

The Application Research of Ontology Parsing and Inquiring Based on Jena^{*}

Li Zhu¹, Qing Yang²

¹Department of Public Basic Courses, Wuhan Conservatory of Music, Wuhan

²Department of Computer Science, Huazhong Normal University, Wuhan

Email: zhuliwuhan@yahoo.com.cn, yangq@mail.ccnu.edu.cn

Received: Jul. 5th, 2012; revised: Jul. 19th, 2012; accepted: Jul. 27th, 2012

Abstract: As the next generation of the Internet, the semantic web provides strong capabilities of data organization and effective information retrieval. How to parse and inquire the ontology is the key technique for the semantic web to achieve intelligent semantic retrieval. The paper gives a deeply research of the ontology parse and inquire method based on Jena, and be used in the compiler principle course ontology.

Keywords: Jena; The Semantic Web; Ontology Parse; Ontology Inquire

基于 Jena 的本体解析与查询应用研究^{*}

朱丽¹, 杨青²

¹武汉音乐学院公共基础课部, 武汉

²华中师范大学计算机科学与技术系, 武汉

Email: zhuliwuhan@yahoo.com.cn, yangq@mail.ccnu.edu.cn

收稿日期: 2012 年 7 月 5 日; 修回日期: 2012 年 7 月 19 日; 录用日期: 2012 年 7 月 27 日

摘要: 语义网作为下一代互联网, 具有强大的数据组织和高效的信息检索能力。如何对本体进行解析和查询是语义网实现智能语义检索的关键技术。文章在 Jena 基础上深入研究了本体解析与查询方法, 并对构建的《编译原理》课程本体进行了解析和查询。

关键词: Jena; 语义网; 本体解析; 本体查询

1. 引言

近年来, 网络数据迅猛增长, 信息量急剧扩大, 然而数据的丰富性与知识的贫乏性之间的矛盾日益加剧。语义网的出现为计算机提供了可理解的语义信息环境, 而基于语义网的智能语义检索的提出为解决信息的查全率与查准率提供了契机。实现智能语义检索的关键问题是本体解析和查询。目前, 国内外对本体解析和查询的相关研究相对有限, 更多关注于本体

构建、本体映射及本体概念语义相似与相关度的研究, 而对 Jena 的研究也只着重于其推理机制。

文献[1]依照软件工程原理, 提出了一种基于 Jena 的循环式本体构建五步法, 解决了本体构建方法不清晰、本体交互性弱问题。文献[2]给出将 Jena 推理机推理出的结果加入知识库, 实现对本体模型的语义检索。文献[3]提出基于 Jena 的一种提取和处理 RDF 层本体处理的方法。针对如何得到更合理的检索结果, 文章在 Jena 的基础上深入研究了.OWL 本体解析和查询方法, 并对构建的《编译原理》课程本体验证了解析和查询方法的有效性。

^{*}资助信息: 国家社会科学基金资助项目(07BY033); 湖北省科技攻关计划基金资助项目(2007AA101C49)。

2. Jena 功能结构

2.1. Jena 框架功能

基于 Java 框架的语义网构建的 Jena^[4]由 HP 公司语义网小组开发, 是用于创建语义 Web 应用系统的 Java 软件包插件。提供了广泛开源的面向 RDF、RDFS、OWL 和 SPARQL 的 Java 库以及基于 RDFS 和 OWL 本体的推理引擎, 能实现对本体模型存储文件的读取操作。由于 Jena 在数据库支持方面的可靠性和持续稳定性, 其一直是各类应用系统开发过程中的首选工具^[3]。Jena 主要框架结构和功能包括:

- 1) RDF 应用程序接口: 能够读写、处理 XML 中的 RDF 数据;
- 2) 本体应用程序接口: 提供对 RDF 和 OWL 本体模型数据的操作和处理。
- 3) 基于规则的推理引擎: 提供对 RDF 和 OWL 本体模型的一致性检测及类、属性和实例的推理, 用户可以自行开发推理规则。
- 4) 存储机制: 允许大量 RDF 本体文件持久化存储到硬盘。
- 5) 查询引擎: 提供的最新的 SPARQL 能对本体模型中的语义数据信息进行检索。
- 6) 服务机制: 通过各种协议, 能够将 RDF 数据发布到其他应用程序中。

