关联数据的消费技术及实现*

□夏翠娟 刘炜

摘要 从关联数据技术实现的角度看，发布和消费是构建关联数据平台和实施关联数据应用应该考虑到的两个重要方面。关联数据的消费技术涉及到数据的访问、获取、发现、查询、交换、传输、处理和利用。而消费的方式和效果取决于关联数据源所提供的消费接口。通过梳理和总结关联数据消费技术的各个方面，调研 DBPedia、Freebase、VIAF 等大型的关联数据集和＜sameAs.org＞、SWSE、Sindice、Swoogle、Falcons 等流行的语义搜索引擎所提供的数据消费接口，总结目前关联数据消费接口的类型以及相应的使用方式，分析其适用范围、优缺点等，以期为关联数据发布方和消费方提供技术参考。

关键词 语义网 关联数据 消费技术

1 概述

Web 上关联数据的发布正呈井喷之势，截至目前，最新版的 DBPedia3.8 已发布了 377 万件事物(Thing)的数据，其中 235 万基于知识本体组织，包括个人(Person)、地点(Place)、机构组织(Organization)、物种和疾病等实体(Entity)，所使用的知识本体包含 359 个类、800 个对象属性、859 个数据类型属性、2347 种本体映射，提供 111 种语言的本地化版本。Freebase 则有 2300 余万实体，而语义搜索引擎 sameAs.org 则抓取了 4300 万实体，LOD 项目中著名的关联数据云已经包含了 328 个关联数据集(Linked Data Sets)，涵盖了生物医学、政府、图书馆、教育机构等多个领域。关联数据已成为整个全球性数据空间不可或缺的一部分，更重要的是，关联数据作为一种分布式环境下，基于语义关系的信息资源集成方式，使得 Web 上分布着饱含语义的、海量的、相互关联的数据，这些数据如果得到充分利用，会产生难以估量的巨大价值。

如何利用 Web 上的关联数据，需要对关联数据的消费(Consuming)方式和技术有深入的了解。发布和消费是关联数据相关技术的两个方面，发布的目的即是为了消费，有的是在内部消费，用于资源管理、整序、发现和重用等，典型的例子如 BBC 的关联数据应用[1]；有的是开放给整个互联网进行消费，如 DBPedia、Freebase 等大型公共关联数据集；有的同时兼具消费者和提供者双重角色，如各种语义搜索引擎。关联数据的消费技术主要涉及关联数据的访问、获取、发现、查询、交换、传输、处理和利用等消费过程中所相关的各类实现方式、技术标准及工具平台。关联数据的消费方式与数据源所提供的消费接口密切相关，目前关联数据源大致可分为两种：语义搜索引擎和关联数据集。本文总结了关联数据的消费技术及相关标准，对上述两种关联数据源所提供的消费接口进行了调研，通过几个案例来说明关联数据消费的实现途径，既考虑到关联数据消费方的技术需求，也为关联数据的发布提供参考方案。

2 研究与应用现状

对于关联数据消费方式和技术的研究与应用主要是国外，国内更多的关注还在关联数据的发布层面，在消费层面的关注较少，黄永文的《关联数据驱动的 Web 应用研究》一文分析了国内外基于关联数据的 7 种 Web 应用类型，最后从用户界面和交互方面、关联关系的有效性、数据融合和模式映射、关

* 本文系国家社科基金项目“关联数据的理论和应用研究(Research on the Linked Data: the Principles and Its Application)”(编号：11BTQ041)的研究成果之一。

联开放数据的许可 4 个方面讨论了关联数据应用面临的挑战[2]。白海燕的《关联数据与 DBPedia 实例分析》一文中以 DBPedia 为例总结出关联数据的三种 Web 获取方式和建立自动关联的两类常见算法[3]。另外在 2012 年 7 月上海图书馆举办的"从文献编目到知识编码:关联数据技术与应用"专题研讨班上,林海青的《关联书目数据:发布、查询、消费及混搭》,对关联数据消费技术如查询和混搭做了普及性的报告[4],夏翠娟的报告《关联数据的实现技术及案例》对关联数据消费的流程、技术、应用体系架构及相关工具平台做了梳理和分析[5]。虽然关联数据的消费在国内的研究层面引起了注意,但进行深入研究与利用的还很少见,较为有影响力的是南京大学 Websoft 研究团队开发的语义搜索引擎 Falcons,可以查询对象、概念、知识本体和文档。

