

面向最终用户的SPARQL Endpoint检索接口与信息呈现*

姜恩波

(中国科学院成都文献情报中心, 成都 610041)

摘要: 自语义网概念提出以来, SPARQL Endpoint作为语义网时代特有的一种服务, 其发展呈现出独特的特征。文章围绕SPARQL Endpoint在用户交互及信息呈现方面存在的问题, 从SPARQL Endpoint的背景与工作原理出发, 对其用户需求表达交互、信息呈现以及系统功能性指标等三个方面的技术进行了对比和分析, 并总结了它们的服务特点。

关键词: SPARQL Endpoint; 用户交互; 信息呈现; 可发现性

中图分类号: G254

DOI: 10.3772/j.issn.1673—2286.2014.11.009

1 问题的提出

1998年, 蒂姆·伯纳斯-李(Tim Berners-Lee)提出了语义网(Semantic Web)的概念。在这后续的十五六年里, 基于语义网的应用层出不穷: 语义搜索、语义推理、本体管理、本体抽取、本体映射、本体存储等等。符合语义网标准的数据资源也不断增长。2012年, Google Knowledge Graph其语义网包含超过570亿个对象, 超过35亿个事实及其关联^[1]。作为关联开放数据中心的DBpedia, 2014年9月发布的最新版本数据中, 包含了用125种不同语言标识的458万个实体, 并外联至3000多万个页面^[2]。但是, 这一切对于普通用户来说, 就像“大峡谷”对面的美丽景色, 可见却不可得。

以DBpedia的SPARQL Endpoint服务为例。如果我们要从维基百科中查询“有哪些人出生在柏林”, 写成SPARQL语句就是: “select ?p where {?p dbpprop: birthPlace "Berlin"@en.}”。而普通用户要走到这一步, 必须具备三方面的知识:

—关于本体的知识: RDF语法、RDF词汇;

—语义查询语言的知识: SPARQL、RDQL、MQL、

Gremlin;

—知识组织体系的语义表达。例如DBpedia用dbpprop: birthPlace表示“出生地”。

可以想见, 对于一般用户来说, 这样系统的可用性趋近于零。当用户想要了解莫扎特的生平, 却不得不先学习SPARQL语法。而这是系统所提供的唯一交互方式。面对DBpedia丰富的知识库, 用户只能望洋兴叹。

用户界面(User Interface)是当前语义网发展的一个短板。在Web2.0时代, 面向普通用户的开发工具层出不穷。AJAX、ExtJS、jQuery、django、mootools给用户带来了极好的应用体验。而在语义网时代, 大多数工具都是基于对三元组的管理与发布, 应用对象是技术人员和本体资源设计与建设人员。而面向普通用户的优秀开发框架与工具却严重缺失^[3]。用户界面不好, 直接影响到系统功能的可用性。而可用性差的直接表现就是用户需要承担的负担大, 需要了解的技术细节多。在当前

* 本研究得到“中国科学院文献情报能力专项项目——开放会议资源服务系统建设”资助。

互联网应用越来越依赖于用户体验的时代,语义网却在理论研究与实际应用间存在了极宽的“鸿沟”。鉴于此,Duane Degler甚至提出“语义网的实现不再需要发展底层的技术,而只需要良好的用户界面与系统可用性”^[4]。

2 背景: SPARQL EndPoint

当前,用户获取语义网服务的途径有三个:Web网站、SPARQL Endpoint以及RDF DUMP。SPARQL EndPoint作为语义网时代特有的一种服务方式已经占据了相当的应用份额。LOD云图中209个关联数据集中已经有接近70%的数据集提供了SPARQL EndPoint服务^[5]。另外,现阶段的SPARQL EndPoint在存在形式与功能上都已然表现出了多样性,为观察和研究语义网用户界面与功能的发展提供了比较理想的途径。

