

# СОВРЕМЕННЫЕ ПРОБЛЕМЫ ПЕРСОНИФИКАЦИИ И ЭКСТРАКЦИИ ДАННЫХ

Е. В. Артамонова

Институт систем информатики СО РАН,  
630090, Новосибирск, Россия

---

УДК 004.738.5:004.942

В настоящее время в Web размещено много данных, структура и семантика которых не формализованы. Статья описывает современные проблемы поиска и извлечения из Web неструктурированных данных, в том числе проблему генерации „информационного портрета“ на основе данных из различных источников данных, а также проблему персонификации данных. В статье кратко описаны также общие принципы построения Linked Data („связанных данных“) как наиболее перспективной основы для решения, Resource Description Framework (RDF) и современный уровень развития концепции „связанных данных“.

**Ключевые слова:** Resource Description Framework, RDF, Linked Data, персонификация данных.

There are much data in the Web, whose structure and semantics are not formalized. The article describes the modern problems of a search of not-structured data in the Web, including the issue of a generation of an 'information portrait' based on the data from different data sources and problems of the data personification. Also the article provides the general principles of the Linked Data conception as the most perspective basis of the solution, Resource Description Framework (RDF) and the modern level of the Linked Data evolution.

**Key words:** Resource Description Framework, RDF, Linked Data, data personification.

**Введение.** В настоящее время способы обработки и хранения данных во всемирной паутине достигли достаточно высокого уровня. Регулярно предпринимаются различные, иногда весьма успешные попытки упорядочивания/структуризации данных, появляются новые методологии, призванные облегчить человечеству комфортное взаимодействие с информационным пространством.

Появились Linked Data технологии, позволяющие связывать между собой неструктурированные данные, находящиеся в разных источниках данных, и по запросу формировать информационный портрет. При этом возможность размещения Linked Data в облачных вычислениях позволяет считать полностью решенной проблему размещения данных большого объема [1].

В настоящей работе описаны этапы развития технологии размещения данных во Всемирной паутине вплоть до настоящего времени и рассмотрен современный уровень развития концепции Linked Data, а также перспективы их дальнейшего развития и ряд прикладных задач по этой теме (см. 1, 2, 3, 4). Далее приведен обзор приложений, разработанных для Linked Data (см. 5), и известных Linked Data проектов (см. 6).

Отдельный интерес представляет проблема формирования информационного „портрета“ по запросу над Linked Data в аспекте персонификации данных (см. 6).

**1. Общие вопросы поиска и обработки информации в Web.** Появление Всемирной паутины (World Wide Web) в корне изменило способ, которым мы обмениваемся знаниями. Благодаря ей публикация данных в рамках глобального информационного пространства стала широко распространена и доступ к данным существенно упростился.

Природа Сети свободно расширяема и общедоступна, и свободный рост является ее ключевой особенностью. Благодаря этому современные поисковые системы индексируют документы в Сети и анализируют структуру связей между ними, после чего пользователь может свободно пересекать это информационное пространство с помощью Web-браузеров.

Высокого уровня достигло управление web-документами, но до сих пор принципы, аналогичные используемым для них, так и не были в полном объеме применены к данным. Данные в Интернете хранятся в форматах XML или CVS, либо представлены в HTML-таблицах.

Для обычной гипертекстовой сети форматом данных является HTML. Но следует учитывать, что природа HTML несовершенна: будучи размеченными в HTML таблицах, данные теряют большую часть своей структуры и семантики. Кроме того, HTML использует нетипизированные ссылки и не дает возможности соединять типизированными ссылками сущности, находящиеся в разных документах.

Но, так или иначе, эволюция в последние годы превратила Web из пространства связанных между собой документов в нечто новое — в мировую информационную сеть, где в равной мере связаны между собой документы и данные. Качественный скачок в развитии произошел благодаря появлению Linked Data (см. 3, 4, 5).

Далее, рассмотрим, например, процесс поиска информации в сети Интернет на любую заданную тему. Сформулировав запрос к поисковой системе, пользователь получает достаточно обширный список релевантных тем, но при этом он сталкивается с рядом трудноустраняемых на текущем этапе развития техники проблем.

Например, предоставленные ссылки могут открывать интернет-ресурсы, содержимое которых дублирует содержимое других ресурсов. Кроме того, выборка может содержать и совершенно не относящиеся к заданной теме данные, попавшие туда случайно (например, омонимы) [2].

Современный исследователь постоянно сталкивается с ситуацией, когда данные, найденные в разных источниках, могут быть представлены в разных форматах, что затрудняет их сопоставление.

Кроме того, данные могут и вовсе опровергать друг друга („будет дождь“ — „ожидается ясная погода“). При обработке таких данных их следует обрабатывать как дубликаты. При этом алгоритм должен однозначным образом определять, какие данные следует считать истинными, а какие ложными.

Далее, если данные не имеют схожей жесткой структуры, то они наверняка имеют и различный синтаксис, что затрудняет их обработку.

В общем, существует целый класс лингвистических и технических проблем, решение которых вывело бы интернет-технологии на принципиально новый уровень.

**2. Resource Description Framework как способ перемещения между объектами Web.** В последнее десятилетие активное развитие получили концепции Resource Description Framework и опирающихся на нее Linked Data (см. 3).

Среда описания ресурса (Resource Description Framework или RDF) — это (в классической терминологии) разработанная консорциумом Всемирной паутины модель для представления данных и в особенности — метаданных. RDF представляет утверждения о

ресурсах в виде, пригодном для машинной обработки. RDF является частью концепции семантической паутины.

Ресурсом в RDF может быть любая сущность — как информационная (изображение, веб-сайт и проч.), так и неинформационная (человек, город или некое абстрактное понятие).

