基于Ontology的大规模知识库构建技术分析*

□ 洪娜 / 中国科学院国家科学图书馆 北京 100190
中国科学院研究生院 北京 100049
□ 张智雄 / 中国科学院国家科学图书馆 北京 100190

摘要：基于Ontology的大规模知识库系统是语义内容应用的基础。文章介绍了四个有代表性的基于Ontology的大规模知识库系统，分别分析了系统的关键技术、特点和性能，并对它们的性能进行了对比分析，最后分析了当前系统的局限、挑战和趋势，以期对国内数字图书馆知识库建设有所帮助。该文为2008年第9期本期话题“知识抽取”的文章之一。

关键词：知识抽取，Ontology存储，知识库，Ontology推理，Ontology查询，性能对比，数字图书馆
DOI:10.3772/j.issn.1673-2286.2008.09.003

1 引言

随着知识技术的发展，基于Ontology的开发和应用越来越多，大规模语义标注、基于Ontology的信息抽取（OBIE）、元数据管理等任务都面临着如何有效存储海量语义内容的问题，基于Ontology的大规模知识库系统的存储规模、查询效率、推理能力都直接影响到语义内容的应用，当前迫切地需要大规模、高性能、可靠的知识库系统来管理这些知识对象，这将对基于知识库的Ontology应用和知识服务起到重要的作用。然而，描述Ontology的RDF、OWL等语言的灵活表示形式既提升了语义的表现程度，也为语义内容的大规模存储和查询带来了挑战，在大规模知识库构建技术中仍然存在大量的问题需要突破和解决。

基于Ontology的大规模知识库构建技术的研究并不是一个全新的领域，学术界和产业界在技术研究和开发方面已经走过了不短的历程，如Ontology的多种存储机制的研究、特殊存储技术的研究、Ontology与关系数据库的映射方法、描述语言的设计和标准化、查询语言的设计和标准化、查询效率的改进、规则集的设计、描述逻辑推理等。目前的研究重点是在现有的较为成熟的存储机制的基础上，研究查询性能和推理性能的改进方法，并开发多种外部调用接口以促进当前研究成果向产业界的应用。

目前，基于Ontology的大规模知识库构建技术的研究和实践已经取得了一定的成果。典型的研究项目有欧洲IST项目的Sesame系统[1]、Ontotext实验室的OWLIM系统[2]、HP实验室的Jena系统[3]、IBM研究中心的Minerva 和SOR系统[4]、Oracle的RDF store系统[5]、Bristol大学的Redland[6]、Franz Inc公司的AllegroGraph 64-bit RDFStore[7]、Maryland大学MIND实验室的 Kowari Metastore[8]、NSF资助项目OWLJessKB[9]等。还有一些研究项目在常规RDF存储策略的基础上开发了特殊的存储技术，如AKT的3store系统[10]、Lehigh大学开发的HAWK（DLDB-OWL）系统[11]、Cornell大学的Mptstore存储技术[12]、Manchester大学开发的Instance Store系统[13]等。

2 大规模知识库构建技术分析

在构建大规模基于Ontology的知识库时，系统设计追求和实现以下三方面基本目标：

（1）能够在可接受的时间内处理大规模的数据，这包括加载时间和查询时间两重效率的要求，加载操作要求大规模知识库能够在可接受的时间内把数据装入内存，查询操作则要求知识库能够在可

* 本文受国家社会科学基金项目“从数字信息资源中实现知识抽取的理论和方法研究”（05BTQ006）和国家“十一五”科技支撑计划课题“网络科技信息监测与评价”（2006BAH03B05）的资金资助。

接受的时间内返回相关概念、术语及其关联关系；

（2）知识库应当提供一个方便的查询接口，对于用户和应用系统来说能够方便的调用；

（3）知识库应当提供足够的推理能力以支持应用的语义需求。

当前能够支持大规模的Ontology长期存储系统主要采用两种存储技术：

（1）本地存储（文件系统）。本地存储方式的优点是数据加载时间和更新时间短，但是查询速度慢，IBM研究中心的马力研究员提出，在本地存储方式下，三元组数据顺序的变化会导致查询时间减慢10倍（甚至更多）[14]。可见本地存储不适宜用于构建大规模的、高效的本体知识库。

（2）关系数据库存储。关系数据库存储方式的查询性能更好，查询响应时间短，而且关系数据库本身的事务处理、查询优化、存取控制、日志和恢复功能也能为Ontology存储提供性能保障。此外，关系数据库还可以把RDF查询嵌入到非RDF查询的SQL查询语言中，这种联合查询可以有效地提高查询性能。所以对于大规模知识库的构建通常采用关系数据库存储技术。