2.2. Jena 体系结构

Jena 具有强大的数据存储和操纵功能, 能实现本体存储文件和数据库的交互性, 并解决本体解析和查询问题, 其体系结构如图 1 所示。

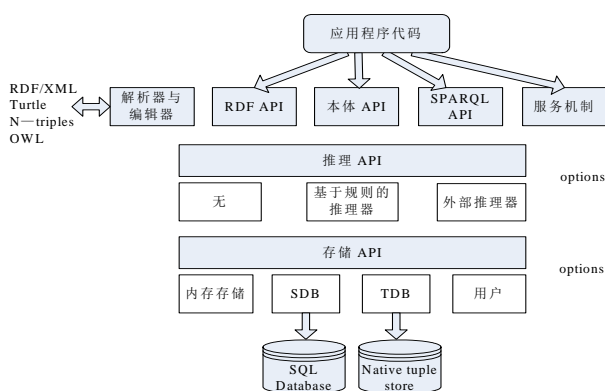


Figure 1. The framework of Jena
图 1. Jena 体系结构

3. 基于 Jena 的本体解析与查询

本体实现了语义层次上网络信息的共享与交换, 是基于语义网的智能信息检索系统的信息组织框架。本体构建完成后, 如何对其进行解析和查询直接关系到信息的获取和检索^[5]。

3.1. 本体解析接口

Jena 包含了大量的应用程序接口和核心数据结构, 主要由 API 和 SPI 两大部分组成, API 提供给用户编程使用, SPI 则为 Jena 提供操作的核心数据结构。Jena^[6-8]支持的持续存储与解析功能由大量的工具包进行调用和处理, 各个工具包又以接口方式进行定义。其中 com.hp.hpl.jena.rdf.model 工具包能识别资源描述框架语言, 提取数据相关的语义信息; com.hp.hpl.jena.ontology 工具包包含大量访问和操纵 RDF、OWL 和 DAML + OIL 本体文件的抽象类方法, 两者均能很好地实现对本体模型中数据的解析。常用的本体解析接口与功能如表 1 所示。

Table 1. Common interface and function of ontology parse
表 1. 常用的本体解析接口与功能

接口	功能
ModelFactory	创建各种标准的本体模型(主要针对本体建模语言、存储方式和推理器), 以实现本体文件的读取
OntDocumentManager	每个本体模型都具备, 协助管理本体文件、导入本体文档以创建本体模型和下载网络上本体
Model	创建资源、属性和值, 增加和删除模型中的陈述以及读取和连结本体模型
OntModel	能获取及操作本体数据
OntClass	定义概念之间、概念和实例间关系(如子类、超类、等价类和不相交类)以及返回能够遍历的迭代器
OntProperty	定义属性之间、属性特性和约束(如定义域和值域), 返回能够遍历的迭代器以及进行相关逻辑判断
DatatypeProperty ObjectProperty	定义属性类的值域是数据对象(实例对象)
Individual	实例定义和相关操作
Restriction	定义属性约束值和相关基数约束(最小、最大基数)

了解了以上本体接口和功能后，便可很容易地使用 Jena 工具包对本体模型进行解析。

3.2. 本体查询接口与语法

Jena^[9,10]支持的查询与推理功能由信息搜索语言 SPARQL 实现。SPARQL^[11-13]作为一种访问 RDF 文档的查询语言和协议，在 Jena 查询引擎 ARQ 的支持下，能对本体模型中的数据信息进行查询。ARQ 查询引擎由 com.hp.hpl.jena.query 工具包进行调用和管理，该工具包能满足 SPARQL 查询语句所遵循的语法，文章以图 2 所示的 RDF 三元组语义数据信息来描述 SPARQL 查询语法。

要获得查询结果，必须首先完成 Jena AQR 运行环境的参数设置，设置代码如下：

```
$ export ARQROOT=~/.ARQ
$ chmod +rx $ARQROOT/bin/*
$ export PATH=$PATH:$ARQROOT/bin
$ sparql
```

Usage: [--data URL] [exprString | --query file]

然后使用 Jena API 执行查询语句，常用的查询接口与功能如表 2 所示。

SPARQL 查询语句的语法规则如下：

```
PREFIX foaf: http://xmlns.com/foaf/0.1/
SELECT ?NoID ?url ?name
WHERE{
    ?NoID foaf:name "John Smith" .
    ?NoID foaf:weblog ?url .
    ?NoID foaf:name ?name .
    FILTER regex(?name, "J", "r")
}
```

