基于 SOA 架构的术语注册和服务系统设计与应用*

欧石燕

摘　要　术语注册和服务系统是各种知识组织系统共建共享的重要平台，是网络知识组织系统（NKOS）由理论走向实际应用的关键环节。通过对国内外术语注册与服务系统及相关研究的调研分析，指出当前我国在术语注册与术语服务方面的研究与应用与国外相比还有较大的差距，提出了一个基于 SOA（Service-Oriented Architecture）构建术语注册和服务系统的解决方案。整个系统的架构从上至下分为四层，分别为任务服务层、工具服务层、组件层和数据层。设计的术语注册和服务系统主要支持基于 RDF 的语义化词表表示格式，并且支持以关联数据的形式显示词表内容数据。此外，本文还详细描述了实现该系统的关键技术，列举了术语注册和术语服务的一些典型应用案例。指出下一步的研究将具体实现一个术语注册和服务原型系统及术语服务客户端，并尝试将客户端程序集成在数字图书馆系统或编目系统中，通过调用术语服务对信息检索和编目提供术语支持。图 4。表 1。参考文献 22。

关键词　术语注册　术语服务　SOA　受控词表　网络知识组织系统

分类号　G250.76

ABSTRACT　Terminology registry with terminology services is an important platform for co-constructing and sharing various knowledge organization systems and a key step for Networked Organization Systems (NKOS) to progress from theory to practical applications. Through reviewing existing terminology registries and terminology services as well as related research projects both at home and abroad, this paper points out that there is still a big gap between home and abroad on the research and application of terminology registries and terminology services. It then proposes a Service-Oriented Architecture (SOA) based solution for building a terminology registry with terminology services, which includes four layers: Task Services Layer, Tool Services Layer, Components Layer and Data Layer. The proposed system is designed to support controlled vocabularies in the RDF-based semantic format and also makes vocabulary data available as linked data. Furthermore, this paper describes the key techniques used to implement the system, and some indicative use cases are also presented. In future, a prototype system will be implemented with terminology service clients, and the clients can be integrated in digital library systems or cataloging systems to provide terminology support for information retrieval and cataloging. 4 figs. 1 tab. 22 refs.

KEY WORDS　Terminology registry.　Terminology services.　SOA.　Controlled vocabularies.　NKOS.

CLASS NUMBER　G250.76

1　引言

术语表、分类表、叙词表、本体等各种词表（即知识组织系统）①在信息资源描述、组织、管理、发现等方面的强大功能已得到图书情报界和相关领域的广泛认可。为促进对这些知识组织工具的有效利用，需要对它们进行组织和管理。早期的做法是在机构内部创建和维护各种印刷版本的词表列表以供用户使用，如欧盟发

* 本文的研究受教育部人文社科基金一般项目“数据关联的语义数字图书馆研究”（编号：10YJA870014）和国家社科基金一般项目“基于 SOA 架构的术语注册和服务系统构建与应用研究”（编号：11BT0023）资助。

① 本文中的词表均指广义的词表，与知识组织系统等义，可互换使用。

布的 Thesaurus Guide[1]。自 1996 年起国外陆续出现了一些以电子格式发布的在线词表列表，如英属哥伦比亚大学图书情报学院的词表索引[2]和 HILT Resource List[3]，遗憾的是这些列表中的大多数并没有得到持久的扩展和维护。20 世纪 90 年代末网络知识组织系统(Networked Knowledge Organization Systems/Services，简称 NKOS)社区①开始了研制术语注册的尝试，知识组织资源的存储、组织、管理和利用开始朝着有序化、规范化和网络化的方向发展。

术语注册是指对各种词表提供权威的、集中控制的存储，以促进词表的发现、重用、管理、标准化和互操作。一个术语注册系统能够列出、描述、识别并且指明在信息系统和信息服务中可用的词表集合，并且提供图形化界面和术语服务以供用户访问和使用词表内容(指词表成员术语、概念及其相互关系)[4]。所谓术语服务是指对词表元数据和词表内容进行浏览、查询、应用的各种 Web 服务的统称[5]。术语服务通过 Web 应用程序接口(API)支持机器对词表及其内容的访问和调用，是在网络环境下对词表进行应用的重要途径。术语注册和术语服务两者相辅相成，前者是后者的前提和保证，后者是前者的目的和应用。

术语注册和服务系统是各种知识组织系统共建共享的重要平台，是网络知识组织系统(NKOS)由理论走向实际应用的关键环节，也是一个国家或领域内重要的信息基础设施。目前国外已经构建了不少术语注册和服务系统或者开展了一些相关项目的研究，如 Open Metadata Registry[6]、OCLC 术语服务[7]、FAO VEST Registry[8]等，与之相比，我国在这方面的研究和建设还比较滞后。国内最早出现的关于术语服务的论文是 2007 年司莉等人对 OCLC 术语服务的介绍[9]；2008 年深圳大学曾新红等人采用 OWL 语言对中文叙词表进行语义化表示并实现了对词表内容的检索[10]，虽然文中提到将来可能开发一套 Web 服务接口以实现机器对机器的术语服务信息交换，但还仅仅是一个展望；2008 年中国科学技术信息研究所史新等人开发了一套基于 Web 服务的汉语科技词系统[11]，这是我国术语注册和服务系统的最早雏形，但该系统只是针对单一的《汉语主题词表》提供可供访问的 Web 服务接口，没有提供术语注册功能，和真正的术语注册和服务系统相比还有一定距离。因此，大力开展术语注册和术语服务方面的研究，构建适用于我国知识组织工具的术语注册和服务系统，是十分必要和迫切的。通过建立术语注册和术语服务机制，可以加强对增长迅速、类型多样、内容复杂、来源不同的各类词表的维护和管理，并可直接通过网络为编目、元数据创建、信息检索、知识组织和管理等各类应用提供方便、快捷、强大的术语支持，让各类知识组织工具在网络环境下发挥更大的效益和价值。

鉴于以上目的，本文提出了一种基于 SOA(Service-Oriented Architecture)架构构建术语注册和服务系统的解决方案，并详细介绍实现该系统的关键技术，讨论术语服务的代表性应用。文章的后序部分按以下结构进行组织：第二部分分析回顾国外的主要术语注册和服务系统及相关研究项目；第三部分介绍词表的表示形式和关联数据化显示；第四部分给出系统的架构设计；第五部分介绍术语服务的代表性应用；第六部分是总结和展望。

