

# 本体的查询与推理研究

王晓慧<sup>1</sup>, 罗 军<sup>1</sup>, 余淑良<sup>2</sup>

(1. 重庆大学 计算机学院, 重庆 400030;

2. 重庆海事局通信信息中心, 重庆 400030)

**摘 要:** 现今, 计算机网络被广泛应用于生活的方方面面, 而从海量的信息中搜寻出人们所需要的还存在诸多问题, 于是产生了本体的概念。而本体的查询和推理是基于本体的应用中重要的组成部分, 研究的目的是为了使知识得以充分表达并且对信息的查询更加精确、完备。首先介绍了本体的概念并建立本体模型, 然后用本体查询语言 SPARQL 对已有模型进行查询并用 SWRL 对模型进行语义规则的扩充; 最后介绍了 Jena, 并对本体模型进行推理, 由此获得了更多知识。结论就是, 在利用 SPARQL 和 Jena 进行查询与推理的过程中, 推理将提高查询能力, 而规则是提高推理能力的关键。

**关键词:** 本体; SPARQL; 查询; SWRL; 语义规则; Jena; 推理

**中图分类号:** TP31

**文献标识码:** A

**文章编号:** 1673-629X(2012)05-0130-04

## Research on Ontology Querying and Inference

WANG Xiao-hui<sup>1</sup>, LUO Jun<sup>1</sup>, YU Shu-liang<sup>2</sup>

(1. College of Computer, Chongqing University, Chongqing 400030, China;

2. Communicated Information Center of Chongqing MSA, Chongqing 400030, China)

**Abstract:** Nowadays, computer network is widely used in various aspects of life. But searching for what people need from the mass information has a lot of questions. Consequently, the concept of ontology has appeared. And then ontology querying and inference is the important component of the application based on ontology. The purposes of this research are to make the knowledge to the fullest expression and querying the information more accurate and complete. Firstly, introduced the concept of ontology and constructed an ontology model. Then used ontology query language, SPARQL, to query the existed model. And also used SWRL to extend the semantic rules. Finally, introduced Jena with which obtained more knowledge by reasoning on the model. Thus concluded, in the process of querying and inference with SPARQL and Jena, inference could develop the ability of querying, and the rules were the key to increase the power of inference.

**Key words:** ontology; SPARQL; query; SWRL; semantic rules; Jena; inference

## 0 引言

计算机网络现已被广泛应用于各个领域, 然而网络内容是适合于人类阅读而不是机器阅读的, 要在海量的信息中找到所需要的还存在诸多问题。Web 创始人 Tim Berners-Lee 于 1998 年 10 月提出了语义 Web<sup>[1]</sup>; 语义 Web 是现有网络的延伸, 并不是独立存在的。信息所包含的意义在语义 Web 环境下可以很好地定义, 这就使得计算机和人能够更好地协作。为了使信息能够很好地加以定义, 人们提出了本体论 (ontology) 概念。目前使用最广泛的是: 本体是共享概念模型的明确的形式化规范说明。

本体语言是用于对本体进行显式的形式化描述。目前 W3C 推荐的本体语言是 OWL, 它是语义 Web 发

展过程中重要的里程碑。然而, 有时本体中复杂的关系难以用本体语言描述清楚, 需要对模型添加语义规则。文中采用 SWRL<sup>[2,3]</sup> 来对本体模型进行扩展, 然后将扩展的语义规则, 加入到本体模型中。语义规则语言 SWRL 是 W3C 规范之一, 利用 SWRL 可将规则以语义方式呈现, 对 OWL 的公理集进行扩展。

而针对本体的查询是最常见的应用之一, 目前 W3C 推荐的查询语言是 SPARQL<sup>[4]</sup>。它是为 RDF 开发的一种查询语言和数据获取协议, 可以用于任何可以用 RDF 来表示的信息资源。SPARQL 构建在以前的 RDF 查询语言 rdfDB、RDQL 和 SeRQL 之上, 并且增添了一些新的特性。它起到了将 Web2.0 和语义 Web 相联系的作用, 是网络数据库的查询语言以及数据获取标准。

