

基于 Web 构架的软件质量评估工具的设计和实现*

万江平, 卓永乐, 徐 晓
(华南理工大学, 广东 广州 510640)

摘 要: 设计并实现了一个基于 Web 的软件质量评估工具, 该工具使用层次分析法和模糊综合评价定量地对软件质量进行评价。讨论了该工具的软件构架选择、系统分析、系统设计, 包括存储结构、数据库结构, 以及 XML 文件格式等, 给出了工具实现的关键算法。

关键词: 软件质量评估; JSP 构架; MVC 模型; 可扩展标记语言

中图法分类号: TP391. 77 **文献标识码:** A **文章编号:** 1001-3695(2005) 03-0205-03

Design and Realization of Web Architecture-based Tool for Software Quality Evaluation

WAN Jiang-ping, ZHUO Yong-le, XU Xiao
(South China University of Technology, Guangzhou Guangdong 510640, China)

Abstract: In this paper, a Web-based tool for software quality evaluation is designed and implemented. The tool evaluated software quality in quantitative way by using the analytic hierarchy process and fuzzy synthetic judgment. Firstly, the software architecture of the tool are discussed. Secondly, the system analysis and the system design including store structure, database structure and the XML file format are presented. Finally, the key arithmetic in the realization of the tool are offered.

Key words: Software Quality Evaluation; JSP Architecture; MVC Model; XML

1 引言

随着信息技术的发展和网络新经济的到来, 软件已经和正在成为国民经济发展的灵魂和核心, 各行各业都离不开软件, 其可靠性、安全性和可维护性对社会经济甚至国家安全的影响日益明显。要有效地把计算机应用于社会生活各方面, 必须加快软件开发的速度和提高软件的质量。软件质量评估正是提高软件质量的关键环节之一。通过软件质量评估, 我们可以找出造成软件质量问题的因素, 从而可以采取相应的措施。软件质量评估也为软件过程改进奠定基础。而开发基于 Web 的软件质量评估工具可以解决评估数据处理周期长, 信息反馈慢等问题, 提高软件质量评估工作的效率和准确性, 也有利于软件质量管理的制度化。本工具采用 Java + XML 等 Web 技术以及层次分析法和模糊综合评价方法具有较好的通用性^[1]。

2 软件构架的设计

2.1 Web 构架的三层体系结构

JSP(Java Server Pages) 有两种常用模式架构: JSP 规范提出了两种用 JSP 技术建立应用程序的方式。这两种方式分别称作 JSP Model 1 和 JSP Model 2, 它们的区别在于处理批量请求的位置不同。在 Model 1 体系中, 如图 1 所示, JSP 页面独自响应请求并将处理结果返回客户, 数据存取都是由 JavaBean

来完成的。Model 1 体系只适合简单应用的需要。如果不加选择地随意运用 Model 1, 会导致 JSP 页内被嵌入大量的脚本片段或 Java 代码。

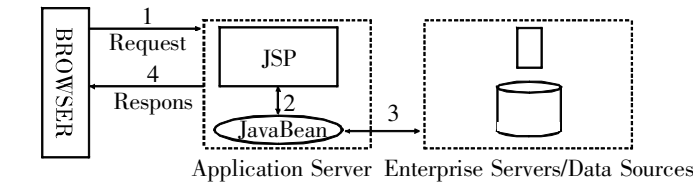


图 1 JSP 模式 1 架构

Model 2 构架, 如图 2 所示, 是一种把 JSP 与 Servlet 联合使用来实现动态内容服务的方法。它吸取了两种技术各自的突出优点, 用 JSP 生成表示层的内容, 让 Servlet 完成深层次的处理任务。Servlet 充当控制者的角色, 负责管理对请求的处理, 创建 JSP 页需要使用的 JavaBean 和对象, 同时根据用户的动作决定把那个 JSP 页传给请求者。JSP 页内没有处理逻辑; 它仅负责检索原先由 Servlet 创建的对象或 JavaBeans, 从 Servlet 中提取动态内容插入静态模板。项目越复杂, 使用 Model 2 构架的好处就越多^[2]。

