基于内容的语义桌面系统设计与实现①

Design and Implementation of Content-Based Semantic Desktop System

王相根　罗铁坚　张　驰　(中国科学院 研究生院 信息学院 北京 100049)

摘　要： 介绍了基于内容的语义桌面系统。系统对桌面数据进行区分，根据桌面内容的特点提供不同的语义查询；采用数据可视化技术展示桌面数据，记录用户的思路；提供数据共享功能，方便用户交换桌面数据；采用插件架构，允许用户方便地扩展功能。

关键词： 语义Web 语义桌面 桌面管理

1　引言

随着计算机的普及和发展，用户电脑中存储的文件越来越多，在文件管理上，用户仍是采用以手工的方式组织文件夹来进行管理。Google Desktop Search 等桌面搜索软件可以快速地找到同搜索关键字匹配的文件，但是搜索结果往往包含大量用户不想要的文件；EndNote 等第三方软件能够对某些方面的文件进行管理，但是由于各个软件采用不同的数据描述格式，数据集成非常困难，用户不能得到一个统一高效的桌面管理平台。因此，Stefan Decker 在 2003 年提出语义桌面[1]，并对其作了如下定义：

定义 1. 语义桌面是一个能够存储各种数字信息(例如文档、多媒体文件、信息等)的独立设备。这些数字信息通过 URI 标注，作为语义 Web 资源，能够以 RDF 图的方式进行访问和查询。经过授权的资源可以被其他用户共享。利用本体来表示用户的思维模型，各种不同的语义桌面系统使用语义 Web 协议来存储和共享数据。

在已有的研究当中，NEPOMUK 项目[2]提出了语义桌面系统应具备的功能，并在此基础上提出了社会化语义桌面的设想，让用户在分布式环境下管理、交换和共享桌面数据。Gnowsis[3]根据用户定义的 PIMOS (Personal Information Model Structures) 来描述桌面数据，通过 Aperture10 抽取数据的语义。Gnowsis 对桌面数据采取统一的管理方式，没有对桌面数据进行区分，无论用户在管理论文相关的文档，还是 MP3 文件，都是采用相同的界面展示数据，提供相同的查询功能进行数据查询。Haystack[4]是一个语义数据浏览器，同时提供了插件机制用来扩展其功能，但是不能同其它的应用程序很好的交互。DeepaMetha[5]提供了类似 Mind Map[6]的方式让用户浏览和访问桌面数据。

用户的桌面数据在内容上存在差别，比如有些文档是关于计算机科学，而有些多媒体文件是关于电影，这就需要用不同的本体[7]和元数据对这些内容进行描述，根据内容的特点设计不同的查询功能和不同的数据展示界面。而已有的研究当中都没有根据用户的工作内容对桌面数据进行区分，没有根据桌面数据的特点提供不同的查询和数据展示方式满足用户的应用需求。因此，本文提出了基于内容的语义桌面系统 CSemDM(Content based Semantic Desktop Manager)，它从用户工作内容的角度，提供不同的本体来描述桌面数据，例如提供计算机科学本体和音乐本体分别来描述论文资料和 MP3 文件；设计不同的可视化界面来展示语义数据，使用户能够直观地获取信息，例如采用 Mind Map 方式显示科研文档；采用插件架构，用户可以方便地扩展其功能。

本文组织如下：第二部分介绍 CSemDM 的系统结构，第三部分以计算机研究插件为例，介绍 CSemDM 的工作方式和实现技术。最后，第四部分总

① 基金项目:国家科技基础条件平台项目(2005DKA64100,2005DKA10201)
收稿时间:2009-02-24

结目前的开发状况，并介绍以后的工作。

2　系统结构

NEPOMUK 项目[2]提出了语义桌面系统的通用架构，这是一个概要的架构。对于 CSemDM，我们采用 3 层体系结构，将数据层和表示层分离，方便开发和维护。系统结构如图 1 所示。底层称为语义数据层，包括本地存储和远端存储两个部分，该层的主要目的是完成元数据的存储。本地存储表示在本地的元数据，而远端存储表示存储在远程服务器上的元数据。用户在数据拥有者的许可下，可以从远端存储下载元数据，并存入到本地存储。这样方便了用户之间的数据共享。中间层称为元数据管理层，该层的核心组件是 Jena，它是由 HP Lab 开发的语义 Web 开发框架。Jena 能够从语义数据层读出和写入元数据，提供元数据检索 API。最上层是交互式用户界面，负责同用户交互。该层采用可视化技术，以不同的数据显示方式展示语义数据，因此用户可以根据数据的特点选择不同的方式来浏览数据，例如浏览论文相关的文件可以用树形结构，按照学科门类来展示。语义数据的特点是描述数据的特征和联系，因此可视化技术能够很好地展示语义数据，用户通过这个界面可以非常容易地浏览数据，获取想要的信息。用户界面从元数据管理层获取数据。

