

Публикация реляционной базы данных электронной библиотеки в Semantic Web. Представление метаданных в виде связанных данных

© Новицкий А.В.

Институт программных систем НАН Украины

ale-nov@yandex.ru

Аннотация

В статье рассмотрена проблема публикации реляционной базы данных электронной библиотеки в Semantic Web, в соответствии с концепцией связанных данных. В качестве сервера для публикации используется D2R Server.

1. Введение

Развитие электронных библиотек (ЭБ) непосредственно связано с развитием технологий доступа к информации и организации информации. Семантические технологии являются одними из наиболее динамично развивающихся технологий среди подходов к представлению информации в машиночитаемом виде.

Рассмотрим задачи, для решения которых применяются семантические технологии. Можно выделить два основных направления. Первое направление связано с развитием и внедрением сервисов, которые имеют определенное семантическое описание (WSMO, OWL-S, SAWSL т.д.) с целью решения задач их автоматической композиции. Однако, это актуально лишь в том случае, когда рассматривается большое количество сервисов. Поэтому целесообразно говорить о новом классе ЭБ как средстве поддержки сложных процессов коммуникации, хранения и обработки информации. В классических ЭБ задачи автоматической композиции сервисов отсутствуют ввиду конечного множества сервисов и обозначенных целей.

Второе направление связано с представлением, поиском и организацией доступа к информации в ЭБ. В рамках решения этой задачи схема ЭБ расширяется метаинформацией об информационных объектах ЭБ. Как правило, в качестве схемы метаинформации используют Dublin Core (DC). Для эффективного поиска в ЭБ метаданные необходимо собрать и представить

соответствующим образом. Semantic Web дает набор технологий, которые позволяют управлять метаданными.

В дальнейшем будем называть поиск на основе реляционной модели данных *реляционным поиском*, а поиск в сетевых моделях данных (к которым относятся семантические сети) - *сетевым поиском*.

2. Linked Data и метаданные

2.1 Linked Data

Для ЭБ технология Semantic Web позволяет решить ряд принципиальных проблем таких как:

- интеграция информации, представленной в различных моделях метаданных;
- обеспечение взаимодействия с другими системами (не только электронными библиотеками);
- удобного и адаптированного поиска с соответствующими интерфейсами для отображения семантики.

Одной из действующих моделей Semantic Web является модель связанных данных - Linked Data (LD). Основные принципы Linked Data изложены в [1]. Преимущество связанных данных заключается в том, что ценность и полезность данных увеличивается по мере увеличения количества связей с другими данными. Основные принципы связанных данных состоят в следующем:

1. в качестве имен сущностей используются URI;
2. для того, чтобы человек мог различать имена, используется HTTP;
3. в URI следует представлять полезную информацию, то есть они должны быть осмыслены;
4. ресурс должен содержать ссылки на другие URI с целью раскрытия дополнительной информации о сущности.

Поэтому, естественным образом, встает вопрос о представлении ресурсов ЭБ в соответствии с концепцией связанных данных. В работе [2] уже описано представление DOI как Linked data, также в [3], [4] описаны выражения элементов общепринятых схем метаданных как Linked data.

В данной работе рассматривается проблема публикации данных ЭБ в соответствии принципам

Linked Data. В качестве примера используется свободное ПО Eprints. Полученные результаты возможно распространить на Dspace и другие подобные системы.

Как известно, в большинстве случаев данные в таких ЭБ хранятся в реляционной базе данных. Поэтому далее будут рассмотрены некоторые основные идеи отображения реляционной базы данных в модель Linked Data. В качестве примера ПО для публикации Linked Data используется D2R-SERVER [5], а в качестве источника данных - реляционная база данных Eprints [6].

2.2 Отображение реляционной базы данных в Linked Data

D2R Server [5] является инструментом для публикации реляционных баз данных в Semantic Web. Он позволяет RDF и HTML-браузерам перемещаться по содержанию базы данных, а приложениям - запрашивать информацию из базы данных, используя язык запросов SPARQL. Детальное исследование использования данного ПО для публикации веб-сайтов как Linked Data можно найти в работе [7].