然而, 查询语言 SPARQL 只能查询本体模型中已存在的数据, 不能对本体进行推理。这样就限制了人们对所需的查询结果的获取。由此, 本体的推理亦

收稿日期: 2011-10-02; 修回日期: 2012-01-05

作者简介: 王晓慧 (1982-), 女, 辽宁丹东人, 硕士研究生, 研究方向为数据库; 罗 军, 硕士, 副教授, 研究方向为数据库。

是语义 Web 应用中不可或缺的内容之一。文中主要使用 Jena 对本体模型进行推理<sup>[5,6]</sup>。Jena<sup>[7]</sup> 是用于创建语义 Web 应用程序的 Java 框架。它为 RDF, RDFS 及 OWL, SPARQL 提供了编程环境, 并且包含一个基于规则的推理引擎。Jena 是由 HP 实验室开发的开放源代码。

## 1 建立本体模型

现以家庭为例创建一个本体。结构如图 1 所示:

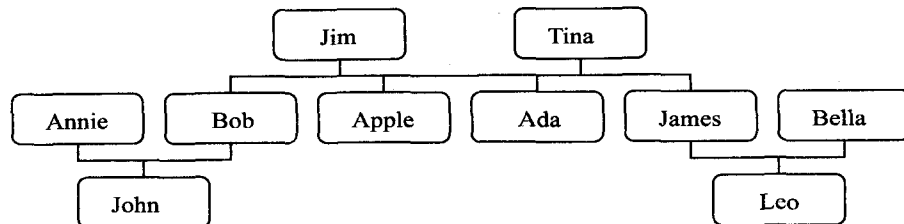


图 1 家庭结构图

首先, 确定概念范围。这里只创建要用到的 hasHusband、hasWife、hasChild、hasGrandchild 四个属性。

使用 Protégé 4 创建本体, 属性和个体代码如下:

```

<!-- http://www.semanticweb.org/ontologies/family.owl#hasGrandchild -->
<owl:ObjectProperty rdf:about="#hasGrandchild"/>
<!-- http://www.semanticweb.org/ontologies/family.owl#hasChild -->
<owl:ObjectProperty rdf:about="#hasChild"/>
<!-- http://www.semanticweb.org/ontologies/family.owl#hasHusband -->
<owl:ObjectProperty rdf:about="#hasHusband">
  <rdf:type rdf:resource="#owl:FunctionalProperty"/>
  <owl:inverseOf rdf:resource="#hasWife"/>
</owl:ObjectProperty>
<!-- http://www.semanticweb.org/ontologies/family.owl#hasWife -->
<owl:ObjectProperty rdf:about="#hasWife">
  <rdf:type rdf:resource="#owl:FunctionalProperty"/>
  <owl:inverseOf rdf:resource="#hasHusband"/>
</owl:ObjectProperty>
<!-- http://www.semanticweb.org/ontologies/family.owl#Bob -->
<rdf:Description rdf:about="#Bob">
  <hasWife rdf:resource="#Annie"/>
  <hasChild rdf:resource="#John"/>
</rdf:Description>
<!-- http://www.semanticweb.org/ontologies/family.owl#Annie -->
<rdf:Description rdf:about="#Annie"/>
<!-- http://www.semanticweb.org/ontologies/family.owl#Jim -->