2 研究综述

国外代表性的术语注册系统有 Taxonomy Warehouse[12]、Lexaurus Bank[13]、FAO VEST Registry、Open Metadata Registry、OCLC 术语服务等。Taxonomy Warehouse 是由 Dow Jones Factiva② 在 2001 年构建的 taxonomy 注册系统，共收集了由 288 个出版商提供的 670 个 taxonomies，是最早建立的术语注册之一，但功能有限，只提供词表

① NKOS 社区是致力于探索和讨论在网络环境下如何使知识组织系统成为支持信息资源的描述和检索的网络交互式信息服务的一个松散研究社区，见 http://nkos.slis.kent.edu/。

② 美国道琼斯公司旗下的一个商业资讯品牌。

的分类浏览和名称检索。Lexaurus Bank是英国Vocabulary Management Group公司开发的一个词表管理系统,支持SKOS、Zthes、IMS VDEX①等格式的词表的输入、输出以及分布式环境下词表的在线创建、编辑和相互映射,能够自动跟踪词表的更新修改并对词表进行完全的版本控制,此外还提供REST模式的Web服务以支持机器对机器的词表访问。FAO VEST Registry是联合国粮农组织建立的一个综合性注册系统,词表大类中存储了90多个与农业和农业管理相关的词表,提供基于词表类型和领域的词表浏览,此外还针对AGROVOC多语言农业词表提供了一组基于SOAP协议的术语服务,实现对该词表中术语及其关系的检索。Open Metadata Registry是在美国自然科学数字图书馆研究项目中构建的一个大型词表和元数据注册系统。是目前最强大的术语注册系统,不仅拥有基本的词表元数据和词表内容检索功能,还支持词表的在线编辑和更新、词表的版本控制、词表更新的自动通知等复杂功能,遗憾的是该系统目前主要是通过可视化图形界面供人类用户使用,还没有提供支持机器访问的术语服务,开发者拟在后序工作中实现[14]。OCLC术语服务是OCLC开发的一个实验性术语服务系统,目前存储了包括LCSH在内的六个词表,支持HTML、MARC XML、Zthes和SKOS四种词表表示格式,采用SRU检索协议和CQL查询语言实现了一组术语服务[15]。目前OCLC的术语服务已有了一些实验性的应用,譬如美国印第安纳大学的OPAC系统采用OCLC术语服务提供了一个查询扩展功能。

除上述专门的术语注册和服务系统外,在一些相关研究项目中也涉及了术语服务的研究和开发,如HILT、STAR和ADL叙词表协议。HILT(High-level Thesaurus)是英国JISC(联合信息系统委员会)和RSLP(研究支持图书馆计划)共同资助的一个研究项目,采用SOAP协议和SRU/SRW协议实现了七个用于术语检索的术语服务,检索结果以SKOS格式表示。STAR(Semantic Technologies for Archaeological Resources)是英国AHRC(艺术与人文研究委员会)的一个研究项目,采用SKOS为词表的表示格式,以SKOS API为词表内容的查询接口,开发了七个术语服务,提供术语查找、相关概念获取、概念扩展等功能。美国亚历山大数字图书馆项目中构建的ADL叙词表协议采用自定义的XML格式表示词表,提供了五个术语服务实现词表的查询和浏览,但是不支持词表的创建、维护、共享和相互映射等复杂操作[16]。

通过对以上术语注册和服务系统及相关研究项目的调研分析,笔者对目前术语注册系统提供的基本功能概括如下:词表的注册和上载、词表元数据的浏览和检索、词表内容(即词表成员术语及其关系)的浏览和检索。有个别复杂的注册系统还提供词表的在线编辑修改、版本控制等高级功能。大部分术语注册系统提供术语服务,使计算机程序能够通过Web服务API访问和调用词表内容。

3 词表的表示

3.1 词表表示格式

词表的表示格式是整个术语注册和服务系统的前提和基础,支持什么样的词表格式决定了整个系统需采用的存储和检索策略。目前术语注册中采用的词表表示格式主要是XML编码格式,但也有极个别系统支持HTML等非XML编码格式,如OCLC术语服务。XML格式又可进一步细分为自定义XML格式和标准XML格式。自定义格式因不具有通用性,只在少数系统中出现,如ADL叙词表协议,大部分系统采用的是标准XML格式,主要有MARC XML、Zthes和SKOS三种。

MARC XML是由美国国会图书馆制定的MARC 21格式的一种XML表示方式,是最早期的词表电子化表示格式。Zthes被称作Z39.50词表描述模型,是一个基于XML格式的词表描

① IMS VDEX是用于教育类受控词表的交换和表示的一种标记语言。

述和传输规范。这两种词表表示格式都是在较早时期制定的，目前已经不能适应网络环境下对词表应用的要求[17]。SKOS 全称是 Simple Knowledge Organization System（简单知识组织系统），是由万维网联盟（W3C）于 2005 年发布的一套词表语义化描述规范，采用 RDF 格式对词表的结构、内容和映射关系进行描述，可用于表示除本体外的几乎所有其他受控词表，是一种适用于网络环境下词表应用的新的表示格式。SKOS 标准包括三部分：用于描述词表基本结构和内容的 SKOS Core 模型；用于描述不同词表概念间映射的 SKOS Mapping；用于描述特定应用的 SKOS-XL。对于简单词表，采用通用的 SKOS Core 模型足以进行描述；但是对于某些复杂词表，如《中国图书馆分类法》和《汉语主题词表》，则还需要对 SKOS Core 模型进行一定程度的扩展，增加特定的类和属性，才能够实现对复杂词表的无损语义化描述。

除了普通的受控词表，还有一种特殊的词表，即本体。本体可采用 RDFS 或者 OWL 语言进行表示。RDFS 是最简单的本体描述语言，缺乏精确的表达能力，一般用于描述简单的知识结构，如 SKOS Core 模型和 DC 元数据标准都是采用 RDFS 语言定义的。OWL 是 W3C 制定的一种复杂的本体描述语言，具有强大的表达和推理能力，目前领域知识本体大都采用 OWL 语言进行描述。

通过对上述几种词表表示格式的分析和比较，笔者推荐采用语义化的词表表示方式，因为语义化的词表表示能为机器读取和理解，适于词表在网络环境下的应用，而且也更容易实现不同词表间的互操作。本文描述的系统是一个面向语义化词表的术语注册和服务系统，所支持的词表分为两类：一类是采用 SKOS 语言表示的普通受控词表，如术语表、叙词表、分类法等；另一类是采用 RDFS 或者 OWL 语言表示的知识本体。