在关系数据库存储技术中采用的RDF长期存取技术主要有两种：

（1）三元组存储技术，即使用关系数据库把RDF存储为一个三列（triples）的表，分别对应每一个statement的主体、谓词和对象。许多RDF存储系统都是采用这种三元组存储的方法，在此基础上改进的典型方法还有，扩展三元组存储技术，即为三元组存储方法再加两个表（literals table和resources table）分别存储literal和resource，由于literal和resource都只被存储了一次，可以大大节省三元组存储的空间，但是这种方式查询比较耗时，因为查询每个statement都要进行三表的联合查询。多数常用的Ontology存储系统都是采用扩展三元组存储技术，如Sesame，Jena和Oracle RDF store。

（2）二进制存储技术，即为Ontology中的每一个类和属性创建一个表，每个类表存储该类的所有实例，每个属性表存储该属性所有的三元组。如DLDB-OWL。由于表的数量和知识库规模成正比，这种存储方式有其明显的缺点，即当Ontology改变时，数据库需要删除和创建新表，schema的更新十分困难。可见，这种方式也不适用于大规模知识库的情况。

3 典型存储系统分析

目前存在着为各种不同存储目标设计的知识库系统，它们分别采用不同的方法实现。笔者对典型的存储系统进行分析和对比，以期能够为Ontology存储应用项目提供参考。

3.1 Sesame

Sesame是欧洲IST项目On-To-Knowledge[1]的一部分，由Aidministrator Nederland b.v[2]开发，Sesame系统设计的目标是开发一个大规模知识仓储系统，以支持RDF和RDFS信息的长期存储、管理、查询和推理。Sesame系统独立于任何的具体存储底层，它支持多种存储方式、多种查询语言、多种推理方式和多种远程调用方式。对RDF数据的存储提供Sesame-memory和Sesame-DB两种数据存储方式。

（1）系统的关键技术

Sesame系统的主体有四个部分：仓储层、SAIL层、访问层和图形接口。仓储层支持文件存储和抽取不同格式文件的RDF形式。SAIL层在仓储层之上，是用于RDF存储和推理的底层系统，它的目的是独立地支持各种类型的存储和推理，包括内存、文件、数据库。存取层主要用于支持外部调用，支持本地和远程两种访问方式包括，HTTP或RMI，图形接口提供RDF的图形化表示，以辅助应用程序调用时进行细致的RDF操作。Sesame的体系结构如图1所示。

（2）系统性能分析

据Lehigh大学的Yuanbo Guo等人在其文章[15]中的分析，以Lehigh大学的Benchmark数据集LUBM为测试数据集，以数据加载时间和查询响应时间为主要指标评估Sesame系统的性能。

从数据加载时间来看，当加载103，074个实例时，Sesame-Memory花费的加载时间为00:00:13（hh:mm:ss），Sesame-DB花费的加载时间为00:09:02（hh:mm:ss），数据库容量为48,333KB。当加载1,316,322个实例时，Sesame-Memory花费的加载

[1] On-To-Knowledge (IST-1999-10132). See http://www.ontoknowledge.org/
[2] http://www.aidministrator.nl/

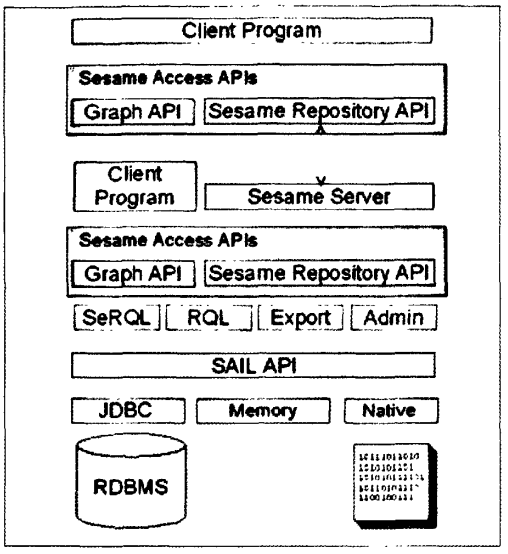

图1 Sesame的体系结构[1]

时间为00:05:40（hh:mm:ss），Sesame-DB花费的加载时间为12:27:50（hh:mm:ss），数据库容量为574,554KB。

从查询响应时间来看，实验分别用不同的查询式进行查询响应时间的测试，Sesame-Memory在处理所有的查询式的速度都是很快的，Sesame-DB在处理一些查询式时响应速度很快，且不受查询结果数据集大小的影响，而一些查询式的处理则很慢，例如，处理那些查询结果为常量的查询式的速度很快，而处理那些没有指定明确的subject或object的URI的查询式，Sesame-DB的查询速度就会明显下降。

