

Java 内容仓库及其在 CMS 中的应用

薛胜军, 成 敏

(武汉理工大学 计算机科学与技术系, 湖北 武汉 430063)

摘 要: Java 内容仓库被设计为一套标准 API 来解决内容管理系统(CMS)领域内容仓库无法统一的问题。它位于应用系统和底层存储层之间, 使用树状存储模型, 提供了诸如内容存取、版本控制、事件、检索和过滤等内容服务。实现了内容访问与存储仓库解耦, 提供了更高的灵活性和交换能力。概述了 Java 内容仓库的概念、原理及其结构, 介绍了其开源实现项目 Apache Jackrabbit, 结合一个新闻发布系统的应用例子说明其在 CMS 中的应用。讨论了 Java 内容仓库技术的优缺点, 并对其未来进行了展望。

关键词: Java 内容仓库; JSR-170 内容管理系统; Apache Jackrabbit

中图分类号: TP312

文献标识码: A

文章编号: 1673-629X(2009)01-0241-04

Research on Java Content Repository and Application in CMS

XUE Sheng-jun, CHENG Min

(College of Computer Science, Wuhan University of Science and Technology, Wuhan 430063, China)

Abstract: The content repository for Java technology is designed to be a set of standard API to solve the problem that the content repository cannot be unified in content manager system (CMS) area. It is located between the application system and the storage layer, using a 'tree-like' storage model, provides many content services like content access, version control, transaction, search and filter. It realized the decoupling of the content access and the storage repository, provided greater flexibility and the ability to exchange. This context outlined the concept of the content repository for Java technology and its principle and structure, and briefed the open source project Apache Jackrabbit, described the role in CMS with a news release system example. It also discussed the advantages and disadvantages and made a prospect for the future at last.

Key words: JCR; JSR-170 CMS; Apache Jackrabbit

0 引 言

人们日常存储各种信息的内容仓库主要基于以下几种方式: 关系数据库、文件系统、XML。数据库处理规范数据类型十分在行, 但是在处理如图像、文档等二进制数据时却不是那么得心应手。文件系统可以弥补这一点, 但它们既没有提供用于搜索信息的查询语言, 也没有提供表示关系或事务的概念。XML 存储又在海量数据及安全控制方面存在缺陷。不同的特性决定了各种内容仓库无法统一, 但随着各个厂家各自的内容仓库数量上的急剧增长, 人们越来越需要一组通用的编程接口来使用这些内容仓库。

被称作 Java 内容仓库的 Content Repository for Java Technology API (JSR-170) 的目标就是提供这样一

个接口。它提供了一套标准的内容仓库 API, 即, 不论底层资源是什么(如, 后端数据存储可以是文件系统、WebDAV 仓库、支持 XML 的系统, 甚至还可以是 SQL 数据库), API 都将相同。它适用于任何兼容 JSR-170 规范的内容仓库。它是一组由 Java Community Process 开发并于 2005 年 6 月完成的规范^[1]。该规范在 javax.jcr 名称空间下提供一个统一的 API, 允许人们以供应商中立的方式访问任何规范兼容的仓库实现。它在数据存储之上提供诸如访问粒度控制、版本控制、内容事件、全文检索和过滤等内容服务。

内容管理系统(CMS)是一个很广泛的概念: 从商业门户网站的新闻系统到个人的 weblog 都可以称作内容管理系统。如果你曾经开发过内容管理系统, 那么你会非常清楚在实现内容系统时所遇到的固有难题。这个领域缺乏一个统一的标准, 许多供应商都有自己的私有仓库引擎。这些困难恶化了这类系统的复杂性和可维护性、增强了厂商锁定、增加了企业市场中对传统系统长期支持的需要。随着企业 weblog 和电

收稿日期: 2008-05-24

基金项目: 国家自然科学基金(60572015)