在国外,从研究方面来看,2009 年 Tom Heath 在其 *Linked Data: Evolving the Web into a Global Data Space* 一书中,系统地论述了关联数据"消费"的概念、消费方式、构建关联数据应用的技术和原则、在关联数据消费过程中应解决的问题如数据质量等,书中用单独一章来谈关联数据的消费,包括目前关联数据消费的现状、关联数据应用的体系架构及其不同的模式、关联数据混搭应用的开发、数据发布者和数据消费者以及第三方应该如何联合起来以促进"数据的网络(Web of Data)"的一致性和整体性。在该书中,将关联数据的消费方式分为通用的和领域的两种,通用的消费方式包括浏览和检索,即基于如 Disco、Tabulator、Marbles 等语义浏览器的浏览,和基于 SWSE、Swoogle[6,7]、Falcons、sig. ma 等语义搜索引擎的检索[8]。使用语义浏览器的关联数据消费者一般是人而非机器,而语义搜索引擎则可同时对人和机器服务,大部分的语义搜索引擎都提供面向人的界面和面向机器的数据消费接口。领域的则根据不同的领域数据源所提供的消费方式而多种多样,取决于数据源(包括关联数据集和语义搜索引擎)提供何种消费接口。另一方面,自 2010 年起,关联数据的消费(Consuming Linked Data,CoLD)作为国际语义网大会(ISWC)的一个专题会议,已连续举行了两届,第三届将在今年继续举行,在 CoLD 专题会上,与会者主要展示一些消费工具和平台的原理、功能及使用方法,如 Christian Bizer 等的用于不同词表映射的 R2R 框架、用于关联关系发现的 SILK 框架、用于关联数据整合的 LDIF 框架等;还有应用案例的演示、经验的分享,如探讨联合消费多个关联数据集的技术方案。在 2011 年的 CoLD 会上,还提到了封闭(Closed)关联数据和绿色(Green)的关联数据的概念。

从技术标准和工具平台方面来看,关联数据作为语义网的一种轻量级的实现方式,早期的语义网研究成果如 RDF 数据模型、RDF 查询语言 Sparql、元数据(Metadata)、知识本体(Ontology)的相关理论和技术是它的基础,致力于语义网研究和应用推广的机构如 W3C、DERI、LATC、还有许多大学研究机构等,不仅参与制定和维护如 RDF、Sparql 语言、Sparql 协议、OWL 本体语言等基础的技术标准,近年来基于这些技术标准不断地开发出各种工具和平台致力于关联数据的发布、消费和利用,就消费来说,从语义浏览器如 Tabulator、Disco,到语义爬虫和语义搜索引擎如 Sig. ma、SWSE、Swoogle、Falcons,到关联发现整合平台如 SILK、LIMES、LDIF 等,这些工具绝大部分是开源的,依据一定的开源协议提供免费下载。

从应用方面来看,Web 上公开的大型关联数据集如 DBPedia、Freebase 由于其海量的数据、多样的消费接口,已成为全球性的数据消费中心,在各个领域得到利用,尤其是政府信息公开、地理信息、生命科学、图书馆档案馆博物馆等,不仅作为数据的提供方,更作为数据的消费方,消费来自 DBPedia、Freebase 和其他领域性关联数据集中的数据,如西班牙国家图书馆的关联书目数据,就用 owl:sameAs 和 rdfs:seeAlso 关联到 DBPedia 和美国国会图书馆的虚拟国际规范档(Viaf)。关联数据的消费也渗入移动领域,一个典型的例子是用于 iPhone 等智能手机的 DBPedia Mobile,它消费 DBPedia 的地点信息帮助旅行者探索某个城市。不仅集成 DBPedia 上现有的 RDF 数据,还支持用户即时发布自己的数据,并以 RDF 格式保存在 DBPedia 上,以供其他用户使用。这在很大程度要归功于 DBPedia 所提供的强大的数据接口,包括数据消费接口和数据输入接口。

3 关联数据的消费技术

从流程来看,关联数据的消费涉及到数据的访问和获取、发现、查询、交换和传输、处理和利用等方面,本文在此对这个过程中所涉及到的基本技术做

一个梳理和总结。

3.1 数据的访问和获取

对关联数据的一种简单直接的访问和获取方式是根据关联数据的四原则，直接访问资源对象的URI来获取关于资源对象的信息。关联数据建立在URI、HTTP等基础的互联网技术之上，每一个资源对象(Object)都有全球唯一的HTTP URI，它集标识功能和定位功能于一身，并且是可"解引(Dereferenced)"的，即可通过访问资源对象的URI来获取关于这个资源对象的信息，这些信息可以是一个html页面，也可以是基于某种序列化格式的RDF数据。在《发布关联数据的最佳实践》一文中提到："解引"可通过HTTP协议的"内容协商(Content Negotiation)"机制来实现，内容协商机制能根据客户端请求的类型(一般在HTTP Header信息中指定)返回相应格式的数据，若是普通浏览器，服务器会自动返回HTML数据，若是语义浏览器，则返回RDF数据。该文还推荐了两种最佳实践，即带#(Hash)的URI的实现方式和303转向的实现方式，如：http://linkeddata.openlinksw.com/about/Berlin#this就是一个带"#"的URI[9]。图1是用PHP实现303转向的一个例子，首先要保证服务器能接受的MIME类型包含客户端所要请求的类型，如application/rdf+xml，如在浏览器中输入http://lod.library.sh.cn/test303/foaf，返回HTML数据，输入http://lod.library.sh.cn/test303/foaf.rdf，则返回RDF/XML数据。

```
$extensions = array('.rdf', '.ttl', '.ntriples', '.html');
$ext = substr($_SERVER['REQUEST_URI'], (strrpos($_SERVER['REQUEST_URI'],'.')));
if (isset($_SERVER['HTTP_ACCEPT']) && strpos($_SERVER['HTTP_ACCEPT'], 'application/rdf+xml') !== false) {
    header('HTTP/1.1 303 See Other');
    header('Location: http://' . $_SERVER['SERVER_NAME'] . $_SERVER['REQUEST_URI'] . '.rdf', 303);
    die;
} else {
        // otherwise redirect to human-readable representation
   header('HTTP/1.1 303 See Other');
   header('Location: http://' . $_SERVER['SERVER_NAME'] . $_SERVER['REQUEST_URI'] . '.html', 303);
   die;
}
```