2.1 SPARQL与SQL的关联

2008年,W3C发布了其推荐标准SPARQL (SPARQL Query Language for RDF) 1.0版本。2013年3月,SPARQL升级到1.1版。SPARQL允许应用程序对分布式RDF数据库进行复杂的查询。SPARQL其重要性相当于SQL对于关系数据库一样。在当前,SPARQL和SQL既独立又有协同工作。与关系型数据库的SQL相比,SPARQL的优势在于能够支持多种查询类型,如SELECT、ASK、CONSTRUCT、DESCRIBE等。例如CONSTRUCT,用户可以根据自己的需要来对检索结果进行一些定制,以自己熟识或本地系统所需要的RDF词汇来组织数据。另外,由于RDF数据库的应用还不是非常普遍。RDF数据集往往保存在传统的关系型数据库中,因此,SPARQL和SQL往往协同工作。例如JENA的sparql2sql。它会把用户输入的SPARQL语句转换为SQL,然后再到底层的关系数据库中进行查询和转换。

2.2 SPARQL EndPoint技术原理

Endpoint是一种以URI进行标识,通过特定协议获取特定Web服务,并返回数据结果的服务方式。SPARQL Endpoint既是一种基于SPARQL语言的服务方式。W3C SPARQL 1.1的服务描述(Service Description)中这样定义SPARQL Endpoint:“侦听各类客户端所发送的SPARQL请求的服务URI。”^[6]图1表

示了SPARQL Endpoint工作原理。

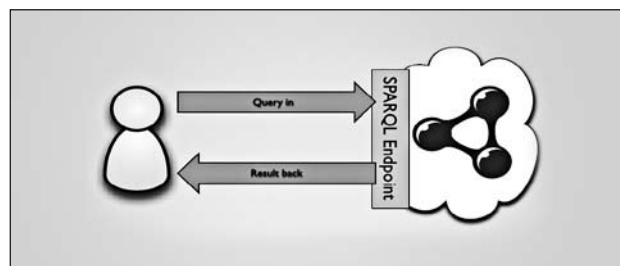


图1 SPARQL Endpoint工作原理示意图^[7]

上图中,用户提出自己的需求表达。这种需求表达在实际应用中存在多种形式。最直接而目前也是最常见的是以各种查询语言(QL Language)拟就的查询语句。这种方式表达能力强,查询结果准确,但是能够使用的对象极少。另外,需求表达也可以通过关键词以及受控自然语言的方式提交。这就需要在用户与SPARQL EndPoint之间存在一个转换机制。这个机制对需求进行解析,如语法、句法分析,然后重新组合成查询语句。查询结果,对于机器用户,只需要返回基于RESTFUL的、特定RDF序列化的数据流即可。对于普通用户,还需要利用特定技术对结果进行组织与呈现。

SPARQL Endpoint是针对RDF数据的语义查询技术。RDF是语义网提出的数据组织标准。它以三元组的方式来描述信息实体(Entity)以及实体之间的关联,以URI作为表征数据的方式,这与之前的基于文本的互联网的服务有着极大的区别。而这些差异都从SPARQL Endpoint服务方式中得以体现。

根据以上的分析我们认为,对SPARQL EndPoint的可用性分析可以从三个方面进行。首先是用户需求表达的交互方式;其次是语义数据查询结果的组织方式;再则是SPARQL Endpoint服务可发现性、服务能力以及执行效率。

3 用户需求表达的交互方式比较

获取用户需求表达是SPARQL EndPoint与用户交互的第一个环节。在这个环节中,用户需要把需求明确、完整地表达出来。然而,目前绝大多数SPARQL EndPoint服务在这个环节是不友好的。这体现在两个方面:首先,用户需要掌握太多的知识和技能才能将需求表达清楚,如前面提到的DBpedia EndPoint服务。另

外,用户的需求不能通过系统提供的交互方式完整地进行表达。鉴于此,不少机构都在努力减少这两方面的缺陷,为用户提供良好的第一印象。这也因此衍生出多种不同类型的SPARQL EndPoint服务。

3.1 模板提示型需求交互方式

这一类SPARQL EndPoint服务是针对第一种缺陷而提出的。顾名思义,机构结合知识库自身的功能以及对查询需求的分析,为用户提供不同类型的SPARQL查询模板帮助用户表达需求。

下图是TSO flint系统提供的一种应用模板。右侧模板包括了三方面的信息。首先是SPARQL语句的输入提示。系统会根据页面左侧用户输入的内容以及SPARQL语法,给出下一步输入的可能选择,并以按钮黑色字显示。其次是RDF数据集的结构及表达信息,如类(Class)及属性(Property)。最后,系统也给出了一些查询用例。用户可以据此来构造特定RDF数据集的查询语句。很明显,这种模板依旧是面向技术人员的,离普通用户仍然很远。