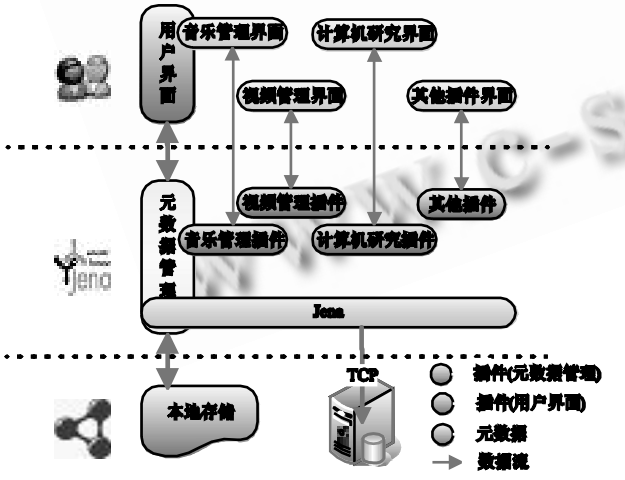

图 1　CSemDM 系统结构

CSemDM 采用插件架构，提供了良好的可扩展性。每一个插件对应一项用户的工作内容，例如计算机研究插件辅助用户完成计算机科研涉及到的文件管理需求。CSemDM 允许用户快速的在各个插件之间切换，各个插件产生的元数据由 W3C 定义的 RDF/XML 来描述，可以直接地被不同插件访问。不同插件提供不同的界面来展示数据，每个插件包含两个部分：用户界面和对应的元数据管理模块。在图 1 当中，元数据管理层的计算机研究模块和用户界面层的计算机研究界面构成了科研插件，其中计算机研究模块完成特定的元数据查询功能，这些功能建立在 Jena 提供的 API 之上。这样实现的原因是不同的插件可能需要实现不同的查询功能，对应到底层实现，就需要不同的元数据查询代码。同时界面和数据管理分离，界面代码可以复用。例如如果视频管理插件和音乐管理插件采用相同的数据展示方式，那么视频管理界面就以直接采用视频插件里的界面代码。

3　系统实现

CSemDM 采用 Eclipse Rich Client Platform (RCP)作为开发基础。Eclipse RCP 可以让我们快速构建跨平台、具备插件机制的应用程序。目前我们已经完成 CSemDM 基础框架的开发，正在设计针对不同桌面内容的插件。我们选择计算机研究插件作为实例来说明其工作方式和开发过程。我们首先介绍该插件用到的本体以及产生的语义数据，然后介绍如何对文档进行标注获取语义数据，再介绍语义查询的设计，最后介绍可视化界面的设计和数据共享的实现。

3.1 本体和语义数据

本体是一系列概念的集合[7]。为了描述文档的主题，我们设计了计算机科学本体，该本体描述了计算机科学当中的概念以及这些概念之间的联系。例如，计算机科学本体包含“计算机科学”、“计算机图形学”这两个概念，它们在本体当中的关系是“计算机科学”包含“计算机图形学”。任何主题是“计算机图形学”的文档，它们的主题也属于“计算机科学”。我们采用目前国际公认的本体来描述文档和作者的其它属性。我们使用 FOAF 本体[8]描述作者属性，这些属性包括姓名、Email、所属组织机构等。为了描述文档的除了主题之外的属性，我们用 Dublin Core[9]本体，它能

够描述文档的路径、作者、发布时间等属性。

用本体进行属性描述之后我们就能得到语义数据，举例来说明语义数据的应用。假如用户想要删除 2008 年之前所写的用于 C++学习的代码，这些文件分布在不同的目录里面，文件数目比较多，即使借助第三方软件，整个删除工作也比较费时，用户需要到各个文件夹选中文件然后再删除，而且用户可能忘了一些文件的存储位置。对于 CSemDM 来说，利用语义数据，通过语义查询返回待删除文件的路径，再删除这些路径所指的文件，两步操作就可以轻松完成。