图1 用PHP实现303转向

对关联数据的另一种访问和获取方式是利用数据集或语义搜索引擎提供的消费接口：如Sparql端点、Restful Web Services接口、OpenSearch/SRU、各种客户端开发库、专用API等，将在本文第4部分详细分析。

3.2 数据的发现

通常把同类数据的集合称作一个数据集(Dataset)，目前Web上的关联数据大多以数据集的形式发布，当Web上的关联数据集越来越多时，就需要引入一种发现机制，作为数据发布者和数据消费者之间的桥梁，如建立一个注册机构或公开目录，提供关于数据集的数据，即数据集的元数据，就是有助于数据被发现的途径之一，尤其是机器的自动发现。VoID(Vocabulary of Interlinked Datasets)是一个基于RDFs的词表，定义了描述关联数据集的元数据方案[10]。VoID包括四个方面的元数据，其中"访问(Access)"元数据用于描述RDF数据使用何种协议访问，结构性元数据用于描述数据集的结构和模式，利于数据的查询和整合，对数据集之间的关系的描述有助于理解多个数据集之间的关系和不同的数据集之间如何整合利用。thedatahub.org就是这样一个为数据集提供注册机制的公开目录，截至目前，thedatahub.org的子集LOD(Linking Open Data Cloud)已建立了一个包含328个相互关联的关联数据集的公开目录，每个关联数据集都用VoID描述，如数据打包下载的URL地址，Sparql端点的URL地址，API的URL地址等信息，用于关联数据的发现和消费。DBPedia也有自己的VoID描述。

3.3 数据的查询

根据关联数据的四原则，数据要尽量采用RDF数据模型。RDF数据通常表现为一堆三元组的集合，描述某个资源的一个或多个三元组称为一个RDF图(Graph)。消费方要查询RDF图中的数据单元并对其进行处理，需要借助专用的RDF查询语言，如W3C的Sparql，Sparql已在2008年1月成为W3C的推荐标准，是目前使用最为广泛的RDF查询语言[11]。Sparql允许从RDF库(通常包含多个RDF图)中查询三元组，与关系数据库相比，RDF库是一个庞大无序的三元组集合，RDF数据没有外键和主键，只有URI，Sparql查询通过定义匹配三元组的RDF图模式(Graph Pattern)来完成，RDF图是用一个全球可定位的HTTP URI来唯一标识，而不是物理意义上的数据库名和表名，如查询URI为<http://lod.library.sh.cn:8080/bib>这个RDF图中dc:creator为"巴金"的所有主语的Sparql语句为：

```
SELECT DISTINCT * WHERE {
   ?s dc:creator "巴金"
}
```

这种方式简单而直接，无需了解底层的数据结构，甚至可以指定多个不同URI(RDF图)同时查询：

```
SELECT ?who ?g ?mbox
FROM <http://example.org/dft.ttl>
FROM NAMED <http://example.org/alice>
FROM NAMED <http://example.org/bob> WHERE {
  ?g dc:publisher ?who.
  GRAPH ?g { ?x foaf:mbox ?mbox }
}
```

Sparql 查询结果也是 URI，而 URI 是 Web 的标准引用格式，通过 URI，可以连接到 Web 上的任何数据，这就突破了关系数据库查询语言一次只能在单个数据库中查询的局限，整个 Web 于 Sparql 语言而言是一个巨大的整体的数据空间。基于 Sparql 语言和 Sparql 协议的 Sparql 端点技术，为查询关联数据集提供了标准的接口，可供人机检索，为大部分关联数据集和语义搜索引擎采用，在此基础上，出现了基于多个 Sparql 端点的联邦 Sparql 检索引擎，可整合不同的数据源的数据。另外，还有一些非标准的 RDF 查询语言，如 Freebase 定义了一种类似于 Sparql 的查询语言 MQL，专门用于查询 Freebase 中的数据。Sindice 为自己的消费接口定义了专用的查询语言 Sindice Query Language。