综合以上分析，Sesame-DB有一个明显的缺点，当数据规模增大时，加载时间会大幅度增长，例如，当数据规模增大25倍，数据加载时间会增长300倍。而查询响应时间则不稳定，和查询式有关。Sesame-Memory的加载时间短，查询响应时间短，但当实例数量超过一定内存容量时便不能加载。通过实验得出以下结论：一般RDF三元组数量在1百万个以内的数据集适宜采用Sesame-Memory，RDF三元组数量在1百万到三百万之间的数据集适宜采用Sesame-DB。

3.2 OWLIM

OWLIM是一个用Java语言开发的高性能大规模语义仓储系统，OWLIM 构建于Sesame框架之上，它基于Sesame的in-memory存储方式做了性能的改进，被封装为Sesame的SAIL（Storage and Inference Layer）层中。OWLIM本身提供SwiftOWLIM和BigOWLIM两个版本，以满足不同的应用需求。SwiftOWLIM主要面向的是存储规模不大但查询、推理效率要求高的应用场合，是OWLIM的标准版。SwiftOWLIM是一个典型的基于内存的OWL知识库，当系统需要进行查询或者推理等操作时，知识库先将所有的数据读取到内存中再进行进一步的处理。由SwiftOWLIM的存储、查询和推理的方式可知，SwiftOWLIM的优势在于其查询和推理效率。BigOWLIM主要面向的是大规模知识库存储，语义理解程度高，同时效率也比较理想的应用场合，是OWLIM的企业版，BigOWLIM并不将所有的数据读入内存，它的优势在于其所能处理的数据规模，但也通过采取多种优化方法提高同时的查询和推理效率。

（1）系统的关键技术

OWLIM支持大规模知识库的存储、查询和推理，其推理的基础组件是RDF规则-衍推引擎（RDF rule-entailment engine.）TRREE[16]，能够支持OWL DLP、OWL Horst级的推理；OWLIM本身又是Sesame和SAIL API的扩展，Sesame的多种调用方式也能够支持OWLIM的调用，例如，OWLIM支持用户的web图形界面调用、应用程序通过Sesame的API调用或嵌入式调用；OWLIM的TRREE还可以通过ORDI（Ontotext实验室的语义数据集成中间件框架）与用户数据集成。OWLIM与Sesame、TRREE、ORDI的关系见图2。

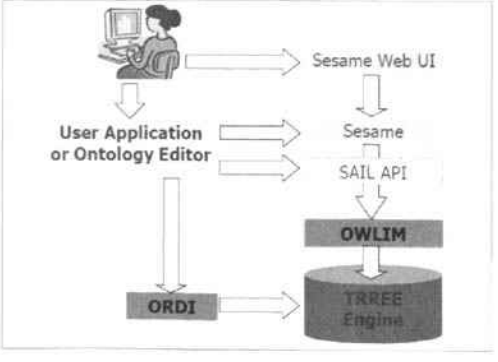

图2 OWLIM与Sesame、TRREE、ORDI的关系[17]

在Sesame系统的框架基础上，OWLIM具有特色的组件是TRREE引擎和ORDI组件。

① TRREE引擎

TRREE是OWLIM的核心组件，OWLIM采用TRREE引擎实现RDFS和OWL推理，OWLIM的推理能力可以通过定义TRREE的规则集来设置，TRREE支持RDF数据的表示、索引、存储和RDF模型，支持三元组的检索和正向推理（forward-chaining）。

② ORDI组件

ORDI组件的功能主要是支持异构资源和推理机之间的互操作，并能够集成数据库和其他结构化数据源。ORDI是一个高性能大规模WSWO仓储库，它采用Sesame、OWLIM和TRREE，并且支持SPARQL查询语言。

（2）系统性能分析

根据Ontotext实验室的报告统计[18]，SwiftOWLIM是目前最快的本体知识库存储系统，用LUBM的数据集进行测试，在32位台式计算机上的处理能力达到100万级的statement，速度达到24Kst./sec。BigOWLIM是目前支持OWL推理的规模最大RDF存储系统，它是目前唯一能达到OWL-Horst级推理能力并能达到3亿以上statement的系统，用LUBM的数据集进行测试，采用小型服务器上的处理10亿级的statement，加载和查询响应LUBM（8000，0）的时间为251413s。详细SwiftOWLIM与BigOWLIM的性能效果如表1所示。

表1 OWLIM处理LUBM（50,0）和LUBM（8000,0）的性能[18]