Рассмотрим, каким образом происходит отображение реляционной базы данных в Linked Data. Как известно, базовыми понятиями реляционной базы данных являются: тип данных, домен, атрибут, кортеж, первичный ключ и отношение. Отношение (relation) - это вся структура целиком, набор записей (в обычном понимании - таблица). Кортеж - это строка, содержащая данные. Более распространённый, но менее формальный термин - запись. Атрибут - это столбец в отношении.

Рассмотрим основные идеи публикации реляционной базы данных в виде связанных данных. Для этого укажем соответствия основных понятий реляционной модели данных и понятий модели связанных данных.

В связанных данных каждый ресурс, который имеет уникальный URI, описывается с помощью модели данных RDF. Ресурсом в RDF может быть любая сущность - как информационный ресурс (например, веб-сайт или изображения), так и не информационный ресурс (человек, город или некое абстрактное понятие). Ресурс состоит из списка утверждений в виде «субъект - предикат - объект», каждое такое утверждение называется триплетом. Для обозначения субъектов, предикатов и объектов в RDF используются URI. Множество RDF-утверждений образует ориентированный граф, в котором вершинами являются субъекты и объекты, а ребра являются предикатами. Схема соответствия понятий моделей приведена в таблице 1.

Для того, чтобы более подробно проиллюстрировать процесс отображения, обратимся к рисунку 1, на котором показано, как элементы отношения представляются RDF-графом. В примере на рисунке 1 представлено некоторое отношение, которое содержит информацию о

метаданных. Данное отношение может быть описано тремя информационными ресурсами, каждый из которых образуется двумя триплетами.

Таблица 1. Соответствие понятий реляционной модели и модели связанных данных

Реляционная база данных	Связанные данные
Тип данных	XML schema datatypes
Атрибут	Предикат, который определяется через общепринятые словари и онтологии
Кортеж	Предикат-объект
Первичный ключ	Субъект
Отношение	Ресурс

Table.Metadata

id (key)	title	date
1	Book about DC	02-02-2010
2	Book about PC	03-02-2011
3	Book about Notebook	25-09-2011

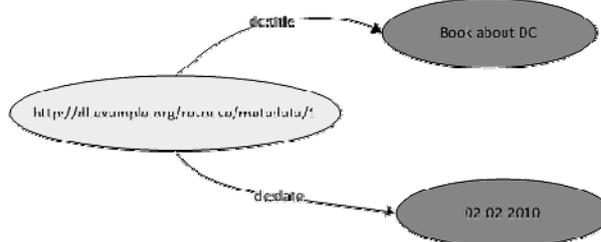


Рис. 1 Отношение и представления кортежа в RDF

eprint	document
eprintid: int	docid: int
rev_number: int	rev_number: int
eprint_status: varchar(255)	eprintid: int
userid: int	pos: int
type: varchar(255)	format: varchar(255)
succeeds: int	formatdesc: varchar(255)
commentary: int	language: varchar(255)
replacedby: int	security: varchar(255)
metadata_visibility: varchar(255)	license: varchar(255)
contact_email: varchar(255)	main: varchar(255)
fileinfo: text	date_embargo_year: smallint
latitude: float	date_embargo_month: smallint
longitude: float	date_embargo_day: smallint
title: text	content: varchar(255)
language: varchar(255)	placement: int
abstract: text	
keywords: text	
coverage: text	
status: varchar(255)	
role: varchar(255)	
entity: text	
date_year: smallint	
full_text_status: varchar(255)	

Рис. 2 Часть схемы реляционной базы данных Eprints 3

2.3 Публикация связанных данных

Рассмотрим в качестве примера отображение реляционной базы данных ПО Eprints в модели связанных данных. Ниже на рисунке 2 представлено подмножество схемы базы данных Eprints 3.2.2. Для публикации из таблиц схемы были выбраны 27, на рисунке представлены две из них.

Как было ранее отмечено, для публикации LD в данной работе используется ПО D2R server. Сопоставление между реляционной моделью данных и LD происходит путем определения специального файла на основе спецификации [8].