3.4 数据的序列化

RDF 只是一种抽象的数据模型，而不是一种具体的数据格式，要使 RDF 数据成为机器可读的数据，就需要对其进行序列化(Serialization)。序列化的 RDF 数据可以通过 HTTP 协议传输和交换，这样应用程序可以对这些数据进行远程获取和处理，有利于不同的操作系统、程序语言之间的互操作。目前有诸如 RDF/XML，RDFa/JSON/JSONP，N3，Turtle 等不同的序列化格式，其中 RDF/XML 和 RDFa 是 W3C 的推荐标准，其他的序列化格式则为满足不同的具体需求而设计。RDF/XML 是 RDF 模型最为经典的序列化方式，与 XML Schema 配合使用，基于 XML 编码的 RDF 甚至能实现不同应用领域之间的互操作，但 RDF/XML 有着不利于人读和写的缺点。RDFa 是将 RDF 三元组嵌入 HTML 文档的一种序列化格式，适用于能方便修改 HTML 文档模板但难以介入系统体系架构的应用，如 Drupal 等内容管理系统[12]，这种序列化方式不能很好地支持内容协商机制。Turtle 是一种纯文本的 RDF 序列化格式，适用于人读和写，N-Triples(N3)是 Turtle 的一个子集，因其规定每一个三元组中的主体、谓词、客体都必须用完整的 URI 来表示，如：<http://lod.library.sh.cn:8080/bib/resource/PersonA000001> <http://www.w3.org/2002/07/owl#sameAs> <http://dbpedia.org/resource/Ba_Jin/>。所以与 Turtle 和 RDF/XML 相比，文件会比较大，但这也是它的优点，因为它的每一行都可以单独解析，同时它也很适合压缩以减少在传输交换过程中的网络流量，这种特性使得 N3 成为适合传输大型关联数据包的一种序列化格式。JSON(JavaScript Object Notation)是一种适合于程序处理的 RDF 序列化格式，大部分程序语言本身就提供处理 JSON 数据的功能，如 JavaScript 和 PHP，将 RDF 数据发布成 JSON 格式可使程序开发员无需安装额外的开发包就能处理 RDF 数据。目前一些关联数据消费接口一般都会提供一种或几种格式的数据以内容协商的方式返回给客户端，如 DBPedia 的 Sparql 端点就提供 RDF/XML，JSON，N3 等多种数据格式。而一些面向机器的消费接口则以 JSON 格式为主。

3.5 数据的处理和利用

在数据消费的过程中，访问和获取、发现和查询、传输和交换的最终目的是为了将数据拿来为我所用。一般有这样几种利用方式：(1)为本地数据建立外部关联，如上文提到的西班牙国家图书馆的关联书目数据；(2)多种数据混搭建立新的应用和服务，如社会书签工具 Faviki，利用 DBPedia、Freebase 等作为背景知识库，通过关联数据的 URI 技术来消除歧义，提供分主题组织的标签导航服务；(3)发现数据之间的关联关系建立知识地图，如生命科学领域的关联数据应用 Diseasome Map，整合来自不同的生命科学领域的数据资源，生成一个相互关联的病毒基因网络；(4)进行语义挖掘和推理发现有用的信息，如 Researcher Map 是一个基于 FOAF 的关联数据应用，通过 DBPedia 和 DBLP 关联数据集中的 RDF Links 来发现德国教授的个人信息。在数据处理和利用的过程中，要对数据查询结果提取和重组、分析和计算，需要基于一些已有的标准规范，也需要一些开发包来支持。在标准规范层面，一些早期的语义网技术的成果如 RDF/XML、Sparql、OWL，其标准的制订和技术的应用已经成熟；在应用开发层面，一些流行的编程语言都有处理 RDF 数据的开发包，如 Java 的 Jena、Sesame、Kowari，PHP 的

RAP，以支持 Sparql 协议向 Sparql 端点发起 Sparql 查询、处理查询结果、读写序列化的 RDF 数据文件、操作原生 RDF 数据库等[13]。还有一些集成的客户端开发库如 PHP 的 ARC2，被 Drupal 等开源平台集成，作为后台关系数据库和前台 Sparql 端点之间的中间件，提供实时的 Sparql 查询到 SQL 查询、SQL 查询结果到 Sparql 查询结果的双向转换。在数据处理的过程中，不可避免会涉及到不同本体或词表间的映射，R2R 词表映射框架可以帮助关联数据应用在 Web 上发现一些未知的术语、术语之间的映射，并利用这些映射将 Web 上的数据转换到应用的宿主词汇表，有助于发现异构数据之间的关系。数据混搭语言 MashQL 提供了另一种整合数据的思路，它将整个 Web 看成一个大的数据库，每个数据源看成是数据库中的一张表，消费者可以不必关心各个数据源所用的词表和技术细节[14]。

4 关联数据的消费接口分析

一个优秀的关联数据集，在发布时就应考虑如何被消费，尽可能地提供便于数据消费者调用的消费接口，大型的关联数据集如 DBPedia、Freebase 就提供多种方式、多种功能、适应不同需求的数据消费接口。一些语义搜索引擎后台有着不断更新壮大的海量 RDF 数据，在 Web 上有供人检索的入口，也有供机器获取数据的开放数据接口。一些数据集成管理平台本身的功能模块可提供不同接口以支持关联数据集的消费。笔者调研的 DBPedia、Freebase、Nature.com、Data.gov、VIAF 等典型的关联数据集，<sameAs.org>、SWSE、Sindice、Swoogle、Falcons 等流行的语义搜索引擎，CKAN、Openlink Software、Tailis Plateform 等大型数据集成管理平台所提供的消费方式和消费接口主要有这几种：批量下载、Sparql 端点、RESTful Web Service 接口、OpenSearch/SRU、专用 Web Service 接口、专用 API、专用客户端开发包、插件（Plugin）/小工具（Widget）。目前关联数据所提供的消费接口有如上所列的多种或全部，每个调研对象的详细情况见表 1、表 2、表 3。