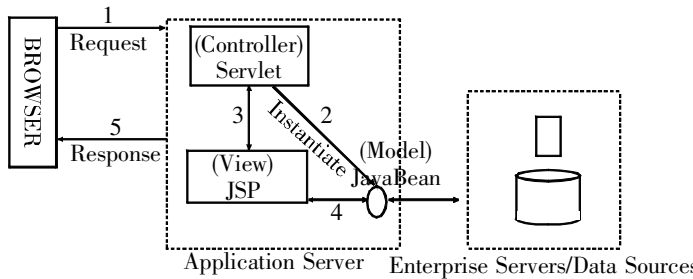


图 2 JSP 模式 2 架构

2.2 MVC 模式

基于 Web 构架的应用系统有两类: 交互式和工作流式。前者需要大量地与使用者交互, 后者只需少量使用者的介入,

甚至可以完全在后台运行。在 B/S 网站式应用程序中,使用者通过提交请求与网站交互,访问设备不断地显示页面相应请求,这是一个典型的交互式应用。架构设计师在设计这样的应用时,普遍采用 MVC(Model- View-Controller) 模式^[3]。

MVC 模式把涉及数据管理和显示的功能分散到不同的对象上,降低对象间的耦合。它把应用分成三部分,分别为模型、视图和控制,并且尽量降低部分间的耦合。每一部分处理特定的任务,并负责完成与其他部分的通信。

2.3 工具的软件构架

本工具的 B/S 三层体系架构采用的 Model 2 构架实现(图 3)。把业务组件分离出来,用 JavaBean 实现;同时用 JSP 作为显示层;对于复杂功能模块的页面控制使用 Servlet 进行选择 and 调用功能。

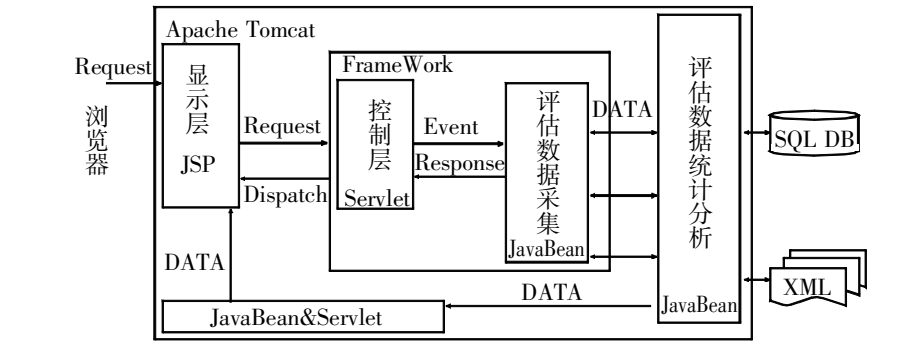


图 3 B/S 三层体系结构

将工具用多层结构表示,分为客户端、交互层、数据采集层和数据处理层。图 4 是本工具的软件构架图。客户端是浏览器,交互层是评估结果输出,数据采集层是用户对待评估项目的评估数据的收集。信息处理层是指标体系中各指标的相对权重的计算和模糊综合评价的模块实现。

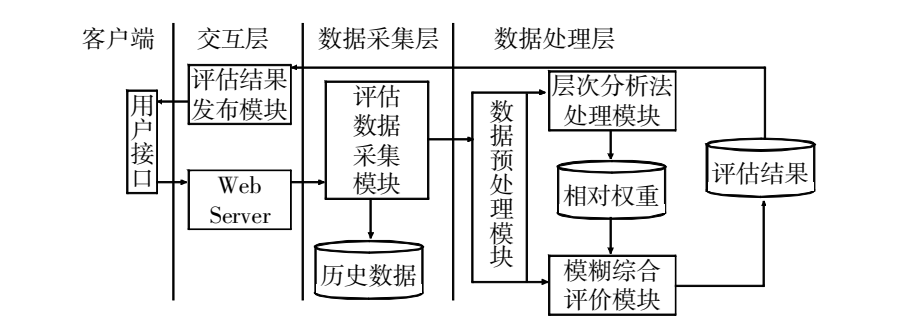


图 4 工具的软件构架

3 系统分析

本工具的功能是以调查问卷的形式,收集用户对某一网站的评估数据(包括评估体系中各指标的重要性比较数据和网站在各项指标的得分),对收集的数据进行预处理,再使用层次分析法确定各个指标相对于上一级指标的相对权重。最后运用模糊综合评价法进行计算,得出对该网站的评估结果(图 5)。