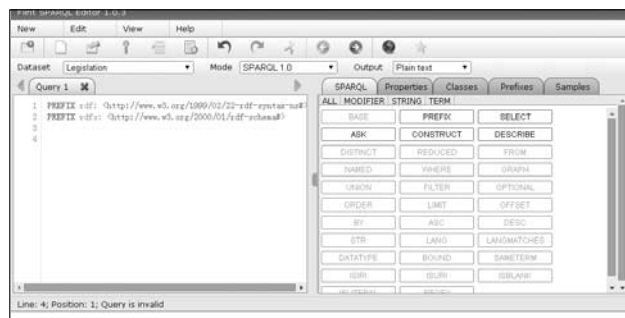


图2 TSO flint SPARQL Endpoint提示模板^[8]

第二种查询模板以Europeana和AAT为代表。前者致力于欧洲各国的文化、艺术品信息的保存和呈现。而后者是一部广泛用于艺术博物馆、图书馆和档案馆的主题词表。这两个系统面向普通用户提供查询模板。例如: Europeana的“18th century Europeana objects from France”和AAT的“Subjects by Contributor Abbrev”。实际应用中,用户点击模板,系统会自动生成相对应的SPARQL查询语句。如果模板中有符合需要的内容,用户就可以在不需要具备SPARQL知识的前提下得到结果。如果用户对SPARQL以及本体有一些了解,还可以在“近似需求”模板的基础上进行修改甚至整合。

很明显,这种类型的SPARQL EndPoint服务,特别是以Europeana和AAT为代表的模板提示型服务,能够减少用户在使用过程中的一部分障碍。但是这种帮助却依然十分有限。Europeana包含了上亿的实体和陈述(Statement),EDM体系更是极其复杂,但是却只提供了15种帮助模板。AAT虽然提供了较多的模板,但是距离满足需求还远远说不上。另外,模板提示型服务更加适合AAT、LCSH、MeSH这类内容较为单一、实体间关系不太复杂的叙词表、分类法体系。而对于DBpedia、Europeana这种跨行业、跨领域,拥有复杂属性、类别的知识库就不太合适。

3.2 基于关键词的需求交互方式

对于普通用户,他们不愿意也不需要去了解后台数据的组织方式是基于文件、关系数据库还是本体。他们也不需要知道从文件到数据库再到语义本体之间的技术飞跃。他们希望的只是最简单的输入得到最准确的结果。这就有了基于关键词的SPARQL EndPoint服务。基于关键词的SPARQL EndPoint服务的最大优点是需求输入界面与传统的文本网络(Web of Document)的信息检索系统相类似。用户在使用过程中基本没有心理障碍。而系统把接收到的检索词转化为SPARQL语句进行查询,对用户屏蔽了部分技术细节。对于需求不是特别复杂的用户来说,如果配合较好的查询结果组织方式,已经能够满足其需求了。当前,大多数语义网搜索引擎大都是以这种方式提供服务,如Google knowledge graph、Falcons、Swoogle、Watson以及DBpedia的关键词搜索。下图是Falcons的查询界面。



图3 Falcons语义搜索引擎检索策略输入界面^[9]

在实际调研中,笔者也发现这种类型的SPARQL EndPoint服务的两个问题。首先,在查询输入界面上依然不能完全屏蔽掉一些用户无法理解的内容。例如几乎所有的搜索引擎检索界面都有这样一些需

要用户勾选的选项: Object、Concept、Ontology、across Ontology、Document、Term、has Class、has Predicated等等。这些信息是后台RDF数据集用来进行知识组织的内部结构与方式。这种选择无疑会让用户对整体检索策略产生不信任感,从而对检索结果产生怀疑。另外一方面,语义网最精华的部分就在于实体、对象之间的关联。如果用户的需求表达没有或者不能够把它们之间的这部分信息包含进去,那么这种查询对语义网的优势就无法体现出来。而基于关键词的SPARQL Endpoint服务在需求表达能力方面明显无法达到这个要求。这也就产生了下面这类基于自然语言的SPARQL Endpoint服务。