批量下载。早期的关联数据集一般会将所有的 RDF 数据打包，在 Web 上提供批量下载地址。作为数据消费方，需要将这些打包的 RDF 数据下载到本地，在本地存储，供本地系统重用这些数据。目前，仍有许多关联数据集沿用这一做法，如 DBPedia、Freebase、CKAN、VIAF、Data.gov.uk 等。DBPedia 以每 2—3 个月更新一次的频率发布了多个版本，2012 年 8 月 6 日发布了 3.8 版，相较于 3.7 版，在数据量、数据的内容范围、组织方式上都有一些增加和改进。CKAN 则提供几种方式让消费方自己制作 CKAN 数据的打包文件。VIAF 提供 RDF/XML，MARC21/XML 等多种格式的打包文件批量下载。有的关联数据发布平台提供 RDF 数据包的打包工具，如 D2R Server 的 Dump-rdf，可将整个关系数据库转换为 RDF 数据包。这种批量下载的消费方式要求数据提供方和数据消费方都要考虑到数据集的更新频率，发布方要考虑到数据发布的版本管理，消费方要密切关注数据发布的动态，考虑是否需要及时更新本地数据。

表 1 关联数据集（Linked Data Sets）的消费接口

	下载	Sparql 端点	Restful WS	OpenSearch /SRU	专用 W S	专用 API	专用客户端开发库	插件、小工具
DBPedia	√	√			√			√
Freebase	√	√	√	√		√	√	√
VIAF			√	√				
Nature.com		√	√	√				
Data.gov.uk	√	√	√			√	√	√
Data.Gov	√	√						

表 2 语义搜索引擎（Semantic Search Engine）的消费接口

	Sparql 端点	Restful WS	专用 WS	专用 API	专用客户端开发包	第三方插件
<sameAs.org>		√			√	√
SWSE	√	√				
Sindice	√		√	√	√	√
Swoogle		√				
Falcons		√				

表 3 工具/平台所提供的消费接口

	Sparql 端点	Restful WS	专用 WS	专用 API	专用客户端开发包	第三方插件
CKAN	√	√		√	√	√
Tailis 平台	√	√		√		
OpenlinkSW	√	√	√		√	√

Sparql 端点是目前最为流行的关联数据消费接口之一，几乎大部分的关联数据集如 DBPedia、na-

ture.com、BBC 都提供这种消费接口，大部分的关联数据发布工具/平台如 D2R Server、Drupal、Pubby、LDIF、Virtuoso 等都提供 Sparql 端点的技术支持，可以说 Sparql 端点是关联数据发布时的标准配置。它既可提供 Web 界面用于人的浏览，也是机器访问和获取数据的接口，人通过普通浏览器访问关联数据集的 Sparql 端点网址（如：http://DBpedia.org/sparql），输入并编辑 Sparql 查询语言，选择所需的数据编码格式，点击一个按钮即可获得返回结果的 RDF 数据。对于机器来说，要通过 Sparql 端点访问或获取某一关联数据集的数据，通常需要借助处理 Sparql 查询的客户端开发库如 Jena 的 ARQ。相比而言，基于 Sparql 端点的消费方式对于消费方来说有较高的技术门槛，要求消费方进行一定的编程工作，对于发布方来说则无此顾虑，因为大多数关联数据发布工具、平台都以提供 Sparql 端点为标准配置。

Restful Web Services 是一种轻量级的 Web Services，利用 URI、HTTP 等简单通用的 Web 标准及技术，对数据进行读取、写入、修改、删除等操作，是目前最为流行的数据接口，大部分的数据集、语义搜索引擎、关联数据管理平台都提供这种数据消费方式，见表 1、表 2、表 3。一种简单的基于 Restful Web Services 的数据消费方式是数据消费方直接访问某一资源对象的 URI，通过内容协商机制，获取该资源对象某种指定编码格式的 RDF 描述。例如，当数据消费方需要获取 VIAF 的某一资源对象数据时，可以直接访问该资源对象的 URI：http://www.viaf.org/viaf/75785466/，若需返回 XML 格式的数据，则访问 http://www.viaf.org/viaf/75785466.xml，若需返回纯 rdf 数据，则访问 http://www.viaf.org/viaf/75785466.rdf。表 4 是 VIAF 的 RESTful Web Service 请求和响应的对照表。

更为复杂的 Restful Web Services 利用 URL 传递事先定义的参数或专用的结构化检索语言向服务器发送数据查询或处理请求，服务器返回既定或由客户端指定的内容和格式。例如 VIAF 的 AutoSuggest 接口，其语法为：http://viaf.org/viaf/AutoSuggest？query=[searchTerms]&callback [optionalCallback-Name]，

表 4　OCLC viaf 的请求/响应对照表[15]

