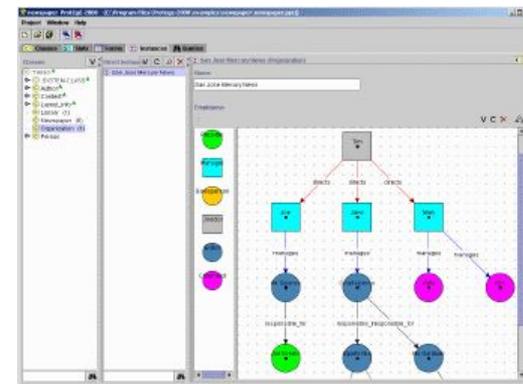


Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N



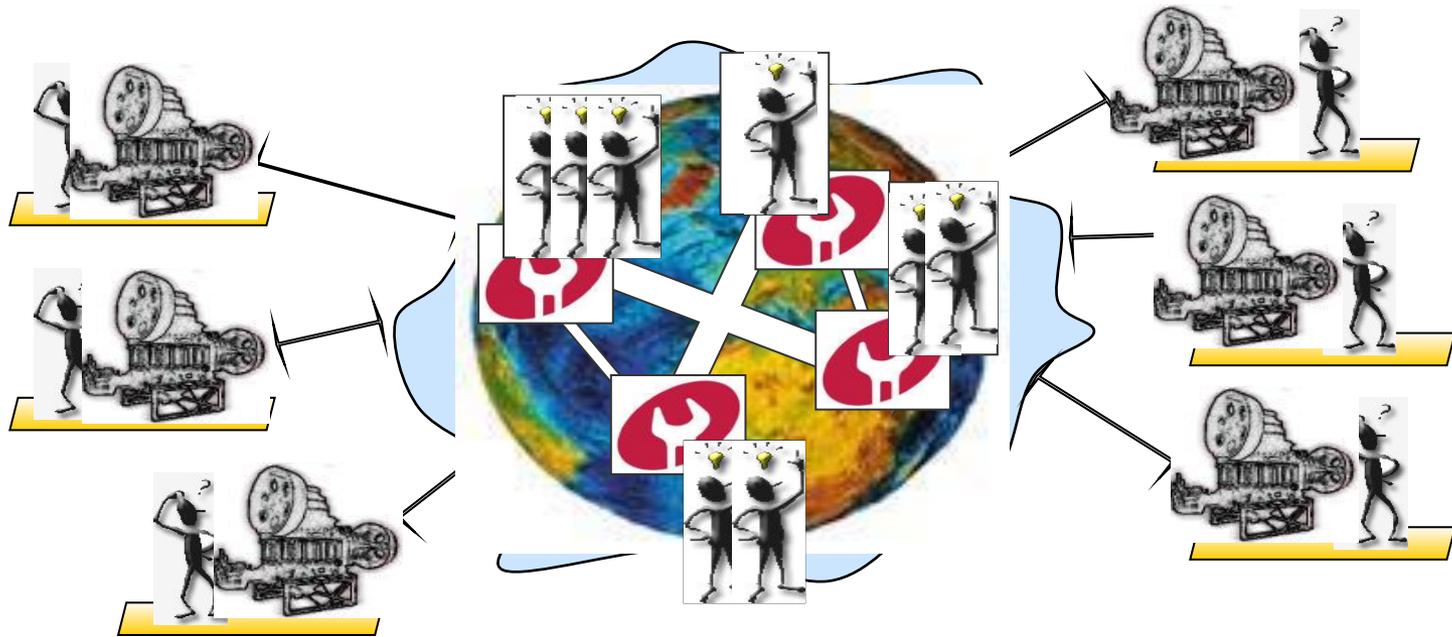
# Web-сервис интеллектуальной обработки данных. Разработка семантического программного адаптера

Терзиян А.В.

Студент гр. ИСПР-00-1

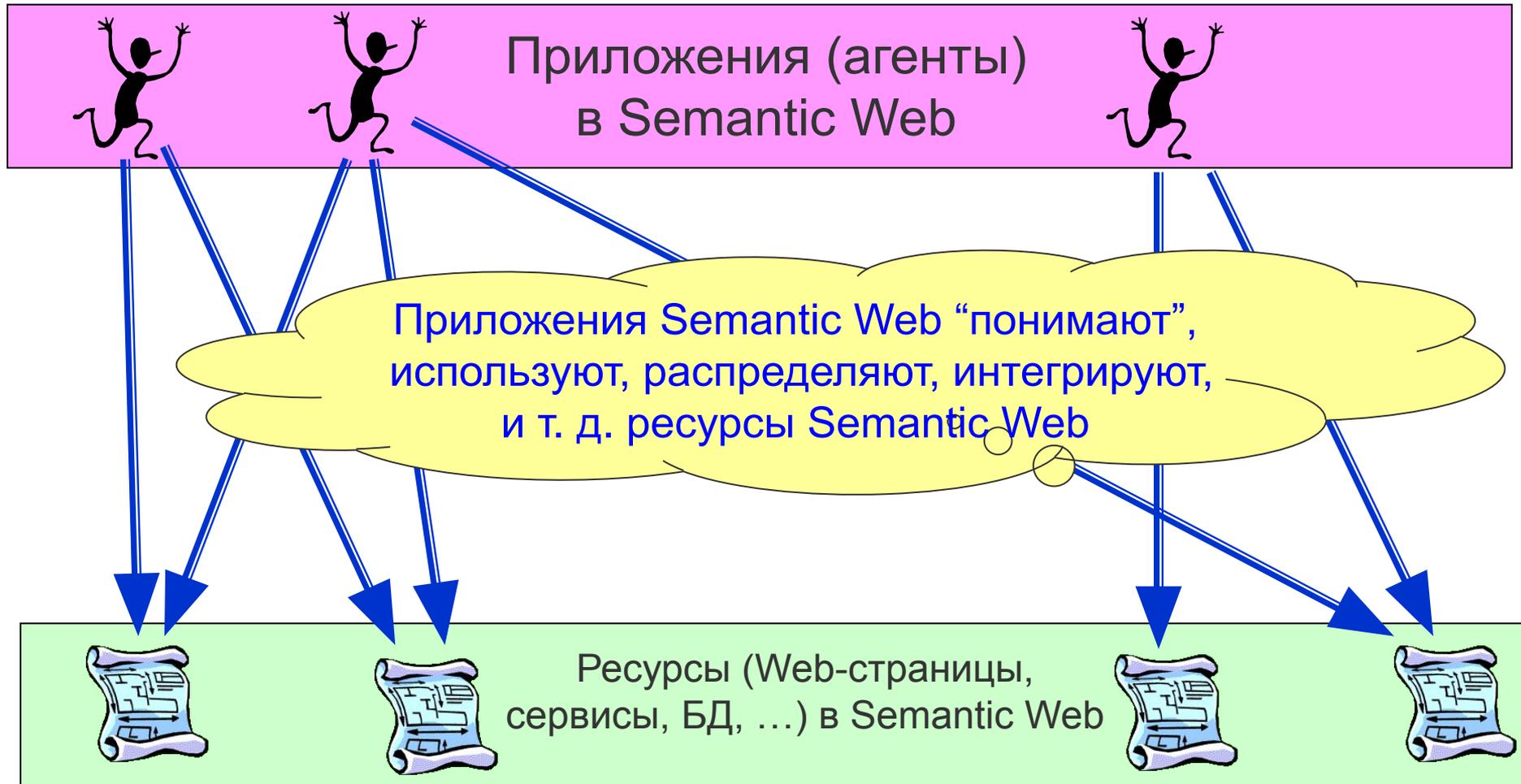
Научный руководитель: доцент каф. ИИ Витько А.В