RDF-схема сопоставляется со схемой реляционной базы данных при помощи конструкций `d2rq:ClassMaps` и `d2rq:PropertyBridges`. Центральным объектом в D2RQ, а также объектом, с которого начинается построение новой схемы отображения, является `ClassMap`. Понятие `ClassMap` представляет собой класс или группу, аналогичную классам RDF. `ClassMap` также определяет способ идентификации экземпляров класса. `ClassMap` имеет наборы `PropertyBridges`, которые определяют свойства экземпляров класса.

Приведем пример файла, отвечающего за отображение. В файле на основе спецификации [8] определено пространство имен для DC и квалификаторы DC. Например, для описания формата файлов используется словарь DCMI Metadata Terms, который определяется пространством имен `@prefix dct:1`.

Задается также соответствие первичных ключей и субъектов, атрибутам отношений и предикатам. Проиллюстрируем разработанную схему отображения.

Формат файла ресурса `dct:format` (формат документа в ЭБ) определяется с использованием сопоставления свойств из различных таблиц:

```
map: eprint__format a d2rq: PropertyBridge;
d2rq: belongsToClassMap map: eprint;
d2rq: property dct: format;
d2rq: column "document.format";
d2rq: join "document.eprintid = eprint.eprintid";
```

Схема отображения заголовка (`title`), типа (`type`), и описания ресурса (`description`) выглядит следующим образом:

```
map: eprint__title a d2rq: PropertyBridge;
d2rq: belongsToClassMap map: eprint;
d2rq: property dc: title;
d2rq: propertyDefinitionLabel "eprint title";
d2rq: column "eprint.title";
```

```
map: eprint__type a d2rq: PropertyBridge;
d2rq: belongsToClassMap map: eprint;
d2rq: property dc: type;
d2rq: propertyDefinitionLabel "eprint type";
d2rq: column "eprint.type";
map: eprint__abstract a d2rq: PropertyBridge;
```

```
d2rq: belongsToClassMap map: eprint;
d2rq: property dc: description;
d2rq: propertyDefinitionLabel "eprint abstract";
d2rq: column "eprint.abstract";
```

Заметим, что построенная таким образом схема отображения базы данных Eprints в модель LD не полна. Для того, чтобы разработанное представление полностью соответствовало концепции связанных данных, а именно, чтобы ресурс был связан с другими ресурсами, необходимо, чтобы объект являлся субъектом для некоторого другого триплета. Для этого в схему дополнительно добавляется конструкция, которая позволяет связывать ресурсы между собой:

```
map: eprint__label a d2rq: PropertyBridge;
d2rq: belongsToClassMap map: eprint;
d2rq: refersToClassMap map: document;
d2rq: property rdfs: seeAlso;
d2rq: join "document.eprintid => eprint.eprintid";
```

На основании вышеизложенных идей была построена модель LD для реляционной базы данных Eprints.

На рисунке 3 приведен вид информационного ресурса с идентификатором 1002, опубликованного при помощи D2R server на основе разработанной схемы отображения.

При разработке отображения была обнаружена проблема, связанная с неправильным проектированием базы данных. Схема реляционной базы данных ПО Eprints оказалась неподготовленной к представлению ее в виде LD для всех атрибутов. Например, атрибут, отвечающий за публикацию статьи в определенном печатном издании, не вынесен в отдельное отношение. При этом невозможно сгруппировать в RDF объект все статьи, опубликованные в издании. Это, в свою очередь, не позволяет создать ресурс, который имеет свой URI и содержит описание неинформационных ресурсов о печатном издании. Причиной данной коллизии является неправильное выделение типа сущностей реляционной базы данных. В данном случае, типом сущности являются элементы DC, которые необходимо принимать в качестве атрибута и вносить в одном отношении только на основе уникальности экземпляра сущности.

Таким образом, при проектировании базы данных для ЭБ необходимо соблюдать следующее ограничение: атрибуты, соответствующие элементам DC могут содержаться в одной таблице только тогда, когда значение этих атрибутов является уникальным для электронного ресурса, описываемого DC. В противном случае, такие атрибуты следует группировать в отдельные таблицы и связывать по ключу. В таблице 2 приведены требования для DC.