3.2 词表内容的关联数据化

关联数据是由万维网创始人蒂姆·伯纳斯·李于 1996 年提出的一个概念，是指通过能够被 HTTP 协议访问的 URI 地址在 Web 上展示、共享、连接数据的一种方式。关联数据的两个基本宗旨是：①采用 RDF 数据模型在 Web 上发布结构化数据；②采用 RDF 链接连接来自不同数据源的数据[18]。作为一种在网络上发布结构化数据的方法，关联数据也可用于展示词表中的成员术语及其相互关系，譬如美国国会图书馆将 MARC XML 格式的 LCSH（美国国会图书馆标题表）转换为 SKOS 格式后以关联数据形式在 Web 上发布[19]。通过对词表内容的关联数据化，能够象访问 Web 文档一样直接通过 HTTP 协议访问词表中的成员术语并沿着术语间的链接（即术语间的关系）在不同词表（或概念体系）间穿行，使所有术语（或概念）构成一张数据网。此外，相对于 Web 文档之间的超链接，术语之间的 RDF 链接更能够揭示术语间的语义关系，有益于人机理解语境信息。词表内容的关联数据化显示如图 1 所示。图 1 中，细箭头指向的内容以 Web 文档的形式显示，根据客户端浏览器的不同，可以是 HTML、RDF/XML 或者 N3/Text 文档；粗箭头是 RDF 链接，沿着 RDF 链接可浏览同一词表或不同词表中的相关术语。

本文所描述的术语注册和服务系统支持词表内容数据的关联数据化显示。为了实现数据关联，所有注册词表及其成员均需采用能够被 HTTP 协议访问的 URI 标识符唯一命名。

对于 RDFS/OWL 本体，因为文档较小且所含成员数量较少，建议采用 Hash URI 地址命名本体中的成员（即概念和属性），如 < http://www.example.com/onto.owl#Concept 或 property >。当访问某个成员的 URI 地址时，HTTP 协议通过自动剥离“#”符号后的片段将对该地址的请求转换为对整个本体文档地址 < http://www.example.com/exonto.owl > 的请求，客户端浏览器将显示本体文档的全部 RDFS 或 OWL 代码。因为本体文档较小，此时浏览文档中对该成员的描述非常方便。对 RDFS/OWL 本体中的成员术语的 Web 访问方式如图 2（a）所示。