3.2 语义标注

CSemDM 需要对文档进行标注才能获取语义数据。CSemDM 为用户提供了拖拽的方式来获取语义数据。用户只需将文档拖到 CSemDM 界面，系统就会抽取相应的语义数据。例如在 Windows 系统，CSemDM 会根据 PDF 文件在操作系统里的属性获取语义数据，如图 2 所示。当然，并非所有文档都包含这些属性，有些属性项可能为空。对于语义 Web 研究来说，如何实现自动化地从非结构化文档中抽取语义数据仍然是一个挑战[10]。CSemDM 采用同用户交互的方式，先抽取可以自动获取的语义数据，同时允许用户补充空缺的属性。这里面会涉及用户体验的问题，Semantic eInk[11]提供了具备更佳用户体验的获取语义数据的方式，CSemDM 会结合这些研究进行改进。

图 2　PDF 文档在 Windows 系统里的属性

3.3 语义检索

利用获取的语义数据和 W3C 的 SPARQL 查询语言，CSemDM 能为用户提供语义查询，实现对数据的精确查找。例如，用户想要查询“作者是来自 W3C、发表于 2002 到 2008 年之间的、主题是关于 Semantic Desktop 的所有论文”，对应的 SPARQL 语句如下：

```
<!--名字空间声明，dc 和 foaf 为本体名字空间缩写-->
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
<!--将文章 id 从数据集中选取出来-->
SELECT ?id
<!--WHERE 子句指明检索条件-->
WHERE {
    <!--作者必须为组织 group 的成员-->
    ?name foaf:member ?group.
    <!--group 的名字为 W3C-->
    ?group foaf:name “W3C”.
    <!--文章的作者姓名是?name 的值-->
    ?id dc:creator ?name.
    <!-- 文章的讨论主题是 Semantic Desktop-->
    ?id dc:subject “Semantic Desktop”.
    <!--文章的发表时间是变量?time 的值-->
    ?id dc:date ?time.
    <!--限定变量?time 的取值范围-->
    FILTER ( ?time >= 2002 && ?time <= 2008 )
}
```

通过语义查询，用户无需关心文件的存放位置，只需要给出数据的特征，例如文件的主题、作者等信息，就可以精确地找到想要的信息。CSemDM 根据数据内容在不同的插件提供不同的查询功能，例如对于论文，用户需要通过论文作者进行查找，而对于 MP3 文件，用户可能需要通过歌词中的关键词进行查找。CSemDM 提供灵活的查询来满足用户的需求。

3.4 Mind Map 用户界面

根据科研工作的特点，我们提供 Mind Map 的方式展示桌面数据，记录用户的思路。Mind Map 可以用来组织想法和设计思路，可以很好地展示数据之间

的联系，使得用户能够方便地管理自己的想法和思路。用户在平时使用桌面系统的时候会按照一定的思路进行操作，例如用户学习 Semantic Web 技术，先学习基础概念和技术，包括本体、RDF、OWL、SPARQL 等，然后再研究实际项目包括 NEPOMUK、Haystack 等，如此一直下去。在这个过程中用户会操作对应的文件，比如学习本体的时候下载本体相关的论文，或者查找系统中已有的本体论文。把这个学习过程以及涉及到的文件用 Mind Map 表现出来，如图 3 所示。CSemDM 会把用户的学习思路记录下来，包括学习中各个阶段所涉及的文档、想法等信息。这样做的好处是可以能够帮助用户管理思路，按照用户的思路对文件等进行组织，以一种符合用户思路的方式管理桌面数据。

图 3　Mind Map

利用 Mind Map 也可以很好的展示数据之间的联系，CSemDM 产生的语义数据本身就描述了数据之间的联系。根据这个特点和 Mind Map 的好处，CSemDM 提供了 Mind Map 界面，使得用户根据自己的思路来组织和管理文档。同时 CSemDM 允许用户保存当前的 Mind Map，使得用户能够保持连续的工作思路。图 4 展示了 CSemDM 的 Mind Map 界面。在这个界面里面，用户还可以完成文件添加、打开、删除等操作。

3.5 数据共享

CSemDM 提供的数据共享机制使得用户之间可以交换各自的桌面数据。这样做是很有意义的，例如对于刚刚进入 Semantic Web 研究领域的人来说，获取专家的 Mind Map，能够帮助他快速地知道需要读