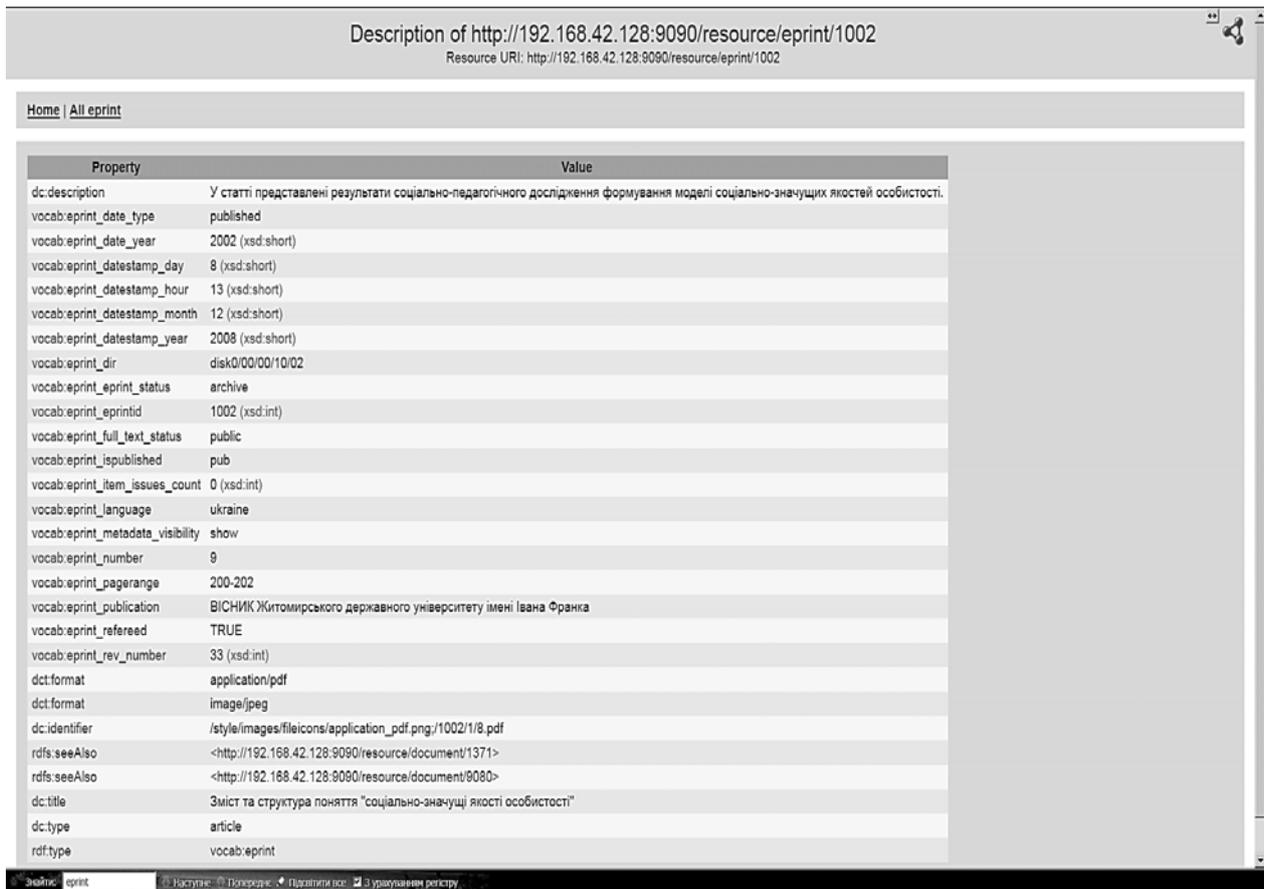


Рис. 3 Информационный ресурс с ИД 1002 опубликованный при помощи D2R server

Таблица 2. Требования для Dublin Core

Элемент дублинского ядра	Принадлежность атрибута к одному отношению	Вынесение атрибута в отдельное отношение
Title	+	-
Creator	-	+
Subject	-	+
Description	+	-
Publisher	-	+
Contributor	-	+
Date	+/-	+/-
Type	-	+
Format	-	+
Identifier	+	-
Source	-	+
Language	+	-
Relation	+	-
Coverage	+	-
Rights	-	+

Следует отметить, что для другого популярного ПО Dspace проблема также является актуальной, поскольку схема базы данных этого ПО похожа на схему Eprints и имеет те же проблемы, связанные с проектированием.

