УДК 004.822:514

МЕТОД ОБРАБОТКИ ДАННЫХ И ЗНАНИЙ В СЕТИ ИНТЕРНЕТ С ПОДДЕРЖКОЙ ШИФРОВАНИЯ

В.А. ВИШНЯКОВ, Д.С. БОРОДАЕНКО

Минский институт управления, Лазо, 12, Минск, 220012, Беларусь

ЕПАМ, Радиальная, 40, Минск, 220070, Беларусь

Поступила в редакцию 3 сентября 2013

С целью эффективности обработки больших объемов данных и знаний в Интернете средствами семантики языка RDF представлен метод отображения реляционных БД на модель RDF. Суть метода заключается в интеграции: модели адаптации реляционных данных для отображения на модель RDF, процедур логического вывода; алгоритмов преобразования запросов на доступ к RDF в запросы SQL, и обновления реляционных данных по запросу RDF. Для поддержки шифрования алгоритм преобразования запросов модифицируется использованием операций объединения для кодирования запросов с неоднородным отображением на реляционную модель. Метод найдет применение для автоматизации обмена данными (с поддержкой шифрования), повышения эффективности поиска релевантной информации в Интернете и защиты информации.

Ключевые слова: обработка данных и знаний, семантический Вэб, отображения реляционных БД, модель RDF, поддержка шифрования, защита информации.

Введение

Для автоматизации управления обработки данных и знаний в Интернете с 1999г. разрабатывается семантическое Вэб-пространство – это надстройка над существующей «Всемирной паутиной», которая призвана сделать размещенную в ней информацию более понятной для компьютеров и интеллектуальных агентов [1]. Машинная обработка возможна в «семантической паутине», благодаря ее важнейшим характеристикам [1]: использованию унифицированных идентификаторов ресурсов (URI), семантических сетей и онтологий. Современные методы автоматической обработки данных в Интернете в большинстве случаев основаны на частотном и лексическом анализе текстового содержимого, которое предназначено для восприятия человеком. В семантическом Вэб-пространстве вместо этого используется стандарт RDF, описывающий семантические сети (графы), в которых узлы и дуги имеют URI [2].

RDF представляет собой способ описания данных в формате субъект–отношение–объект, в котором в качестве любого элемента этой тройки используются только идентификаторы ресурсов. Модель данных RDF опирается на следующие базовые понятия [2]: графовая структура данных, словарь идентификаторов URIref, типы данных, литералы, факты, правила логического следования. Однако большинство накопленной информации, хранимой в Интернете, представлено в реляционных БД, доступ к которым семантическими средствами затруднен.

Методика эксперимента

Для решения двух задач внедрения технологии RDF (интеграции с существующими реляционными БД и повышения производительности обработки данных) может быть

использована система хранения RDF-данных, сочетающая подходы на основе отображения реляционных схем данных на модель RDF, на основе таблицы триплетов «субъект–предикат–объект». Для ее построения необходимо разработать модель адаптации реляционных данных для отображения на структуру RDF и обеспечить обработку RDF-запросов на доступ и обновление реляционных данных.

Модель адаптации реляционных данных представим в виде двойки [3]: {M, N}, где M – отображения реляционной модели данных на модель RDF, позволяющего создавать утверждения RDF на основе значений полей записей реляционных таблиц, (реляционная таблица соответствует классу RDF-ресурсов, запись – RDF-ресурсу, значение первичного ключа – субъекту, имя поля – предикату, значение – объекту утверждения RDF); N – единое пространство имен (первичных ключей) для всех RDF-ресурсов, отображенных из записей реляционных таблиц, а также RDF-ресурсов, описываемых утверждениями, хранимыми в таблице триплетов, что позволит вносить в нее утверждения, использующие в позициях субъекта, предиката и объекта, любые RDF-ресурсы;

Обработку RDF-запросов определим тройкой {Aq, An, P}, где Aq – алгоритм преобразования запросов к данным RDF в запросы SQL, An – алгоритм обновления реляционных данных по запросу RDF, R – разбор и преобразование RDF-запросов и команд обновления данных в запросы и команды к реляционной СУБД на стандартном языке SQL.