Task and Test ref	Tool, Version	Hardware & conf.(CPU,RAM,-Xmx)	Mill.stat.	Time(sec)	Speed st/sec	Inference	Comment
LUBM(50),[here]	BigOWLIM v0.92b	P4 3Ghz,2GB RAM,Xmx256,JDK1.5;owl-horst	6.9	1 317	5 232	OWL-Horst +/-	
LUBM(50)	BigOWLIM v0.92b	P4 3Ghz,2GB, Xmx350;owl-horst	6.9	1 055	6 531	OWL-Horst +/-	
LUBM(50),[20]	SwiftOWLIM v.2.9b4	P4 3Ghz,2GB, Xmx900;owl-horst	6.9	277	24 874	OWL-Horst +/-	
LUBM(50),[20]	SwiftOWLIM v.2.9b4	2xOpt270,2Ghz,12GB, Xmx2000,JDK1.5_64bit	6.9	112	61 518	OWL-Horst +/-	
LUBM(8000)	BigOWLIM v0.92b	2xOpt270,2Ghz,16GB, (4cOpt2g), Xmx12000,JDK1.5_64bit;owl-horst	1 060	251 413	4 216	OWL-Horst +/- complete	

尽管OWLIM采取一种相对简单的存储和查询方案，但经实践证明，OWLIM能够有效地、可靠地存储数据，并且已被多个应用项目所采用。如OWLIM被用于KIM语义标注平台的语义仓储库；TopBraid Composer系统集成OWLIM作为其推理机；BigOWLIM用于生命科学领域中的临床研究元数据管理系统AstraZeneva的存储，以及LifeSKIM平台的大规模基因交互数据集成系统；GATE 4.0集成SwiftOWLIM作为其Ontology存储服务系统[17]。

3.3 Jena

Jena是惠普实验室提供的开放源码工具，它是用于开发语义网应用系统的一个Java框架。Jena系统设计的目标是为RDF、RDFS、DAML、OWL提供一个程序开发环境，为应用程序提供多种途径的、表示灵活的RDF模型，以便于应用程序方便的访问操作。Jena支持大规模的语义仓储和查询引擎，对OWL 数据的存储提供Jena-memory和Jena-DB两种数据存储方式，它可以将数据存储在内存或数据库中。Jena的长期存储以三元组方式为基础策略，通过对三元组建立索引提高检索性能[19]。

（1）系统的关键技术

Jena提供了将RDF数据存入关系数据库的接口，其输入、输出、创建、导航、操作和查询等接口可以用于访问和维护数据库里的RDF数据。Jena独立于底层存储模式，即在处理数据时，应用程序只需通过Jena的API访问数据库，而不必直接操作数据库，也不必知道数据库的存储模式。Jena的查询接口支持RDQL和SPARQL查询语言。在Jena2中提供许多不同推导规则的规则集，这些规则集可以被应用于三元组表中存储的数据，也可以应用于内存中的前向推理、后向推理或联合推理，但Jena本身的推理能力有限，效率也不高。

系统的关键模块有：

① 三元组长期存储模块

初期的Jena产品Jena1采用扩展三元组存储方

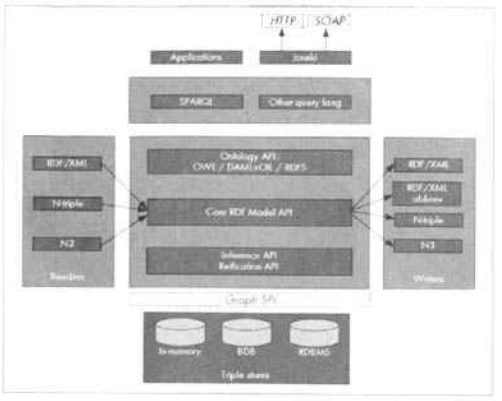

图4 Jena的组成结构[20]

法，即为三元组存储方法再加两个表（literals table和resources table）；在Jena2中，为了改善查询的效率，采用了反常规三元组存储方法，将一些短的literal和resource URIs直接存储在三元组表中，只有超过一定阈值的literal和resource URIs被单独存放在各自的表中。另外，增加一个标记位来标记这些项在三元组表中还是在一个独立的表中。反常规的方法比常规方法使用更多的存储空间，但在查询短的statement相关值时可以只查询一个表，总体上采用了两倍于常规方法的空间但提高了一倍的查询效率。

② 查询模块

Jena的查询处理主要是将查询式转化成查询三元组（subject-predicate-object）的形式，再将其转化成SQL语句的联合查询，再通过匹配三元组来找到满足查询条件的方式实现，系统查找返回所有满足查询条件的三元组。