Утверждение, высказываемое о ресурсе, называется триплетом. Триплет RDF имеет вид „субъект–предикат–объект“. Для обозначения субъектов, отношений и объектов в RDF используются URI. Множество RDF-утверждений образует ориентированный граф, в котором вершинами являются субъекты и объекты, а ребра отображают отношения.

Структура RDF модели предполагает использование наборов триплетов данных (объекта, субъекта и соответствующего предиката). Это позволяет строить сложные запросы и работать с данными из разных источников, в том числе, неструктурированными. Запрос от пользователя в некоем поле объектов можно представить как некую „точку интереса“ данного пользователя. В идеально простом случае этой „точкой интереса“ может быть однозначно определяемый объект. Например, пользователь подбирает информацию о городе Москва в РФ. Тогда в качестве ответа он получает уникальный объект со своим идентификатором, по которому через все доступные предикаты этого объекта можно получить информацию обо всех свойствах объекта без учета степени их вложенности.

Внешне такой граф можно представить в виде шара (или круга), где в центре расположен собственно объект, а по краям окружности расположены его субъекты. Ребрами данного графа будут связывающие их предикаты. Поскольку в данном случае мы не рассматриваем вложенность и цикличность объектов, высота этого графа равна 1.

В дальнейшем, если при построении графа используется алгоритм, учитывающий вложенность данных, а также решающий проблемы, описанные в 1, то в итоге полный портрет может иметь достаточно сложную структуру.

В случае если точка интереса не описана однозначным образом, то при генерации результата приходится принимать во внимание уже не точку, а некую область интереса.

Например, пользователь осуществляет поиск объектов, имеющих свойство „красный“. В этом случае поиск возвращает множество результатов, в качестве которых могут быть представлены как свойства (атрибуты) объекта, так и набор объектов.

Специфика RDF модели предоставляет возможность, публикуя данные в Web, легко включать в структуру логические взаимосвязи (там, где они есть), используя в описании сущностей ссылки на другие сущности (URI).

Для того чтобы источник данных стал частью глобальной сети данных, должны быть установлены RDF-ссылки на соответствующие объекты в других источниках данных. Поскольку каждый источник данных может содержать достаточно много ссылок на различные объекты, RDF-ссылки могут быть созданы автоматическим или полуавтоматическим способом.

Языком запросов к данным, представленным по модели RDF (и одновременно протоколом для передачи этих запросов и ответов на них) является **SPARQL** (SPARQL Protocol and RDF Query Language).

SPARQL является рекомендацией консорциума W3C и одной из технологий семантической паутины. Предоставление SPARQL-точек доступа (англ. SPARQL-endpoint) является рекомендованной практикой при публикации данных во Всемирной паутине. Неоспоримое достоинство SPARQL в том, что позволяет пользователям писать глобально однозначные запросы.

Существуют различные среды генерации RDF-ссылок. С помощью декларативных языков можно определить, какие именно RDF-ссылки должны быть созданы, какие применены матрицы подобия для сравнения объектов и каким образом учитываются конкретные свойства в рамках общей совокупности:

— **Silk** среда работает с локальными и удаленными SPARQL точками доступа и предназначена для использования в распределенных средах без необходимости реплицировать наборы данных локально [3];

— **LinQL** среда работает над реляционными базами данных и предназначена для использования вместе с базой данных на картографических инструментах RDF, таких как D2R Server или Virtuoso.

При генерации RDF ссылок бывает полезно использовать способы, упрощающие задачу. Например, в ряде областей действуют популярные соглашения о наименованиях. Например, идентификаторы ISIN относятся к финансовой сфере, числа ISBN и ISSN — к области публикации и т. д. Тогда, если оба набора данных, которые требуется соединить RDF-ссылкой, относятся к одной и той же схеме идентификации из числа общеизвестных, задача о генерации ссылки заметно упрощается. В этом случае неявные связи между наборами данных легко сделать явными — такими как RDF-ссылки. Именно такой подход используется для создания связи между различными источниками данных в облаке LOD.

Если же никаких общих схем присвоения имен не выявлено, то RDF-ссылки устанавливаются, опираясь на сходство объектов в каждом из наборов данных. Правда, в этом случае создание ссылок требует большого объема работ по обнаружению дубликатов и линкованию данных (data linking) в сообществе баз данных и сообществе представления знаний.

Например, чтобы установить RDF-связи между данными из двух наборов данных, содержащими сведения о музыкантах, можно использовать алгоритм, основанный на аналогии подобия, где используются метрики подобия для сравнения имен музыкантов и названий альбомов/песен [4].

RDF модель следует понимать не только как способ перемещения между объектами Web, но и как полезный инструмент в ряде современных исследований.

Здесь нелишним будет упомянуть и о ведении в настоящее время исследований принципиально иной направленности, использующих возможности и преимущества RDF в части моделирования и анализа систем [5].

Например, в ИСИ СО РАН накоплен положительный опыт применения RDF для разработки электронных архивов и исторической фактографии. В работе реализован новый подход, основанный на сквозной записи информации. Это позволило существенно сократить время поиска информации.

В качестве результата представлена новая система Polar, предназначенная для создания специализированных баз данных, а также систем управления базами данных. В основе системы лежит развитие теоретических разработок 1980-х гг.

Polar позволяет описывать информацию на основе концепции RDF. Также в работе представлены специализированные алгоритмы индексирования триплетов, которые позволяют строить „простые портреты“, причем быстрее, чем аналогичные „портреты“, созданные при использовании специализированной для работы с RDF системы Open Link Virtuoso.

**3. Linked Data и технические особенности их построения.** По завершении этапа первичного накопления данных во Всемирной паутине стало очевидным, что для продук-

тивной работы с этими данными необходимы современные методы публикации данных, поддерживающие их структурированность и обеспечивающие между ними взаимосвязь.