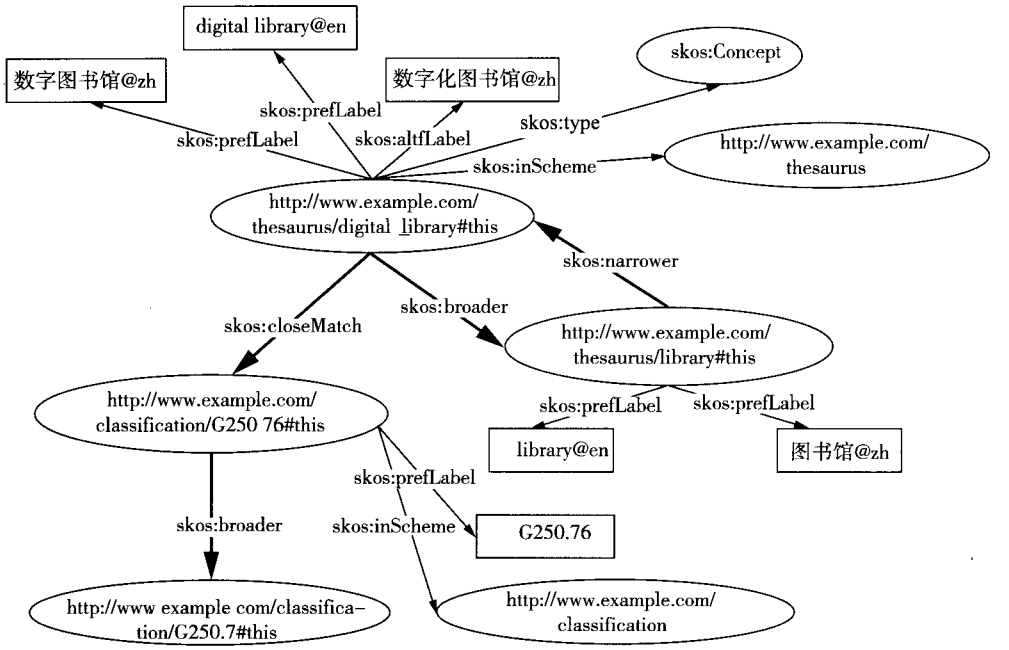


图 1　词表内容的关联数据化示意图

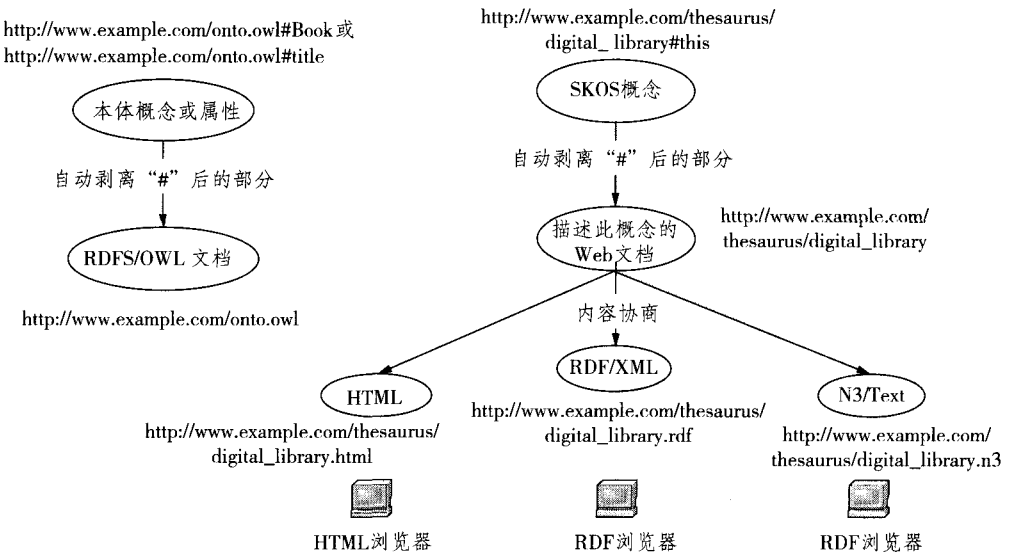
对于 SKOS 词表,因为文档较大且所含成员数量较多,建议采用 Slash URI 地址命名 SKOS 词表中的 SKOS 概念,如 < http://www. example. com/thesaurus/Concept > 。对于 Slash URI 地址的访问需采用 303 重定向方式进行,即 HTTP 协议自动将 SKOS 概念的 Slash URI 地址重定向到描述该概念的 Web 文档(如 HTML、RDF/XML、N3/Text 文档)的 URI 地址,如 < http://www. example. com/thesaurus/Concept. rdf 或 Concept. html 或 Concept. n3 >,具体采用哪种表示形式由 HTTP 协议的内容协商机制①根据客户端浏览器的情况来确定。如果客户端是普通的 HTML 浏览器,将发送 Concept. html 表示给客户端;如果客户端是支持 RDF 数据的 RDF 浏览器(如内嵌在 Firefox 中的 Tabulator RDF 浏览器),将发送 Concept. rdf 给客户端。采用 303 重定向方式的缺点是将不可避免地造成延时,为了避免该弊端,一个解决的方法是在 SKOS 概念的 URI 地址之后添加一个 Hash 后缀,将 Slash URI 地址转换为 Hash URI 地址,如 < http://www. example. com/thesaurus/Concept#this > 。当访问 SKOS 概念的 Hash URI 地址时,HTTP 协议自动剥离“this”后缀,将对 SKOS 概念 URI 地址的请求转换为对描述该概念的 Web 文档的请求,即对地址 < http://www. example. com/thesaurus/Concept > 的请求。该 URI 地址有多种表示形式,内容协商机制将选择最适合的表示形式返回给客户端浏览器。对 SKOS 词表中的成员术语的 Web 访问方式如图 2(b)所示。

4　系统架构设计

本文所描述的术语注册和服务系统采用基于 SOA 的架构模式。SOA(Service-Oriented Architecture),即面向服务的架构,是一种构造分布式系统的架构方法和设计原则,是将异构平台

① 内容协商是 HTTP 规范中定义的一个强大机制,它能够使同一个 URI 地址服务于 Web 资源的多个不同表示,从而能够将最适合的表示发送给浏览器。

上应用程序的不同功能实体通过它们之间定义良好的接口和规范按照松耦合的方式整合在一起的一个组件模型[20]。采用基于 SOA 的架构能够提高系统各种功能组件的重用性，有利于系统集成，使系统的扩展和更新更加容易，并且提高了系统的互操作性，有利于支持多线程并发的、组合的、更新频繁的、实时的术语服务。但需要说明的是，采用基于 SOA 的体系结构有时要以牺牲效率为代价，因此需要根据合理的设计和划分服务，使系统的综合性能达到最优。



（a）RDFS/OWL本体成员：Hash URI访问方式　（b）SKOS词表成员：Hash URI与内容协商相结合的访问方式

图 2　词表内容关联数据化的 Web 访问方式示意图

基于 SOA 的架构体系由服务、组件和对象三种不同粒度的功能实体构成，其核心是服务。服务是由一个或多个组件构成的粗粒度实体，向外界提供统一的接口，能够通过网络来访问，向服务请求者提供某种功能。组件是由多个对象构成的较细粒度的实体，能够提供独立功能并且可以同其他组件交互，而对象则是封装了状态和操作的更细粒度的实体[21]。根据 SOA 体系架构原则，整个术语注册和服务系统的架构从上至下分为四层（见图 3）。

任务服务层：该层的功能边界直接相关于特定的上层业务任务或流程。任务服务的复用潜力较小，主要作为一个服务组合中的控制器部分，负责组装其他和过程更加无关的服务[21]。整个术语注册和服务业务流程被划分为六个任务服务，每个任务服务组装一系列粒度更小的工具服务或组件，完成一定的业务流程。

工具服务层：介于任务服务和组件之间的中间层。每个工具服务致力于提供可复用的、横切的工具功能，可以封装多个组件，能够被多个任务服务调用[21]。术语注册和服务系统中的工具服务分为三大类：一类是对词表内容进行操作的服务；另一类是对词表内容进行验证的服务；第三类是对其他关系型数据进行操作的服务。

组件层：组件是指那些能够在各种服务中被反复使用的功能实体，一般是指应用程序的最小功能单元，也可看作是最小粒度的服务。在术语注册和服务系统中所用到的关键组件包括 RDF/OWL/SKOS 数据操作组件和 RDF/OWL/SKOS 数据验证组件。

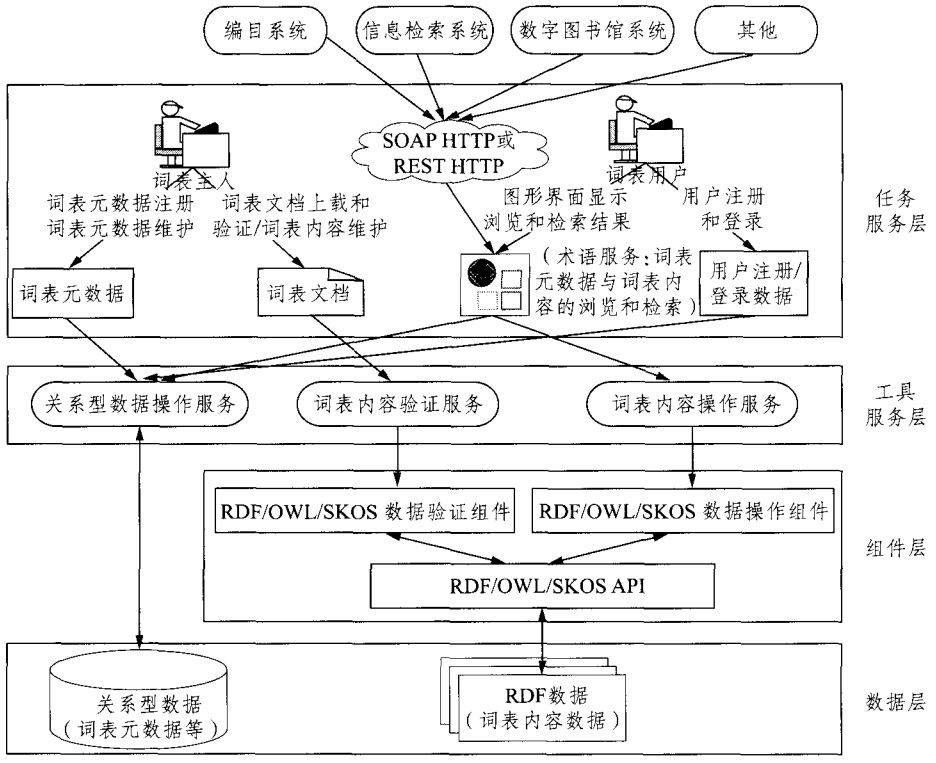


图3 基于SOA的术语注册和服务系统体系架构示意图

数据层:该层是数据存储层。术语注册和服务系统中所涉及的数据主要有两类:一类是词表内容数据,即RDF数据;另一类是关系型数据,包括词表元数据、用户注册数据、用户评论数据等。

