知识服务与开源软件

北京师范大学管理学院
耿骞

图书馆服务

- 从文献服务到信息服务
- 从信息服务到知识服务

图书馆知识服务

- 知识发现
- 知识组织（知识关联）
- 知识检索
- 知识传递

图书馆知识服务

- 学科馆员与学科化服务（参考咨询）
- 知识处理系统（语义检索，用户建模）
- 均需要建立一个有效的知识体系（非文献体系）

本体的构成

- 对应本体的具体构造过程，可以用下面的公式形象地给出：

  本体 = 概念（ Concept）
  ＋ 属性（ Property）
  ＋ 公理（Axiom）
  ＋ 取值（Value）
  ＋名义（Nominal）

Perez分类法组织本体

- 类（classes）或概念（concepts）
- 关系（relations）
- 函数（functions）
- 公理（axioms）
- 实例（instances）

Perez分类法组织本体

- 类（classes）或概念（concepts）
  指任何事务，如工作描述、功能、行为、策略和推理过程等，本体中的这些概念通常构成一个分类层次。

Perez分类法组织本体

- 关系（relations）
  在领域中概念之间的交互作用，形式上定义为n维笛卡儿积的子集：
  R:C1×C2×…×Cn。如子类关系（subclass-of）。在语义上关系对应于对象元组的集合。

Perez分类法组织本体

- 函数（functions）
  一类特殊的关系。该关系的前n-1个元素可以唯一决定第n个元素。形式化的定义为F:C1×C2×…×Cn-1→Cn。如Mother-of就是一个函数， mother-of(x,y)表示y是x的母亲。

Perez分类法组织本体

- 公理（axioms）
  代表永真断言，是定义在“概念”和“属性”上的限定和规则。如概念乙属于概念甲的范围

Perez分类法组织本体

- 实例（instances）
  指术语某概念类的基本元素，即某概念类 所指的具体实体。

本体描述语言

- 本体的描述语言解决的是概念的形式化问题。自上个世纪90年代以来，一些基于AI的本体实现语言陆续被提出，如KIF，Ontolingua，CycL，Loom，OCML，Flogic。后来，随着Web的发展，又出现了一系列基于Web的本体语言，也叫做本体标记语言，如SHOE，XOL，RDF，RDF-S，OIL，DAML，DAML+OIL，OWL。

本体构建工具

- 根据这些工具所支持的本体描述语言，大致可以分为两类。
  第一类包括Ontolingua，OntoSaurus，WebOnto等。
  第二类包括Protégé系列，WebODE，OntoEdit，OilEd等

本体构建工具

- 第一类工具的共同点是，都基于某种特定的语言(Ontolingua基于Ontolingua语言，OntoSaurus基于LOOM语言，WebOnto基于OCML语言)，并在一定程度上支持多种基于AI的本体描述语言。

本体构建工具

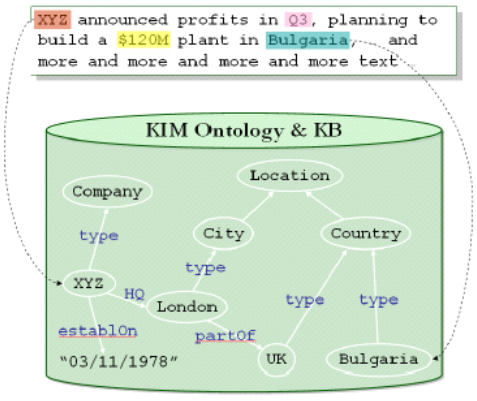
- 第二类工具最大的特点是独立于特定的语言，可以导入/导出多种基于Web的本体描述语言格式(如XML，RDF(S)，DAML+OIL等)。其中，除了OilEd是一个单独的本体编辑工具外，其他都是一个整合的本体开发环境或一组工具。它们支持本体开发生命周期中的大多数活动，并且因为都是基于组件的结构，很容易通过添加新的模块来提供更多的功能，具有良好的可扩展性。

KIM

- KIM是OntoText实验室的研究项目。该项目的研究成果KIM Platform（Knowledge and Information Management Platform）提供了一个语义服务平台构架和在此构架上的应用，包括：网页内容的半自动的语义标注、本体部署、基于内容的语义索引和检索和知识导航和知识问答。

KIM

- KIM将目标定位于解决以下两个基本问题：一个是识别文档中的命名实体，另一个是用命名实体来标引和检索文档。



KIM Ontology

- KIM Ontology属于顶级轻量级本体，用于定义命名实体的类型，采用RDFS语言描述，其RDF(S)文件kimo.rdf可以从网上下载，并能在Protégé本体编辑工具中编辑。