```

```

<rdf:Description rdf:about="#Jim">
  <hasChild rdf:resource="#Bob"/>
  <hasChild rdf:resource="#James"/>
  <hasWife rdf:resource="#Tina"/>
  <hasChild rdf:resource="#Apple"/>
  <hasChild rdf:resource="#Ada"/>
</rdf:Description>
<!-- http://www.semanticweb.org/ontologies/family.owl#Tina -->
<rdf:Description rdf:about="#Tina">
  <!-- http://www.semanticweb.org/ontologies/family.owl#James -->
  <rdf:Description rdf:about="#James">
    <hasWife rdf:resource="#Bella"/>
    <hasChild rdf:resource="#Leo"/>
  </rdf:Description>
  <!-- http://www.semanticweb.org/ontologies/family.owl#Bella -->
  <rdf:Description rdf:about="#Bella">
    <!-- http://www.semanticweb.org/ontologies/family.owl#Apple -->
    <rdf:Description rdf:about="#Apple">
      <!-- http://www.semanticweb.org/ontologies/family.owl#Ada -->
      <rdf:Description rdf:about="#Ada">
        <!-- http://www.semanticweb.org/ontologies/family.owl#Leo -->
        <rdf:Description rdf:about="#Leo"/>
      </rdf:Description>
    </rdf:Description>
  </rdf:Description>
  <!-- http://www.semanticweb.org/ontologies/family.owl#John -->
  <rdf:Description rdf:about="#John"/>

```

## 2 对模型的查询

现在用 SPARQL 语言来查询本体模型。查询语句:

```
PREFIX family: <http://www.semanticweb.org/ontologies/family.owl#>
```

```
select ? BobWife
```

```
where { family:Bob family:hasWife ? BobWife }.
```

输出结果为:

```
{ BobWife      |
```

```
=====
```

```
| family:Anna |
```

因为模型中显式说明了 Bob 的妻子是 Anna, 所以查询结果为 Anna。

查询语句(这里省略掉前缀说明部分, 下同):

```
select ? AnnaHusband
where { family: Anna family: hasHusband ? An-
naHusband }
```

输出结果为:

```
| AnnaHusband |
```

```
=====
```

可见 SPARQL 没能给出隐含存在的知识。

再查询:

```
select ? AnnaChild
where { family: Anna family: hasChild ?
AnnaChild }
```

输出结果为:

```
| AnnaChild |
```

```
=====
```

由上面的描述可知 Bob 的妻子是 Anna, 孩子是 John。很显然, John 也是 Anna 的孩子, 但查询结果未能给出。

### 3 语义规则的扩充

SWRL 采用 Horn 子句的形式, 将规则与 OWL 知识库相结合, 作为本体的一部分, 和本体推理引擎一起工作。SWRL 的格式包括规则体和规则头, 其含义是, 当规则体中的条件都满足时, 规则头中的条件就一定满足。空的规则体表示永远为真, 即满足所有解释; 空的规则头表示永远假, 即不满足任何一个解释。规则体中的各个原子必须都要满足, 是合取关系。在已经建立的本体模型中, 描述了 hasWife、hasHusband、hasChild 三个属性, 而若直接描述属性 hasGrandchild, 则将失去其语义, 它可通过采用 SWRL 得到<sup>[8]</sup>。其语法为:

```
hasChild( Jim, Bob ) ∧ hasChild( Bob, John ) -> has-
Grandchild( Jim, John )
```

可以将上式中的个体用变量表示, 以使该规则更具有普遍性:

```
hasChild( ? x, ? y ) ∧ hasChild( ? y, ? z ) -> has-
Grandchild( ? x, ? z )
```

这里选中 protégé 的 Reasoner 中的 Pellet, 加入规则, 执行后选中 Jim, 可见:

```
hasGrandchild John
hasGrandchild Sam
```

### 4 Jena 的推理分析

Jena 的推理机制<sup>[9]</sup>如图 2 所示, 它使用 ModelFactory 将数据集和推理机组合起来, 形成一个新的推理模型。通过调用 bindSchema 将推理机与本体数据绑