Харьков, 2004

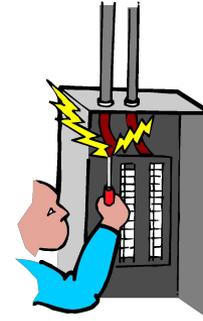
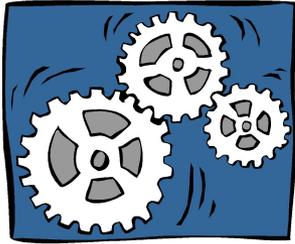


# Мотивация и цель работы

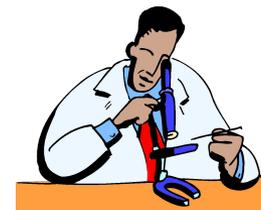
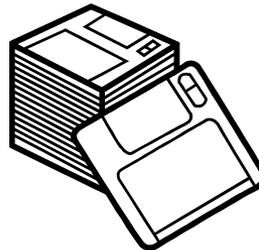
# Semantic Web: первоначальная концепция



# Новая концепция: Global Understanding eNvironment (GUN)



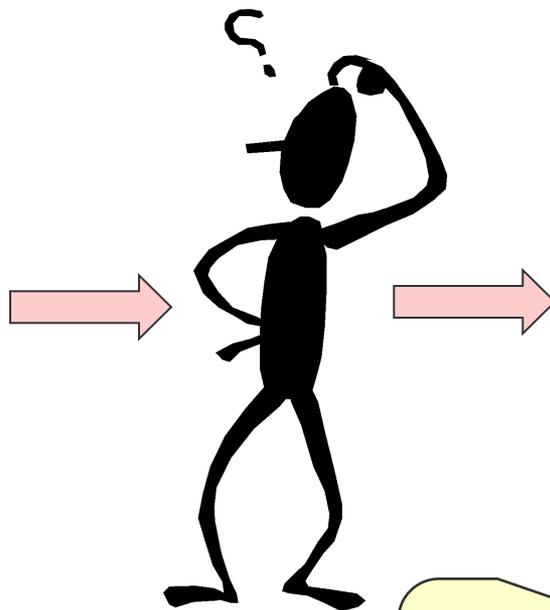
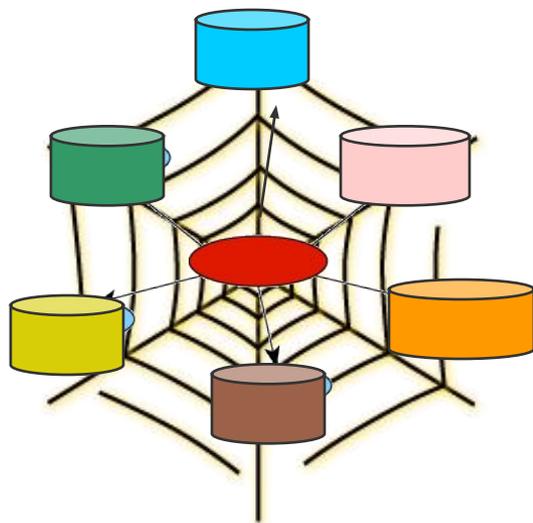
Как сделать так, чтобы объекты  
окружающего мира (например,  
производственной сферы)  
стали доступными из WWW и  
«понимали» друг-друга в  
случае необходимости ?



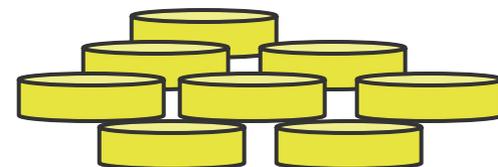


# Извлечение знаний из распределенных и разнородных массивов данных о ресурсах в Web-или GUN-среде – одна из основных задач Web-сервисов по интеллектуальной обработке данных

Распределенные и разнородные массивы данных

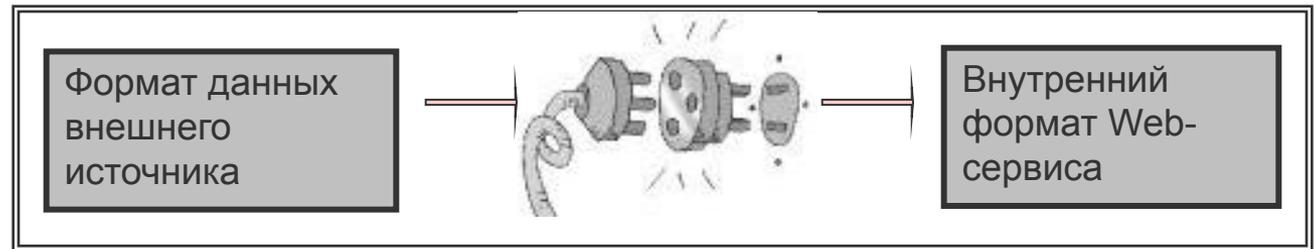


Знания  
(например, модели  
диагностики)