По мере востребованности в конкретных приложениях от системы хранения RDF-данных также может потребоваться поддержка дополнительных возможностей. Набор алгоритмов, входящих в метод, обеспечивает поддержку следующих расширений:

– реификация (представление в виде самостоятельных ресурсов) утверждений RDF [4];

– применение правил логического вывода при преобразовании RDF запросов для учета в результатах выполнения запросов отношений подкласс-суперкласс, заданных предикатом *rdfs:sub Class Of* (так, вышеупомянутое создание единого пространства имен равносильно включению всех отображаемых классов ресурсов RDF на суперкласс, *rdfs: Resource;*

– применение правил логического вывода для учета подотношений, определенных при помощи предиката *rdfs: sub Property Of,* указывающего, что все утверждения, верные для подотношения, также верны и для базового отношения;

– применение правил логического вывода для учета транзитивных отношений, входящих в класс предикатов *owl'TransitiveProperty* (примером практического применения транзитивного отношения может быть выборка всех комментариев к заданному сообщению вне зависимости от уровня вложенности).

На рис. 1 представлена структура разработанного метода семантического доступа к данным на основе отображения реляционных БД на модель данных RDF. Суть метода заключается в интеграции: новой модели адаптации реляционных данных для отображения на модель данных RDF; процедур логического вывода на основе известных алгоритмов; обработке RDF-запросов на доступ и обновление данных [5]. Для реализации обработки разработаны новые алгоритмы преобразования запросов к данным RDF в запросы SQL и обновления реляционных данных по запросу RDF.

Следует отметить, что существующие системы хранения RDF-данных ограничиваются применением правил логического вывода на уровне приложения, что упрощает реализацию таких систем, но существенно снижает производительность обработки запросов. Например, выполнение запроса с учетом правил для подклассов и подотношений в подобной системе подразумевает перебор всех возможных комбинаций подклассов и подотношений, используемых в RDF-запросе, и выполнение отдельного запроса SQL для каждого варианта.

Разработанный метод семантического доступа, в отличие от аналогов, использует реализацию логического вывода на уровне хранимых процедур реляционной СУБД. Механизм хранимых процедур позволяет в процессе обновления данных создавать и поддерживать вспомогательные структуры, обеспечивающие выборку данных с учетом всех заданных правил логического вывода посредством одного запроса SQL. Наиболее известный пример такой структуры – транзитивное замыкание, сводящее проверку истинности транзитивного отношения до одной операции выборки.

Модель адаптации реляционных данных не накладывает дополнительных ограничений на используемую схему реляционной базы данных сверх ограничений стандарта SQL. Любая

таблица *T* в первой нормальной форме может быть отображена для доступа при помощи RDF-запросов. Таким образом, любая существующая база данных может быть адаптирована для доступа через RDF, не теряя при этом обратной совместимости с существующими SQL-запросами.