KIM Ontology

- **Thing**
  - Entity
    - Abstract
    - Happening
    - Object
  - EntitySource
    - Recognized
    - Trusted
  - LexicalResource
    - Alias
    - NERLexica

KIM Ontology

KIM Ontology公理：

<X, locatedIn, Y> and <Y, subRegionOf, Z > => <X, locatedIn , Z>

KIM Ontology

- 从KIM Platform1.2.12.16以后，KIMO 被以下ontology所取代：Proton、KIMLO和KIMSO。Proton由SEKT 项目维护，该本体用owl语言编辑，有300多个class和100个properties，覆盖了语义标注、索引、检索领域最通用的上层概念，具体细节可参考http://proton.semanticweb.org。

KIM Ontology

Proton具备如下特点：

- 领域无关
- 轻量级逻辑定义
- 和通用的元数据标准兼容
- 对具体的命名实体具有较好的覆盖率

KIM Knowledge Base

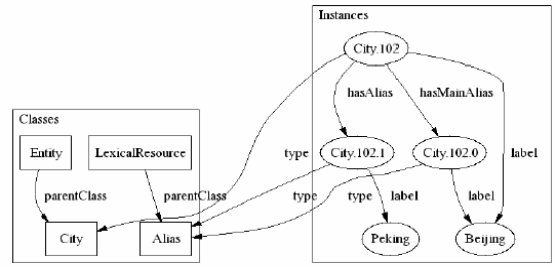
- KIM Ontology定义了实体的类型、关系和属性，而实体的具体描述保存在KIM Knowledge Base中。可以把KIM Ontology看作是KIM Knowledge Base的模式（Schema），两者都应该存储在语义数据库中，该语义数据库存储工具能够支持知识推理、检索、甚至版本控制、访问控制、事务处理等功能。

KIM Knowledge Base

KIM KB中包含了两类实体描述信息：

- 一类是可信任实体，是从可信任数据源导入的；
- 另一类是机器自动抽取的，

KIM KB 实体描述实体



KIM KB 的内容

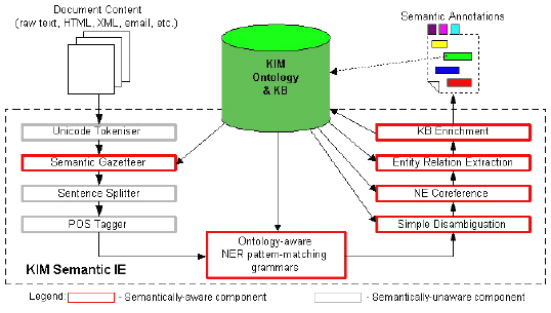
- KIM KB包含77,500个实体和110,000个别名。这些实体中数量最多的是地理实体和机构实体。地理实体从GNS(GEOnet Names Server)和其他数据源中导入，这些地理数据涉及50,000多个地点，包括大陆，全球地区，282个国家，所有的首都和4700个城市。

KIM KB 的内容

- KB中包含了8400多个世界机构实体，包括7900个公司（包括5000多家上市公司以及5500多个管理职位），140家股票交易机构。只有在全球有影响力的公司数据才被导入到KB中，公司数据来源于Google目录和Hoovers Online 的公司列表。

KIM语义信息标注

- 识别命名实体并建立关联
- 句法模式匹配
- 简单排歧
- 实体间拼写相互参照



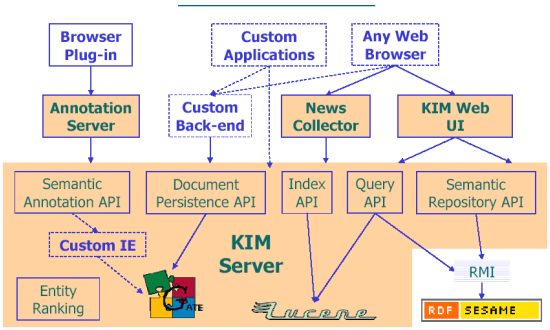
KIM技术基础

- GATE
- Sesame
- Lucene

KIM体系结构

KIM平台 包括以下四个部分：

- KIM Ontology
- KIM World KB
- KIM Server
- Front-ends



本体推理

- 本体的推理以描述逻辑为基础

  描述逻辑（Description Logic）又称为术语逻辑（Terminological Logic）或类KL-ONE系统，由Branchman于1977年提出，并实现了第一个DL系统KL-ONE，其最初的研究动机是为了知识表示中的语义网络提供形式化的基础。

本体推理

一个标准的描述逻辑系统由四个部分组成：

- 表示概念和关系(Role)的构造集；
- Tbox，包含概念定义(如: Father ManhasChild.Person,即Father被定义为“有孩子的男人”)及公理(如: Father Person)；
- ABox,包含概念断言(如: Student (Zeke))和关系断言(如: hasFriend(Zeke,Bob))；
- TBox和ABox上的推理机制。