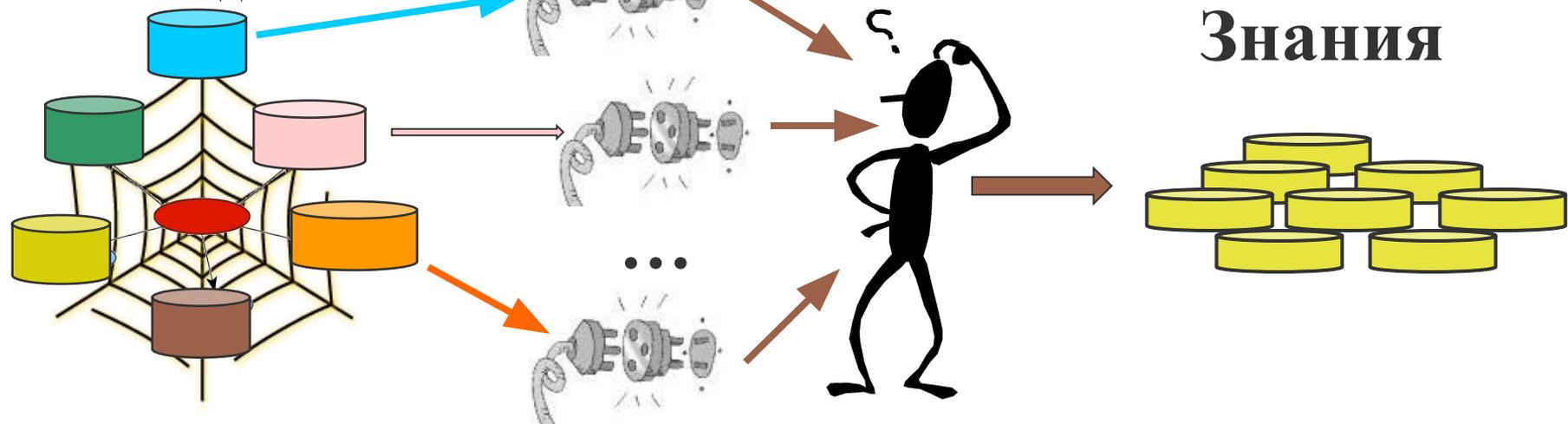


Web-сервис по  
интеллектуальной  
обработке данных

**Однако, если Web-сервис понимает только один формат входных данных, тогда ему необходим адаптер для работы с каждым новым массивом данных**



**Распределенные и разнородные массивы данных**

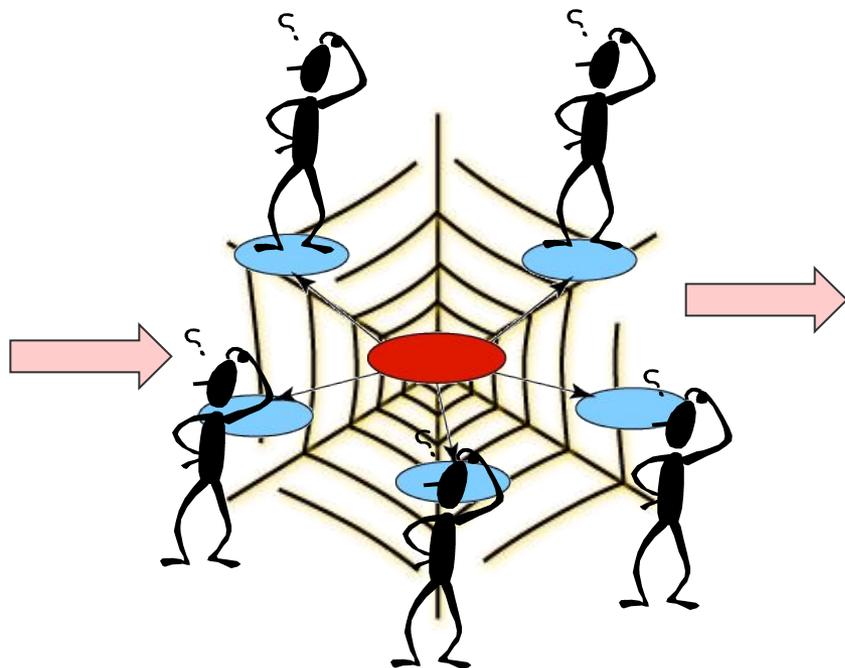
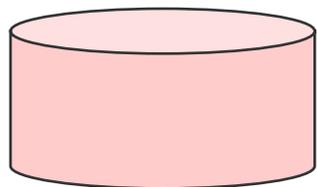


Адаптер

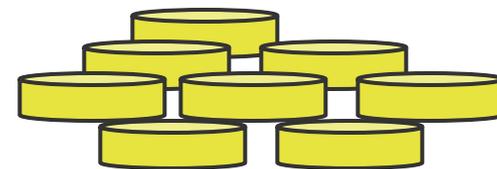
**Знания**

С другой стороны, данные могут браться из одного массива данных, но обрабатываться несколькими распределенными и разнородными Web-сервисами...