③ 推理模块

Jena自身包含的推理机基本上是一种CLISP配合本体领域产生式规则的前向推理系统。因此，它的运行效率不是很高。不过通过DIG接口，允许Jena前端挂接到后台不同的推理引擎（RACER、FaCT、Pellet等）这样更专业的推理机上。

（2）系统性能分析

Jena提供了支持HSQLDB、MySQL、PostgreSQL、Derby、Oracle、Microsoft SQL Server的程序接口。有些第三方机构也提供Jena对其他数据库接口的支持。

在Maciej Janik和Krys Kochut的实验中，Jena处理的性能效果是：数据集LUBM(50, 0)加载的内存消耗为1828Mb，系统内存一般只能支持LUBM(10, 0)级别数据的查询。Jena的大规模处理性能较差，对RDF和OWL的查询和推理的响应都非常慢，有些查询甚至在运行了几个小时后仍不能终止。Jena不适合用于大规模的装载、查询和推理。

但是鉴于Jena是基于Java的完全开源的Ontology查询、存储和推理系统，在规模不是很大的应用领域采用Jena可以降低系统开发的综合成本和系统开发的难度，因而目前采用Jena系统进行开发的项目也有不少，例如Ingenta的MetaStore Project就采用Jena作为Ontology的存储系统[3]。

3.4 SOR

SOR（Scalable Ontology Repository）是由IBM在其前身IBM Minerva系统[21]的基础上开发的一个基于关系数据库的Ontology存储、推理和查询系统。SOR系统开发的目标是能够解决IBM的Ontology管理和使用周期中的一系列关键问题，是用于IBM产品资源知识库的原型系统，因此SOR系统更具有企业级应用的实际价值。SOR使用关系数据库存储Ontology和它的大规模实例（可能是上亿个三元组），在其之上执行有效的推理并支持方便易用的SPARQL查询接口。

（1）系统的关键技术

SOR是IBM Minerva系统的扩展和改进，SOR系统支持大规模存储，它的基本设计原则是：在关系数据库和Ontology之间要建立映射关系，如图5的OWL解析器完成此功能；设计将SPARQL查询转化为关系数据库查询语言的组件以支持检索，如图5的查询适应组件完成此功能；支持不同能力的推理引擎，如图5推理组件。

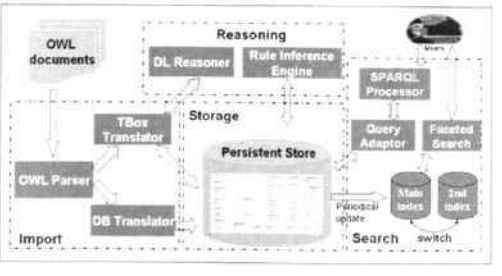

图5 SOR的体系结构图[4]

[3] http://www.ingentaconnect.com

SOR系统由四个模块组成，分别是：装载模块、推理模块、存储模块和查询模块。

装载模块包括OWL解析器和两个转换器。解析器负责对OWL文档解析，下一步数据库转换器将所有的ABOX断言装载到后备数据库中，TBOX转换器的功能有两层，一个是将所有的TBOX公理装载到描述逻辑推理器中，另一个是负责从描述逻辑推理器中取得推理结果存入到数据库中。

推理模块由一个描述逻辑推理机和一个规则推理引擎组成。首先，描述逻辑推理机推导出类和属性中所有的包含关系；接着，规则推理引擎在规则的基础上处理ABOX推理。目前，推理规则采用SQL语言实现，除了自行开发的结构化推理算法，SOR还可以通过DIG接口采用成熟的Racer和Pellet推理机进行TBOX推理。

存储模块不但要负责存储原始数据还要存储描述逻辑推理器和规则推理引擎处理后的推断结果。由于推理和存储被认为是一个完整的Ontology存储和查询系统中不可分割的两个组件，设计有效的RDBMS是执行有效推理的基础。当前，可以作为SOR的后台数据库有IBM DB2、Derby、SQL Server和Oracle。

查询模块，SOR支持的查询语言是标准的SPARQL，SPARQL查询请求的响应是通过对数据库的SQL查询的结果的返回。Faceted Search功能是为了实现Ontology关系的查询而专门设置的一个功能模块，它通过建立实例之间的索引关系实现简单关系的查询。Faceted Search模块支持知识库实例关系的查询。这对于用户挖掘出知识库中的潜在知识具有很大的意义。

（2）系统性能分析

IBM开发成员对SOR的存储性能进行了测试，通过测试定义的查询式Q1-Q8，采用SPARQL查询语言，在4M的测试数据集上对SOR和其前身系统Minerva分别进行查询速度测试，见表2所示。从测试结果可以看出SOR的查询时间都明显比Minerva提高很多。