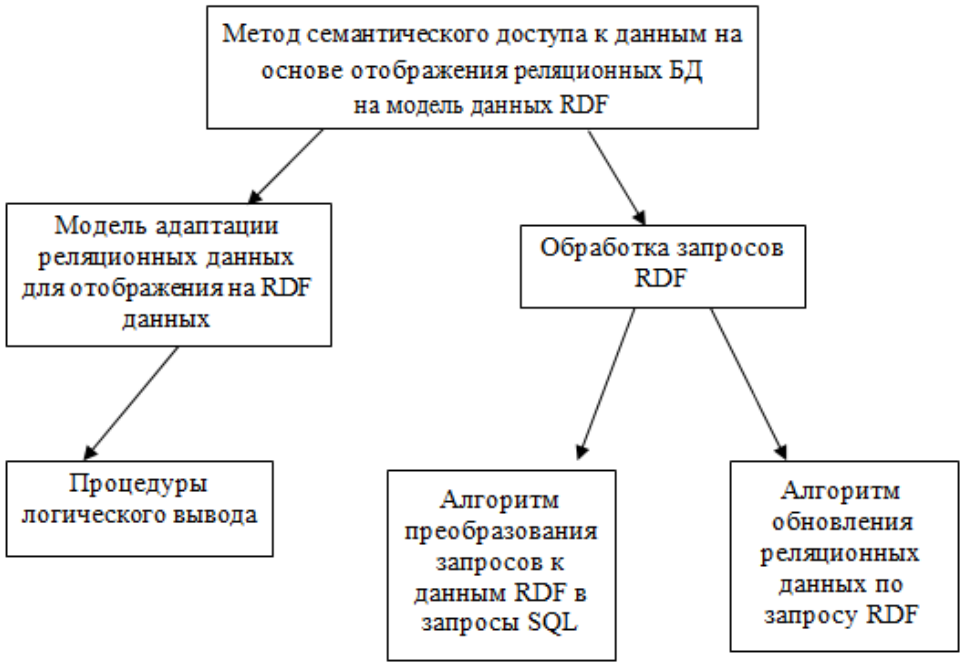

Рис. 1. Структура метода семантического доступа к данным на основе отображения РБД на модель RDF

Процесс адаптации включает добавление в базу данных атрибутов, внешних ключей, таблиц и хранимых процедур, необходимых для преобразования запросов RDF и поддержки дополнительных возможностей, предлагаемых разработанной системой, таких как реификация утверждений и логический вывод на правилах для *rdfs:sub Class Of rdfs: sub Property Of* и *owhTransitiveProperty* [6]. Разработанная модель может быть представлена в виде следующей последовательности шагов.

1. Ввести n множеств кортежей T_i, представляющих таблицы реляционной базы данных:

$$T_1 = \left\{ \left\langle a_{t_1 1}, ..., a_{t_1 m_1} \right\rangle \right\},$$
$$T_2 = \left\{ \left\langle a_{t_2 1}, ..., a_{t_2 m_2} \right\rangle \right\}, \quad (1)$$
$$T_n = \left\{ \left\langle a_{t_n 1}, ..., a_{t_n m_n} \right\rangle \right\}.$$

2. Для каждого реляционного атрибута $a_{t_i j}$ выбрать соответствующее RDF-отношение $p_k \in P$, где P – множество отношений RDF, построить отображение отношений RDF на реляционные таблицы и атрибуты:

$$M : \{p_k\} \to \left\{ \left\langle T_i, a_{t_i j} \right\rangle \right\}. \quad (2)$$

3. Создать единое пространство имен для ресурсов RDF, отображенных из записей таблиц, и ресурсов, описываемых в таблице триплетов.

3*а*. Создать таблицу ресурсов *R*, отображенную на суперкласс *rdfs .'Resource*, с автоматически генерируемым первичным ключом **id**(*R*), так что для любого определенного на *rdfs:Resource* RDF-отношении *рщ:*

$$M\,(p_{Rj}) = (R, a_{Rj}). \quad (3)$$

3*б*. Заменить первичные ключи id(T_i) таблиц T_i, отображенных на подклассы класса *rdfs .'Resource,* на внешние ключи, ссылающиеся на таблицу ресурсов *R*, так что:

$$\mathbf{id}(R) = U_{i=1}^{n} id(T_i), \quad \forall_i \neq j \mathbf{id}(T_i) \cap \mathbf{id}(T_i) = 0 \tag{4}$$

Обновить существующие внешние ключи с учетом замененных значений первичных ключей.

4. Зарегистрировать хранимые процедуры логического вывода на правилах для *rdfs:sub Class Of* для обновления таблицы ресурсов и поддержки целостности внешних ключей при выполнении операций над таблицами подклассов T_i.