Accept Header	Response Format	URL
/ application/xhtml+xml，text/html	HTML	/
text/xml	application/xml	XML
/viaf.xml application/rdf+xml	RDF XML	/rdf.xml
application/marc21+xml application/marc21+html	MARC21 XML	/marc21.xml
application/unimarc+xml application/unimarc+html	UNIMARC XML	/unimarc.xml
application/rss+xml	RSS	/rss.xml

返回数据的格式为 JSON。DBPedia 的 Lookup Service，其语法为：http://lookup.dbpedia.org/api/search.asmx/<API>？<parameters>。其中<API>是指从 DBPedia 申请的 API Key，用于权限控制。还有 Freebase 的各种 Search API 如根据关键词查找实体的 Search Service、获取一个实体的概要信息的 Topic API、获取实体的简单文本型描述信息的 Text Service，获取图像实体的拇指图的 Image Service。其中 Search Service 的语法为：https://www.googleapis.com/freebase/v1/search？q=bob&key=<YOUR_API_KEY>。Sindice 的 Search API（v3）使用自定义的查询语言详细定义了十多种参数，包括关键词、词表及属性、返回结果的格式等，如：http://api.sindice.com/v3/search？q=Rome&fq=class：city&format=json。Sindice 基于其自定义的 Sindice Query Language 的 API 支持关键字匹配、三元组查询和各种过滤策略。Tailis 平台的 Augment Service API 能对来自不同数据集的返回结果进行合并处理。Tailis 平台的 Full Text Searching API 不仅可以利用自定义的全文检索语言通过在 URL 传递事先定义好的参数对数据集中的数据进行全文检索，还可以传递 Sparql 查询语言来查询数据集中的数据，甚至可以对数据集中的数据进行修改和删除。这种方式对于消费方来说使用方便，但发布方需要考虑到数据传输中的安全问题，如是否需要数据加密和权限控制。

OpenSearch/SRU。OpenSearch 的是 Amazon.com 子公司 A9 公司所提出的一种分享查询结果的简单的格式标准，2005 年 3 月首次在 O'Reilly 新兴

技术会议上提出 OpenSearch1.0 版本，目前使用版本为 1.1 版。OpenSearch 与国会图书馆的结构化查询规范 SRU 结合，同样是利用 URL 传递参数返回指定格式的结果数据，但这里的参数不是自定义而是标准的结构化查询语言 CQL，如 CKAN 和 VIAF 提供 OpenSearch 接口，而 nature.com 则提供 OpenSearch 和 SRU 整合的消费接口[16]。VIAF 的 OpenSearch 查询举例：http://viaf.org/viaf/search?query=cql.any+all+%22[searchTerms]%22+&maximumRecords=100&startRecord=[startIndex]&sortKeys=holdingscount&httpAccept=application/rss%2bxml，VIAF 还提供一个可视化的界面，只需要选择一些参数的值，系统会自动生成包含 CQL 查询的 OpenSearch URL。这种方式建立在 OpenSearch、SRU、CQL 等标准和技术之上，对于消费方来说，无需进行复杂的编程运算，又能灵活方便地直接获取关联数据集中的数据单元，但需要掌握 CQL 语言。对于发布方来说，可以植入自定义的认证和授权规则，有利于数据的安全和版权控制。

专用 Web Service 接口。如 DBPedia 所提供的“公开分面 Web Services 接口”，http://DBpedia.org/fct/service 建立在 Openlink Software 公司的 Virtuoso 分面 Web Services 接口基础上，该 Web Services 接口定义了一系列的 HTML 标签，在这些标签中可以写入 Sparql 语言。当在本地 Web 应用的 HTML 网页上嵌入这些标签时，服务器会返回所请求结果的 XML 格式数据，供本地网页用自定义的 XSLT 为用户提供数据展示界面[17]。

专用 API。这种方式是发布方通过 API 来精心包装自己的数据，甚至定义专用的查询语言，开发出专用于特定数据获取和处理的方法以供消费方在编程时调用。一般会详细定义输入参数和返回结果，还有对返回结果进行处理的方法，同时有可靠的授权控制机制如 API Key、Web ID 等。如 CKAN 的各种 API，允许消费方传入指定的属性值，返回特定的结果。还有 Freebase 基于其 MQL 的各种 API。这种方式需要发布方进行大量的开发工作，将数据访问、查询和处理过程封装起来，对用户透明，只根据用户的输入参数按需提供结果数据，而消费方需要深入了解 API 的使用方法，对发布方和消费方都有较高的技术门槛。

专用客户端开发包。一些大型的关联数据集为了满足不同消费者的个性化需求，为数据消费提供无限的可能，开发出专用的客户端开发包，供程序员编程使用。如 Freebase 提供 JavaScript、Flash、Python、Perl、PHP、Ruby、Clojure、Java、NET、Objective-C 10 种语言的客户端开发包，同时提供基于 JavaScript 的 Arce 集成开发平台，有着大量利用 Freebase 的数据构建本地应用的开发工具。CKAN 也提供专用的客户端开发库，目前有 Java、PHP、JavaScript、Python、Rubby，PERL6 种语言的客户端开发库，利用这些客户端开发包，消费方可以灵活地获取、使用、甚至修改 CKAN 中的数据，进行复杂的计算。语义搜索引擎 sameAs.org 提供 Java 开发包 sameas4j，Sindice 提供 Java 和 Pyhton 等简单的工具包。

