

Т.В. МОНАХОВА
**ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ ОПИСАНИЯ
ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ**

Монахова Т.В. Онтологическая модель описания экспериментальных данных.

Аннотация. В данной статье рассматривается вопрос о наиболее удобном и эффективном представлении данных, полученных экспериментальным путём, для хранения их в базах данных. Также приводится обзор широко применяемых методов организации данных с целью выбрать из них наиболее подходящий для рассматриваемой задачи. При этом рассматривается вопрос о целесообразности хранения экспериментальных данных в базах данных.

Ключевые слова: онтология, база данных, экспериментальные данные, организация данных.

Monakhova T.V. Experimental data description ontology model.

Abstract. The question of most convenient and effective experimental data presentation for database storage are wrote in this paper. A widely used data organization methods review are also presented for choosing the most suitable one for this subject. This paper also included a description of the data storage in databases.

Keywords: ontology, database, experimental data, organization of data.

1. Введение. Довольно часто данные, требуемые для построения функциональной модели предметной области и разрабатываемой информационно-системы, получены экспериментальным путём и нуждаются в упорядочении и определённой формализации. В настоящее время существует большое разнообразие способов организации данных, что позволяет выбрать оптимальный метод.

2. Обзор и сравнение методов организации данных. Одной из часто применяемых являются нотации IDEF, позволяющие представить данные о системе в виде набора взаимосвязанных функций. Это удобно, но в ряде случаев неприменимо. При таком подходе также можно отобразить на полученных диаграммах данные, требуемые для выполнения той или иной функции, указать части системы, выполняющие эти функции, а также ряд других параметров.

Другим широко известным способом представления данных является DFD – Data Flow Diagram, или диаграммы потоков данных. Как видно из названия, такие диаграммы отображают передачу данных между элементами системы. Часто эта методология входит в число нескольких, поддерживаемых каким-либо CASE-средством.

Также распространено представление данных о системе или предметной области в виде xml-документа. Такие документы весьма

напоминают html-страницы, так как тоже создаются с использованием набора тэгов и тоже могут просматриваться в стандартном интернет-браузере. При помощи языка xml можно представить как структуру проектируемой или исследуемой системы, так и её функции, представив их в виде древовидной структуры.

Довольно известна также нотация UML, позволяющая объединить в одной диаграмме элементы системы, функции, ими выполняемые и протекающие в системе события. Одним из несомненных плюсов методологии UML является возможность автоматизированного построения программного кода информационной системы на основе данных диаграмм, что существенно ускоряет и упрощает работу над системой.

Кроме перечисленных способов организации данных нельзя не упомянуть метод представления данных при помощи онтологий OWL. Онтология определяет множество сущностей, описывающих и представляющих предметную область и логические выражения соотношений терминов друг с другом. Онтологии OWL применяются для описания классификаторов в базах данных и приложениях для совместного использования информации предметной области, позволяя описать объекты с различной сложностью структуры.

Как и для других способов представления данных, для онтологий существуют специальные средства для их построения и поддержки. Эти средства можно представить в виде следующей таблицы.

Таблица 1. Системы работы с онтологиями

Название системы	Описание	URL-адрес
Средства создания онтологий		
Ontolingua	Поддержка совместной разработки	http://www.ksl.stanford.edu/software.html
WebOnto	Поддержка совместного просмотра	http://webonto.open.ac.uk
Protégé	Создание, просмотр, поддержка	http://protege.stanford.edu
OntoSaurus	Web-браузер баз знаний LOOM	http://www.isi.edu/isd/ontosaurus.html
ODE, WebODE	Создание, методология Methonthology	http://deliciciss.dua.fi.open.es/webODE/
KADS22	Проектирование моделей знания по методологии	http://www.swi.psy.uva.nl/projects/kads22.ind

	CommonKADS	
OntoEdit	Разработка и поддержка онтологий	http://www.ontopeuse.de/products/ontoedit
OdEd	Поддержка рас-суждения	http://oded.musn.ac.uk
i'com	Поддержка интел-лектуального кон-цептуального мо-делирования	http://www.inf_unity_it/~francoes/icom/
Объединение и сравнение онтологий		
PROMPT	Приложение объ-единения к Protege	http://protege.stanford.edu
Chimaera	Инструмент объ-единения на основе редактора онтоло-гий Ontolingua	http://www.ksl.stanford.edu
OntoMerge	Инструмент объ-единения и рас-суждения	http://cs-wwwcs.yale.edu/homes/dvm/detranslation.html
OntoMorph	Преобразование символических знаний	http://www.isi.edu/~hans/onto
OBSERVER	Информационная система на основе взаимодействия онтологий	http://lsdis.cs.uga.edu/method
FCA-Merge	Метод сравнения онтологий	http://www.aifb.uni
ONION	Система компози-ции онтологий	http://www.sees.anticweb
Аннотирование Web-ресурсов		
SHOE's Knowledge Annotator	Описание содер-жания Web-страниц	http://www.cs.umd.edu
MnM	Аннотация на ос-нове онтологий	http://kmi.open.ac.uk
Metabrowser	Создание и про-смотр метаданных	http://metabrowser/spirit.su