3.3 基于自然语言 (NL) 的需求交互方式

基于自然语言的SPARQL Endpoint虽然不是目前最好用的SPARQL Endpoint服务,但无疑却是最有发展前途的一种SPARQL Endpoint服务。原因在于这种模式能够有效地弥补前面所说的两种缺陷。

在实际服务过程中,用户与系统的交互是通过提交(受控)自然语言进行的。系统对用户需求的处理分为三步。首先是句法、词法分析。系统对用户输入的自然语句进行句法和词法的分析,得到主干词汇以及词汇之间的依赖关系。第二步将主干词汇与知识库中的实体进行比较、消歧、确认或替代。这一步得到规范化的命名实体(Named Entity)。第三步则利用命名实体之间的依赖关系生成SPARQL查询语句。为了提高准确率,系统会生成一些备选查询(candidate query),用户从中选择能够准确表达自己需求的查询。

基于自然语言(NL)的SPARQL Endpoint服务是当前研究的热点,也是今后发展的方向。用户不再需要了解RDF、SPARQL,也不需要知道知识库有什么内容以及这些内容是怎样联系到一起的,只需要把自己的需求用平白的语句描述出来即可。但是,由于人工智能领域对自然语言的识别与处理本身还不完善,因此,这类服务当前最大的问题就在于对复杂的用户需求表达尚不能非常准确地分析并构造查询。针对这一情况,在实际用户查询的构建过程中,系统都通过一些措施对用户表达进行某种引导、限制或通过与用户交互等方式尽可能地准确构造语义查询。以Querix为例。它在用户的提问方式上进行了一些限制,要求用户要以特定的词汇开始提问,例如what、how many、give me等等。另外,根据

需求生成一系列SPARQL查询语句之后,提供了与用户交互的过程,让用户选择最符合需要的查询。如图4。

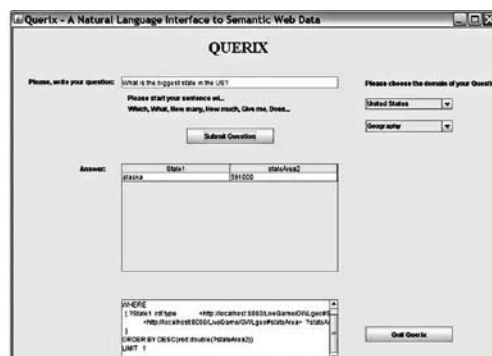


图4 Querix SPARQL Endpoint提问界面^[10]

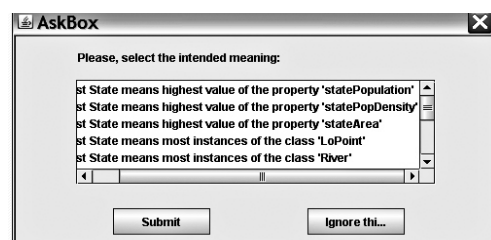


图5 Querix SPARQL Endpoint交互界面^[10]

前面介绍的四种类型的SPARQL Endpoint服务,就需求表达能力来说,无疑是第一种纯粹基于SPARQL语言的服务最强。但其用户可用性最低。后面三种服务类型,虽然表达能力稍有降低,但是在需求表达的可用性上却逐次递增,为更多的用户所青睐。对于SPARQL Endpoint服务来说,通过多种方式获得用户需求表达之后,下一步就需要提供良好的查询结果组织界面,把语义网的信息组织特点完美地展现给用户。

3.4 SPARQL Endpoint需求交互方式总结

由此可见,四种SPARQL Endpoint用户需求交互在可表达性和易交互性方面各自具有不同的特征。以表格表示如下:

表中1、2、3、4等表示特定需求交互方式在可表达性能力和易交互性方面的能力。数字越小表示其这类能力越强大,反之亦然。第二行斜线左侧的数字表示可表达性能力,右侧数字表示易交互性的能力。由此不难看出,基于自然语言的SPARQL Endpoint用户需求交互方式由于其可在表达性和易交互性方面的均衡发展,使得其成为当前SPARQL Endpoint的热门与今后的发展方向。

表1 SPARQL Endpoint可表达性与易交互性比较

交互方式 指标	纯粹SPARQL QUERY交互	模板提示型 需求交互	基于关键词 需求交互	基于自然语 言需求交互	交互方式 指标
可表达性	1 4	3 3	4 1	2 2	易交互性