Для решения проблем, описанных в 1, была разработана концепция Linked Data [6].

Linked Data, согласно стандартному определению, есть совокупность коллекций взаимосвязанных наборов данных во Всемирной паутине, базирующихся на RDF модели представления данных (см. 2) с применением HTTP протокола.

В 2006 году Бернерсом-Ли были сформулированы правила публикации связанных данных в Интернете, соблюдение которых позволит любому публикуемому набору данных стать частью глобального инфопространства:

1. Используйте URI как имена для вещей: концепция Linked Data использует в качестве определения сущности его URI, причем любая сущность, включенная в Linked Data, может быть найдена соответствующим клиентским приложением.

2. Включайте ссылки на другие связанные URI.

3. Предоставляйте пользователям, просматривающим URI, полезную информацию, используя стандарты RDF, SPARQL.

Технически Linked Data можно воспринимать как использование Интернета для создания типизированных связей между данными, относящихся к самым различным источникам.

По аналогии с классическими методами интеграции, эти источники данных напоминают не имеющие никаких налаженных способов коммуникации разнородные (гетерогенные) информационные системы, представленные своими базами данных, взаимодействие между которыми на уровне данных затруднительно.

Linked Data должны быть машиночитаемы, т.е. значение таких данных должно быть определено в явном виде, что необходимо для связи с другими наборами данных.

Как сказано выше, Linked Data опираются на документы, содержащие данные в RDF формате, используемом для создания типизированных утверждений (предикатов) и, далее, связывания с их помощью любых пар объектов и субъектов.

В построении Linked Data применяются две основные технологии — URI (единые идентификаторы ресурсов) и HTTP как протокол передачи гипертекста.

URI, в отличие от традиционно применяемых для адресов объектов URL, предоставляют более универсальные средства для выявления любой сущности объекта.

Так как объекты идентифицируются через URI, основанных на HTTP:// схемах, для просмотра связанного объекта достаточно просто назвать URI по протоколу HTTP. Протокол HTTP предоставляет простой и универсальный способ извлечения ресурсов — они могут быть сериализованы в поток байтов или переданы в виде описания.

Web-данные, опираясь на общую структуру, имеют ряд свойств, аналогичных свойствам классических Web-документов. Они используют для доступа к данным HTTP-протокол, для идентификации ресурсов — HTTP URI, а также применяют RDF триплеты для описания ресурсов. Это позволяет объединять данные в глобальный граф, охватывающий по мере их появления все новые источники данных.

Такие данные могут быть любого типа, и опубликовать их может любой желающий. Благодаря этому технология Linked Data становится все более популярной, а ее принципы публикации применяются все большим числом владельцев данных. Таким образом, возникло то, что называют сетью данных — глобальное пространство данных, содержащее миллиарды утверждений (триплетов).

Далее представляется полезным вкратце остановиться на некоторых особенностях формирования Linked Data:

- словари для связывания данных;
- псевдонимы (*Aliases*) для отслеживания разных ресурсов, описывающих один и тот же объект;
- метаданные для повышения качества публикуемых данных с точки зрения пользователя;
- сериализация для передачи данных по сети.

*Словари.* Связывание объектов Linked Data осуществляется с помощью специализированных словарей, позволяющих описывать объекты и связи между ними. Такие словари являются сборниками классов и свойств. Они выражаются в RDF, но используют термины из OWL (Ontology Web Language) и RDFS (RDF Vocabulary Definition Language). Иначе говоря, соответствия между связанными словарями задаются, когда RDF триплеты соединяют классы и свойства одного словаря с классами и свойствами другого.

Каждый поставщик данных может при публикации использовать свой предпочтительный словарь, поскольку паутина данных полностью открыта для любых словарей, используемых параллельно. Несмотря на это, считается хорошим тоном не создавать каждый раз новые словари с новой терминологией данных, а по возможности повторно использовать термины из известных RDF словарей, таких как FOAF, SIOC, SKOS, DOAP, vCard, Dublin Core, OAI-ORE или GoodRelations. Это позволяет значительно упростить для клиентских приложений обработку связанных данных.

Если же ввод новой терминологии неизбежен, то следует делать ее самодокументированной (см. ниже). Это позволяет клиентам получить RDF схему и OWL определения терминов, а также сопоставления терминам в других словарях.

Таким образом, связи между наборами данных устанавливаются через использование общих и специальных словарей, содержащих термины, характерные только для одного источника данных, с соответствиями, установленными между ними.

*Псевдонимы (Aliases).* Разные поставщики информации часто вводят различные URI для обозначения одного и того же объекта, поскольку не знают друг о друге. Например, DBpedia использует URI <http://dbpedia.org/resource/Berlin>, чтобы определить Берлин, в то время как GEONAMES для идентификации Берлина использует <http://sws.geonames.org/2950159/> URI.

Если URI относятся к одному и тому же реальному объекту, их называют URI псевдонимами (*alias*). URI псевдонимы — общие для всей сети данных. URI псевдонимы разыменовываются к различным описаниям одного и того же реального объекта, обеспечивая тем самым многообразие взглядов на каждую сущность и играя важную социальную роль в сети данных.

Для контроля над тем, что разные поставщики информации описывают одну и ту же сущность, обычно используются owl:sameAs ссылки на все известные URI псевдонимы.

*Метаданные.* Информация о создателе данных, дата их создания и метод создания могут оказаться полезными для того, чтобы пользователи могли оценить качество и полезность опубликованных данных. Поэтому такую информацию следует публиковать вместе с данными в качестве метаданных различных типов. Кроме того, издатели данных при желании имеют возможность предоставить дополнительные технические метаданные об их наборе данных и его отношениях взаимосвязей с другими наборами данных.

Как правило, метаданные представляются в терминах Dublin Core [7] или Semantic Web Publishing словаря.