插件、小工具。在 Web2.0 时代，插件和小工具以其灵活、小巧，可自由嵌入等特点风行一时，这些优点在语义网世界继续发扬光大，如 thedatahub.org 的基于 Chrome 浏览器的插件 JSONView for Chrome，可以在 Chrome 浏览器上浏览它的数据，还有 Google Refine CKAN Extension，DBPedia 的 DBpedia Lookup 服务插件，可以装在本地服务器上，其功能是通过关键词或简写查找某一实体的 URI。Freebase 用于移动设备上的 App 如 Freebase Explorer Android App。Sindice 的基于 Drupal 的 MOAT module，可以在 Drupal 站点中消费 Sindice 提供的数据。

5 技术实现案例

案例 1：直接利用外部数据的 URI 为本地数据添加外链。

要向本地数据集——名人规范档中的实体添加到 DBPedia 的外链，用 owl:sameAs 作关联。首先了解到 DBPedia 中对某一人(Person)的实体的 URI 命名规则是“http://dbpedia.org/resource/”+人名，若是中国人，人名是姓名的拼音，如巴金的 URI 是 http://dbpedia.org/resource/Ba_Jin/，这样，就可以把本地数据集中的巴金用 owl:sameAs 链接到 DBPedia 的巴金，用三元组来表示，即：<http://lod.library.sh.cn:8080/bib/resource/PersonA000001>owl:sameAs <http://dbpedia.org/resource/Ba_Jin/>。其他的人则可以由此规则类推，只要计算出本地数据集中所有人物实体的姓名的拼音，如矛盾的拼音是

Mao_Dun，其 URI 就是 http://dbpedia.org/resource/Mao_Dun/，采用这种数据消费方式，需要数据发布方有明晰的 URI 命名规则或者提供根据关键词查询 URI 的服务，数据消费方也要对目标关联数据集的 URI 命名规则或 URI 获取方式有所了解，才能准确地获知某一资源对象的 URI。

案例 2：利用 Restful Web Services 获取数据。

要获取本地数据集中的实体在 VIAF 中的 URI，首先了解到 VIAF 的 URI 命名规则是“http://viaf.org/viaf/”＋VIAF ID，只要获取了某一实体的 VIAF ID，就可以知道这一实体在 VIAF 中的 URI。这时可以用 VIAF 的 AutoSuggest 接口，获取本地数据集中某一实体在 VIAF 中的同一实体的 VIAF ID，以巴金为例，在 URL 中传递参数 bajin，http://viaf.org/viaf/AutoSuggest？query＝bajin，返回的结果为 JSON 格式的数据，如下：

```
{" query" :" bajin",
" result":[{" term":" Bajin"," viafid":" 19673501"," nla":" 000036730226"," nlilat":" 000102513"," nkc":" jo2002104188"," bnf":" 11889753"," selibr":" 176382"," dnb":" 118914421"," lc":" n79133113"}]
}
```

上述结果中包括 viafid 这个属性的值 19673501，此即巴金的 VIAF ID，由此可知巴金在 VIAF 中的 URI 为 http://viaf.org/viaf/19673501。可以用程序读取所返回的 JSON 数据，取得 VIAF ID 的值。

案例 3　基于 Java 利用 ARQ 客户端编程向多个 Sparql 端点获取数据。

已知 DBPedia 的 Sparql 端点的 URL 地址为：http://dbpedia.org/sparql。利用 ARQ 开发包，它支持指定关联数据集的 Sparql 端点的地址，就可以远程访问和获取数据。如下为部分程序片段：

```
import com.hp.hpl.jena.query.QueryExecution;
import com.hp.hpl.jena.query.QueryExecutionFactory;
import com.hp.hpl.jena.query.ResultSet;
import com.hp.hpl.jena.query.ResultSetFormatter;
String query =
"    PREFIX rdfs:＜http://www.w3.org/2000/01/rdf-schema#＞ \n"＋
"    PREFIX owl:＜http://www.w3.org/2002/07/owl#＞ \n"＋
"    select distinct ?sameAs where \n"＋
"    {?s rdfs:Label" Lu Xun". \n"＋
"    ?s owl:sameAs ?sameAs \n"＋
"    } LIMIT 100";
QueryExecution x = QueryExecutionFactory.sparqlService("http://dbpedia.org/sparql", query);
try {
    ResultSet results = x.execSelect();
    for (; results.hasNext();) {
    // 在此处理查询结果
    }
}
finally {
    x.close();
}
```