作者简介: 薛胜军(1956-), 男, 教授, 博士生导师, 主要研究领域为计算机网络、计算机支持的协同工作、人工智能、智能交通(ITS)。

子企业文档管理的日益流行,CMS 行业对标准化内容仓库接口的需求比以往任何时候都更加显著。因此,使用 Java 内容仓库技术开发 CMS 将成为一种趋势。

1 Java 内容仓库模型原理和存储结构

1.1 Java 内容仓库模型原理

图 1 描述了使用 JSR - 170 模型的原理。Java 内容仓库位于 CMS(或入口应用)和底层的内容存储工具(比如 RDBMS、文件系统、LDAP 服务器、XML 或者其它数据存储机制)之间。从 API 的角度看,JCR 的功能类似于 RDBMS 中的 JDBC。

在应用系统运行的时候,它可以操作内容仓库(Content Repository)1,2,3 中的任意一个。而开发该应用系统时则完全不用关心数据是如何存储的,它可以存储于关系数据库,文件系统,XML,甚至远程内容仓库——只要操作的内容仓库支持(或需要 JSR - 170 驱动间接支持)JSR - 170。目前只有文件系统可以直接支持 JSR - 170,其他内容仓库则需要 JCR 连接桥来支持^[1]。

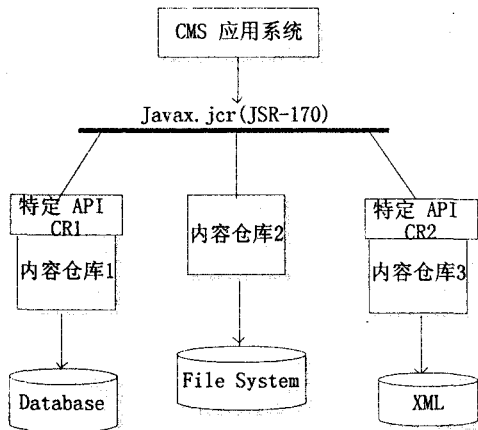


图 1 JSR - 170 模型原理图

1.2 Java 内容仓库存储结构

Java 内容仓库使用“树形结构”保存信息,树由节点和属性组成。如图 2 所示,圆形代表节点,方形代表属性。一个节点有且仅有一个父亲,有任意数目的孩子(子节点)和任意数目的属性。一个属性有且只有一个父亲(它也是节点),它没有子节点,由一个名字和一个或多个值组成。属性值的类型可以是:布尔(Boolean)、日期(Date)、双精(Double)、长整(Long)、字符串(String)或流(Stream)。只有属性可以被用来存储信息,节点则被用来创建树内部的“路径”。在某种程度上,这棵树类似于文件系统的结构,节点是目录,属性是实际的文件^[2]。在实际应用中,这种“树形结构”可以概括所有存储类型结构。这种树状的存储结构非常适合 CMS 的后台存储、全文索引、搜索等操作。

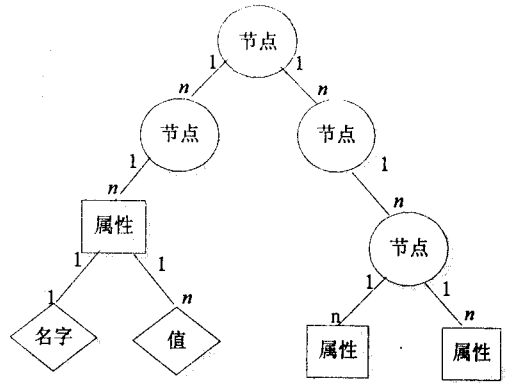


图 2 Java 内容仓库存储模型图

每个节点有且只有一个主节点类型 (primary node type)。主节点类型定义节点的特征,比如允许节点具有的属性 and 子节点。除了主节点类型之外,节点还可能具有一个或多个 mixin 类型。mixin 类型很像装饰器 (decorator),为节点提供额外的特征。具体来说,JCR 实现可以提供三种预定义的 mixin 类型:

mix:versionable,它允许节点支持版本控制。

mix:lockable,它为节点启用锁定功能。

mix:referenceable,它提供一个自动创建的 jcr:uuid 属性,该属性给予节点惟一的、可引用的标识符。

Java 内容仓库的功能被划分为几个级别,每个级别提供一组特定的特性(见表 1)。

表 1 Java 内容仓库的功能划分^[3]