Для описания рабочих процессов преобразования данных используются термины из Open Provenance Model.

Издатели данных могут также настроить альтернативные способы доступа, помимо разыменовываемых URI [8] это могут быть SPARQL endpoint или RDF выводы.

Словарь взаимосвязанных наборов данных (Vocabulary Of InterLinked Datasets) задает термины и предоставляет наилучшие способы предоставления статистической метаинформации о наборах данных и наборах ссылок, соединяющих их.

*Сериализация.* Процессом сериализации называется перевод какой-либо структуры данных в последовательность битов.

Стандартным форматом сериализации Linked Data является RDF/XML [9, 10].

Кроме того, Linked Data можно сериализовать как RDFa, он позволяет встраивать RDF триплеты в HTML. При этом издателям данных RDFa следует использовать в качестве атрибута, чтобы назначить URI объектам и дать возможность другим поставщикам данных установить RDF ссылки на них.

А если необходим человеческий контроль над RDF данными, можно использовать альтернативные, более доступные для чтения варианты внутренних сериализаций — Notation3 или его подраздел Turtle [11].

**4. Публикование Linked Data.** Ввиду перспективности Linked Data как технологии, способной со временем расширить возможности работы с информацией, хранящейся в Интернете, в настоящее время активно ведутся разработка и внедрение средств для публикации Linked Data [12].

Сообразно перечисленным выше (см. 3) принципам Linked Data, публикование набора данных в Web в качестве Linked Data состоит из следующих этапов:

1. Для всех объектов, включенных в набор данных, следует назначить URI.
2. Для всех URI необходимо, соответственно, обеспечить разыменование в RDF представлении по протоколу HTTP.
3. Для других источников данных в Интернете следует установить RDF ссылки. Это позволит перемещаться между наборами данных.
4. Для опубликованных данных следует обеспечить метаданные, что позволит оценить качество этих данных и выбирать между различными видами доступа.

Существующие в настоящее время средства публикования Linked Data либо обслуживают содержимое RDF хранилищ (например, Linked Data on the Web), либо обеспечивают просмотры источников данных, которые даже не являются официально созданными RDF источниками.

С помощью этих программных средств владельцы данных могут не вдаваться в технические подробности, например, о том, полностью ли соблюдены правила публикования Linked Data и является ли согласованным содержимое источника.

Все инструменты поддерживают разыменования URI в описаниях RDF.

Также некоторые инструменты нужны для обеспечения SPARQL запроса доступа к обслуживаемым наборам данных и публикования RDF выводов.

*D2R server* [13] является инструментом для публикации не-RDF реляционных баз данных в виде Linked Data. Используя декларативный язык, издатель данных определяет соответствие между реляционной схемой базы данных и целевым RDF словарем. На ос-

новании этого соответствия сервер D2R публикует выходы Linked Data с базой данных и позволяет клиентам обращаться к базе данных через SPARQL протокол.

*Virtuoso Universal Server.* The OpenLink Virtuoso server обеспечивает обслуживание RDF данных через Linked Data интерфейс данных и endpoint SPARQL.

Данные RDF могут либо храниться непосредственно в Virtuoso, либо могут быть непосредственно созданы из не-RDF реляционных баз данных на основе отображения (mapping).

*Talis Platform* представляет собой доступную через HTTP услугу. Talis Platform предоставляет собственную (встроенную) память для RDF / Linked Data. При наличии прав доступа содержимое каждого хранилища Talis платформы доступно через endpoint SPARQL или серии REST API, которые, в свою очередь, придерживаются важнейших принципов Linked Data.

*Pubby* сервер можно использовать в качестве расширения к любому RDF-хранилищу, поддерживающему SPARQL. Pubby преобразует URI запросы в SPARQL DESCRIBE запросы, лежащие в основе RDF-хранилища. Кроме RDF, Pubby также дает возможность быстрого просмотра HTML хранилища данных, принимает на себя обработку 303 URI редиректа, а также обеспечивает согласование содержания между двумя представлениями.

*Triplify* инструментарий используется разработчиками для расширения существующих веб-приложений над Linked Data с пользовательским интерфейсом. Используя шаблоны SQL запросов, Triplify обслуживает Linked Data и JSON просмотр над базой данных приложения.

*SparqPlug* является сервисом, который извлекает связанные данные в Интернете из старых HTML документов, которые не содержат данных RDF. Служба работает, опираясь на сериализацию в HTML DOM как RDF, и позволяет пользователям определять SPARQL запросы, которые преобразуют его элементы в RDF граф по своему выбору.

*OAI2LOD server* является оберткой Linked Data для серверов документов, которые поддерживают Open Archives OAI-PMH протокол.

*SIOC экспортеры.* В рамках проекта SIOC были разработаны обертки для Linked Data для нескольких популярных движков блогов, систем управления контентом и дискуссионных форумов, таких как WordPress, Drupal и PHPBB.

*Vapour.* Сервис, позволяющий издателям отлаживать их Linked Data сайт — это Vapour validation service. Vapour проверяет, что опубликованные данные соответствуют важнейшим принципам Linked Data, а также лучшим практикам сообщества.

**5. Обзор Linked Data приложений.** В то время как гибридные приложения взаимодействуют с заранее определенным набором данных, Linked Data работают с несвязанным информационным пространством. Это означает, что по мере появления в Интернете новых источников данных результаты запросов над Linked Data будут расширяться.

Соответственно, ввиду неуклонного роста популярности Linked Data, многие разработчики предпринимают усилия по созданию и совершенствованию приложений, использующих эту сеть данных. В целом все приложения этого направления можно разбить на 3 категории: поисковые движки Linked Data, браузеры Linked Data и проблемно-ориентированные Linked Data приложения.