Web-массивы  
данных

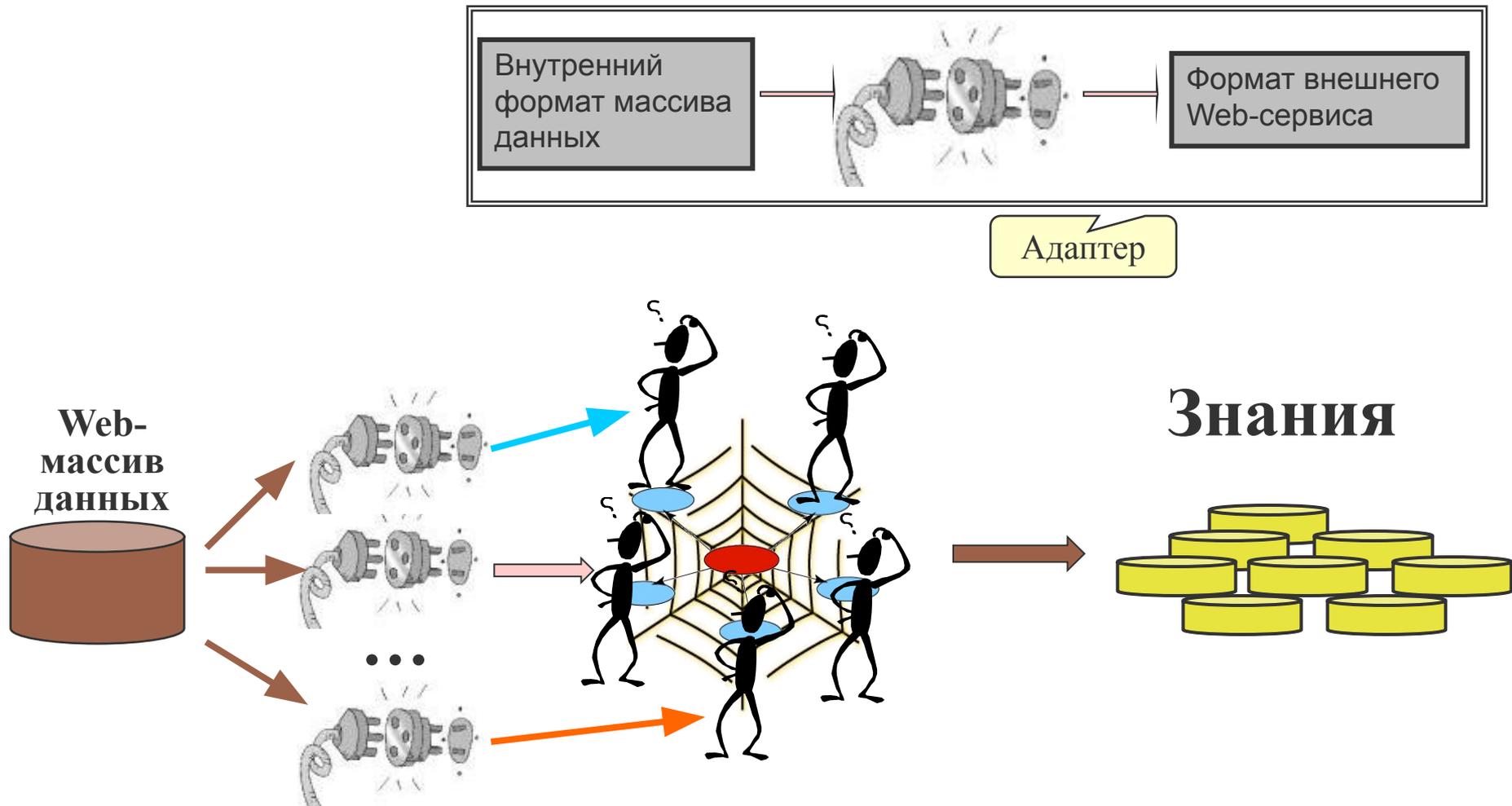


Знания



Сеть разнородных Web-сервисов по интеллектуальной обработке данных

# ... и это подразумевает, что нам необходимы адаптеры для работы с разными Web-сервисами



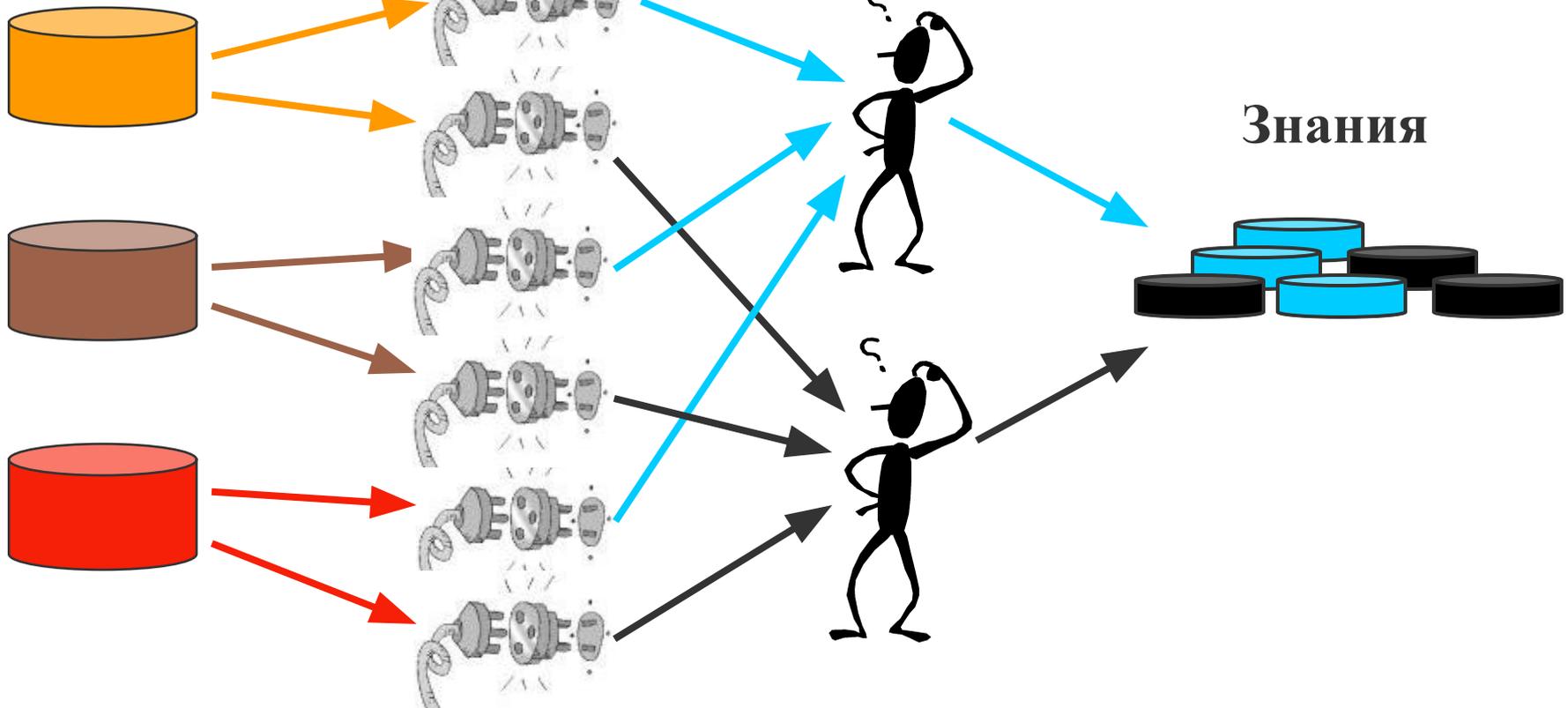
В общем случае для оперирования с  $N$  разнородными массивами данных и  $M$  разнородными Web-сервисами нам необходимо  $M \times N$  адаптеров