4 查询结果数据的呈现方式

不同类型的SPARQL Endpoint对查询结果的组织方式也大相径庭。好的数据呈现方式不仅能够清晰、直观地将数据展现在用户眼前，而且还为用户进一步的应用给出了提示与安排，无疑会提高用户体验的满意度。当前SPARQL EndPoint服务通常会使用这样一些技术对查询结果进行组织：排序、分页、数据导出、分面（Facet）、可视化（Visualization）、Mashup等。

4.1 基本类型数据呈现方式

这种情况占据了当前SPARQL Endpoint服务用于组织查询结果的大多数。其体现出来的特征就是基本不对查询结果进行组织。结果数据通过简单的TABLE、TR、TD标签进行排列；对页面颜色、排版、字体、字号等美化元素基本忽略。缺省情况下，系统根据具体查询语句来确定是否对结果进行排序。语句中没有order by就不会对结果进行排序，按照索引的顺序进行显示。对查询结果有时会提供分页功能或通过“更多”予以提示，但很少会有1、2、3、4、5这样的页码跳转功能。几乎所有系统都会提供不同格式的数据（RDF、文本、HTML、JSON）显示和导出服务。如下图。这种类型的数据组织方式基本不是面向普通用户，而是为机器用户的数据采集提供查询语句的验证环境。

Concept
http://www.w3.org/2004/02/skos/core#Concept
http://xmlns.com/foaf/0.1/Person
http://schema.org/Person
http://wikidata.dbpedia.org/resource/Q215627
http://www.w3.org/2002/07/owl#Thing
http://wikidata.dbpedia.org/resource/Q5
http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Agent
http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#NaturalPerson
http://dbpedia.org/ontology/Agent
http://dbpedia.org/ontology/Athlete
http://dbpedia.org/ontology/Person
http://dbpedia.org/ontology/TennisPlayer
http://schema.org/CreativeWork
http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#InformationEntity
http://dbpedia.org/ontology/Software
http://dbpedia.org/ontology/VideoGame
http://dbpedia.org/ontology/Work

图6 DBpedia SPARQL Endpoint查询结果

4.2 基于分面 (Facet) 的数据呈现

分面是指事物的多维度属性。例如一本书的信息就包含主题、作者、出版机构、出版年代等分面。基于分面的数据呈现把检索与分面浏览两种功能结合在一起，让用户可以通过选择或者修改具体分面的参数值来对结果进行选择 and 过滤，用户的选择和挖掘空间变大了。典型的分面检索浏览系统有DBpedia、Faceted Wikipedia Search System、Freebase Parallax和AutoSPARQL。下面以DBpedia的分面检索系统为例进行介绍。



图7 DBpedia分面检索示意图^[11]

图中用户的需求是“Which rivers flow into the Rhine and are longer than 50 kilometers?”（流入莱茵河且长度超过50公里的河流）。页面右侧3处表示系统对需求的解析结果，包括主干词的获取以及命名实体的匹配；4处表示根据这一分析结果查询到的数据信息；而页面左侧2处则表示根据用户需求分析出来的分面信息：信息类型、河流入口、长度以及名称。而水面面积则是系统根据“河流”这一概念给出的另外一个属性分面。可以想象，系统还可以给出一些诸如“发源地”、“流经国家”等更多的分面。至此，用户得到了第一批数据结果。如果希望修改河流名称或长度等内容，可以对左侧的相应分面重新赋值。

分面信息的获取,是在系统对用户需求分析的基础上得到的。为了让这个结果更加准确,AutoSPARQL提供了另外一种方式。如下图所示,系统在初步分析用户需求的基础上,提出一些问题让用户选择yes或no,从而逐步获得用户的真正需求,形成检索模板。如图8。

```

PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?band ?label ?homepage ?genre WHERE {
  ?band a dbo:Band .
  ?band rdfs:label ?label .
  OPTIONAL { ?band foaf:homepage ?homepage } .
  ?band dbo:genre ?genre .
  ?genre dbo:instrument dbpedia:Electric_guitar .
  ?genre dbo:stylisticOrigin dbpedia:Jazz .
}
ORDER BY ?label LIMIT 100

