ИНТЕГРАЦИЯ РЕЛЯЦИОННЫХ БАЗ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ RDF ГРАФОВ

Рафиков М.Р.
Научный руководитель: Тузовский А.Ф.
Томский политехнический университет, Институт кибернетики
mrr1@tpu.ru

Введение

В настоящее время, информация большинства организаций хранится в базах данных, к которым может быть предоставлен доступ в локальной или глобальной компьютерной сети. Информация об одних и тех же объектах организации может содержаться в различных базах данных, которые могут иметь разную структуру. Для получения полной информации об объекте из различных баз данных требуется выполнить их интеграцию. Данная задача может быть решена путем преобразования данные из различных источников в единый формат. В качестве такого формата могут быть использованы языки, разработанные и стандартизированный организацией W3C, в рамках разработки следующего поколения веб-сети – Semantic Web.

Единое описание информации

В технологиях Semantic Web предложен единообразный формат представления разнородной информации – язык RDF (Recourse Description Framework) [1]. Данный язык RDF описывает информацию в виде триплетов «subject-predicat-object», которые являются простыми логическими утверждениями. В триплетах: subject – это некоторый ресурс (понятие), заданный с помощью URI; predicat – это некоторое свойство данного ресурса, заданное с помощью заданное с помощью URI; object – это значение данного свойства у данного ресурса, заданное с помощью заданное с помощью URI или текстовой строки. Каждый триплет можно рассматривать в виде именованной дуги (predicat), связывающей вершинs subject и object. Набор RDF утверждений называется RDF графом. URI (Universal Resource Identifier) являются глобально уникальными идентификаторами любых сущностей, а не только ресурсов WWW. Основными преимуществами формата RDF является:

1. RDF формат является универсальным способом описания информации. Стандартные формы представления данных (реляционные базы данных, электронные таблицы и т.п.) могут быть преобразованы в RDF-формат.
2. RDF графы могут очень просто интегрироваться путем простого их объединения (слияния). При этом вершины, которые имеют одинаковые URI просто объединяются.
3. Для работы с RDF данными разработаны другие языки. Например, такие, как SPARQL – язык запросов к RDF данным и языки RDFS и OWL для описания семантических моделей (онтологии, словари), на основе которых формируются и обрабатываются RDF данные.

Преобразование реляционных данных в RDF

Для приведения реляционной базы данных в RDF формат используется платформа D2RQ [2]. В составе D2RQ присутствуют средства для отображения и преобразования данных из реляционных СУБД в RDF граф с помощью инструмента «generate-mapping». Также есть возможность получить данные в виде триплетов.

После преобразования появляется возможность работать с реляционной базой данных как с RDF графом, а именно: составлять SPARQL запросы и использовать полученный граф в Jena [3].

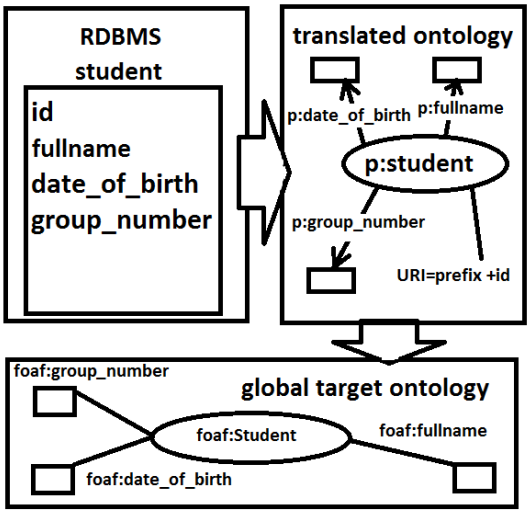

Рисунок 1 - Двухэтапное Отображение СУБД в целевую онтологию

Также D2RQ позволяет посмотреть полученные RDF class, как отдельное приложение Semantic Web. Для этого используется встроенный в платформу, D2R-server. Он показывает информацию, построенную после mapping в HTML и RDF вариантах.