定来具体化推理机, 具体化的推理机可通过 bind 来绑定不同的实例数据, 而所有的推理过程都在 InfGraph 实现。

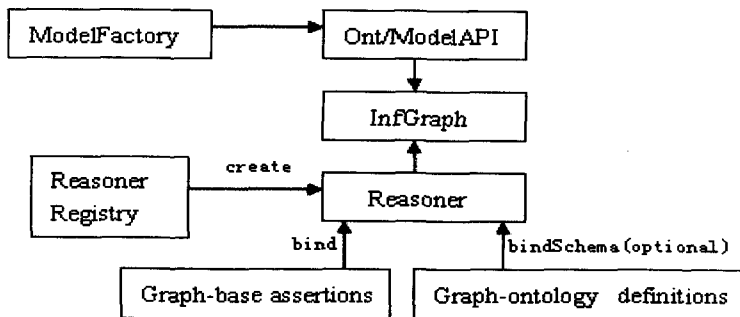


图 2 Jena 推理机制图

前面给出了 hasHusband 和 hasWife 这对 OWL 互逆属性, 现在对推理机添加规则:

```
[ rule1: ( ? pa owl: inverseOf ? pb ) ( ? a ? pa ?
b ) -> ( ? b ? pb ? a ) ]
```

创建该推理模型的关键代码如下:

```
OntModel om = ModelFactory. createOntologyModel
();
```

```
File myfile = new File( " D:/DOWNLOAD/owl-ex-
ample/family. owl" );
```

```
FileInputStream myin = new FileInputStream ( my-
file );
```

```
om. read( myin, null );
```

```
String rules = " [ rule1: ( ? pa owl: inverseOf ? pb )
( ? a ? pa ? b ) -> ( ? b ? pb ? a ) ] ";
```

```
Reasoner myreasoner = new GenericRuleReasoner
( Rule. parseRules( rules ) );
```

```
InfModel myinf = ModelFactory. createInfModel
( myreasoner, om );
```

此时查询 Anna 的丈夫, 查询语句:

```
String queryString = " PREFIX family: < http://
www. semanticweb. org/ontologies/family. owl#> " + " se-
lect ? AnnaHusband " + " where { family: Anna family:
hasHusband ? AnnaHusband } ";
```

输出结果为:

```
| AnnaHusband |
```

```
=====
```

```
| family: Bob |
```

此处应注意的是, 如果仅声明属性 hasWife 的逆属性是 hasHusband, 而没有声明属性 hasHusband 的逆属性是 hasWife, 则查询结果为空。可见 Jena 要求互逆属性对中的两个属性要分别都进行声明。

查询 Anna 的孩子, 添加规则:

```
[ rule2: ( ? a http://www. semanticweb. org/ontolo-
gies/family. owl#hasWife ? b ) ( ? a http://www. seman-
```

ticweb. org/ontologies/family. owl#hasChild ? c) ->( ?  
b http://www. semanticweb. org/ontologies/family. owl#  
hasChild ? c) ]

查询语句:

```
String queryString = " PREFIX family: < http://  
www. semanticweb. org/ontologies/family. owl#>" + " se-  
lect ? AnnaChild " + " where { family: Anna family: has-  
Child ? AnnaChild }";
```

输出结果:

| AnnaChild |

=====

| family: John |

同理查询 Kate 的孩子, 输出结果:

| KateChild |

=====

| family: Lucy |

| family: Lily |

| family: James |

| family: Bob |

可见, 通过对 Jena 添加规则后, 原本体模型中蕴含的知识都可以经过推理得出。

## 5 结束语

文中在创建了家庭的本体模型后, 对模型进行了查询。而查询本身是数据驱动的, 即只能查询存在的数据, 并不具备推理能力。而描述逻辑提供的推理能力是有限的<sup>[10,11]</sup>, 文中使用 SWRL 对本体模型进行了语义规则的扩展, 结合 Protégé 中内置的推理机 Pellet 对其进行推理, 得出更复杂的关系知识。最后使用 Jena 推理机对模型进行推理。在 Jena 推理的过程中, 重点是编写规则, 有针对 OWL 属性的规则, 如 owl: inverseof 等, 也有自定义规则, 这些规则对新知识的获取