5. Создать хранимые процедуры и вспомогательные структуры данных, необходимые для поддержки дополнительных возможностей алгоритма преобразования запросов.

5*а*. Зарегистрировать хранимые процедуры для прочих случаев логического вывода на правилах для *rdfs:sub Class Of.*

5*б*. Для представления RDF-данных, не отображенных на реляционную схему, и реификации утверждений RDF создать таблицу триплетов S, отображенную на *R* так, что:

$$id(S) \subset id(R) \tag{5}$$

5*в*. Для поддержки логического вывода на правилах для *rdfs: sub Property Of* добавить атрибуты a_s различения подотношений, ссылающиеся на записи в таблице ресурсов *R,* хранящие идентификаторы URIref соответствующих отношений, для каждого атрибута $a_p(T_i)$, отображенного на отношение, для которого определены подотношения:

$$a_s : \{\langle id(T_i), a_p \rangle\} \to P. \tag{6}$$

5*г*. Создать таблицы транзитивных замыканий $T^{+}_{a_t}$ и зарегистрировать хранимые процедуры логического вывода на правилах для *owl'TransitiveProperty* для каждого атрибута a_t(T_i), отображенного на транзитивное отношение p_t, такое, что

$$\langle a, p_t, b \rangle \wedge \langle b, p_t, c \rangle \Rightarrow \langle a, p_t, c \rangle. \tag{7}$$

Результаты и их обсуждение

На рис. 2 приведен пример схемы базы данных, полученной в результате применения всех вышеперечисленных изменений к схеме БД системы обмена сообщениями (Member, Message – отображенные на RDF таблицы *Ti,* Resource – таблица ресурсов; Statement – таблица триплетов; Part – таблица транзитивного замыкания для отношения *dct'AsPartOf* part_of_subproperty – атрибут различения подотношений отношения *dct:isPartOf*).

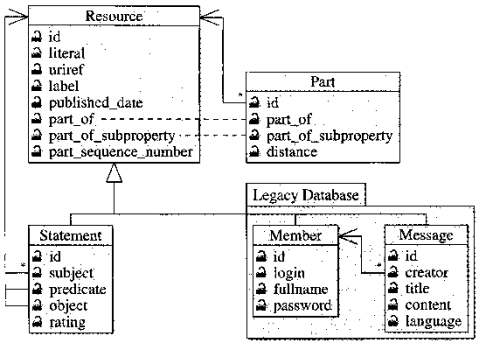

Рис. 2. Схема модели адаптации реляционных данных для отображения на модель данных RDF на примере системы открытой публикации Samizdat

Таблицы Member и Message – отображенные на RDF таблицы исходной реляционной схемы, соответствующие сущностям «пользователь» и «сообщение». Таблица Resource представляет суперкласс *rdfs .'Resource,* ее атрибуты literal, uriref и label используются в преобразовании запросов для различения типов ресурсов, а атрибуты publishecLdate и part_of отображаются на отношения *dcdate* и *dct:isPartOf*, действительные для всех ресурсов. В таблице Statement хранятся стандартные триплеты *(rdf.'subject, rdf'.predicate, rdf-.object),* расширенные специфичным для системы Samizdat атрибутом rating, содержащим определяемый пользователями рейтинг истинности реифицированных утверждений RDF. Таблица Part содержит транзитивное замыкание отношения *dctiisPartOf.* Поскольку для данного отношения определена возможность различения подотношений через атрибут part_of subproperty таблицы Resource, этот атрибут также отражен и в таблице Part; атрибут distance – стандартный атрибут, используемый при вычислении транзитивных замыканий.