级别	功能	特性
级别 1	提供对仓库的读访问	对节点和属性的读访问 对属性值的读访问 输出到 XML/SAX 支持 XPATH 语法的查询服务
级别 2	在级别 1 的基础上提供写功能	增加和移除节点和属性 对属性值的写操作 从 XML/SAX 输入数据
可选级别	在级别 2 的基础上定义实现五种附加功能	版本控制 事务支持 SQL 查询 内容锁定和监视

Apache Jackrabbit 是 JSR - 170 的开源参考实现,提供级别 1,2 和可选功能。Apache Jackrabbit 完整实现了 JCR API 的内容库。目前 Jackrabbit 发布的版本是 2.0,该版本被认为足够稳定,可以被用在产品环境。除了实现 JSR - 170 中定义的所有特性,Jackrabbit 还加入了额外的功能(如 SessionListeners 或 CustomNode 注册),以及一个项目套件,它包括:JCA 连接器、taglib、WebDAV 接口、虚拟文件系统和 JDBC 后端。

2 基于 JSR - 170 的新闻发布系统的开发

2.1 系统内容仓库模型

图 3 描述了一个新闻发布系统的内容仓库模型。

每个 root node(根节点)的子节点都代表了一个 news 实体。与这个 news 实体有关的数据都存储在 news Entity 节点的属性里,属性类型包括文本内容和二进制文件(图片和视频等)。

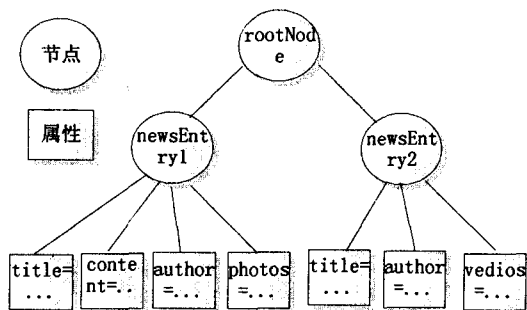


图3 新闻发布系统存储模型图

2.2 内容仓库配置

Jackrabbit 需要两个参数来配置一个内容仓库实例:内容仓库主目录和内容仓库配置文件。这两个参数可以通过两种方式设置,一种是在仓库实例创建时直接传到 Jackrabbit 里去,一种是间接地通过设置 JNDI object factory。

2.3 系统实现

下面简要介绍在内容仓库已配置好的前提下,基于 JSR-170 如何实现新闻发布系统的一些基本功能如内容存取、搜索、导入导出数据、添加二进制内容和版本管理。

(1) 初始化内容仓库。

JSR-170 用 TransientRepository 获得仓库、工作区和根节点:

```
//初始化内容仓库
Repository r1 = new TransientRepository();
Session session = r1.login(new SimpleCredentials("userid",
"".toCharArray()));
//从 session 中获得根节点
Workspace w1 = session.getWorkspace();
Node root = session.getRootNode();
```

(2) 添加和删除新闻内容。

要添加一个内容,需要向仓库添加内容节点。例如要添加一个名为 news 的节点,它包含 title 和 author, content 等属性:

```
Node news = root.addNode("news");
news.setProperty("title", new StringValue("today's headline"));
p.setProperty("author", new StringValue("chengmin"));
p.setProperty("content", new TextValue("xxx...yyy"));
session.save();
```

上面代码段的最后一行代码将保存会话。添加和设置节点以及节点属性只能修改临时的会话存储层。

要将这些变化保存到仓库中,则必须用 session.save() 保存会话。可以在目标节点上调用 Node.remove() 来删除内容节点。

(3) 新闻内容存取。

JSR-170 提供了两种存取内容的方法:遍历存取和直接存取。遍历存取包括用相对路径在内容树中进行遍历,直接存取允许用绝对路径直接跳到节点,如果节点是可以引用的,则用 jcr:uuid 直接跳到节点。

(4) 四级标题。

JCR 的 XPath 搜索工具提供了获得特定内容条目的更好方式。从树形结构来看,工作区模型非常类似于 XML 文档,所以 XPath 是查找节点的理想语法。XPath 查询是通过 QueryManager 对象执行的。查询的过程与通过 JDBC 存取记录类似。