Вопрос выбора оптимального варианта представления данных имеет существенное значение для проектирования информационной системы. Также важно избежать большого количества преобразований данных, поскольку при этом возникает проблема неоднозначности результатов трансформаций. Кроме того, вполне возможно, что какое-либо необходимое преобразование данных окажется попросту невоз-

можным. Это заставит вернуться к одному из ранее полученных вариантов представления, а возможно, и к выбору первоначального формата данных.

Для накопления, хранения и использования экспериментальных данных наиболее применимы базы данных, представляющие собой структурированные совокупности данных различных типов. Данные в базах хранятся и обрабатываются в соответствии со специализированными форматами баз данных. В некоторых из этих форматов (например, dBase) можно определить только структуру хранения данных, в других же (например, Paradox) на данные можно наложить дополнительные ограничения.

3. Средства проектирования баз данных. Для работы с различными форматами баз данных, а также для проектирования баз данных используются специализированные программные средства. К примеру, простую базу данных можно создать на языке Pascal при помощи объектно-ориентированной системы Turbo Vision. При этом данные хранятся в объекте типа, производного от TCollection – типа, используемого для хранения набора объектов определённого типа. По своей сути, Turbo Vision является иерархией типов. Программа, написанная с использованием средств Turbo Vision, представляет собой набор объектов, взаимодействующих друг с другом при помощи механизма обработки событий. Очевидно, что при помощи этой системы можно создать не только базу данных, но и любое другое приложение.

Кроме того, существуют средства проектирования более сложных баз данных. К примеру, такие продукты предлагает ряд известных компаний: Oracle, Microsoft или IBM. Также можно использовать хорошо себя зарекомендовавшие MySQL и PostgreSQL, хоть они и не обладают таким широким, как у средств IBM, набором функций. Среди достоинств этих пакетов можно назвать взаимодействие со скриптами CGI, что делает их весьма полезными при работе с Internet. Кроме того, MySQL является широко популярным продуктом вследствие его бесплатного распространения.

4. Онтологическая модель организации данных. Для большей наглядности приведём пример построения онтологической модели описания экспериментальных данных. Допустим, что мы имеем некий набор однотипных данных о каких-либо событиях, полученных в различных подразделениях какой-либо научной организации. Структуру таких данных можно представить в виде xml-файла или онтологии owl.

Рассматриваемые данные, представленные в xml, выглядят ниже-следующим образом. Первая строка является обязательной для всех

xml-документов и сообщает программе-обработчику используемую версию языка xml и кодировку документа, в данном случае – utf-8. Далее указывается корневой элемент (один для каждого документа) описываемой структуры, после него – его вложенные элементы и элементы, входящие в состав этих дочерних элементов. Для каждого из элементов можно указать атрибуты, описывающие его. Таким образом, наш документ, составленный на языке xml версии 1.0 в кодировке utf-8, описывает структуру данных, состоящих из трех блоков, каждый из которых состоит из четырех элементов данных, характеризующихся двумя параметрами, одинаковыми для всех элементов. Значения параметров, разумеется, могут различаться для каждого рассматриваемого элемента данных.

```
<?xml version="1.0" encoding="utf-8" ?>
<Data>
  <Part1>
    <Data_item1 par1="1" par2="2"/>
    <Data_item2 par1="1" par2="2"/>
    <Data_item3 par1="1" par2="2"/>
    <Data_item4 par1="1" par2="2"/>
  </Part1>
  <Part2>
    <Data_item1 par1="1" par2="2"/>
    <Data_item2 par1="1" par2="2"/>
    <Data_item3 par1="1" par2="2"/>
    <Data_item4 par1="1" par2="2"/>
  </Part2>
  <Part3>
    <Data_item1 par1="1" par2="2"/>
    <Data_item2 par1="1" par2="2"/>
    <Data_item3 par1="1" par2="2"/>
    <Data_item4 par1="1" par2="2"/>
  </Part3>
</Data>
```

Листинг 1. Xml-представление данных.