```

- search for “bands with a genre which mixes electric guitars and Jazz”
- resource: dbpedia:Faols answer: YES
- resource: dbpedia:Hot_Chip answer: NO
- resource: dbpedia:Metalwood answer: YES
- resource: dbpedia:Polvo answer: YES
- resource: dbpedia:Uzric_Tentacles answer: YES
- resource: dbpedia:New_Young_Pony_Club answer: NO
- select “genre” as property to return and “homepage” as additional property
- click on “label” column head to order by it and adjust limit

图8 AutoSparql分面检索界面^[12]

4.3 图形化呈现数据方式

基于图形化的数据组织是将查询结果通过图形直观地表示出来。检索结果或者显示多个实体间的关系,或是显示一个实体的属性等内容。当前应用比较广泛的图形化SPARQLEndPoint工具有SILK、RelFinder、LODLive、LOV (Linked Open Vocabularies)、InPhO等。RelFinder是一款查询实体间关系,并以图形化显示关联信息的工具。下图显示了“World Wide Web Consortium”、“Resource Description Framework”以及“Tim Berners-Lee”三个实体在DBpedia中的关系。黑色线条表示用户当前选中的关系。

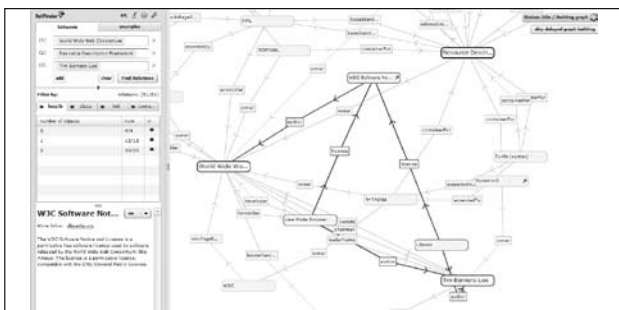


图9 RelFinder SPARQL Endpoint工作示意图^[13]

LODLive、LOV则是类似于关键词的检索方式。以用户输入的关键词为中心显示它的各种属性以及与

其他实体的关系。

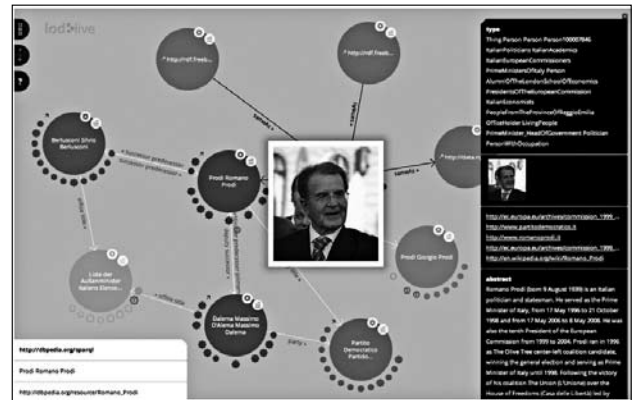


图10 LODLive 检索结果可视化示意图^[14]

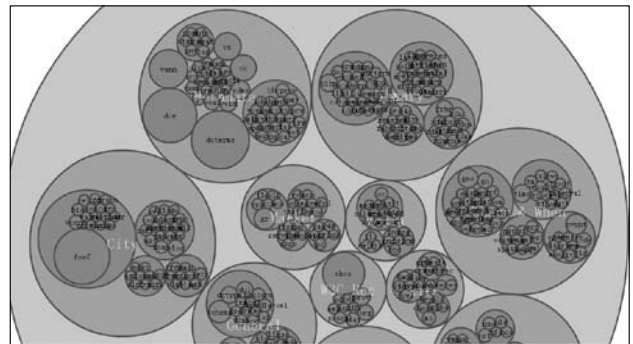


图11 LOV可视化示意图^[15]

图形化工具的优点毋庸置疑,通过点和线把实体之间的复杂关系清晰、直观、完整地显示出来。另外,对于不同领域、不同类别的知识可视化方案和工具很多,用户接受起来没有任何负担。因此,在语义网应用中,基于图形化的方式来组织、展现数据的案例越来越多。