3 разнородных массива данных

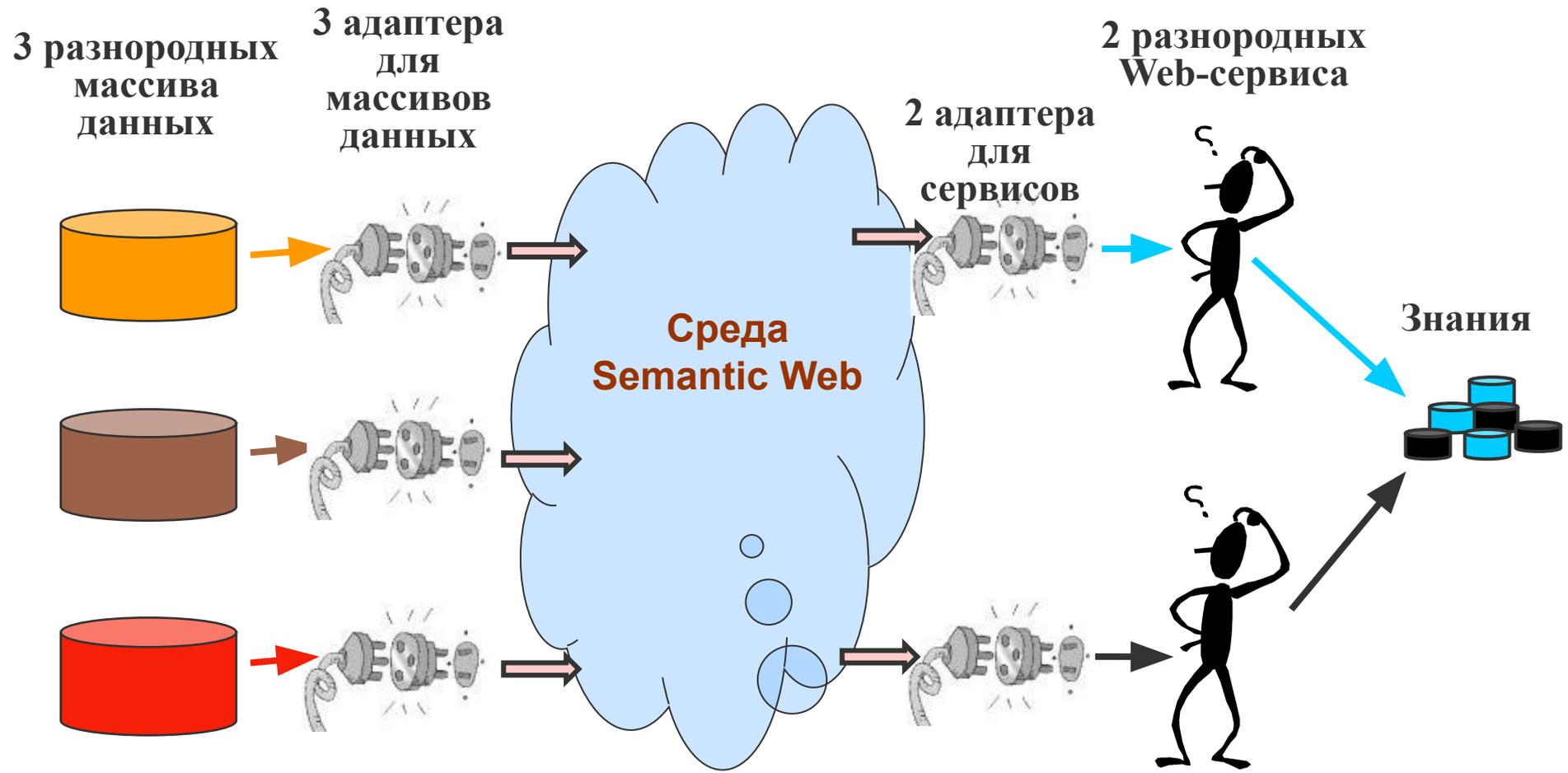
6 адаптеров

2 разнородных Web-сервиса

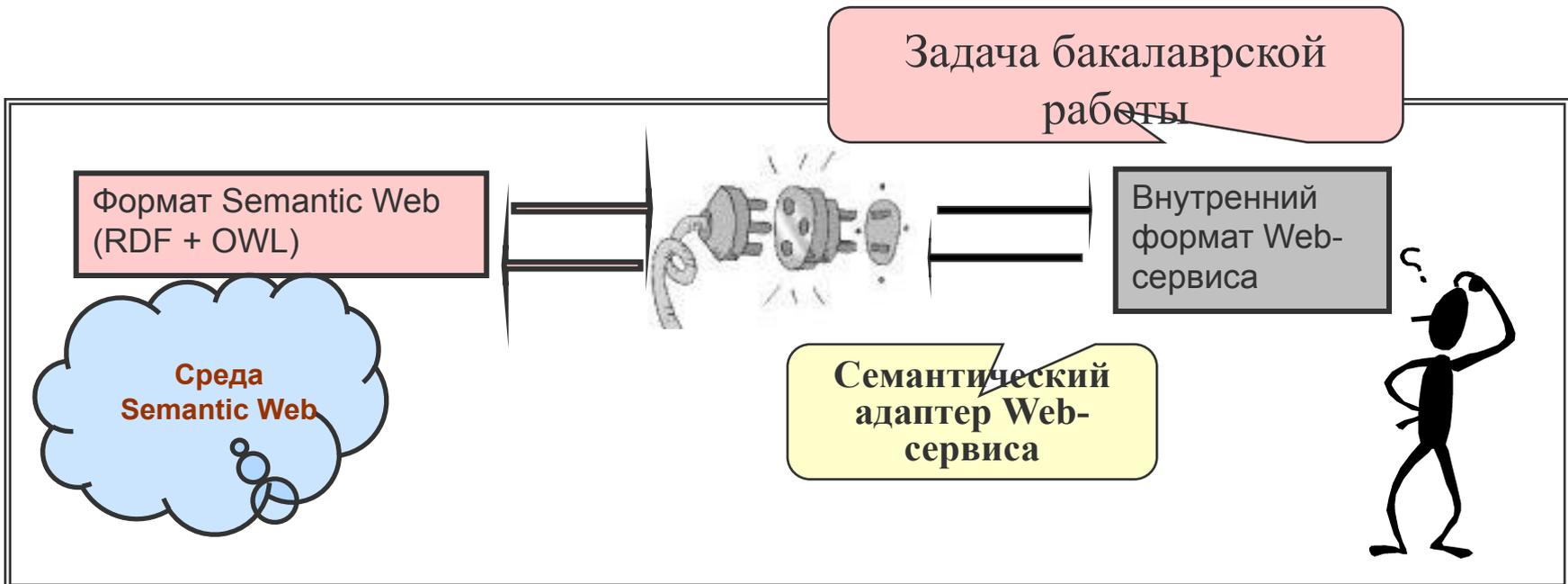
Знания



Для уменьшения стоимости адаптации решением может быть использование стандартных форматов данных (например, стандарты W3C для Semantic Web : RDF, RDFS, OWL) в качестве посредника между источниками данных и сервисами, что в худшем случае будет требовать  $M + N$  (семантических) адаптеров



# Таким образом, нам необходимо реализовать два типа семантической адаптации: (1) преобразование массивов данных в среду Semantic Web и (2) наоборот



# Основные цели работы

- Разработка способов семантического представления обучающих выборок
- Разработка программного адаптера, реализующего функции:
  - преобразования входных данных для Web-сервиса из стандартизованного формата OWL в массив данных, соответствующий внутреннему представлению информации в Web-сервисе;
  - преобразование выходных данных, полученных в результате работы Web-сервиса из внутреннего формата в стандартизованный;
- Экспериментальное тестирование адаптера на примере данных, подготовленных в среде Protégé.

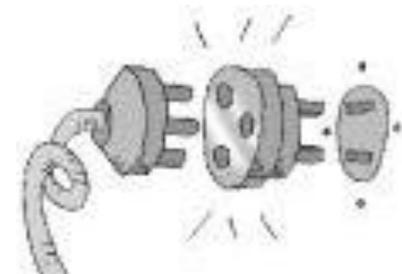
# Описание обучающих выборок в семантическом формате

- **целевой атрибут** представляется самым высшим классом в иерархии классов (owl:Class);
- **возможные значения целевого атрибута** представляются подклассами (owl:SubClassOf) класса целевого атрибута;
- **атрибуты** описываются свойствами класса целевого атрибута. Возможны два варианта:
  - использовать простые свойства (owl:DatatypeProperty), значения атрибутов задаются простыми значениями стандартных типов данных;
  - использовать объектные свойства (owl:ObjectProperty), значения атрибутов задаются экземплярами заранее определенных «дополнительных» классов значений;
- **примеры в выборке** представляются в виде экземпляров класса целевого атрибута.

# Схема преобразования OWL-Table I

**Вариант 1: значения атрибутов в онтологии  
являются символьными строками**