进行测试的查询式分别为：

Q1: SELECT * WHERE {?x pim:PartKey '34389'}

Q2: SELECT ?x WHERE {?x pim:hasSupplier pim:SUP1}

Q3: SELECT * WHERE {?x pim:hasParent ?y . ?ypim:hasSupplier pim:SUP2}

Q4: SELECT * WHERE {?x pim:hasParent ?y . ?ypim:hasSupplier pim:SUP3 . ?x pim:SCode ?bFILTER (?b != 'N')}

Q5: SELECT * WHERE {?x ?y ?z . ?x pim:PartKey'34389'}

Q6: SELECT * WHERE {?x ?y pim:ABC}

Q7: SELECT ?x {?x a pim:C1 . ?x a pim:C2}

Q8: SELECT ?x {?x a pim:C3 . ?x a pim:C4 . ?xpim:hasSupplier ?y . ?x pim:hasManufacturer ?zFILTER (?y != ?z)}

表2 SOR系统的查询效率测试[22]

Query	Result size	Runtime(ms)	
		Minerva	SOR
Q1	1	245,812	146
Q2	137	162,172	21
Q3	6,181	245,594	646
Q4	1,341	19,430	484
Q5	33	3,233,633	16
Q6	386,038	364,586	18,088
Q7	19	18,375	4,709
Q8	25	330,845	16,755

4 基于Ontology的大规模知识库系统性能对比分析

为了更清晰的对以上大规模知识库仓储系统进行性能的分析和比较，本文从存储规模、查询速度、数据库支持、分析推理能力、支持查询语言、支持描述语言、开发语言和版权状态以及开发机构八个指标，对上述主要系统做简要的评估，见表3。

5 总结

本文通过对4个大规模知识库系统采用的关键技术和性能的分析，笔者认为，在系统查询性能和推理能力上的平衡是目前技术突破的难点。这两方面对大规模Ontology存储的需求通常是不能同时满足的，推理能力的增长会导致查询响应时间的增长。所以，在当前的系统开发中，不同的系统大多数都

表3 典型大规模仓储系统性能对比

	Sesame-Memory/Sesame-DB	SwiftOWLIM/BigOWLIM	Jena-Memory/Jena-DB	SOR
存储规模	1million 以内 /1million～3million triples	100万级的statement/10亿级的statement	无法支持海量数据，最大加载UniProt (650M)	120M triples以上
查询速度	快/不稳定	快/较快4 538 statements/sec	大规模数据查询速度较低	较快
数据库支持	PostgreSQL, MySQL, Microsoft SQL Server,Oracle	PostgreSQL, MySQL, Microsoft SQL Server,Oracle	HSQLDB,MySQL, PostgreSQL,Derby, Oracle,Microsoft SQL Server	IBM DB2, Derby,SQL Server,Oracle
分析推理能力	OWL Tiny	OWL DLP,OWL Horst	OWL Lite	OWL lite
支持查询语言	SeRQL,RDQL, RQL	SPARQL	RDQL,SPARQL	SPARQL
支持描述语言	RDF	RDF,OWL	RDF, OWL,DAML+OIL	OWL
开发语言、版权状态	Java 开源	Java开源/商用	Java 开源	Java商用
开发机构	Aduna B.V等	Ontotext Labs	HP Labs	IBM

明确各自的存储目标和重点，很少有系统的存储目标重点既需要处理大规模，又需要达到丰富语义的推理能力。

虽然，采用关系数据库存储大规模知识库是目前可行的主要存储策略，但其也有以下局限：

（1）广泛采用灵活的三元组结构存储Ontology及其实例，但是这种方式带来的问题是对三元组表的查询经常转换为多重的、自组织的联合查询，查询效率有待提高。

（2）基于规则引擎的数据库推理是不完全的。多数系统能够将推理和推理结构显式地存储于数据库中，采用规则推理，例如Sesame，OWLIM和Jena。但基于规则的推理是不完全的，如Jena的规则引擎只能处理内存中的RDF模型。

为了更大程度地实现大规模语义的存储、查询和推理，基于Ontology的大规模知识库构建技术今后的发展可能会趋向于以下方面：

（1）分布式存储。随着语义网技术的发展，各种语义应用对信息和资源的共享要求越来越高，因此，大规模知识库分布式存储的需求也随之被提出。分布式存储意味着存储机制的标准化、web service的服务方式、异构资源的集成查询和推理以及高性能的网络技术的全面支持。这对基于Ontology的大规模知识库构建技术来说是发展趋势，也是挑战。

（2）新的存储机制。现有的基于内存的存储机制和基于三元组的数据库存储机制在某些性能方面出现了瓶颈，这是由它们自身的局限性所决定的。关系数据库设计的初衷并不能深度地揭示语义关系，目前采用它作为知识库的底层数据结构只是在现有技术之上的一种解决方案，所以大规模知识库的存储如要达到质的飞跃，需要寻求全新的存储机制，由于各种复杂因素和技术水平的制约，这不是短期内所能解决的问题，需要付诸长期的研究和探索。

参考文献

[1] Sesame[EB/OL]. [2008-06-25].http://www.openrdf.org/.