(5) 用 XML 导入和导出新闻内容。

为了确保跨 JCR 实现的移植性,可以使用 JSR-170 提供的标准的 XML 导入和导出工具。通过使用这些工具,符合规范的供应商仓库内容可以很容易地转移到另一个符合规范的供应商仓库。使用 XML 进行序列化的另一个优势是:可以用传统的 XML 解析工具操纵导出的仓库。例如只需要以下三行代码就可以执行导出:

```
File outputFile = new File("systemview.xml");
FileOutputStream out = new FileOutputStream(outputFile);
session.exportSystemView("/root", out, false, false);
```

然后可以把生成的 XML 文件导入另一个新仓库:

```
File inputFile = new File("systemview.xml");
FileInputStream in = new FileInputStream(inputFile);
session.importXML("/", in, ImportUUIDBehavior.IMPORT_UUID_CREATE_NEW);
session.save();
```

(6) 添加二进制附件^[4]。

JCR 除了支持常用数据类型如字符型、布尔型和长整型等外,还支持其他类型,包括二进制图片等流类型。下面的代码中,将一个图片文件 rose.gif 添加到 news 节点的子节点 file 节点中。文件数据本身被保存为 pic 节点。实际的图片文件数据包含在 jcr:data 属性中。

```
File file = new File("rose.gif");
MimeType mt = MimeType.getDefaultTable();
//确定内容类型
String mimeType = mt.getContentTypeFor(file.getName());
if(mimeType == null)
mimeType = "application/octet-stream";
```

```
//添加 file 节点
Node fileNode = roseMode.addNode(file.getName(),"file");
//添加 pic 节点
Node picNode = fileNode.addNode("jcr:content","pic");
//设置内容类型
resNode.setProperty("jcr:mimeType", mimeType);
//设置内容编码方式
resNode.setProperty("jcr:encoding","");
//用 FileInputStream 装入文件
resNode.setProperty("jcr:data", new FileInputStream(file));
```

(7) 系统版本管理。

JSR-170 支持许多可选特性,包括访问控制、事务、锁定和版本管理。这些特性本身都可以是个完整的主题,所以必须简要地总结一下,只介绍它们当中最流行的那一个:版本管理。在最简单的情况下,只需将 mix:versionable 混合类型添加到任何节点,就可以执行版本管理^[2]。在节点上,可以用一组类似 CVS 操作的方法实现版本管理:

```
n.checkout(); //版本签出
n.save(); //版本保存
n.checkin(); //版本签入
```

JCR 中的其他操作包括:更新、合并和恢复以前版本等,在此省略。

3 Java 内容仓库在 CMS 开发中的优劣分析

3.1 使用 Java 内容仓库所带来的好处

从系统架构角度来说,Java 内容仓库在系统中的实际作用类似于 JDBC 在数据库系统中的作用。所不同的是,JDBC 是基于数据库,而 Java 内容仓库则是基于内容仓库,而这个内容仓库可以是 RDBMS、文件系统、LDAP 服务器、XML 或其它数据存储机制。利用 Java 内容仓库技术开发 CMS 将带来以下好处:

(1) 提供统一接口,解决了 CMS 领域内容仓库 API 无法兼容的问题。

JSR-170 提供了一组统一接口,打破了传统 CMS 领域技术上的分立局面,降低了 CMS 系统的复杂度,增强了可维护性。使用 JSR-170,对于开发者来说,无需了解厂家的仓库特定的 API,只需要该仓库兼容 JSR-170 就可以通过 JSR-170 访问。

(2) 提供了更高的灵活性和交换能力,解决了分布式集成问题。

由于提供了一个抽象层,JCR 在混合和匹配 CMS 产品以及内容存储工具方面具有了更高的灵活性和交换能力。此外,它可以简化分布式应用不同部分之间的集成,也就是说,一个简单的仓库就可以为多个客户端服务(前提是他们使用 API),反过来,一个单一的应