Рисунок 2 - SPARQL запрос на D2R-server

Для обработки RDF графов используется фремверк Jena для Java. Большое преимущество в этом дает D2RQ, который может использоваться как элемент приложения для отображения RDF графов.

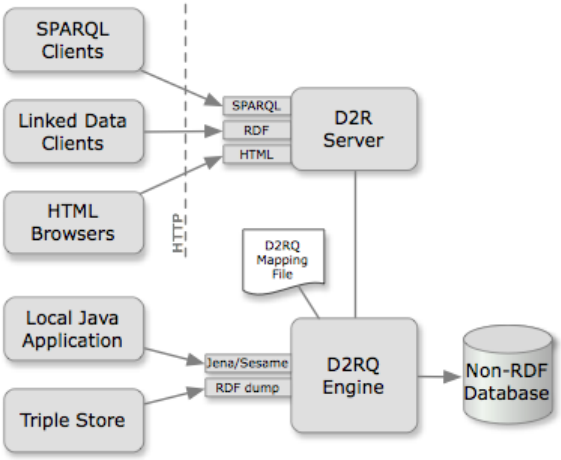

Рисунок 3 - Общая схема D2R-server

Общая модель данных

Для подготовки к выполнению глобальных запросов к данным необходимо собрать все полученные после “generate-mapping” RDF графы. Для этого следует создать каталог метаданных, которое представляет собой RDF хранилище.

Выполнение SPARQL запросов к набору

Система Jena ARQ[5] предоставляет базовую функциональность для потокового выполнения SPARQL запросов и реализует набор итераторов для выполнения потоковых и материализованных связываний, левых связываний, объединений, фильтров. Пользователи системы интеграции выполняют запрос к посреднику, который далее выполняет подзапросы к подходящим источникам данных, чтобы сформировать составить, результат запроса. Система является масштабируемой относительно количества источников данных и количества и размера онтологий.

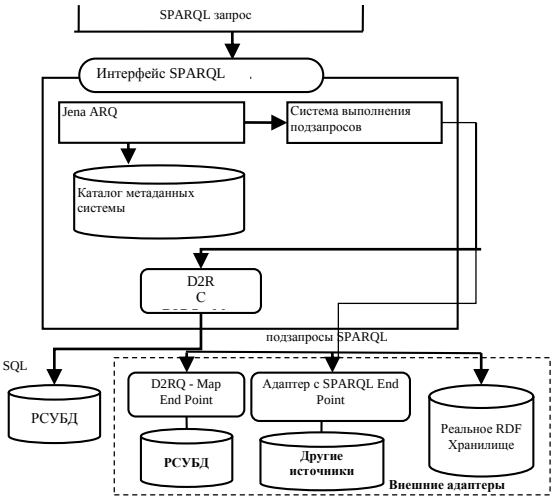

Рисунок 4 - Структурная система выполнения запросов к набору

Заключение

При использовании стека технологий D2RQ, Jena можно добиться решения задачи интеграции данных. Решение данной задачи позволит с точки зрения пользователя использовать одну интегрированную информационную систему, предоставляющую всю информацию из подсистем, которые, например, могут быть системами баз данных, мультимедийными серверами, хранилищами документов, наборами электронных таблиц, экспертными системами или Web сервисами.

Список использованных источников

1. RDF http://www.w3.org/RDF/[Электронный ресурс] Режим доступа: свободный

2. D2RQ http://d2rq.org/ [Электронный ресурс]. – Режим доступа: свободный

3. Apache Jena http://jena.apache.org/ [Электронный ресурс]-Режим доступа свободный

4. Langegger A. A Flexible Architecture for Virtual Information Integration based on Semantic Web Concepts (phd thesis) // Institute for Application Oriented Knowledge Processing, 2010

5. ARQ - A SPARQL Processor for Jena http://jena.apache.org/documentation/query/ [Электронный ресурс]-Режим доступа: свободный