[2] OWLIM[EB/OL]. [2008-06-25].http://www.ontotext.com/owlim/index.html.

[3] Jena[EB/OL].[2008-06-25].http://jena.sourceforge.net/.

[4] LU Jing, MA Li, ZHANG Lei, BRUNNER J S, WANG Chen, PAN Yue, YU Yong. SOR: A Practical System for Ontology Storage, Reasoning and Search[EB/OL].[2008-06-25]. http://www.vldb.org/conf/2007/papers/demo/p1402-lu.pdf.

[5] RDF Support in Oracle[EB/OL].[2008-06-25]. http://www.oracle.com/technology/tech/semantic_technologies/pdf/semantic_tech_rdf_wp.pdf.

[6] Redland[EB/OL].[2008-06-25].http://librdf.org/.[EB/OL]

[7] AllegroGraph 64-bit RDFStore[EB/OL].[2008-06-25].http://agraph.franz.com/allegrograph/.

[8] METASTORE K[EB/OL].[2008-06-25]. http://kowari.org/.

[9] OWLJessKB[EB/OL].[2008-06-25].http://edge.cs.drexel.edu/assemblies/software/owljesskb/.

[10] HARRIS S, GIBBINS N.3store: Efficient Bulk RDF Storage[EB/OL].[2008-06-25].http://km.aifb.uni-karlsruhe.de/ws/psss03/proceedings/harris-et-al.pdf.

[11] PANG Z, HEFLIN J. DLDB: Extending Relational Databases to Support Semantic Web Queries[EB/OL].[2008-06-25]. http://www.cse.lehigh.edu/~heflin/pubs/psss03-poster.pdf.

[12] Mptstore[EB/OL]. [2008-06-25]. http://mptstore.sourceforge.net/

[13] BECHHOFER S, HORROCKS I, TURI D. Implementing the Instance Store[EB/OL].[2008-06-25].http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-115/02-bechhofer.pdf.

[14] MA L, YANG Y, QIU Z, XIE G, PAN Y, LIU S. Towards a complete owl Ontology benchmark[C]//Proc. of European Semantic Web Conference, 2006:125-139.

[15] GUO Yuanbo, PAN Zhengxing, HEFLIN J. An Evaluation of Knowledge

(下转70页)

社会文明”。它所包含的实质内涵就是团结协作，不仅图书馆员与图书馆要团结协作，而且图书馆员与馆际之间、与社会之间也要团结协作。用户驱动作为图书馆2.0的核心特征，也是未来图书馆发展的驱动模式。但图书馆2.0环境下个性化服务必须提升其共享性，充分利用图书馆2.0技术的社会性将个性化服务提升为图书馆用户全体可分享的层次。这主要通过两种方式予以实现，首先是增强图书馆个性化服务的社会性，即服务不再仅仅由一个图书馆来提供，而是充分利用他人的成果，甚至商业性网站的服务来融合并拓展。其次是图书馆用户之间充分实现共享，形成社会性网络。在图书馆2.0这一大环境下，图书馆员必须牢固树立馆际协作意识，在信息资源建设和提供方面，要做到博采众长、为我所用，充分保证能为用户提供全面的信息资源。这依赖于图书馆员之间以及图书馆员与用户之间也要建立良好的合作交流关系。《中国图书馆员职业道德准则（试行）》中“发扬团队精神，树立职业形象；实践馆际合作，推进资源共享；拓展社会协作，共建社会文明”对协作方面有所提及，这也是职业伦理基本准则中“客观公正原则、公共存取原则”所要求的。

参考文献

[1] LibraryCrunch.[EB/OL].[2008-01-25]. http://www.librarycrunch.com/2005/12/what_is_library_20.html.

[2] 范并思，胡小菁.图书馆2.0:构建新的图书馆服务[J].大学图书馆学报,2006(1):3.

[3] 沙勇忠.信息伦理学[M].北京:北京图书馆出版社,2004:83,95-100.

[4] 德国之声深度报道:什么是Web2.0?[EB/OL].[2008-01-25]. http://www.jisuanji.com.cn/info/5490-1.htm.

