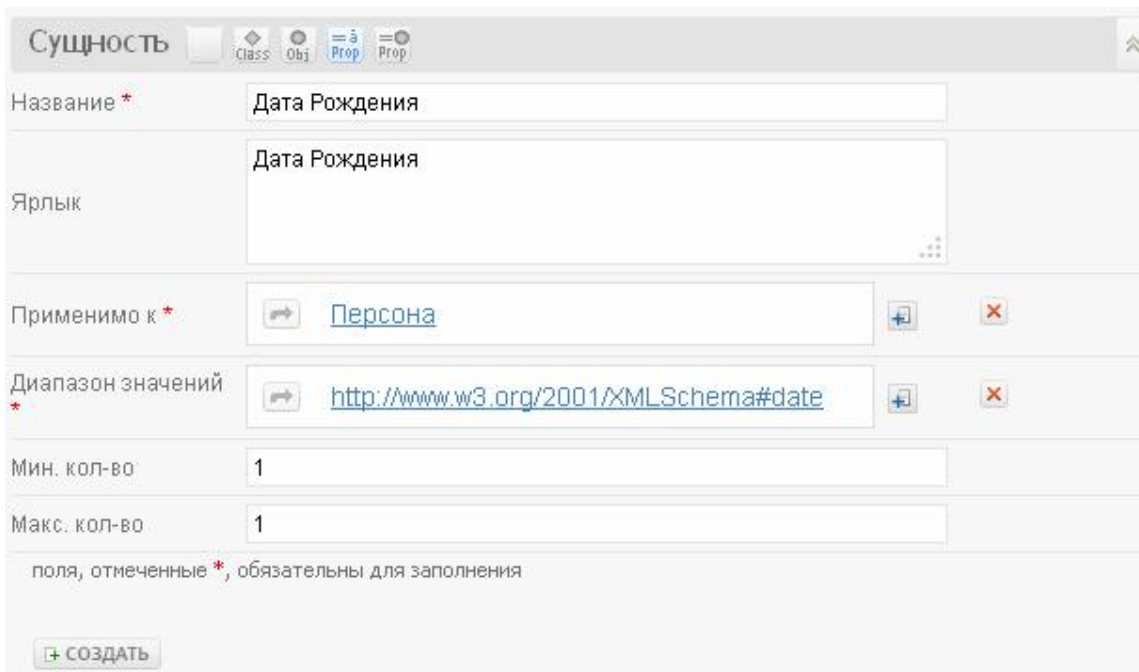
Применимо к

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 ниже списке в положение «Свойства». В списке появится наше свойство – «Адрес»:



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.



Сущность	Class	Obj	Prop	Prop
Название *	Дата Рождения			
Ярлык	Дата Рождения			
Применимо к *	Персона			+ -
Диапазон значений *	http://www.w3.org/2001/XMLSchema#date			+ -
Мин. кол-во	1			
Макс. кол-во	1			
поля, отмеченные *, обязательны для заполнения				
<input type="button" value="+ СОЗДАТЬ"/>				

Свойства готовы.

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

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 внизу выбираем переключатель «Свойства-указатели на объект», и нажимаем кнопку «+» («Добавить»). Заполняем форму создания свойства:

Сущность Class Obj Prop Prop

Название *

Ярлык

Применимо к * + ×

Диапазон значений * + ×

Мин. кол-во

Макс. кол-во

поля, отмеченные *, обязательны для заполнения

+ СОЗДАТЬ

Создание индивидуальных объектов

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

Сущность Class Obj Prop Prop

Тип * + ×

Название *

Ярлык

поля, отмеченные *, обязательны для заполнения

+ СОЗДАТЬ

Обратите внимание, что в заголовке формы активен переключатель типа сущности «Объект».

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

Class Obj Prop Prop + + + +

<input type="checkbox"/>	Наименование	Сущность
<input checked="" type="checkbox"/>	Иванов И.И.	Иванов

Выбрав этот объект, мы сможем отредактировать все его свойства, написать к нему комментарии, прикрепить файлы, и посмотреть результат в wiki-представлении.

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:

The screenshot shows a web form titled 'Сущность' (Entity) for editing an entity named 'Иванов'. The form includes the following fields and controls:

- Тип ***: A dropdown menu showing 'Персона' (Person).
- Название ***: A text input field containing 'Иванов'.
- Ярлык**: A text input field containing 'Иванов И.И.'.
- Адрес**: A text input field containing 'Вишневая, 1' with a green '+' button to the right.
- Адрес**: A text input field containing 'Восточная, 4'.
- Дата Рождения ***: A date picker showing '1' for the day, 'Январь' for the month, and '1970' for the year.

Below the form, there is a note: 'поля, отмеченные *, обязательны для заполнения' (fields marked with * are mandatory for filling). At the bottom, there are two buttons: 'СОХРАНИТЬ' (Save) and 'УДАЛИТЬ' (Delete).

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

Просмотр информации о классах в wiki-представлении

Нажав на ссылку «посмотреть в wiki-представлении», расположенную в верхней части формы редактирования свойств сущности, мы увидим следующую страницу (пример для ООО «Альфа» из онтологии, используемой в этом тестовом примере):

The screenshot shows a wiki page for 'ООО Альфа'. The page includes a search bar at the top right with the text 'Поиск' and a magnifying glass icon. The main content area is divided into several sections:

- Статья**: A section header.
- ООО "Альфа"**: The main title of the page.
- ООО "Альфа" —**: A subtitle or description.
- Комментарии**: A section for comments, showing a comment from 'Сергей Горшков' dated '22.01.2014' with the text 'Комментарий о компании'.
- Файлы**: A section for files, showing a file '22.01.2014 Сергей Горшков - cim10_010825c.rdf'.
- Properties Table**: A table listing properties of the entity 'ООО "Альфа"':

ООО "Альфа"	
Тип:	Компания
Тип:	NamedIndividual
Адрес:	Уральская, 5
Имеет Учредителя:	Иванов И.И.
Имеет Учредителя:	Петров П.П.

Щелчком по ссылкам на любую сущность (например, на И.И. Иванова) можно перейти на соответствующую ей страницу, где будет отображена обратная ссылка, а также ссылки на другие элементы семантической модели:

Статья

Иванов

Иванов И.И. —

Другие связи

- [ООО "Альфа" Имеет Учредителя](#)

Иванов И.И.	
Тип:	Персона
Тип:	NamedIndividual
Адрес:	Вишневая, 1
Адрес:	Восточная, 4
Дата Рождения:	1970-01-01

Эти страницы доступны неавторизованным пользователям [onto.pro](http://www.onto.pro); на ней также выводятся комментарии и файлы, прикрепленные в [Onto.pro](http://www.onto.pro) к данному элементу онтологии. Таким образом, создавая информационную модель, мы одновременно делаем ее доступной для просмотра неавторизованными пользователями, в виде wiki-представления.

Мы можем искать термины в нашей wiki, зайдя по адресу [http://www.onto.pro/\[имя онтологии\]](http://www.onto.pro/[имя онтологии]), и введя интересующее слово в строке поиска. Его результаты выглядят так:

Поиск

tutorial

- [ООО "Альфа"](#)