用也可以更简便地访问不同的仓库。对于使用 CMS 的公司则无需花费资金用于在不同种类 CMS 的内容仓库之间进行转换。

(3) 实现了内容访问与存储仓库解耦,延长了 CMS 系统存储方案生命周期。

内容访问与存储内容分离后,应用程序或者内容存储机制中的某些部分发生变化后不会影响其它部分。这样 JCR 就可以降低购买 CMS 相关产品的风险:取缔过时或者达不到性能要求的软件时,底层内容仓库几乎没有什么改变或者根本没有变化^[5]。此外,如果能够构建或者购买用于 Legacy 应用的 JCR 桥,原有(legacy)存储方案的生命周期将加长。对于 CMS 厂家来说,无需自己开发内容仓库,而专注于开发 CMS 应用。

3.2 JCR 技术普及中的限制

由于 JCR 是一项处于发展过程中的技术,在现阶段普及于 CMS 领域还存在一些限制:

(1) JCR 在现阶段 CMS 市场中实现的困难。

由于市场上的内容仓库缺乏统一的 API,JCP 要想在市场中立足,必须采取两个重要步骤:必须重写 CMS 应用,以便通过 API 请求或者提交内容;必须建立与通用企业内容存储技术的连接器或者扩展。当然上述目标并非一朝一夕的事,因为直到人们认同 JCR 是一项关键技术,CMS 市场和内容存储公司才会投资实现 JCR。

(2) 实施 JCR 带来的风险问题。

采用 JCR 的 CMS 厂商可能同样会面对微软向 XML 格式转化时遇到的问题:如果微软开放自己的格式,公司就必须推进升级周期、满足客户需求,并保证遵循标准所带来的综合利益,这些都是有风险的。

内容管理系统(CMS)脱离过去单一的专利模式是必然趋势,它正在向更灵活、互操作性更强的体系结构转化,从而摆脱不同厂商的限制,实现 CMS 栈不同层的互操作,而 JCR 是这个进程中的重要一步。这有利于各个公司建立自己的 CMS 方案,有利于开发人员在其上的操作。此外,虽然某些公司在前进的道路上会遇到困难,它对整体 CMS 产业是有益的。

4 JSR-170 的应用前景

尽管 JSR-170 已经于 2005 年 5 月完成,Java 内容仓库的工作并没有终止^[6]。JSR-283 是 JSR-170 的后继版本,将聚焦于功能增强,如远程支持,客户端/服务器协议映射和扩展内容模型的能力。同时还存在着一些 JSR 之外的想法和项目:绑定/映射框架,它可

(下转第 247 页)

分量中缺少的舌体可在 S 分量和 V 分量中得到。设 P_H 为分量 H 的二值图像, P_S 为分量 S 的二值图像, P_V 为 V 分量二值图像, 则处理后图像为 P, 可以通过公式 (4) 进行图像计算, 得出舌图初始分割效果 (图 1(d))。

$$P = P_H - P_S + P_V \quad (5)$$

从最终分割的效果可以看出其效果优于 RGB 模型, 不仅能分割出舌体, 而且保持了齿痕舌边缘凹凸不平的特征, 并且除了边缘的少量噪声外舌表面保持整体统一的像素值。只要对这张图像进行边缘平滑后取最大面积的连通区就可以得到完整的舌边缘。这种处理的效果不仅很好地分割了舌体而且在为后面的边缘检测和进一步的齿痕识别是一个良好的开端。