[5] 向我开炮吧:小钟脑子里“有用”的Web2.0[EB/OL].[2008-01-25]. http://blog.sina.com.cn/u/492279530100089g.

[6] 曹丽涛.网络环境下图书馆参考咨询的特征[J].江南大学学报(人文社会科学版),2005(3):110-113.

[7] 老槐.图书馆学五定律之2.0版[EB/OL].[2008-01-25].http://blog.sina.com.cn/s/blog_4bbb1fdf010008eh.html.

[8] 周庆山.传播学概论[M].北京:北京大学出版社,2004:251,112-130.

作者简介

杨瑞坤，福州大学图书馆馆员，发表论文多篇。通讯地址：福州大学图书馆 350002

Study on Self-Construction of Librarian2.0 Professional Ethics

Yang Ruikun / Library of Fuzhou University, Fuzhou, 350002

Abatract: With a review of the change of library service in the environment of library2.0, this paper introduces the basic principles of professional ethics in library, and finally analyzes the strategies of librarian2.0 professional ethics construction.

Keywords: Library2.0, Librarian2.0, Professional ethics, Professional moral, Web 2.0, Digital library

（收稿日期：2007-12-20；责任编辑：虞敏）

(上接25页)

Base Systems for Large OWL Datasets[EB/OL].[2008-06-25]. http://swat.cse.lehigh.edu/pubs/guo04c.pdf.

[16] TRREE[EB/OL]. [2008-06-25]. http://www.ontotext.com/trree/index.html.

[17] OWLIM Pragmatic OWL Semantic Reposotory[EB/OL].(2008-04).[2008-06-25].http://www.ontotext.com/owlim/OWLIMPres.pdf.

[18] OGNYANOFF D, KIRYAKOV A, VELKOV R, YANKOVA M. D2.6.3 A scalable repository for massive semantic annotation[EB/OL].SEKT project, Jan 2007 .[2008-06-25].http://www.ontotext.com/publications/SEKT_D2.6.3.PDF.

[19] WILKINSON K, SAYERS C, KUNO H, REYNOLDS D. Efficient RDF Storage and Retrieval in Jena2[EB/OL].[2008-06-25]. http://www.cs.uic.edu/~ifc/SWDB/papers/Wilkinson_etal.pdf.

[20] SEQUEDA J. JENA Architecture[EB/OL] .[2008-06-25].http://www.cs.utexas.edu/~jsequeda/files/feb72007/JENAArchitecture.ppt.

[21] ZHOU Jina, MA Li, LIU Qiaoling, ZHANG Lei, YU Yang, PAN Yue. Minerva: A Scalable OWL Ontology Storage and Inference System[EB/OL].[2008-06-25].http://www.dis.uniroma1.it/~degiacom/didattica/semingsoft/SIS05-06/seminari-studenti/07-04-04%20-%20SIS%20-%20Antonello%20Ercoli%20Alessandro%20Pezzullo%20-%20IBM%20Minerva/articoli/Minerva/Minerva_A_Scalable_OWL_Ontology_Repository.pdf.

[22] BRUNNER J S, MA L, WANG C, ZHANG L, WOLFSON D C, PAN Yue, SRINIVAS K. Explorations in the Use of Semantic Web Technologies for Product Information Management[EB/OL].[2008-06-25]. http://www2007.org/papers/paper776.pdf.

作者简介

洪娜（1980-），女，中国科学院国家科学图书馆，在读博士研究生，发文5篇。通讯地址：北京市海淀区中关村北四环西路33号，中国科学院国家科学图书馆 100190

张智雄（1971-），男，中国科学院国家科学图书馆研究馆员、博士生导师，发文60余篇。通讯地址：同上

Technical Analysis of Constructing Ontology-based Scalable Knowledge Base

Hong Na / National Science Library, Chinese Academy of Sciences, Beijing, 100190; Graduate School of the Chinese Academy of Sciences, Beijing, 100049

Zhang Zhixiong / National Science Library, Chinese Academy of Sciences, Beijing, 100190

Abstract: Ontology-based scalable knowledge base system is the foundation of semantic content. In this paper, based on the introduce of current typical ontology-based scalable knowledge base system, the author analyzes the key technology, character and performance of each system, and then makes a performance contrast analysis of them. This paper also analyzes limitations, challenges and trends of current typical system. It expected to be helpful in the construction of knowledge base system of Chinese digital library.

Keywords: Knowledge extraction, Ontology storage, Knowledge base, Ontology inference, Ontology query, Performance compare, Digital library

（收稿日期：2008-07-13；责任编辑：贾延霞）