4.4 Mashup

当前大部分SPARQL Endpoint服务都只提供本地知识库里的信息,极少去把与用户需求相关的其他数据集信息进行整合与显示,即Mashup。基于Mashup的数据组织方式能够极大地扩展用户的视野,是当前可用性最好的一种数据呈现方式。以联合国粮农组织的AGRIS为例。严格说AGRIS并不是一个纯粹意义上的SPARQL Endpoint服务,而是一种汇集多个SPARQL Endpoint数据的整合服务。AGRIS分别从IFPRI (International Food Policy Research Institute)、Nature、DBpedia、Global Biodiversity Information Facility以及Biodiversity

International等多个SPARQL Endpoint以及本体中查询和抽取数据^[16],并利用类似于Exhibit的技术将这些数据整合后有条不紊地展现给用户。



图12 AGRIS查询结果mushup示意图

4.5 SPARQL Endpoint数据呈现方式总结

毫无疑问, RDF语义数据有一定的特殊性, 因此对于其查询结果的信息呈现技术与方式就显得更加重要。上述这些技术应用得越多, 组织越协调的系统, 也就越能把用户长久地留在服务中。下面以图表来说明各项技术在当前SPARQL Endpoint服务中的普遍程度或者说实现的容易程度。

表2 SPARQL Endpoint数据呈现技术应用比较表

排序	分页	不同RDF格式显示	结果下载	分面	图形可视化	MASHUP
1	1	1	1	2	3	4

表格中的数字表示该项技术在SPARQL Endpoint查询结果呈现的应用普遍程度。数字越小表示应用得越普遍, 反之表示这项技术越复杂, 越不容易应用到数据呈现中。但是也表示这项技术会给用户带来越好的用户体验。

5 系统功能性指标

如果说SPARQL Endpoint用户需求表达方式以及查询结果的信息呈现方式偏向于外部用户界面, 那么SPARQL Endpoint的可发现性、服务能力则更多地偏向于内部功能性指标。这一类指标通常无法直接通过用户界面获得。

随着大量SPARQL Endpoint工具的发布, 让用户

感到困惑的首先是每个SPARQL Endpoint究竟包括了什么类别的数据、数据量有多少、数据是由什么机构发布的、权威性如何、第三方应用许可协议是什么等等。可发现性指标指的是SPARQL Endpoint是否以及如何把这些信息向用户予以公开。当前用于发布SPARQL Endpoint服务概况的方式主要有两种。一种是通过VOID词表; 另外一种则是通过W3C SPARQL 1.1中的Service Description (SD)。国外专业机构对DataHub上的430个SPARQL Endpoint进行了扫描, 结果只有14.15%的SPARQL Endpoint提供了SD服务描述说明; 13%的SPARQL Endpoint提供了基于VOID的描述服务。大英图书馆基于SPARQL 1.1 Service Description提供了自己的服务描述。



图13 大英图书馆SPARQL Endpoint服务描述^[17]

但是比较遗憾的是, 基本没有见到SPARQL Endpoint服务把这其中一些可以用来揭示自己服务内容信息放到服务页面上供用户了解。这些信息更多的是提供给机器用户采集、分析或者以用户很难理解的RDF格式进行描述。这样就给用户了解和使用SPARQL Endpoint带来了障碍。

服务能力指的是SPARQL Endpoint对SPARQL查询语言的支持能力。随着用户需求的变化, SPARQL查询语言的内容也不断丰富和扩展。2013年, W3C发布了SPARQL 1.1版本, 增加了联合查询、更新RDF数据、管理RDF图等多项功能。为了适应这一情况, 各机构的SPARQL Endpoint也逐步开始升级, 全面或部分支持SPARQL 1.1的功能。

6 结论

系统可用性在语义网发展过程中作为“非核心”问

题曾经一度被忽略。SPARQL Endpoint服务作为语义网特有的一种服务方式,其发展水平从某种程度上也代表了语义网的发展状况。语义网的用户包括两类:机器用户与普通用户。SPARQL Endpoint则要在这两类用户提供服务。对于机器用户,SPARQL Endpoint只需要提供基于RESTFUL的数据流信息。而对于普通用户,从Web 2.0时代起就对用户体验方面有着越来越高的要求。因此,如何更好地提供良好的系统可用性,满足普通用户在查找语义网数据过程中的各种应用需求是当前SPARQL Endpoint也是语义网发展极其重要的课题。