Проиллюстрировано в среде Protégé-2000



# Задание атрибутов в Protégé 2000

The screenshot displays the Protégé 2000 interface for editing an OWL ontology. The main window title is "OWL\_2 Protégé 2.0.1 (U:\Protege-2000\OWL\_2.pprj, OWL Files)". The menu bar includes "Project", "Edit", "Window", "OWL", and "Help". The toolbar contains various icons for file operations and editing. The "Properties" tab is active, showing the configuration for the class "Not\_To\_Play\_Tennis (type=owl:Class)".

**Asserted Hierarchy:** A tree view on the left shows the class hierarchy: owl:Thing (parent), Weather\_Conditions (child), and Not\_To\_Play\_Tennis (child of Weather\_Conditions). Play\_Tennis is also listed as a child of Weather\_Conditions.

**Class Configuration:** The "Not\_To\_Play\_Tennis" class is selected. The "Name" field contains "Not\_To\_Play\_Tennis". The "Documentation" field is empty. The "Annotations" table is empty.

**Asserted Conditions:** The "Asserted" tab is active. It shows a list of conditions. A condition is listed: "Weather\_Conditions" with a "NECESSARY" relationship. The "NECESSARY & SUFFICIENT" relationship is also visible.

**Properties at Class:** The "Properties at Class" tab is active. It shows a list of properties: Windy, Outlook, Temperature, and Humidity. The "Name" field is empty.

# Задание типов и возможных значений атрибутов

The screenshot shows a software interface for defining attribute types and values. The interface is divided into several sections:

- Name:** A text field containing "Humidity".
- Documentation:** An empty text area.
- Annotations:** A table with two columns: "Property" and "Value".
- Domain defined:** A checked checkbox.
- Domain:** A dropdown menu showing "Weather\_Conditions".
- Range:** A dropdown menu showing "Symbol".
- XML Schema Datatype:** A dropdown menu showing "string".
- Allowed Values:** A text area containing "high" and "normal".
- Options:** Two unchecked checkboxes: "Allows multiple values" and "Inverse Functional".