*Специфика разработки приложений.* При разработке Linked Data приложений следует принимать во внимание ряд особенностей:

1. Web-данные являются самодокументируемыми. Если приложение встречает Linked Data, описываемые посредством незнакомого словаря, оно должно разыменовывать URI, чтобы найти их определение.

2. Данные разделяются на основе их форматирования и презентационных свойств.

3. По сравнению с WebAPI, опирающимися на гетерогенные модели данных и интерфейс доступа, использование HTTP как стандартизированного механизма доступа к данным вкупе с RDF как стандартизированной модели данных, значительно упрощает доступ к данным.

4. Приложения не должны быть реализованы на стандартном определенном наборе данных, новые данные могут быть добавлены во время выполнения запроса по RDF ссылкам.

Существует возможность выбора между двумя шаблонами идентификации сущностей — 303 URI или hash URI. Любой из них позволяет отличать сущности реального мира от документов, описывающих эти сущности.

*Browsers.* Существуют Linked Data приложения, с помощью которых можно просматривать данные в одном источнике данных, после чего по ссылкам переместиться в другие, связанные источники.

Иначе говоря, так же, как традиционные веб-браузеры позволяют пользователям перемещаться между HTML страницами, следуя по гиперссылкам, браузеры Linked Data дают возможность перемещаться между источниками данных по ссылкам, представленным триплетами RDF. При этом необходимо, чтобы данные были представлены в доступной для пользователя форме. Человеку приходится не только перемещаться по ссылкам, но также и анализировать представленную перед ним массу данных.

Например, приложение *Tabulator* (разработка 2006–2008 годов), перемещая пользователя в пространстве Web of data, использует „режим портрета“ (или, иначе, „режим эскиза“), то есть отображает для него при этом только часть имеющихся данных [14]. При этом отслеживается продвижение пользователя по ссылкам и формируется соответствующий личности именно этого пользователя „шаблон интереса“, после чего можно вызвать любое количество похожих шаблонов. В итоге из результатов запроса формируется таблица, содержимое которой можно проанализировать и отобразить традиционными способами представления данных (в том числе стандартными браузерами). Таким образом, *Tabulator* способен объединять данные из разных источников.

Примерно по этой же схеме действует ряд других браузеров (таких как *Marbles* и проч.)

В то же время ряд других авторов сомневается в том, что граф-ориентированный вывод RDF-данных целесообразен, что, возможно, со временем выльется в развитие альтернативных концепций.

*Поисковые устройства и индексирование данных.* Традиционно поисковые устройства — это сервисы или приложения, с которых начинается навигационный процесс, инициированный в браузерах.

Для работы с Linked Data, ввиду их специфики, разработаны и используются специальные поисковые устройства. Принцип их действия в целом напоминает получение данных по запросу из локальной базы данных.

Благодаря этим поисковым Linked Data „движкам“, Linked Data, следуя RDF ссылкам, широко распространились по Интернету. Это позволяет, в числе прочего, формировать и запросы над агрегированными данными.

Вообще говоря, эти поисковые устройства могут быть разделены на две категории:

1. Поисковые системы, ориентированные на человека (напр. *Falcons* и *SWSE*). Пользователю предоставляются интерфейсные возможности, приближенные к традиционным. Имеется поле поиска, в котором он вводит ключевые слова, далее приложение возвращает список результатов, которые могут иметь отношение к запросу. При этом, помимо простого списка, пользователю доступен более детальный интерфейс с краткой информацией о сущности каждого объекта, и здесь уже используется структура Linked Data. *Falcons* позволяет искать документы (поиск RDF документов, содержащих запрашиваемое слово — вариант, близкий к традиционной практике), концепты (поиск свойств и классов в опубликованных web онтологиях), объекты (конкретные объекты, например, персоналии или достопримечательности), и каждый вид поиска имеет свое представление данных результата. Существенным будет отметить, что, даже относясь к разным объектам, документы Web и данные Web формируют одно управляемое информационное пространство. Также пользователь может переходить по ссылкам из HTML документа к документу в Web of Data и назад.

2. Ориентированные на приложение индексы (напр. *Swoogle*, *Sindice* и *Watson*). Группа сервисов, разработанная для удовлетворения потребностей приложений над Linked Data. Они предоставляют API, с помощью которых приложения могут разыскивать RDF-документы на Web, которые ссылаются на соответствующие URI или содержат ключевые слова. Благодаря этим сервисам, каждому новому приложению нет нужды сканировать и индексировать все части Web of data, которые могут понадобиться. Приложению достаточно просто запросить эти индексы и получить указатели на потенциально подходящие документы, после чего приложение может их извлечь и должным образом обработать. *Sindice* ориентирован на поиск документов, а *Swoogle* и *Watson* — на поиск онтологий (концепций).

*Проблемно-ориентированные приложения.* Помимо браузеров и поисковых движков, существует еще и определенное количество специализированных web-сервисов.

Такие сервисы классифицируются как проблемно-ориентированные приложения. В качестве наиболее популярных и развитых можно перечислить следующие:

1. *Revuu* — web-сайт ([revuu.com](http://revuu.com)), который предоставляет возможность просмотра данных и формирует рейтинг этих данных, опираясь на принципы Linked Data и Semantic Web стековой технологии. *Revuu* публикует Linked Data в процессе обработки, а также использует и другие Linked Data из других источников. Например, при просмотре и анализе фильма *Revuu* сайт пытается сопоставить его с соответствующей записью в *DBpedia*. Если такое совпадение установлено, то дополнительная информация о фильме извлекается из *DBpedia* и отображается в HTML-формате на страницах сайта (т. е. в легкодоступной для человеческого восприятия форме). Ссылки при этом формируются на уровне RDF, благодаря чему пользователь видит информацию, скомпонованную из разных источников, и может обратиться к связанным HTML страницам.