起到了关键作用。但 Jena 在推理的过程中在内存等方面的代价较大, 而在大规模本体中, 很难将大量属性之间的关系全部显式声明; 另外, Jena 要与 SWRL 协同工作需要借助 Pellet, 因此, 有必要引入外部推理机来弥补 Jena 在这些方面的不足<sup>[12]</sup>。下一步的研究主要是在 Jena 的基础上借助外部推理机实现大规模本体推理的工作。

## 参考文献:

- [1] Berners-Lee T, Hendler J, Lassila O. The Semantic Web[J]. Scientific American, 2001, 284(5): 34-43.
- [2] SWRL: A Semantic Web Rule Language Combining OWL and RuleML[EB/OL]. 2011-10-01. <http://www.w3.org/Submission/SWRL>.
- [3] 王海林. SWRL 推理规则在平面几何证明中的应用[J]. 计算机技术与发展, 2010, 20(9): 218-226.
- [4] 高志强, 潘越, 马力, 等. 语义 Web 原理及应用[M]. 北京: 机械工业出版社, 2009.
- [5] 王继东, 张瑜, 李娜. 基于本体的语义检索技术研究及实现[J]. 计算机技术与发展, 2009, 19(10): 134-137.
- [6] 李永超, 罗钧曼. 语义 Web 中的本体推理研究[J]. 计算机技术与发展, 2007, 17(1): 101-107.
- [7] Jena-A Semantic Web Framework for Java[EB/OL]. 2011-09-25. <http://jena.sourceforge.net/index.html>.
- [8] 朱创录. 语义 Web 推理的研究与实现[J]. 科学技术与工程, 2010, 10(1): 273-278.
- [9] Jena 2 Inference Support[EB/OL]. 2011-09-30. <http://jena.sourceforge.net/inference/index.html>.
- [10] 韩亚洪, 刘永革. 本体的查询与推理机制研究[J]. 计算机工程与应用, 2005, 41(9): 82-85.
- [11] 王金环, 李宝敏. 基于本体 DL 的语义推理研究[J]. 计算机技术与发展, 2009, 19(11): 94-100.
- [12] 宋晓峰, 唐发根. 基于本体的推理技术的相关研究[EB/OL]. 2011-09-09. <http://www.paper.edu.cn>.

(上接第 129 页)

- [J]. IEEE Signal Processing Letters, 2008(15): 124-125.
- [4] 孙仲康, 周一宇, 何黎星. 单多基地有源无源定位技术[M]. 北京: 国防工业出版社, 1996: 223-226.
- [5] 谢恺, 钟丹星, 邓新蒲, 等. 一种空间时差定位的新算法[J]. 信号处理, 2006, 22(2): 129-135.
- [6] 钟丹星, 邓新蒲, 周一宇. 一种基于 WGS-84 地球面模型卫星测时差定位算法[J]. 宇航学报, 2003, 24(6): 569-573.
- [7] 彭朝阳, 叶浩欢, 柳征, 等. 一种数据残缺时的三星时差定位目标跟踪算法[J]. 航天电子对抗, 2009(5): 7-8.
- [8] So H C, Hui S P. Constrained Location Algorithm Using TDOA Measurements[J]. IEICE Trans on Fundamentals of Electronics, Communications and Computer Sciences, 2003(12): 3291-3293.
- [9] 张健, 李鹏. 改进的无线传感器网络 TDOA 定位算法[J]. 计算机工程与应用, 2010, 46(16): 117-118.
- [10] 谢恺, 周一宇, 薛模根, 等. 基于平方根 UKF 的自由段目标跟踪算法[J]. 国防科技大学学报, 2007(4): 112-113.
- [11] Xiong K, Zhang H Y, Chan C W. Performance evaluation of UKF based nonlinear filtering[J]. Automatica, 2006, 42(2): 261-270.
- [12] Zhao Huibo, Pan Quan. PF-UKF-RJMC MC Approaches for Radar Target-tracking[C]//International Conference on Information Technology and Computer Science. [s. l.]: [s. n.], 2009: 373-374.