术语注册和服务系统是一个复杂的系统,如果从无到有地进行开发将是一项非常浩大的工程。采用基于SOA架构体系的一大优势是可以利用已有的和新开发的工具或组件共同"搭建"一个新系统。目前已经存在着许多现成处理RDF/OWL/SKOS数据的工具,这些工具在系统中可直接作为组件或工具服务进行调用,从而减少系统开发的难度和复杂度,下文将对可选用的组件或服务进行详细介绍。整个术语注册和服务系统采用Web服务的方式实现。Web服务技术由于具有良好的封装性、松散的耦合性、协议规范的标准性以及高度的可集成性得到了业界广泛支持而成为目前实现SOA架构的理想方式,其好处是能够实现一个中立平台来获得服务并获得良好的通用性。

4.1 任务服务层

整个术语注册和服务流程被划分为六个任务服务,其中词表注册和文档上载、词表维护、词表文档浏览和下载、词表浏览和检索是核心服务。每个任务服务的功能详细描述如下:

(1) 用户注册和登录服务

- 对新用户提供注册功能,验证并存储用户提交的注册信息;
- 对注册用户提供登录功能,验证登录信息;
- 允许用户对注册信息进行修改和更新。

(2) 词表注册和文档上载服务

- 提供词表元数据注册功能,按照预定义的词表元数据标准提供所要注册的词表的元数据,并对提交的元数据进行验证和存储;

• 提供词表文档上载功能，默认支持 RDF/XML 序列化格式的词表文档的上载，并对上载的词表文档的格式和句法进行验证；

• 扩展支持其他序列化格式（如 N3、N-Triple 和 Turtle）的词表文档的上载和验证。

（3）词表维护服务

• 修改和更新已注册的词表元数据；

• 更新已上载的词表文档的版本和相应的词表元数据；

• 删除已注册的词表元数据及相应的词表文档；

• 对同一词表的不同版本进行版本控制；

• 扩展支持对词表内容的在线修改和更新；

• 扩展支持词表间的自动映射和集成。

（4）词表文档浏览和下载服务

• 允许用户浏览并下载免费的词表文档全文，默认以 RDF/XML 序列化格式显示；扩展支持以其他序列化格式（如 N3、N-Triple、Turtle）显示和下载词表文档。

（5）词表浏览和检索服务

• 浏览和检索词表的元数据；

• 浏览和检索词表的内容，即词表成员术语、概念及相互间关系；

• 以 Web 图形界面显示浏览和查询结果；

• 以 Web 服务的形式发布术语服务，供机器通过 HTTP 协议访问和调用；

• 支持以关联数据的形式显示词表内容的浏览和检索结果。

（6）系统管理服务

• 管理注册用户，对注册账户进行删除、修改、禁止和激活等操作；

• 管理注册词表，对注册词表的元数据进行删除、修改、锁定和解锁等操作。

4.2 工具服务层和组件层

本节主要对词表内容操作服务和词表内容验证服务及其构成组件进行介绍。关系型数据操作服务是通用服务，主要是通过数据库查询语言 SQL 实现对关系型数据的增加、删除、修改和查询的操作，在此不作累述。

（1）词表内容操作服务

词表内容操作服务的功能是对词表内容进行读、写和输出操作。在术语注册和服务中，使用的主要是读操作和输出操作，包括对词表文档的各种序列化格式（如 RDF/XML、N3）进行读取和解析，对词表成员（即术语、概念以及相互间关系）进行浏览和检索，以某种序列化格式将词表内容进行输出。如果系统还要支持对注册词表的内容进行在线编辑和修改，那么还需用到写操作，即修改、添加、删除词表成员的操作。词表内容操作服务封装了三个组件：RDF 数据操作组件，SKOS 数据操作组件和 OWL 数据操作组件。

RDF 数据操作组件的功能是读取和解析 RDF 或 RDFS 文档并对 RDF 数据进行读写和输出操作，需通过针对 RDF 数据的 API 来实现。虽然 OWL 数据和 SKOS 数据本质上也是一种 RDF 数据，在 RDF 数据层面也可采用 RDF 数据操作组件进行操作，但是这两种数据已经各自有更高语义层面的 API 可供使用，因此 RDF 数据操作组件主要用于处理 RDFS 本体文档。目前存在着多种开源 RDF API，针对 Java 语言的比较多，比较著名的有 Jena 和 Sesame。Jena 是 HP 实验室开发的一个开源的语义网工具包，包含了支持 RDF/RDFS/OWL 的 API、SPARQL 查询引擎、RDF/XML 解析器、RDF 数据持久化存储等组件；Sesame 是荷兰 Aduna 公司在欧盟研究项目 On-To-Knowledge 中开发的一个面向 RDF 和 RDFS 的开源存储、查询和推理框架。

OWL 数据操作组件的功能是读取和解析 OWL 文档并对 OWL 数据进行读写和输出操作，需通过针对 OWL 数据的 API 来实现。比较著名的开源 OWL API 有三个：Jena 中所带的 OWL API，本体编辑工具 Protege 3.x 版中所使用的 OWL API，以及由英国曼彻斯特大学主要开发和维护的 OWL API 1.0、2.0 和 3.0。上述 OWL API 都是 Java API，各有优缺点，在使用时可根据实际情况进行选择。