2.4 Поиск информации в сети связанных данных

Полученная схема связанных данных построена на основе реляционной схемы. Запросы к такой схеме, представленной в семантической модели данных (сетевой поиск), транслируются в запросы к реляционной схеме (реляционный поиск). Рассмотрим некоторые примеры запросов и ответов.

Пример 1.
 SELECT?x?y
 WHERE {
 ?x rdfs: seeAlso?y.
 ?x dc: type "article".
 }
 LIMIT 3

Результат выполнения данного запроса будет иметь следующий вид:

x	y
<http://192.168.42.128:9090/resource/eprint/61>	<http://192.168.42.128:9090/resource/document/4030>
<http://192.168.42.128:9090/resource/eprint/61>	<http://192.168.42.128:9090/resource/document/11313>
<http://192.168.42.128:9090/resource/eprint/311>	<http://192.168.42.128:9090/resource/document/2417>

Пример 2.
 SELECT?s?p?o
 WHERE {
 ?s?p?o.
 ?s dc: type?o
 }

Результат будет иметь следующий вид:

s	p	o
< http://192.168.42.128:9090/resource/eprint/129 >	< http://purl.org/dc/elements/1.1/type >	"Article"
< http://192.168.42.128:9090/resource/eprint/13 >	< http://purl.org/dc/elements/1.1/type >	"Article"
< http://192.168.42.128:9090/resource/eprint/15 >	< http://purl.org/dc/elements/1.1/type >	"Book_section"
< http://192.168.42.128:9090/resource/eprint/59 >	< http://purl.org/dc/elements/1.1/type >	"Article"

Следует подчеркнуть, что D2R Server не поддерживает вывод знаний при запросах.

3. Заключение

В работе рассмотрены идеи публикации реляционных баз данных в виде связанных данных. Подход проиллюстрирован на примере отображения базы данных Eprints в модель Linked Data, поддерживаемую D2R Server. Рассмотрены некоторые ограничения, которые необходимо соблюдать при проектировании баз данных для ЭБ для того, чтобы сделать возможной их публикацию в виде связанных данных. Дальнейшая работа связана с проектированием виртуальных таблиц и их отображением в модель Linked Data.

Литература

- [1] Berners-Lee, T.: Linked Data. In: World Wide Web Consortium (W3C). 2009. - <http://www.w3.org/DesignIssues/LinkedData.html>
- [2] Erickson, J.: DOIs, URIs and Cool Resolution. In: Bitwacker Associates. 2010. - <http://bitwacker.wordpress.com/2010/02/04/dois-uris-and-cool-resolution/>
- [3] Ed Summers, A.: LCSH, SKOS and Linked Data. Proc. Int'l Conf. on Dublin Core and Metadata Applications, 25-33 (2008)
- [4] Baker, T.: Tutorial: Dublin Core -Building blocks for interoperability. In : Dublin Core and Linked Data, Tokyo (2010). - <http://www.meta-proj.jp/ev-1/ev-p3.pdf>
- [5] Chris Bizer, R.: D2R Server - Publishing Relational Databases on the Semantic Web. In: WWW4 - research application server of the Lehrstuhl für Wirtschaftsinformatik. - <http://www4.wiwiwiss.fu-berlin.de/bizer/d2r-server/>
- [6] University of Southampton: EPrints - Digital Repository Software. - <http://www.eprints.org/>
- [7] Wang, X.: Investigating the Use of Linked Data for Exposing the Data from the Catalhoyuk Web Site., Southampton (2009). - <http://2tree.brinkster.net/Resource/Dissertation/Mat er.pdf>.
- [8] Chris Bizer, R.: The D2RQ Plattform v0.7 - User

Manual. In: WWW4 - research application server of the Lehrstuhl für Wirtschaftsinformatik. <http://www4.wiwiwiss.fu-berlin.de/bizer/d2rq/spec/>

Publishing of a Relational Database of a Digital Library on the Semantic Web

© Oleksandr Novytskyi

The paper considers a problem of publishing of a relational database of a digital library in the Semantic Web in accordance with the concept of Linked Data. As a publishing server the D2R Server is used.