Разработанная система хранения RDF-данных полагается на ручное отображение отношений RDF на таблицы и атрибуты. Конфигурация отображения загружается из файла в формате YAML, пример такой конфигурации для схемы БД, приведенной на рис. 2, представлен на рис. 3.

```
ns:
s : ‘http: //www.nongnu.org/samizdat/rdf/schema#’
rdf : ‘http: //www.w3.org/1999/02/22-rdf-syntax-ns#’
dc : ‘http: //purl.org/dc/elements/1.1/’
dct : ‘http: // purl. org/dc/terms/’
map:
' dc: :date': {Resource: published_date}
‘dct::isPartOf': {Resource: part_of}

- ‘rdf :: subject : {Statement: subject}
‘rdf:: predicate’ :{Statement:predicate}
‘rdf::object' : {Statement: object}
's::rating': {Statement: rating}
's::login': {Member: login}
's::fullName': {Member: full_name}
'dc::creator' : {Message: creator}
'dc:-.title': {Message: title}
's::content': {Message: content}
'dc::language': {Message: language}
```

Рис. 3. Элемент модели адаптации реляционных данных (отображение отношений RDF на реляционные таблицы и атрибуты)

В первой части конфигурации, выделенной ключевым словом **ns,** перечислены пространства имен, используемые далее для сокращенной записи идентификаторов URIref. В основной части конфигурации, выделенной ключевым словом **map,** дано отображение идентификаторов предикатов (сокращенных с помощью заданных выше пространств имен) на поля таблиц реляционной схемы БД.

Для поддержки шифрования требуется адаптировать модуль хранения RDF к более широкому спектру проблемных областей. В этом случае алгоритм преобразования запросов модифицируется возможностью использования операций объединения для кодирования запросов с неоднородным отображением на реляционную модель. Конфигурирование реляционной схемы приводится к общему виду, включая поддержку композитных ключей и более гибкие хранимые процедуры для логического вывода и реификации утверждений.

Заключение

В результате выполненной разработки получен новый метод (с поддержкой шифрования [7]) обработки данных и знаний в Интернете. Он может быть реализован программно, а для обслуживания большого количества пользователей (десятки, сотни тысяч) и аппаратно. Метод найдет применение для автоматизации обмена данными и знаниями (в том числе шифрованными) предприятия, университета [8] с другими учреждениями, повышения эффективности поиска релевантной информации в Интернете и защите информации.

METHOD OT DATA AND KNOWLEDGE PROCESSING IN INTERNET WITH CIPHERING SUPPORT

V.A. VISHNIAKOV, D.S. BORODAENKO

Abstract

The method for mapping relational DB to the RDF model for efficiently automation processing large volume of data and knowledge in Internet on semantic of RDF language is represented. The idea of this method in the integration: the adaptation model of relational data mapping to the RDF model, logical inference procedures; algorithms of queries translating on accesses to RDF into SQL queries and processing updates of relational data based on RDF queries. Algorithm of queries translation for the ciphering support is modifying with unity operations for queries coding with inhomogeneous mapping on relational model. This method can be used for data and knowledge automation exchange (with coding support), high efficiently of Internet relevant information search and information defense.

Список литературы

1. Berners-Lee T., Hendler J., Lassila O. // Scientific American. May 2001. P. 28–37.
2. Resource Description Framework (RDF): Concepts and Abstract Syntax [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/rdf-concepts/. – Дата доступа: 04.07.2012.
3. *Бородаенко Д.С.* // Докл. БГУИР. 2010. № 2 (48). С. 84–89.
4. RDF Semantics [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/rdf-mt/ – Дата доступа: 04.07.2012.
5. *Вишняков В.А., Бородаенко Д.С.* // Экономика и управление. 2011. № 2 (26). С. 79–83.
6. RDF Vocabulary Description Language 1.0: RDF Schema. [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/rdf-schema/.
7. *Вишняков В.А.* // Матер. междунар. научн. конф. «Информационные технологии и системы-2013»: материалы международной научной конференции, Минск, 18 октября 2013 г. С. 130–131.
8. *Бородаенко Д.С.* // Информатизация образования. 2010. № 4. С. 35–42.