Обратите внимание, что в примере не указаны конкретные названия частей структуры представляемых данных, самих элементов данных и их параметров. Это сделано для того, чтобы избежать навязывания читателю каких-либо определённых жёстких рамок представления. Ведь сами данные могут варьироваться в зависимости от конкрет-

ных нужд пользователя, их можно по-разному сгруппировать, а кроме того, выбрать различные атрибуты в качестве заслуживающих упоминания в документе, или же вовсе указать все из них, что тоже возможно, хотя и сделает документ более запутанным и менее наглядным.

Далее рассмотрим представление данных в формате owl-онтологии. Для этого приведём фрагмент онтологии рассматриваемой предметной области.

```
<Rdf:RDF
  Xmlns = " http://www.w3.org/TR/2004/REC-owl-guide-200402
10/data * "
  Xmlns:vin = " http://www.w3.org/TR/2004/REC-owl-guide-200402
10/data * "
  Xml:base = " http://www.w3.org/TR/2004/REC-owl-guide-200402
10/data * "
  Xmlns:comp = " http://www.w3.org/TR/2004/REC-owl-guide-200402
10/comp * "
  Xmlns:owl = " http://www.w3.org/2002/07/owl * "
  Xmlns:rdf = " http://www.w3.org/1999/02/22-rdf-syntax-ns * "
  Xmlns:rdfs = " http://www.w3.org/2000/01/rdf-schema * "
  Xmlns:xsd = " http://www.w3.org/2001/xmlschema * ">
< Owl: Ontology rdf:about = "">
<Rdfs:comment> пример онтологии OWL </rdfs:comment>
< Owl:priorVersion rdf:resource = " http://www.w3.org/TR/2003/PR-
owl-guide-2003121 5/data "/>
< Owl:imports rdf:resource = "http://www.w3.org/TR/2004/REC-owl-
guide-200402 10 comp "/>
<Rdfs:label> Пример онтологического описания эксперименталь-
ных данных</rdfs:label>
<owl:Class rdf:ID="Data"/>
```

Листинг 2. Фрагмент онтологии owl.

На примере данного фрагмента рассмотрим некоторые из основных частей онтологии. Самой первой частью любой типичной онтологии является декларация пространства имён. Такие декларации определяют пространства имён, которые используются в данном документе, а также задают пространство имён, используемое по умолчанию.

Далее после тега owl:Ontology следует заголовок онтологии. Здесь можно добавлять комментарии, управлять версиями и включать в документ другие онтологии.

Кроме того, для описания данных в онтологии пользователем создаются соответствующие классы. Любой объект в owl рассматривается как член класса owl:Thing. Также поддерживается класс owl:Nothing, пустой, как видно из его названия. Кроме того, создадим основной класс Data.

Кроме классов онтология поддерживает объекты, т.е. присутствует объектно-ориентированный подход. Помимо этого поддерживаются различные виды свойств и типов данных.

Онтологии довольно часто используются в базах данных для классификации данных. Они удобны тем, что позволяют классифицировать несколько предметных областей и с их помощью можно описать структуру различной степени сложности. Кроме того, существуют инструментальные средства, поддерживающие автоматизированное использование онтологий.

5. Заключение. Из всего вышесказанного мы можем сделать вывод об удобстве таких методик представления данных, как xml и owl и т.д. Систематизировав данные умозаключения, мы увидим, что полученные экспериментальным путём данные логично хранить в базе данных, и при этом удобно использовать для их описания и классификации онтологии owl.

Литература

1. *Воробьёв В.И., Монахова Т.В.* Функциональное моделирование системы информационной защиты //Труды СПИИРАН. СПб.: СПИИРАН, 2004.
2. *Воробьёв В.И., Монахова Т.В.* Проектирование систем защиты с применением онтологий//Труды СПИИРАН. СПб.: СПИИРАН, 2004.
3. *Воробьёв В. И., Фаткиева Р. Р., Перминов С. В.* Применение онтологического моделирования для поиска информационных аномалий//Материалы XI Международной научно-практической конференции «Информационная безопасность» Ч.2: 22 – 25 июня 2010, Таганрог: Изд-во ТТИ ЮФУ, 2010. – с. 15 – 18.
4. *Ковтанюк Ю.С.* Программирование на Turbo Pascal. М.: Эксмо, 2008. 592 с.
5. *Вей О.М., Проскудина Г.Ю.* «Обзор инструментов инженерии Онтологий». Институт программирования.
6. *Наталья Ф.Ной(Natalya F.Noy), Дэбора Л.МакГиннесс(Deborah L.McGuinness)* «Разработка онтологий» Стэнфордский Университет, Стэнфорд, Калифорния.
7. www.w3c.org
8. www.escc.ru
9. <http://sherdim.ru>