Property	Value
----------	-------

Domain defined

Domain

Range

XML Schema Datatype

Allowed Values

Allows multiple values

Inverse Functional

# Задание примеров из обучающей выборки

The image shows a screenshot of the Protégé software interface. The top menu bar includes 'OWLClasses', 'Properties', 'Forms', 'Individuals', and 'Metadata'. On the left, the 'Classes' panel shows a hierarchy: 'owl:Thing' (parent), 'Wheather\_Conditions' (child), and 'Not\_To\_Play\_Tennis (1)' and 'Play\_Tennis (1)' (children of 'Wheather\_Conditions'). The 'Not\_To\_Play\_Tennis (1)' class is selected. The main 'Display Slot' area shows the selected class and its 'Direct Instances', which includes 'Sample\_1'. A floating window titled 'Sample\_1 (type=Not\_To\_Play\_Tennis)' is open, displaying a form for this instance. The form has tabs for 'Name', 'Labels', 'SameAs', and 'DifferentFrom'. The 'Name' tab is active, showing the instance name 'Sample\_1' in a text field. Below the name field is a 'Documentation' text area. Further down, there are four dropdown menus for instance properties: 'Outlook' (set to 'sunny'), 'Temperature' (set to 'hot'), 'Humidity' (set to 'high'), and 'Windy' (set to 'false').

# Преобразование из OWL в таблицу

```
...  
<N rdf:ID="Sample_1">  
  <Outlook>sunny</Outlook>  
  <Temperature>hot</Temperature>  
  <Humidity>high</Humidity>  
  <Windy>>false</Windy>  
</N>  
...  
<P rdf:ID="Sample_3">  
  <Outlook>overcast</Outlook>  
  <Temperature>hot</Temperature>  
  <Humidity>high</Humidity>  
  <Windy>>false</Windy>  
</P>  
...
```

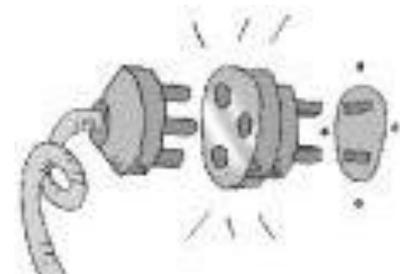


Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N

# Схема преобразования OWL-Table II

**Вариант 2: значения атрибутов в онтологии  
являются элементами некоторых классов**

Проиллюстрировано в среде Protégé-2000



# Создание дополнительных классов для возможных значений атрибутов

The screenshot displays the Protégé 2.0.1 interface for editing an OWL ontology. The main window title is "OWL\_1 Protégé 2.0.1 (U:\Protege-2000\OWL\_1.pprj, OWL Files)". The menu bar includes "Project", "Edit", "Window", "OWL", and "Help". The toolbar contains various icons for file operations and editing. The "OWLClasses" tab is active, showing the "Asserted Hierarchy" on the left. The hierarchy is as follows:

- owl:Thing
  - Values
    - Humidity\_Values
    - Outlook\_Values
    - Temperature\_Values
    - Windy\_Values
  - Weather\_Conditions
    - Not\_To\_Play\_Tennis** (selected)
    - Play\_Tennis

The main workspace shows the details for the selected class "Not\_To\_Play\_Tennis (type=owl:Class)". It has two tabs: "Name" and "Labels". The "Name" tab is active, showing the class name "Not\_To\_Play\_Tennis" in a text field. Below it is the "Documentation" section, which contains the text: "Class of wheather samples not suitable for playing tennis". To the right of the main workspace is the "Annotations" section, which is currently empty. At the bottom of the interface, there are two panels. The "Asserted" panel shows "Asserted Conditions" with a list containing "Weather\_Conditions" and a "NECESSARY" relationship. The "At Class" panel shows "Properties at Class" with a list of properties: "Windy", "Humidity", "Temperature", and "Outlook".

# Задание примеров со значениями атрибутов, являющимися экземплярами классов

The screenshot displays the Protégé OWL editor interface. On the left, the 'Classes' panel shows a hierarchy starting with 'owl:Thing', followed by 'Values', and then several subclasses: 'Humidity\_Values (2)', 'Outlook\_Values (3)', 'Temperature\_Values (3)', 'Windy\_Values (2)', 'Wheather\_Conditions', 'Not\_To\_Play\_Tennis (5)', and 'Play\_Tennis (9)'. The 'Not\_To\_Play\_Tennis' class is selected.

The main workspace is divided into two panes. The top pane, titled 'Display Slot', shows the 'NAME' property for the selected instance 'Sample\_1'. Below it, the 'Direct Instances' list includes 'Sample\_1', 'Sample\_14', 'Sample\_2', 'Sample\_6', and 'Sample\_8'. The bottom pane shows the instance 'Sample\_1 (type=Not\_To\_Play\_Tennis)' with its properties: 'Name' (Sample\_1), 'Documentation' (1-st sample of the training set), 'Outlook' (sunny), 'Humidity' (high), 'Temperature' (hot), and 'Windy' (false). Each property value is shown in a text field with a small icon to its left, and the 'Outlook', 'Humidity', 'Temperature', and 'Windy' fields have additional control buttons (V, C, +, -).

# Преобразование из OWL в таблицу

...  
<N rdf:ID="Sample\_1">

<Outlook rdf:resource="#sunny"/>

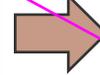
<Temperature rdf:resource="#hot"/>

<Humidity rdf:resource="#high"/>

<Windy rdf:resource="#false"/>

</N>

...



Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N

# Добавление классифицированного примера В ОНТОЛОГИЮ

The screenshot displays the Protégé 2.0.1 interface for editing an ontology. The main window is titled "OWL\_1 Protégé 2.0.1 (U:\Protege-2000\OWL\_1.pprj, OWL Files)". The interface is divided into several panes:

- Classes:** A tree view on the left showing the ontology structure. The "Wheather\_Conditions" class is expanded, showing subclasses "Not\_To\_Play\_Tennis (6)" and "Play\_Tennis (9)".
- Display Slot:** A central pane showing the selected class "Unseen\_sample" with its type "Not\_To\_Play\_Tennis". It includes a "Direct Instances" list with "Unseen\_sample" highlighted.
- Properties:** A right-hand pane showing the "Unseen\_sample" instance with its properties: "Outlook" (rain), "Humidity" (high), "Temperature" (hot), and "Windy" (false). Each property has a corresponding input field and a set of control buttons (V, C, +, -).