2. *DBpedia Mobile* — браузер для запуска на iPhone или аналогичном мобильном устройстве; ориентирован на туристов. Исходя из данных GPS о текущем местоположении мобильного устройства, приложение формирует подборку о близлежащих достопримечательностях, фото и ревью.

3. *Talis Aspire* представляет собой веб-приложение, ориентированное на управление списками ресурсов и разработанное для преподавателей вузов и студентов. Пользователи могут создавать списки через обычный веб-интерфейс, далее приложение создает RDF-триплеты, которые сохраняются в хранилище Linked Data. Таким образом, опираясь на

принципы Linked Data, пункты, представленные в одном списке, оказываются связанными с соответствующими элементами, размещенными в списках в других учебных заведениях. Тем самым через действия пользователей, не являющихся специалистами в аспектах Linked Data, создается паутина научных данных.

4. BBC Programmes and Music — приложение, разработанное для нужд Британской вещательной корпорации (BBC). Использует Linked Data для внутренних нужд как легкую технологию интеграции данных. BBC запускает многочисленные радиостанции и телевизионные каналы, каждый из которых использует свою систему управления контентом, что изначально создает определенные трудности. BBC начала использовать Linked Data технологию вместе с DBpedia и MusicBrainz, используя их как контролируемые словари для подключения контента на нужную тему. При этом разные части контента размещены в разных репозиториях, и его общий объем увеличивается за счет использования дополнительных данных из облака Linking Open Data. На основании этих связей BBC Programmes и BBC Music строят Linked Data сайты по всей имеющейся музыке и программам, связанным брендами.

5. DERI Pipes. Разработан на базе Yahoo Pipes. DERI Pipes формирует „портреты“ („коллажи“) на уровне платформы данных, совместно включая несколько источников данных для формирования нового канала. В результате агрегации возникают сложные операции, такие как консолидация идентификатора, отображение схемы, RDFS или OWL, с преобразованием данных через SPARQL CONSTRUCT операций или шаблоны XSLT [15].

Таким образом, большая часть разработанных сегодня проблемно-ориентированных приложений не затрагивает глобальной проблематики и предназначена для „притирки“ данных, полученных из разных Linked Data источников.

**6. Известные Linked Data проекты.** Дальнейшее развитие концепции Linked Data привело к появлению некоторого числа достаточно успешных проектов, ставящих своей целью выборку структурированных данных из Web of data по запросу пользователя (см. Linking Open Data и DBPedia).

*Linking Open Data.* Наиболее наглядным примером принятия и применения важнейших принципов Linked Data стал проект Linking Open Data (LOD) [16].

Целью проекта LOD является заполнение Web of data следующим образом:

- обнаружение подходящих наборов данных с открытыми лицензиями;
- конвертация их в RDF формат с соблюдением правил Linked Data;
- опубликование их в Интернете.

Активное разрастание проекта объясняется не только интересными перспективами его развития, но и его общедоступностью: практически любой желающий может опубликовать свой набор данных и связать его с подходящими данными из других наборов. LOD размещены в облачных вычислениях, поэтому проблема размещения большого объема данных считается априори решенной.

Структуру LOD легко проиллюстрировать в виде связанной диаграммы, где каждый узел является набором данных. Чем больше связей между двумя наборами данных, тем более широкой линией отображается соединяющая их дуга.

Часть узлов являются связующими. Например, узел GEONAMES объединяет RDF описания огромного количества географических объектов по всему миру, благодаря чему он постепенно развился в центр, к которому подключено большое количество других наборов данных.

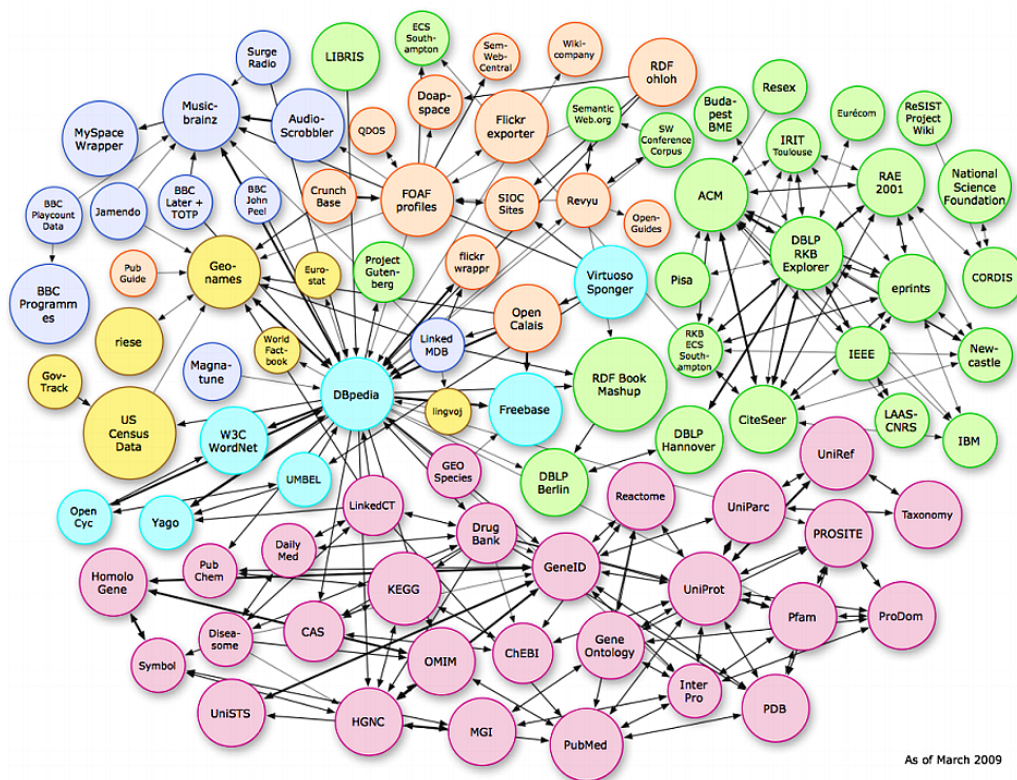


Рис. 1. Linked Open Data — диаграмма связей между наборами данных, размещено в „облачных“ вычислениях