SKOS 数据操作组件的功能是读取和解析 SKOS 文档并对 SKOS 数据进行读写和输出操作，需通过针对 SKOS 数据的 API 来实现。目

前针对 SKOS Core 模型的 SKOS API 有两个:一个是由欧盟研究项目 SWAD-Europe 开发的 Java API,支持以 Web 服务的形式访问 SKOS 表示的叙词表,但是该 API 的功能有限,实用性不高;另一个是由 JISC 研究项目 CO-ODE 和欧盟研究项目 Sealife 联合开发的 Java API, 基于 OWL API 2.0 实现,基本上实现了对基于 SKOS Core 模型的 SKOS 数据的各种读写操作。此外,SKOS 数据也是一种 RDF 数据,也可采用 RDF 数据操作组件输入和输出 SKOS 词表文档,并通过 RDF 查询语言 SPARQL 来查询词表文档中的特定内容。这种方式更加自由灵活,不受 SKOS API 功能的限制,但是从理论上来说, SPARQL 查询的方式要比直接通过 API 解析 SKOS 文档的方式要慢一些。而且如果术语注册系统要支持词表内容在线编辑和修改等写操作,仍需通过基于 SKOS API 的 SKOS 数据操作组件来实现。

(2) 词表内容验证服务

词表内容验证服务的功能是对上载的词表文档的格式和句法进行验证,以保证注册词表的正确性和权威性。上载的词表文档均是以某种序列化格式表示的 RDF 文档,因此词表文档首先要遵循相应序列化格式的 RDF 句法规则,譬如 RDF/XML 文档需符合 RDF/XML 句法规则,N3 文档需符合 N3 的句法规则。除遵循 RDF 句法外,SKOS 词表还需遵循 SKOS 语言的规则,OWL 本体还需遵循 OWL 语言的规则,因此还需分别对它们进行 SKOS 验证和 OWL 验证。大多数 SKOS 验证器和 OWL 验证器中往往已包含了对 RDF 句法的验证,因此无需单独进行 RDF 验证。但是如果这些验证器中没有包含该验证(如 SKOS 2005 Validator),则需首先进行 RDF 验证。对于 RDFS 本体,只需进行 RDF 验证。词表内容验证服务封装了三个组件:RDF 验证器、OWL 验证器和 SKOS 验证器。

RDF 验证器的功能是对提交的以某种序列化格式表示的词表文档进行 RDF 句法验证。W3C 提供了一个 RDF 验证服务①, 能够对 RDF/XML 文档的句法进行验证并且对文档进行解析,输出 RDF 三元组和 RDF 图形表示。但是这个验证服务目前还不支持 N3 等其他序列化格式。

OWL 验证器的功能是对提交的 OWL 本体的句法进行验证,即验证 OWL 文件是否符合某种 OWL 子语言的句法规则。比较有名的 OWL 验证器是欧盟研究项目 WonderWeb 开发的 WonderWeb OWL-DL Validator②, 它能够验证 OWL 本体采用哪种子语言描述且是否符合该子语言的句法规则。OWL 本体的验证还包括语义验证,即检查 OWL 本体中描述的内容是否具有一致性,可以采用推理机来进行。因为 OWL Full 子语言不支持逻辑推理,因此语义验证只能针对 OWL Lite 和 OWL Full 本体。OWL 语义验证的过程比较复杂,建议在术语注册和服务系统中只对 OWL 文档的句法进行验证。

SKOS 验证器的功能是对提交的 SKOS 词表进行验证。目前 W3C 推荐了两个 SKOS 验证器③:SKOS 2005 Validator 和 SKOS 2009 Validator。SKOS 2005 Validator 是对 SKOS 词表的完整性和兼容性进行验证,目前还处于高度实验阶段,并且不支持对 RDF 句法的验证,因此在使用该验证服务之前,需事先使用 RDF 验证器验证 SKOS 词表文档 RDF 句法的正确性。SKOS 2009 Validator 是 SKOS 叙词表管理系统 PoolParty④提供的一个 SKOS 叙词表完整性⑤检测器。

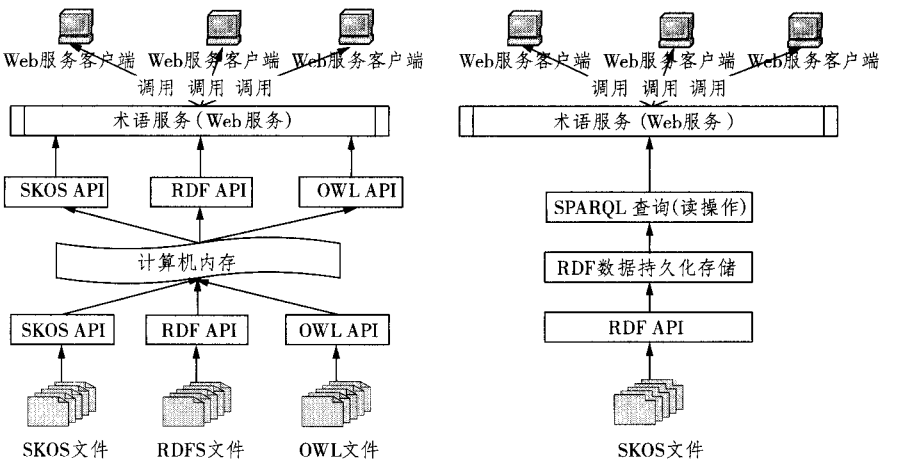
4.3 数据层

本节主要对词表内容数据(即 RDF 数据)的存储进行介绍,关系型数据(如词表元数据)的存储在此不作累述。

① W3C 的 RDF 验证器的地址是 http://www.w3.org/RDF/Validator/。
② WonderWeb OWL Ontology Validator 的地址是 http://www.mygrid.org.uk/OWL/Validator。
③ W3C 的 SKOS 验证器的地址是 http://www.w3.org/2004/02/skos/validation。
④ PoolParty 是奥地利的一家公司开发的一个 SKOS 叙词表管理系统,见 http://poolparty.punkt.at/。
⑤ SKOS 词表的完整性是指 SKOS 数据是否符合 SKOS 模型。

最简单的RDF数据存储模式是将其直接存储在文件系统中，每次调用数据时，针对不同格式的词表文档（即SKOS词表、RDFS/OWL本体），选择通过相应的API（指RDF/SKOS/OWL API）来读取整个文档，将数据装载到计算机内存中，然后通过特定API对不同格式的数据进行增加、删除、查询、修改的操作，从而提供术语服务。基于文件的词表内容数据存储模式如图4(a)所示。在图4(a)中，可选择采用HP实验室的Jena API读入和操作RDFS/OWL数据，该API同时支持RDF、RDFS和OWL三种标记语言；选择采用CO-ODE&Sealife SKOS API读入和操作SKOS数据。