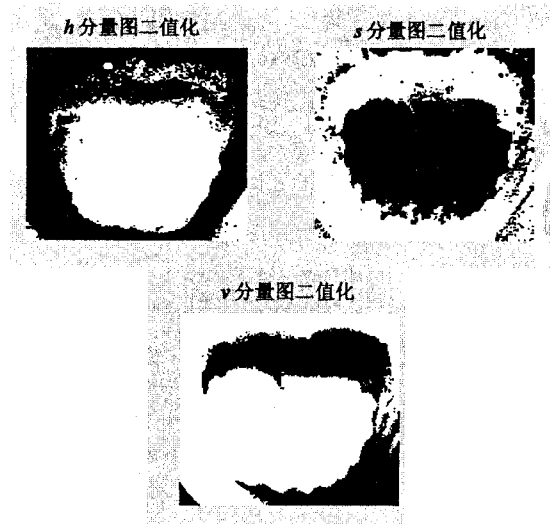


图 4 分量处理效果图

2 实验结果分析

舌体分割一直是中医数字化的难点和热点^[6~8], 如何分割出理想的舌体图像引起许多学者的研究。文中采取的方法是为了得到齿痕舌体边缘特征。此方法用 Matlab 实验, 对要求齿痕识别的一类图像有很好地分割效果, 不但克服了唇部和脸部色彩的相近性, 而且很好地保留了齿印的特征并且计算量少、速度快。但因样本有限所以在后继的工作中应该实验更多的样本, 并通过研究一定的特征实现更详细的分类后, 再进一步研究舌体分割和识别的方法。

参考文献:

- [1] 李乃民. 中医舌诊大全[M]. 北京: 学苑出版社, 1995: 1-525, 1224-1347.
- [2] 金芬芳. 齿痕舌的现代研究概况(综述)[J]. 北京中医药大学学报, 2002, 25(1): 57-59.
- [3] 张永涛. 数字舌图的分析方法与齿痕舌上的应用研究[D]. 北京: 北京中医药大学, 2005: 24-30.
- [4] 陈群, 林雪娟, 徐志伟. 中医舌象计算机识别技术的研究概述[J]. 辽宁中医杂志, 2006, 33(2): 151-153.
- [5] 马超. 中医舌诊图像分割和特征提取方法研究[D]. 重庆: 重庆大学, 2007: 21-39.
- [6] 余兴龙, 谭耀麟, 竺子民, 等. 中医舌诊自动识别方法的研究[J]. 中国生物医学工程学报, 1994, 13(4): 336-342.
- [7] 刘关松, 许建国, 高敦岳. 舌图像自动分割方法[J]. 计算机工程, 2003, 29: 63-74.
- [8] 赵忠旭. 基于数学形态学和 HIS 模型的彩色舌图像分割[J]. 北京工业大学学报, 1999, 25(2): 67-71.

(上接第 244 页)

以将 Java 类转换为一个 JCR 树, 反之亦然 (类似 ORM, 后端用 Java 内容仓库替代数据库), 建构于 JCR 之上的 WebDAV 服务器等。已经出现了用于不同产品的 JSR-170 连接桥, 如 Alfresco、BEA Portal Server 和 IBM Domino^[3]。很显然, JCR 的应用前景看起来一片光明。

5 结束语

分析了 Java 内容仓库的存储模型和原理结构, 结合一个新闻发布系统的例子说明了其在 CMS 领域的应用, 分析了该技术的优点和限制并对其前景进行了展望。

参考文献:

- [1] Patil S. What is Java Content Repository[EB/OL]. [2006-10-04]. [http://www.onjava.com/pub/a/onjava/2006/10/](http://www.onjava.com/pub/a/onjava/2006/10/04/what-is-java-content-repository.html?page=1)

04/what-is-java-content-repository.html?page=1.

- [2] Barik T. Introducing the Java Content Repository API[EB/OL]. [2006-06-06-27]. <http://www.ibm.com/developerworks/java/library/j-jcr/?S-TACT=105AGX52&S-CMP=cn-a-j>.
- [3] Leau C. 集成 Java 内容仓库和 Spring[EB/OL]. 胡键泽. [2008-02-12]. <http://www.infoq.com/cn/articles/spring-modules-jcr>.
- [4] Sommers F. Catch Jackrabbit and the Java Content Repository API[EB/OL]. [2006-06-03]. <http://www.artima.com/lejava/articles/contentrepository.html>.
- [5] Nuescheler D, Boye J. JSR-170: What's in it for me? From CMS Watch[EB/OL]. [2008-03-03]. <http://www.cmswatch.com/Feature/123>.
- [6] Nuescheler D. JSR 170 Overview: Standardizing the Content Repository Interface, From OSCOM[EB/OL]. [2006-10-24]. <http://oscom.org/events/oscom4/proposals/jsr170.html>.