LOD, в числе прочего, позволяет работать с реляционными БД и API. Для того чтобы использовать их в качестве наборов данных, необходимо сначала сгенерировать вокруг каждого из таких объектов некую обертку, что позволяет включить их в LOD. Соответственно, каждый такой объект необходимо сначала просмотреть, а после проанализировать. Это обстоятельство делает затруднительным определение точного размера сети данных.

*DBPedia*. В качестве примера успешного проекта, предназначенного для извлечения структурированных данных из существующей базы данных, рассмотрим DBPedia [17].

DBPedia создавалась с целью получения в структурированном виде данных, хранящихся в базе проекта „Википедия“ (Wikipedia) [18]. DBPedia данных состоит из RDF триплетов, извлеченных из „infoboxes“, часто видных на правой стороне статей „Википедии“.

Успешность проекта DBPedia легко проиллюстрировать статистически. По данным на сентябрь 2014 года базы данных проекта содержат

- описания более чем 4,58 млн понятий (из которых 4,22 млн классифицированы в соответствии с онтологией);
- 38 млн меток и аннотаций на 125 языках;
- 25,2 млн ссылок на изображения;
- 29,8 млн ссылок на внешние веб-страницы;
- 50 млн внешних ссылок на другие базы данных RDF-формата;
- 80,9 млн категорий „Википедии“;

— 3 млрд RDF-триплетов (из них 580 млн были взяты из английского раздела „Википедии“ и 2,46 млрд извлечены из разделов на других языках).

Существует специфическая проблема обработки синонимов при извлечении информации из „Википедии“. Например, понятие „место рождения“ может быть сформулировано в английском языке как „birthplace“ и как „placeofbirth“. Соответственно, поскольку любые понятия могут быть выражены в шаблонах разными способами, SPARQL запрос проходит по обоим вариантам для получения более достоверного результата.

Например, запрос для получения данных об объекте „Alexander Pushkin“ выглядит следующим образом:

```
PREFIX foaf:    <http://xmlns.com/foaf/0.1/>
SELECT ?mbox
WHERE
  { ?x foaf:name "Alexander_Pushkin" .
    foaf:mbox ?mbox }
```

Для облегчения поиска при сокращении количества синонимов был разработан специальный язык — DBpedia Mapping Language, а у пользователей DBpedia появилась возможность повышать качество извлечения данных с помощью сервиса Mapping.

**7. Проблема формирования полного информационного „портрета“ по запросу над LOD. Персонафикация данных.** Рассмотрим проблему получения данных по запросу над Linked Data.

Несмотря на то, что объем данных, представленных как Linked Data, значителен, и на то, что язык запросов над Linked Data SPARQL позволяет формулировать глобально однозначные запросы, а также на многие другие аспекты, проблема полноты предлагаемых по запросу данных стоит достаточно остро.

Рассматриваемую проблему можно разбить на 2 основных задачи — получение интересующих нас сторонних данных и экстракция данных, то есть формирование из полученного набора данных некоего логически связанного резюме (краткого конспекта).

Можно осуществлять сбор данных различными способами, но логически напрашивается, что в качестве собственно сторонних данных следует использовать LOD. Тогда управление им будет включать извлечение этих данных, верификацию на выявление дубликатов и противоречащих друг другу данных, удаление противоречивых данных из сформированного по нашему запросу набора данных и управление связями между отдельными частями этих данных.

Извлечение может быть обеспечено, например, DBPedia. Тут возникает ряд проблем, рассмотренных ниже.

*Проблема верификации данных.* Когда в рассматриваемой базе содержится 100 000 триплетов, все их можно проверить за обозримое время. Кроме того, мы условно можем полагать их статическими. В случае, когда количество триплетов превосходит 50 млрд, проверить их простым перебором уже не представляется возможным. Это подводит нас к необходимости разработать быстрый и надежный способ верификации данных.

*Проблема полноты набора данных.* Отдельную сложность представляет добавление в набор данных тех данных, для которых в рассматриваемом информационном поле (например, в DBPedia) отсутствует триплет.

*Проблема формирования запроса и информационного „портрета“.* Когда исходных данных много, возникает проблема, в каких терминах сформулировать запрос и как орга-

низовать выборку именно тех данных, которые нужны в данном случае (информационный „портрет“).

Интуитивно понятно, что точность формулирования запроса определяет уровень соответствия между ожиданиями пользователя и полученными в результате данными. Соответственно, чем точнее запрос, тем полезнее должны быть получаемые данные.

Если запрос достаточно простой, и мы точно знаем и можем описать нашу „точку интереса“, то и выборка данных может быть представлена либо одним триплетом, либо несколькими, содержащими похожую информацию. Тогда информационный портрет можно графически представить в виде некоего шара, в центре которого размещена наша „точка интереса“, а вокруг — некая дополнительная информация о ней, которая связана с ней одним предикатом.

*Персонификация.* В широком значении данного слова „персонификация“ есть присваивание некой абстракции человеческих или личностных качеств. В рассматриваемом нами случае „персонификация“ есть взаимосвязь между особенностями личности, формулирующей запрос, и генерацией результата (подбором данных).