Монахова Татьяна Вячеславовна — аспирант лаборатории вычислительных систем и проблем защиты информации СПИИРАН, младший научный сотрудник Четвёртого центрального научно-исследовательского института Министерства Обороны Российской Федерации (4 ЦНИИ МО РФ). Область научных интересов: моделирование информационных систем. Число научных публикаций - 3. panthornator@yandex.ru; 4 ЦНИИ МО РФ,

ул. М.К. Тихонравова, 39, Юбилейный, 141091, РФ. Научный руководитель – В.И. Воробьев.

Monakhova Tatyana Vyacheslavovna — postgraduated student, Laboratory of Computing Systems and Information Protection Problems, SPIIRAS. Junior researcher, 4 Central Science Institute of the Ministry of Defense, Russia. Research interests: information systems modeling. The number of publications — 3. panthernator@yandex.ru; 4 CSI MD, Tikhonravova st., 39, Yubilejnyj, 141091, Russia. The supervisor of studies is V.I. Vorobyev.

Рекомендовано лабораторией информационно-вычислительных систем СПИИРАН, заведующий лабораторией Воробьев В.И, д-р техн. наук, проф.
Статья поступила в редакцию 25.12.2012.

РЕФЕРАТ

Монахова Т.В. **Онтологическая модель описания экспериментальных данных.**

В данной статье рассматривается выбор оптимального метода организации данных, полученных путём проведения эксперимента. Такие данные обычно характеризуются постоянным набором параметров, значения которых меняются для каждого эксперимента.

На сегодняшний день существует довольно большое разнообразие методов представления данных, и мы должны выбрать среди них наиболее подходящий для описания экспериментальных данных. После исключения из рассмотрения методов, ориентированных на описание функций, мы получаем уже меньшее количество вариантов для выбора. Из этого подмножества наиболее подходящим кажется метод представления данных в виде онтологии owl. На нём мы и остановимся более подробно.

Онтология позволяет создавать различные классы и их объекты, что очень удобно для организации экспериментальных данных в виде объектов с определённым набором характеристик. Такие данные могут быть сохранены для последующего использования. Лучше всего для этого подходят базы данных.

Базы данных тоже различаются по своей структуре, и для их построения есть различные инструментарии, как и для работы с ними. В данной статье мы не рассматривали каждое из этих средств подробно, остановившись лишь на самых известных и применяемых. К таким средствам можно отнести MySQL, PostgreSQL, продукты компании IBM, и, для простых баз данных, Turbo Vision.

Как следует из всего вышесказанного, данные, полученные экспериментальным путём, имеют характерную структуру, что позволяет описать их в виде owl-онтологии. В свою очередь, owl-онтологии удобны для работы с базами данных. Кроме того, в базах данных удобно хранить экспериментальные данные и использовать их. Очевидно, что данные, являющиеся результатами экспериментов, весьма целесообразно сперва упорядочить в виде owl-онтологии, а затем сохранить в базе данных.

SUMMARY

Monakhova T.V. **Experimental data description ontology model.**

An optimal method of experimental data organization is described in this paper. Such data usually characters by constant set of parameters. This parameters values are changed for each experiment.

There are very big variety of data representation methods now, so we have choose a most suitable for experimental data one. If not review function oriented methods, we get littlest number of variants for choice. Data representation by an owl ontology method seems to be most suitable one from this subset. We will consider this method more in detail.

There are different classes and objects of this classes may be created in ontology, and this is very convenient for experimental data organization with a determined set of characteristics. Such data may be saved for a following usage. Databases are better for it.

Databases also may be different by it's structure and there are different program tools for it's creating and working with databases. We will not consider each of this programs in detail in this paper, we will describe only most popular ones. These tools are MySQL, PostgreSQL, IBM products, and, for simple databases, Turbo Vision.

As follow from wrote, data that got as an experiment result, has a character structure and it is allow to describe this data as an owl ontology. In turn, owl-ontology are convenient for work with databases. Also an experimental data are convenient to store and use in databases. It is obviously that experimental results data are very expediently to order by owl ontology at first, and save in database then.