The "Documentation" field for the "Unseen\_sample" instance contains the text: "The sample from the testing set, which has been classified by using Naive Bayes method".

# Укрупненный алгоритм работы семантического адаптера

**Шаг 1. Создаем онтологическую модель для подаваемого на вход адаптера OWL-документа.**

**Шаг 2. Обрабатываем свойства модели**

*Цикл* по всем свойствам: {

*если* свойство типа данных или объектное свойство,

*то* свойство является атрибутом;

*если* атрибут типа данных,

*то* обрабатываем возможные значения;

*если* атрибут объектный,

*то* находим класс возможных значений, переходим к этому классу и обрабатываем его экземпляры как возможные значения атрибута

}

# Укрупненный алгоритм работы семантического адаптера (продолжение)

## Шаг 3. Обрабатываем классы и экземпляры модели

*Цикл* по всем классам: {  
  *если* класс неабстрактный и не является классом значений,  
  *то* класс соответствует значению целевого атрибута;  
  *Цикл* по всем экземплярам класса: {  
    заполняем выборку, представленную во внутреннем формате Web-сервиса:  
    - класс экземпляра заносим в целевой атрибут;  
    - значения всех свойств заносим в определяющие атрибуты. }  
}

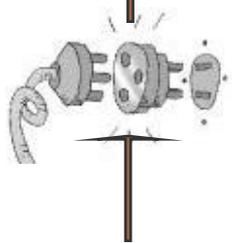


# Задания, выполненные адаптером

## (2): Обновление OWL-файла классифицированными примерами

```

- <Not_To_Play_Tennis rdf:ID="Unseen_sample">
  <Humidity rdf:resource="#high" />
  <Outlook rdf:resource="#rain" />
  <Temperature rdf:resource="#hot" />
  <Windy rdf:resource="#false" />
  <rdfs:comment>The sample from the testing set, which has been classified by using Naive Bayes method</rdfs:comment>
</Not_To_Play_Tennis>
    
```



Классифицированный пример в OWL формате

Outlook	Temperature	Humidity	Windy	Class
rain	hot	high	false	N

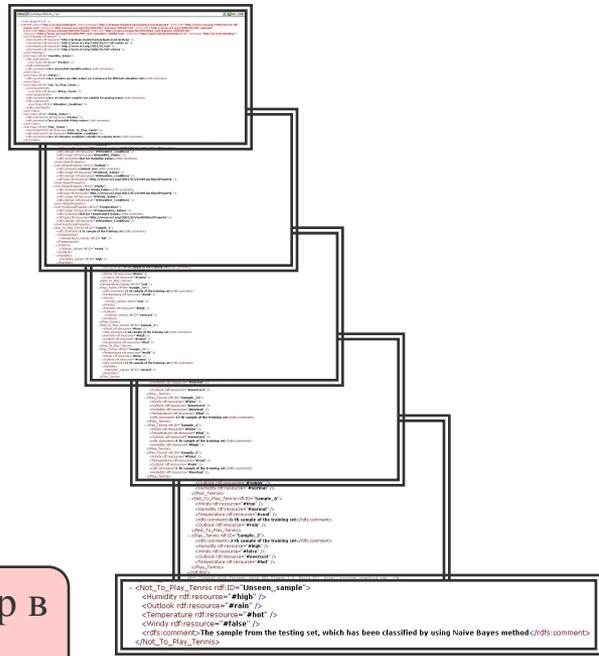
Классифицированный пример в табличной форме



Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

+

Outlook	Temperature	Humidity	Windy	Class
rain	hot	high	false	?



расширенная OWL онтология

# Результаты работы (1)

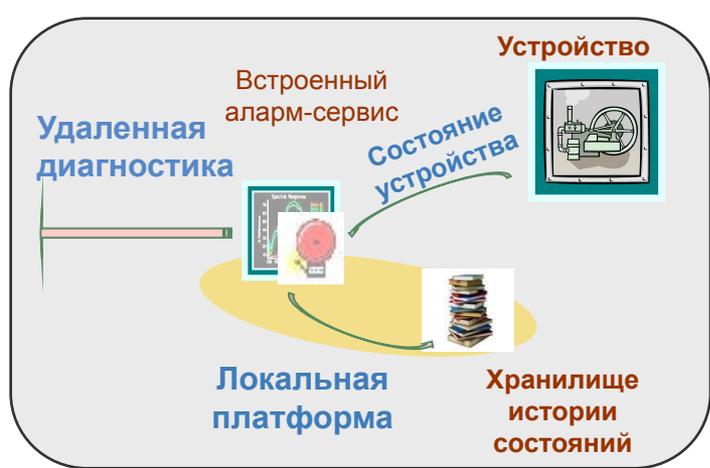
- Исследованы современные **Интернет-технологии**, связанные с Semantic Web, Web-сервисами, мультиагентными системами, интеллектуальной обработкой данных, автоматизированным промышленным мониторингом и удаленной диагностикой.
- Взята за основу концепция **GUN**, предполагающая проактивность Web-ресурсов и адаптеры для привязки ресурсов к Web.
- Был обоснован выбор стандартов представления информации в глобальных информационных системах, определенные W3C-консорциумом, таких как **RDF, RDFS и OWL**.

# Результаты работы (2)

- Был предложен способ **универсального представления обучающих и тестирующих выборок** для Web-сервиса в формате онтологий, закодированных средствами языка OWL.
- Был разработан **механизм семантической адаптации**, реализующий преобразование данных из OWL в табличное представление и обратно и позволяющий в перспективе совместно использовать (интегрировать) разнородные сервисы диагностики.

# Результаты работы (3)

- Был разработан **семантический адаптер** как программный модуль на языке **Java™ 2 SDK Standard Edition Version 1.4.2\_03** в среде **NetBeans™ IDE 3.6**. Была использована библиотека **JENA Semantic Web Framework Version 2.1**.
- Адаптер был экспериментально протестирован на примерах данных, подготовленных в **Protégé-2000**.
- Работа выполнялась как часть комплексной работы (5 человек).
- По результатам работы были опубликованы тезисы доклада на **8-м Международном Молодежном Форуме** «Радиоэлектроника и молодежь в XXI веке» и готовится статья.



# Спасибо за внимание !