参考文献

- [1] Introducing the Knowledge Graph: things, not strings.[EB/OL].[2014-09-22]. <http://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html>.
- [2] DBpedia Version 2014 released.[EB/OL].[2014-09-22].<http://blog.dbpedia.org/2014/09/09/dbpedia-version-2014-released/>.
- [3] Andraz T, Zemanta User Interfaces for Semantic Web: Do They Have to Be Ugly?.[EB/OL].[2014-10-01].<http://www.slideshare.net/andraz/semtech2010-do-semanticwebuserinterfaceshavetobeugly>.
- [4] Duane Degler, Semantic Web Interface Design: Opportunities and Techniques.[EB/OL].[2014-08-03].http://www.designforcontext.com/files/dd_design_semtech_2010-06-22.pdf.
- [5] State of the LOD Cloud.[EB/OL].[2014-07-09].<http://lod-cloud.net/state/>.
- [6] SPARQL1.1 Service Description.[EB/OL].[2014-09-05]. <http://www.w3.org/TR/sparql11-service-description/>.
- [7] SPARQL Query Basics and Federated Queries.[EB/OL].[2014-07-05]. http://www.slideshare.net/dunken69/sparql-basic-and-federated-queries?qid=1f3ec141-f91b-4e1e-949c-38a3f2f56775&v=qf1&b=&from_search=1.
- [8] TSO flim SPARQL Endpoint.[EB/OL].[2014-10-05]. <http://openuplabs.tso.co.uk/demos/sparqleditor>.
- [9] Falcons Web Site.[EB/OL].[2014-08-03].<http://ws.nju.edu.cn/falcons/objectsearch/index.jsp>.2014-10-11.
- [10] Querix SPARQL Endpioit Interface.[EB/OL].[2014-08-03]. <https://files.ifi.uzh.ch/ddis/oldweb/ddis/research/talking-to-the-semantic-web/querix/>. 2014-07-02
- [11] 同10
- [12] DBpedia Facet Search Interface. [EB/OL].[2014-06-22]. <http://wiki.dbpedia.org/FacetedSearch>.
- [13] Jens Lehmann,LorenzB uhmnn. AutoSPARQL:Let Users Query Your KnowledgeBase.[EB/OL].[2014-08-03].http://jens-lehmann.org/files/2011/autosparql_eswc.pdf.
- [14] RelFinder Web Site.[EB/OL].[2014-10-12].<http://www.visualdataweb.org/relfinder.php>.
- [15] Lodlive Web Site.[EB/OL].[2014-09-25].<http://en.lodlive.it/>.
- [16] Linked Open Vocabularies (LOV). [EB/OL].[2014-08-11].<http://lov.okfn.org/dataset/lov/>.
- [17] Agris:How it works.[EB/OL].[2014-10-12].<http://agris.fao.org/content/how-it-works>.
- [18] British Library SPARQL Service Description.[EB/OL].[2014-08-25]. <http://bnb.data.bl.uk/sparql>.

作者简介

姜恩波, 1972年生, 中国科学院成都文献情报中心副高级工程师, 研究方向: 网络信息组织、关联数据, E-mail: jiangeb@clas.ac.cn。

致谢: 本文是于2014年在美国做访问学者期间完成的。在此过程中, 我的导师——肯特州立大学图书馆学与信息科学学院曹蕾教授 (<http://www2.kent.edu/slis/people/~mzeng/>) ——给予了悉心的指导与帮助, 在这里特表示衷心感谢。

End User-Oriented SPARQL Endpoint Technology and Information Presentation

JIANG EnBo

(Chengdu Documentation and Information Center, CAS, Chengdu, 610041, China)

Abstract: SPARQL Endpoint, a service occurred in Semantic Web era, has presented its specific characters since the concept of Semantic Web had been put forward. After a brief introduction of the deficiency, background and operating principle of SPARQL Endpoint, the article compared and analyzed the current service characteristics from three aspects below: user request interaction, information presentation of query results and the system functional indicators and summarized their specific characteristics.

Keyword: SPARQL Endpoint; User interaction; Information presentation; Discoverability

(收稿日期: 2014-11-11)