Наиболее наглядно персонификация данных, полученных из Сети, может быть реализована на примере авторизованного пользователя, изначально заполнившего личную анкету. Например, если известно, что запрос сформирован лицом мужского пола возрастом 30 лет, неженатым, то его информационный портрет, составленный по запросу „Чехия“, предположительно может включать данные о субъекте „хоккей“ и не включать данные о субъекте „шоппинг“. Если атрибуты пользователя будут другими, то и, соответственно, информационный портрет должен быть составлен иначе.

Представляется интересной задача, позволяющая решить проблему формирования полного информационного портрета по запросу, составленному пользователем, с учетом условий, накладываемых персонификацией.

**Заключение.** В связи с активным развитием Web в последнее время перед научным сообществом встал ряд вопросов оптимальной организации информационного пространства.

В настоящий момент формирующийся по запросу в Web список источников данных имеет ряд несовершенств: не учитывает особенности личности, сформировавшей запрос, не позволяет отделить „нужные“ данные от „ненужных“, не позволяет избежать дублирования результатов и, в большинстве случаев, не формирует на основе экстрагированных из разных источников данных единый готовый документ.

Появившиеся сравнительно недавно Linked Data — это, в определенном смысле, наиболее передовые на сегодняшний день способы размещения и подключения структурированных данных в Web, позволяющие связывать данные аналогично тому, как классический HTML позволяет связывать документы.

Эволюция концепции Linked Data породила ряд приложений принципиально нового типа, совместное использование которых позволяет формировать выборку Linked Data по запросу пользователя.

В перспективе представляется интересным рассмотреть проблему формирования полного информационного Linked Data „портрета“ с точки зрения возможности персонификации запрашиваемых данных.

## Список литературы

1. Платонов Ю. Г., Артамонова Е. В. Метод Business Community и „облачные“ вычисления (Cloud computing) // *Фундаментальные исследования*. 2013. № 4. Ч. 5. С. 1089–1093. [Электрон. рес.]. [http://www.rae.ru/fs/?section=content&op=show\\_article&article\\_id=10000577](http://www.rae.ru/fs/?section=content&op=show_article&article_id=10000577) (дата обращения: 02.04.2015).
2. Батура Т. В., Мурзин Ф. А. *Машинно-ориентированные логистические методы отображения семантики текста на естественном языке*. Новосибирск: Изд. НГТУ, 2008.
3. Volz J., Bizer C., Gaedke M., Kobilarov G. *SILK. A Link Discovery Framework for the Web of Data* [Electron. res.]. [http://events.linkedata.org/ldow2009/papers/ldow2009\\_paper13.pdf](http://events.linkedata.org/ldow2009/papers/ldow2009_paper13.pdf) (дата обращения: 02.04.2015).
4. Raimond Y., Sutton C., Sandler M. *Automatic Interlinking of Music Datasets on the Semantic Web / Proc. of the Linked Data on the Web Workshop (LDOW2008), Beijing, China, April 2008*. [Electron. res.]. <http://events.linkedata.org/ldow2008/papers/18-raimond-sutton-automatic-interlinking.pdf> (дата обращения: 01.04.2015).
5. Марчук А. Г. *PolarDB — система создания специализированных NoSQL баз данных и СУБД // Моделирование и анализ информационных систем*. 2014. Т. 21. № 6. С. 169–175.
6. Berners-Lee T., Bizer C., Heath T. *Linked Data — The Story So Far // Integrated Computer-Aided Engineering*. New York. 2012. N 19 (1). P. 93–109.
7. Carroll J. *Dublin Core, the Primer and the Model Theory* [Electron. res.]. <http://lists.w3.org/Archives/Public/w3c-rdfcore-wg/2002May/0040.html> (дата обращения: 02.04.2015).
8. Cyganiak R., Bizer C. *PUBBY. A Linked Data Frontend for SPARQL Endpoints*. [Electron. res.]. <http://www4.wiwiss.fu-berlin.de/pubby/> (дата обращения: 02.04.2015).
9. Beckett D. *RDF/XML Syntax Specification (Revised) — W3C Recommendation 10 February 2004*. [Electron. res.]. <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/> (дата обращения: 01.04.2015).
10. Gandon F., Schreiber G. *RDF 1.1 XML Syntax — W3C Recommendation 25 February 2014*. [Electron. res.]. <http://www.w3.org/TR/rdf-syntax-grammar/> (дата обращения: 01.04.2015).
11. Berners-Lee T. *Notation 3 Resources*. [Electron. res.]. <http://www.w3.org/DesignIssues/N3Resources> (дата обращения: 01.04.2015).
12. Bizer C., Cyganiak R., Heath T. *How to publish Linked Data on the Web*. [Electron. res.]. <http://www4.wiwiss.fu-berlin.de/bizer/pub/LinkedDataTutorial/> (дата обращения: 02.04.2015).
13. Bizer C., Cyganiak R. *D2R Server — Publishing Relational Databases on the Semantic Web / Poster at the 5th International Semantic Web Conf. (ISWC2006), 2006*.
14. *Decentralized Information Group. How to use the Tabulator*. [Electron. res.]. <http://dig.csail.mit.edu/2005/ajar/ajaw/Help.html> (дата обращения: 03.04.2015).
15. Тидвэлл Д. *XSLT. 2nd Edition*. СПб: Символ-Плюс, 2009.
16. *The Linking Open Data cloud diagram*. [Electron. res.]. <http://http://lod-cloud.net> (дата обращения: 12.04.2015).
17. *DBPedia*. [Electron. res.]. <http://dbpedia.org/> (дата обращения: 12.04.2015).
18. *Wikipedia*. [Electron. res.]. <https://ru.wikipedia.org/> (дата обращения: 12.04.2015).

*Артамонова Елена Валерьевна — аспирант  
Института систем информатики СО РАН,  
e-mail: artamonova.elena.v@gmail.com*

*Дата поступления — 06.05.2015*