其中，PREFIX 为定义 foaf 的域名空间，?NoID、?url 及 ?name 是查询的变量，FILTER regex(?name, "J", "r")

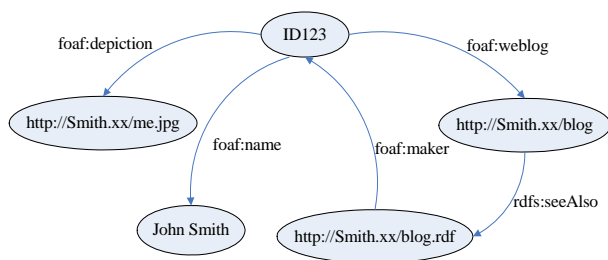


Figure 2. The semantic data information of RDF triple
图 2. RDF 三元组语义数据信息

Table 2. Common interface and function of ontology inquire
表 2. 常用的本体查询接口与功能

接口	功能
QueryFactory	创建由文件或字符串组成的文本查询语句，返回查询结果对象
QueryExecution	为本体模型构建查询执行类对象
ResultSetFormatter	输出查询结果

是查询筛选条件，表示查找包含字母“J”或“r”的姓名(筛选条件也可以是表达式，如 FILTER(?age>=30))。其查询返回的结果如下：

NoID	url	name
ID123	http://Smith.xx/blog	John Smith

此外，SPARQL 也支持更为复杂的查询，如 OPTIONAL、UNION，以及定义查询结果显示，如 DISTINCT、LIMIT(显示 N 条结果)、OFFSET(显示前 N 条结果)和 ORDER BY。其语法规则如下：

```
SELECT ?name ?depiction
WHERE{
    ?NoID foaf:name ?name .
    OPTIONAL {
        ?NoID foaf:depiction ?depiction .
    }
}
```

ORDER BY DESC(?name)LIMIT 3

或者使用：{查询条件 1}UNION{查询条件 2}。

4. 实现技术

4.1. 代码实现

要对本体进行解析和查询，必须先在领域专家的指导下构建相关本体^[14,15]。本文已预先构建好了《编译原理》课程本体，并以.Owl 文本文件的形式存储，对该本体应用 Jena 工具包进行解析和查询，部分程序代码实现如下：

1) 在内存中建立本体模型，并读取本体文件。

```
OntModel
ontModel=ModelFactory.createOntologyModel(OntModelSpec.Owl_DL_MEM);
ontModel.getDocumentManager().addAltEntry("http://www.onto.cn/bianyiyi", "file:H/BT/BianYi_Ontology.owl");
ontModel.read("http://www.onto.cn/bianyiyi");
```

其中 OntModelSpec 为本体模型的配置标准，OWL_DL_MEM 表示本体建模语言为 OWL DL，无推理器，内存存储。

2) 获取本体中所有概念、概念间的继承关系及实例。

```
for(Iterator i=(Iterator)ontModel.listClasses();i.hasNext());{
    OntClass c=(OntClass)i.next();
    If(!c.isAnon())
    {
        //获取所有概念和子概念
        System.out.println("class: "+c.getLocalName());
        for(Iterator j=c.listSubClasses();j.hasNext();){
            OntClass subc=(OntClass)j.next();
            if(!subc.isAnon())
            {
                System.out.println("has subclasses: "
                    +subc.getLocalName());
            }
        }
        //获取所有实例
        for(Iterator k=c.listInstances();k.hasNext();){
            Individual inst=(Individual)k.next();
            if(!inst.isAnon()){
                System.out.println("has individuals:"
                    +inst.getLocalName());
            }
        }
    }
}
```

3) 查询本体模型中相关的语义数据信息。

```
InputStream in = new FileInputStream(new
File("BianYi_Ontology
.owl"));
String queryString =
"PREFIX dc: < http://www.onto.cn/bianyi /dc> " +
"SELECT ?concept ?url ?eul " +
"WHERE { " +
" ?concept dc:identify \"正规集\" . " +
" ?concept dc:consist ?url . " +
" ?url dc:equal ?eul . " +
"}";
Query query = QueryFactory.create(queryString);
```

```
//执行查询并获得结果
QueryExecution qe = QueryExecutionFactory.create
(query, model);
ResultSet results = qe.execSelect();
//输出查询结果
ResultSetFormatter.out(System.out, results, query);
qe.close();
```