6　总结与展望

基于关联数据技术框架的数据消费可以做到真正意义上的分布式和跨数据源的查询，实现基于语义的全 Web 集成。各种大型公开的关联数据集和语义搜索引擎提供了海量的富含语义的结构化数据，成为 Web 上的公共数据中心，多样化的数据消费接口为数据消费者利用数据带来了无限的可能。但是，目前关联数据的消费接口，除了 Sparql 端点外，其他的方式包括 Restful Web Services 接口，尤其是各种专用的 API，因其使用方法各不相同，有的还异常复杂，尚未形成统一的标准，对消费方来说颇为不便，成为阻止关联数据得到有效利用的障碍之一。在系统和平台的开发上，进入此领域的大型商业公司并不多，为大英图书馆提供关联数据体系架构的 Tailis 公司在 2012 年宣布停止提供关联数据相关服务，反而在大学和研究机构，涌现出越来越多的开源项目，但对于普通消费者来说，开源软件的使用仍然存在着较高的技术门槛。随着关联数据的发布越来越呈现出分布式、虚拟化、关联化的特征，关联数据的消费也将更多地依赖于接口的标准化。另一方面，数据来源跟踪和数据质量检测的机制的建立，是来自技术之外的挑战。

参考文献

1　刘炜. 关联数据：概念、技术及应用展望. 大学图书馆学报，2011(2)：5－12
2　黄永文. 关联数据驱动的 Web 应用研究. 图书馆杂志，2010(7)：55－59

3 白海燕. 关联数据及 DBPedia 实例分析. 现代图书情报技术,2010(3):33－39
4 林海青. 关联书目数据:发布、查询、消费及混搭. "从文献编目到知识编码:关联数据技术与应用"专题研讨班. 2012,7. [2012－08－21]. http://conf. library. sh. cn/sites/default/files/LBD 的查询消费及混搭_林海青. pdf
5 夏翠娟. 关联数据的技术实现及案例. "从文献编目到知识编码:关联数据技术与应用"专题研讨班. 2012,7. [2012－08－21]. http://conf. library. sh. cn/sites/default/files/LD 的技术实现及案例_夏翠娟. pdf
6 Li Ding, etc. Swoogle: A Search and Metadata Engine for the Semantic Web, the Thirteenth ACM Conference on Information and Knowledge Management , November 2004.
7 Li Ding, etc. Finding and Ranking Knowledge on the Semantic Web, in the Proceedings of the 4th International Semantic Web Conference, November 2005
8 Tom Heath, Christian Bizer. Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1－136
9 Diego Berrueta , Jon Phipps. Best Practice Recipes for Publishing RDF Vocabularies. [2012－08－31]. http://www. w3. org/TR/swbp-vocab-pub/
10 Keith Alexander, etc. Describing Linked Datasets with the VoID Vocabulary. [2012－08－31]. http://www. w3. org/TR/void/.
11 Olaf Hartig, Christian Bizer, Johann-Christoph Freytag. Executing Sparql Queries over the Web of Linked Data. Lecture Notes in Computer Science, 2009, Volume 5823/2009:293－309.
12 夏翠娟等. 关联数据的发布技术及实现——以 Drupal 为例. 中国图书馆学报,2012(1):49－57
13 Christian Bizer, Daniel Westphal. Developers Guide to Semantic Web Toolkits for different Programming Languages. [2012－08－21]. http://www. wiwiss. fu-berlin. de/suhl/bizer/toolkits/06062006/
14 Mustafa Jarrar, Marios D. Dikaiakos. A Data Mashup Language for the Data Web. [2012－08－31]. http://events. linkeddata. org/ldow2009/papers/ldow2009_paper14. pdf
15 OCLC. Virtual International Authority File : Using the API. [2012－08－26]. http://www. oclc. org/developer/documentation/virtual-international-authority-file-viaf/using-api
16 Tony Hammond. nature. com OpenSearch: A Case Study in OpenSearch and SRU Integration. [2012－08－13]. http://www. dlib. org/dlib/july10/hammond/07hammond. html
17 Openlink Software. Virtuoso Facets Web Service. [2012－08－24]. http://virtuoso. openlinksw. com/dataspace/dav/wiki/Main/VirtuosoFacetsWebService

作者单位:上海图书馆,上海,200031
收稿日期:2013 年 2 月 5 日

Technologies and Implementation of Consuming Linked Data

Xia Cuijuan　Liu Wei

Abstract: Publishing and consuming are both important for Linked Data implementation. The technologies of consuming Linked Data involve accessing, discovering, querying, exchanging, transmitting, and processing. How to consume Linked Data depends on data consuming interface provided by Linked Data source. Survey has been conducted for the data consuming interface provided by the Linked Data sets such as DBPedia, Freebase, Viaf, etc. and the semantic search engine such as 〈sameAs. org〉, SWSE, Sindice, Swoogle, Falcons, etc. , the strengths and weaknesses of every type of interface are also analyzed.

Keywords: Semantic Web; Linked Data; Consuming Technologies

(接第 28 页)

Data Curation Education and Career Development at Abroad

Ye Lan

Abstract: The paper provides a snapshot of the theoretical research and practical progress on the data curation education and career development at abroad, and summarizes five characteristics of the data curation education and career development in foreign countries: data-centric courses and programs are new and under development; data management curriculums vary; data curation education and training programs are aimed at all levels of data professionals; there exits kinds of cooperation models; courses and training programs reflect the actual demand for job skills. Finally, some reflections on the development of data curation education and training programs in China are presented.

Keywords: Data Curation Education; Data Management Curriculum; Training Programs; Professional Skills