本体推理

描述逻辑的推理功能集中在：

- 一致性检测
- 满足性检测
- 包含检测
- 实例检测

描述逻辑推理机Racer

- Racer （Renamed Abox and Concept Expression Reasoner）是一个基于描述逻辑SHIQ的推理机，最初由德国汉堡工业大学和加拿大康考迪亚大学从1999年开始开发。Racer支持以下推理查询：概念的一致性检测、概念的可满足性检测、检测Tbox中所有不一致的概念、概念自动分类、实例查询等。

描述逻辑推理机Racer

- 作为推理服务器，Racer提供两种接口，一种是基于TCP协议的Socket接口，Racer提供了专门的基于java的API包Jracer，用户可以在自己的应用程序中通过Jracer和Racer服务器通讯；另一种是基于HTTP协议的DIG(DL Implementation Group)接口。

描述逻辑推理机Racer

Racer的功能分为以下几类：

- Knowledge Base Management
- Tbox Management
- Abox Management
- KB Declareations
- Evaluation Functions
- Retrieval

nRQL

- Racer在Abox query的基础上，开发了nRQL(new racer query language)语言，允许用户在检索式中使用变量，进行更复杂的检索。

nRQL

- 一个nRQL检索式由三部份构成：retrieve标志、检索目标（query-head）、检索条件式（query-body）。

例如（retrieve (?x) （?x |Movie|））

其含义是检索Movie类中的所有实例。

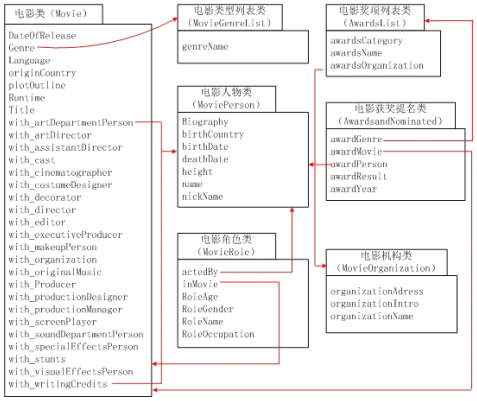
nRQL

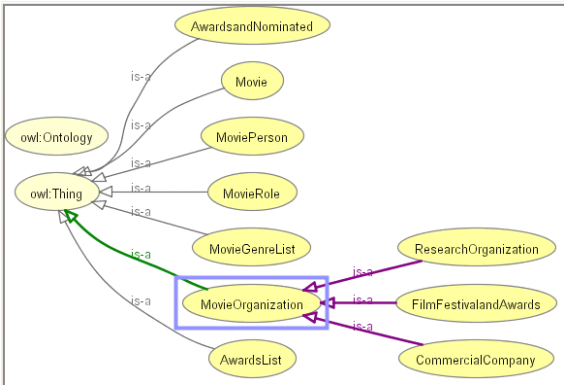
- nRQL的完整语法如下见：

http://www.cs.concordia.ca/~haarslev/racer/racer-queries.pdf

基于本体的知识检索

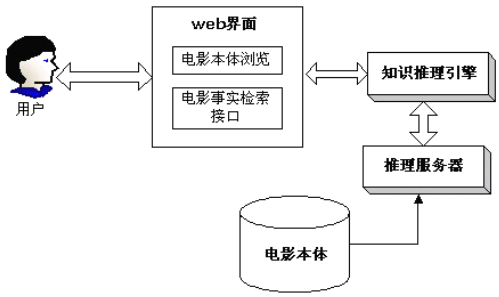
- 使用KIM和Racer，建立电影数据本体，建立了一个基于本体的知识检索系统。





基于本体的知识检索

- 实例数据建立
  电影本体框架（类、属性、关系）搭建好之后，就需要在框架之下添加实例数据，在KIM中，实例数据放在Knowledge Base，实例大多是从权威数据集导入或自动识别。



检索示例

- 获得第77届奥斯卡最佳电影奖的影片
- 电影《百万宝贝》的导演是谁?
- 第77届奥斯卡最佳女主角奖的获得者及其获奖作品

检索示例

- **RQL Query：**
- **(retrieve (?x ?y) (and (?x |Movie|)**
  (?y | AwardsandNominated|)
  (?y ?x | awardMovie|)
  (?y (string= | awardYear| "2004"))
  (?y (string= | awardResult| "award"))
  (?y | Best_Film_Award| |awardGenre|)

检索示例

- **Racer Answers：**
- **(((?X |Million_Dollar_Baby|) (?Y |movie_RDFResource_32|)))**

谢谢