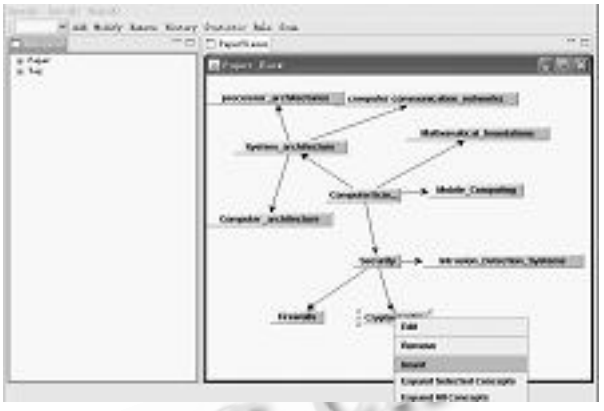

图 4　CSemDM 的 Mind Map 界面

哪些论文，需要掌握哪些工具。数据共享方便了用户之间的学习和交流，以 RDF/XML 描述的语义数据不仅可以方便的在 CSemDM 程序之间共享，也可以方便的同别的语义 Web 应用程序共享，因为 RDF/XML 定义了统一的数据表示标准。为了实现数据共享，我们提供数据共享服务器，用户可以上传自己的语义桌面数据，同时在别的用户的许可下获得别人的语义桌面数据。

4　结论

CSemDM 利用语义 Web 技术，根据桌面内容的特点，记录数据的语义，帮助用户管理桌面数据。用户可以将论文主题、工作思路等信息记录在语义数据当中，使得 CSemDM 能够理解数据的语义，能够提供精确的语义检索来替代基于关键字的检索，减轻用户桌面管理的工作量，提高工作效率。同时，CSemDM 提供了数据共享能力，促进用户之间的交流。插件机制能够使用户方便地扩展其功能，扩充管理范围。

目前，CSemDM 还在开发完善中，以后我们会将其开源并建立开发社区，让更多的开发者参与开发，丰富插件。同时我们还会研究语义数据获取方法，目前的方法需要比较多的用户操作。对于共享数据，我们希望能够对用户上传的数据进行挖掘来获得有用的信息，例如根据用户的语义桌面数据对语义 Web 的研究方向进行关注度排行，给研究人员参考。

参考文献

1 Sauermann L.The gnowsis-using semantic Web technologies to build a semantic desktop [Diploma Thesis]. Technical University of Vienna, 2003.

2 Groza T, Handschuh S, Moeller K, Grimnes G, Sauermann L, Minack E, Mesnage C, Jazayeri M, Reif G, Gudjonsdottir R. The NEPOMUK project - on the way to the social semantic desktop. Proc. of I-Seman tics' 07. Austria:JUCS, 2007.201－211.

3 Sauermann L, Grimnes G, Kiesel M, Fluit C, Maus H, Heim D, Nadeem D, Horak B, Dengel A. Semantic desktop 2.0: The gnowsis experience. Proc. of the ISWC Conference. 2006.887－900.

4 Dennis Q, David H, David K. Haystack:A Platform for Authoring End User Semantic Web Applications. 2nd International Semantic Web Conference (ISWC2003). LN CS 2870. Heidelberg: Springer-Verlag, 2003.738－753.

5 Richter J, Volkel M, Haller H. Deepamehta-a semantic desktop. Proc. of the 1st Workshop on The Semantic Desktop-Next Generation Personal Information Management and Collaboration Infrastructure at the International Semantic Web Conf., Galway, Ireland, 2005.

6 Buzan T, Buzan B. The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential. New York: Plume, 1996.43－48.

7 Tom G What is an ontology? International Journal of Human-Computer Studies, 1995,43(4-5):907－928.

8 Grimnes G, Edwards P, Preece A. Learning Meta-descriptions of the FOAF Network. Proc. 3rd International Semantic Web Conf. (ISWC 2004). LNCS 3298. Heidelberg: Springer-Verlag, 2004.152－165.

9 Weibel S. The Dublin Core: A simple content description format for electronic resources. NFAIS Newsletter, 1999,40(7):117－119.

10 Paul F, Li Z, Kurt M, Steven Z, Mohammad Z. Automated Template-Based Metadata Extraction Architecture. ICADL 2007. LNCS 4822. Heidelberg: Springer-Verlag, 2007.327－336.

11 Marcus L, Kinga S, Andreas D, Nadir W, Beat S, Moira N. Pen and paper-based interaction with the semantic desktop. Proc. of DAS 2008, 8th IAPR International Workshop on Document Analysis Systems. Nara, Japan, 2008.954－958.