对于少量的RDF数据来说，基于文件的存储是一种非常有效的存储方法，但并不适于处理大规模数据，因为随着数据量的增多，每次装载数据到内存的时间也越长，而且对机器内存的要求也越高。在术语注册系统中上载的RDFS/OWL本体是不含实例数据的类模型，数据量较少，因此可采用基于文件的存储。但是对于SKOS词表，一般数据量较大，这种存储方式不很理想。海量数据的理想存储方式是采用持久化的存储，即将RDF三元组固化到关系型数据库中或某种文件格式中。遗憾的是目前的SKOS API并不支持这种存储，也不支持对SKOS数据的SPARQL查询，虽然开发者正在朝这个方向努力[22]。另一种解决方案是把SKOS数据完全当作RDF数据来处理，采用支持持久化存储的RDF存储系统（如Jena/Sesame RDF API）对SKOS数据进行存储，然后通过SPARQL查询语言实现对SKOS数据的检索和浏览。基于持久化存储的词表内容数据存储模式如图4(b)所示。



(a) 基于文件的词表内容数据存储模式　　(b) 持久化的词表内容数据存储模式

图4　词表内容数据的存储模式

5　术语服务的应用

术语注册和术语服务在信息检索、信息浏览、信息发现、自动翻译、语义推理、编目和元数据创建、知识组织等许多领域都有着重要的应用。下面对一些代表性应用进行简单介绍和讨论。

5.1　编目和元数据创建

在编目和元数据创建中，首先可利用术语注册平台浏览和检索词表元数据，发现合适的词表以供使用。其次可将术语服务客户端内嵌在编目工具中，在编目时通过调用各种术语服务，如辅助分类、特定类型术语查找等，来减少编目工作量，提高编目的速度和精度。表1给出了案例1和案例2中使用的两个术语服务的详细描述。

表1 术语服务

服务标识符		matchNotationByTerm(String skosURL/title/alternative, String conceptLabel)
输入参数	参数1	SKOS词表的URI标识符/正式名称/其它名称 如"http://www.example.com/CT","汉语主题词表","CT"
	参数2	概念标签(即术语),如"数字图书馆"、"数字化图书馆"
输出值		与输入术语相匹配的分类号,如"G250.76"
功能描述		辅助分类:根据已知的主题词,返回与该主题词相匹配的分类号。如果没有发现与该主题词相匹配的分类号,寻找与主题词的上位概念相匹配的分类号,如此循环往复,直到找到匹配的分类号或者直到顶层概念。如果直到顶层概念仍未发现匹配的分类号,返回值为空。
作用的词表		相互映射的《汉语主题词表》和《中国图书馆分类法》
服务标识符		searchAllTerms(String str, [String termType])
输入参数	参数1	任意字符串,如"南"
	参数2	术语类型,如"中国地名"(该参数为可选参数)
输出值		返回与输入字符串相匹配的术语,如果没有精确匹配的字符串,返回不同程度前向匹配的术语(即概念标签),以及相应词表的URI标识符或名称,如"南京","南昌","南宁"等。(如果没有第二个参数,默认搜索所有的注册词表)
功能描述		特定类型术语查找:在多个词表中通过字符串匹配查找特定类型的术语
作用的词表		含有某种术语类型的注册词表,如语种词表、地名词表或所有注册词表

案例1:图书馆要为一批文献资源编目,每个资源已采用《汉语主题词表》中的规范术语进行了标引,现在需按照《中国图书馆分类法》对这些资源进行分类。通过调用"辅助分类"术语服务,输入每个资源的主题词,术语服务返回与之相匹配的分类号。因此编目员无需重新审阅每个文献的内容进行分类,只需对返回的类号进行验证或者在此基础上稍加修改即可,大大提高了分类的速度。

案例2:图书馆要为一批资源创建元数据,其中涉及到许多国内外地名。如果编目员对某个地名的拼写或覆盖范围不熟悉,可在编目工具内嵌的检索框中输入地名术语的一部分或全部字符串,如"New","特定类型术语查找"术语服务将返回与输入的字符串模糊匹配的一组地名及其详细描述,如"New York"、"New Hampshire"、"New Jersey"等,编目员可从中选择合适的地名,无需记忆或者离线查找。

5.2 信息检索

在搜索引擎、数字图书馆等信息检索系统中可内置术语服务的客户端,通过调用术语服务获取用户输入检索词的同义词、相关术语或上位词,从而对检索词进行扩展,以扩大检索范围。反之,也可通过术语服务获取输入检索词的下位词来缩小检索范围。

案例1:用户想查找有关"vog"的文献资料,但是通过数字图书馆系统检索该词没有发现任何检中记录。此时调用"相关术语查找"术语服务发现"vog"与"volcano gases"相关,换用"volcano gases"作为查询词,则顺利地检索到了相关文献。

案例2:用户想查找"采用计算机进行系统仿真"的文献资料。他先用"系统仿真"作为检索词,但是搜索引擎给出的检索结果太多,很难从中立即发现他所要的记录。因此他调用"检索词精炼"术语服务,发现"系统仿真"有两个下位词"计算机仿真"和"计算机化仿真",他换用这

两个词进行检索，立即找到了相关文献。反之，如果用户使用“计算机仿真”和“计算机化仿真”两个检索词没有发现合适的检索记录，可利用“检索词扩展”术语服务获取它们的上位词“系统仿真”，改用上位词作为检索词以扩大检索范围。

5.3 知识组织

词表是重要的知识组织工具，利用词表中呈现的复杂而规范的知识组织结构，能对信息检索系统返回的数量众多的检索结果进行分类和组织，也能构建强大、系统的资源导航工具。

案例1：搜索引擎或其他信息检索系统往往返回成百上千条检索结果，用户很难从中立即找到所需的记录，譬如用户想寻找有关“禽流感预防接种”的资料，输入“flu vaccinations”这个检索词，搜索引擎返回了113条检索结果。如果利用叙词表、知识分类体系（taxonomy）、分类法中对概念或术语的层次化分类，可进一步将检索结果按照主题进行分类，方便用户对检索结果的快速浏览和定位。譬如将“flu vaccinations”的检索结果分为四个类：Health assessments (68)，Occupational health(12)，Symptoms(14)，Virus (14) 和 Bird Flu(5)，用户可以直接跳到“Bird Flu”类中寻找所需记录，无需再浏览其他类别的记录。