4.2. 测试结果

通过 Jena 工具包所提供的类和方法，能很好地对本体模型进行解析和查询。文章使用软件开发工具 Eclipse SDK3.5.2 中引入 Jena2.5.2 工具包，对《编译原理》课程本体进行解析的部分显示结果为：

```
class:文法的二义性
has individuals(1):文法二义性的消除
has individuals(2):文法二义性
class:单词符号
has subclasses(1):正规式与正规文法转化方法
has subclasses(2):正规集
has subclasses(3):正规文法
has subclasses(4):正规式
class:文法的分类
has individuals(1):文法__正规文法
has individuals(2):文法__3 型文法
has individuals(3):文法__2 型文法
has individuals(4):文法__右线性文法
has individuals(5):文法__上下文有关文法
has individuals(6):文法__0 型文法
has individuals(7):文法__无限制文法
has individuals(8):文法__左线性文法
```

此外，调用查询语句输出的部分显示结果为：

```
有穷自动机 dc:identify 正规集
正规式 dc:consist 正规文法
左递归 dc:consist 递归规则
右递归 dc:consist 递归规则
句柄 dc:consist 句型
规范规约 dc:consist 规约
规则 dc:equal 产生式
规范推导 dc:equal 最右推导
最左规约 dc:equal 规范规约
```

最左直接短语 dc:equal 句柄

语法树 dc:equal 推导树

确定有穷自动机 dc:equal DFA

非确定有穷自动机 dc:equal NFA

预测分析法 dc:equal LL1 分析法

规范规约分析法 dc:equal LR 分析法

在查询结果中, Jena 推理引擎帮助获得更多更详细的语义信息。实验数据结果显示, 将 Jena 工具包引入到本体模型领域中, 能获得较好的解析和查询结果。

5. 总结

语义网的发展为语义信息检索系统提高信息处理的准确性和有效性带来了契机, 由于本体能提供明确的语义, 解决网络资源信息的共享和交换, 因而本体被作为基于语义网的智能语义检索系统实现的关键技术。文章在 Jena 的基础上对本体的解析与查询做了深入研究, 并在构建的《编译原理》课程本体中说明了方法的实现。文章具有一定的实用性, 希望对本体研究学者带来一些帮助。此外, 如何使用 Jena 中的推理引擎获得更多信息也是下一步继续研究的工作。

参考文献 (References)

- [1] 向阳, 王敏, 马强. 基于 Jena 的本体构建方法研究[J]. 计算机工程, 2007, 33(14): 59-61.
- [2] 梁晔, 刘宏哲. 运用 Jena 对本体模型进行推理及其应用[J]. 北京联合大学学报, 2009, 23(3): 23-27.
- [3] 盛秋艳, 印桂生. 基于 Jena 的动态语义检索方法[J]. 计算机工程, 2009, 35(16): 62-64.
- [4] Jena. A semantic web framework for Java. <http://jena.sourceforge.net>
- [5] 王继东, 张瑜, 李娜. 基于本体的语义检索技术研究与实现[J]. 计算机技术与发展, 2009, 19(10): 134-137.
- [6] 耿科明, 袁方. Jena 推理机在基于本体的信息检索中的应用[J]. 微型机与应用, 2005, 10: 62-64.
- [7] 李刚, 钱省三, 叶春明. 基于 Jena 的知识检索系统的优化研究[J]. 微计算机信息, 2011, 8: 23-24.
- [8] 吴振生, 孙秀迪, 李新云等. 基于本体推理在行业信息化知识库中的应用[J]. 计算机工程, 2008, 34(18): 59-61.
- [9] 严武军. 基于 Jena 规则推理数字图书馆信息检索系统研究[J]. 电脑开发与应用, 2010, 2: 40-42.
- [10] 王晓慧, 罗军, 余淑良. 本体查询与推理研究[J]. 计算机技术与发展, 2012, 22(5): 130-133.
- [11] D. Vallet, M. Fern and P. Castells. An ontology-based information retrieval model. Heraklion: Proceeding of the 2nd European Semantic Web Conference, 2005.
- [12] P. Eric, S. Andy and P. L. Hewlett. Ssparql query language for RDF. <http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115>
- [13] P. McCarthy. Search RDF data with SPARQL. <http://www.ibm.com/developerworks>
- [14] 陆建江, 张亚非, 苗状等. 语义网原理与技术[M]. 北京: 科学出版社, 2007: 89-130.
- [15] 甘健侯, 姜跃, 夏幼明. 本体方法及其应用[M]. 北京: 科学出版社, 2011: 24-56.