案例2：《中国分类主题词表》是按照主题分类一体化的思想将《汉语主题词表》和《中国图书馆分类法》进行映射后构成的一个综合词表。因此可利用《中国分类主题词表》的知识组织体系构建一个以学科为基础的知识导航工具。通过对文献资源进行主题分类和标引，根据类号和标引词在词表中的等级关系和相关关系对文献资源进行组织，实现分类导航和相关资源的导航。

6 结论和展望

术语注册和术语服务是一个国家或领域内重要的信息基础设施，目前国外已经或正在开展这方面的研究，已有一些实验型系统面世，我国在该领域的研究还相对滞后，因此研究和构建我国自己的术语注册和服务平台是一项迫切的任务。

本文提出了一个基于SOA（Service-Oriented Architecture，面向服务的体系架构）的术语注册和服务系统构建方案。基于SOA的架构能够实现软件设计的粒度化，因此可利用已有的和新开发的程序模块、工具或服务“搭建”一个新的系统，从而减少系统开发和实现的难度和复杂度。本文所设计的术语注册和服务系统主要支持基于RDF的语义化词表表示格式，并且支持以关联数据的形式显示词表内容数据。本文对实现术语注册和服务系统的各种关键技术进行了分析和介绍，并且列举了术语注册和术语服务的一些代表性应用案例。

在下一步的研究中将具体实现一个术语注册和服务原型系统以及相应的术语服务客户端，并尝试将客户端程序集成在数字图书馆系统或编目系统中，通过调用术语服务对信息检索和编目提供术语支持。通过术语注册和术语服务，能够促进各种知识组织工具在网络环境下的应用，发挥它们在编目、元数据创建、信息检索、信息浏览、主题标引、知识组织等领域的巨大应用潜力。

参考文献：

[1] EUROBroker S. Thesaurus guide: Analytical directory of selected vocabularies for information retrieval, 1992 (2nd version) [R]. Luxembourg: European Communities, 1993.

[2] HILT vocabulary resources [OL]. [2011 - 01 - 28]. http://hilt.cdlr.strath.ac.uk/Sources/vocabulary.html.

[3] Stephenson M. Indexing resources on the WWW: database indexing, controlled vocabularies & thesauri [OL]. [2011 - 01 - 28]. http://www.slais.ubc.ca/resources/indexing/database1.htm.

[4] Golub K, Tudhope D. Terminology registry scoping study (TRSS): Final report [R/OL]. UK: Joint Information Systems Committee (JISC), 2009. [2011 - 01 - 28]. http://www.jisc.ac.uk/media/documents/programmes/sharedservices/trss-report-fi-

nal. pdf.

[5] Tudhope D, Koch T, Heery R. Terminology services and technology: JISC state of the art review [R/OL]. UK: Joint Information Systems Committee (JISC), 2006. [2011-01-28]. http://www. jisc. ac. uk/Terminology _ Services _ and _ Technology_Review_Sep_06.

[6] Open metadata registry [OL]. [2011-01-28]. http://metadataregistry. org/.

[7] OCLC terminology services [OL]. [2011-01-28]. http://www. oclc. org/research/activities/termservices/default. htm.

[8] FAO VEST registry [OL]. [2011-01-28]. http://aims. fao. org/vest-registry/browse-by-vocabularies.

[9] 司莉,徐丽晓,吴钢,等. OCLC术语服务研究:背景、进展与启示[J]. 中国图书馆学报,2007(1):58-61.

[10] 曾新红,林伟明. 中文叙词表本体的检索实现及其术语学服务研究[J]. 现代图书情报技术,2008,(2):8-13.

[11] 史新,乔晓东,张志平,等. 汉语科技词系统的Web服务研究与实现[J]. 现代图书情报技术,2008(12):37-42.

[12] Taxonomy warehouse [OL]. [2011-01-28]. http://www. taxonomywarehouse. com/.

[13] Lexurus bank [OL]. [2011-01-28]. http://www. vocman. com/lexaurusbank.

[14] Hillmann D,Sutton S,Phipps J,et al. A metadata registry from vocabularies up:The NSDL registry project [C/OL]//Baker, T. & Solorio, J. Proceedings of 2006 International Conference on Dublin Core and Metadata Applications: metadata for knowledge and learning. Colima, Mexico: Dublin Core Metadata Initiative. 2006:65-75. [2011-01-28]. http://arxiv. org/ftp/cs/papers/0605/0605111. pdf.

[15] Vizine-Goetz D. Terminology services [OL]. [2011-01-08]. http://www. oclc. org/research/presentations/vizine-goetz/cendi-nkos-isko. ppt.

[16] Janée G, Ikeda S, Hill L. The ADL thesaurus protocol (version 1.0) [OL]. [2011-01-28]. http://www. alexandria. ucsb. edu/thesaurus/specification. html.

[17] 王军,张丽. 网络知识组织系统的研究现状和发展趋势[J]. 中国图书馆学报,2008(1):65-69.

[18] Berners-Lee T. Linked data design issues [OL]. [2011-01-28]. http://www. w3. org/DesignIssues/LinkedData. html.

[19] Summers E, Isaac A, Redding C, et al. LCSH, SKOS and linked data [C/OL]//Dekkers M & Neuroth H. Proceedings of the 2008 International Conference on Dublin Core and Metadata Applications. Singapore: Dublin Core Metadata Initiative. 2008:25-33. [2011-01-28]. http://dcpapers. dublincore. org/ojs/pubs/article/viewFile/916/912.

[20] Barry D K. Service-oriented architecture (SOA) definition [OL]. [2011-01-28]. http://www. service-architecture. com/web-services/articles/service-oriented_architecture_soa_definition. html.

[21] Erl T. SOA服务设计原则 [M]. 郭耀,译. 北京:人民邮电出版社,2009.

[22] Jupp S, Bechofer S, Stevens R. A flexible API and editor for SKOS [C/OL]//Aroyo, L. et al. The Semantic Web: Research and Applications, Proceedings of the 2009 European Semantic Web Conference, Lecture Notes in Computer Science 5554. Berlin: Springer. 2009: 506-520. [2011-01-28]. http://citeseerx. ist. psu. edu/viewdoc/download? doi=10. 1. 1. 140. 9760&rep=rep1&type=pdf.

欧石燕 南京大学信息管理系教授。通讯地址:南京市汉口路22号南京大学信息管理系。邮编:210093。

(收稿日期:2011-02-14)