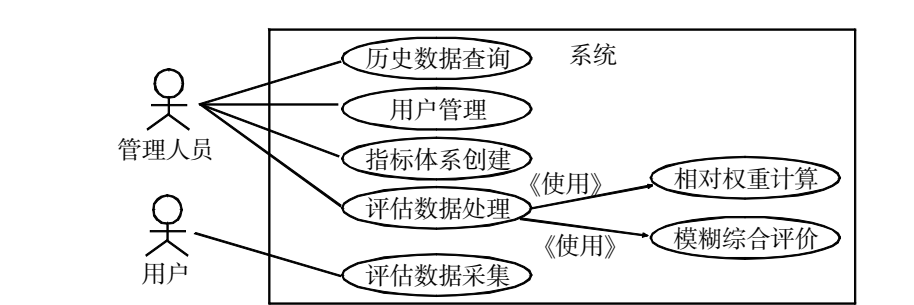


图 5 工具用例图

4 系统设计

4.1 数据存储结构

采用二叉树数据结构表示层次结构中的多叉树。二叉树

中节点的左儿子节点表示多叉树中节点的第一个儿子节点;二叉树中节点的右儿子节点表示多叉树中节点的下一个兄弟节点。

4.2 数据库结构

本工具数据库采用 Microsoft SQL 2000。数据存放在数据库 MyDB 中见表 1~表 7。

表 1 评估结果信息表

Item	Model	UserSum	Excellent	Good	Middle	Passed	Bad	Result
Char(20)	Char(20)	Int	Float	Float	Float	Float	Float	Char(2)

表 2 评估模型信息表

Mode	LayerSum	NodeSum	Le ave Sum
Char(20)	Int	Int	Int

表 3 评估模型指标信息表

Model	Node	Order	Branch	Brother_no	Layer_no
Char(20)	Char(20)	Tinyint	Tinyint	Tinyint	Tinyint

表 4 用户信息表

User	Pass word	Sex	Age	Tel	Major	Company	Duty	WorkingAge	Email
Char(10)	Char(20)	Char(6)	Char(3)	Char(11)	Char(20)	Char(50)	Char(20)	Char(3)	Char(20)

表 5 指标重要性比较信息表

User	Model	Node1	Node2	RelativedWeigh
Char(10)	Char(20)	Char(20)	Char(20)	Char(3)

表 6 模糊综合评价信息表

User	Model	Item	Node	Excellent	Good	Middle	Passe d	Bad
Char(10)	Char(20)	Char(20)	Char(10)	Float	Float	Float	Float	Float

表 7 历史数据信息表

User	Item	Model
Char(10)	Char(20)	Char(20)

4.3 指标体系 XML 文件格式

评估指标编辑器采用 XML 文档存储指标体系的数据。数据以二叉树的节点为单位进行存储。先序历遍二叉树,得到一个节点序列。XML 文档就是按照这个顺序存储所有节点。在数据恢复的过程中,工具按顺序读取 XML 文档中的节点信息,以先序历遍的顺序重新建立与层次图对应的二叉树。

XML 存储格式:

```
<? xml version = "1.0" encoding = "UTF - 8" ? >
< Model >
< ModelName > 网站质量评估指标体系 < /ModelName >
< BinaryTree >
< NodeSumLayer > 1 < /NodeSumLayer > // 层次图第一层中节点数
< NodeSumLayer > 3 < /NodeSumLayer > // 层次图第二层中节点数
...
< NodeItem > // 一个节点的信息
< NodeName > 节点 A < /NodeName > // 节点名称
< NodeDefinition / > // 节点信息备注
< NodeOrder > 1 < /NodeOrder > // 节点在所在层次的排行
< NodeBranch > 3 < /NodeBranch > // 节点的分支数目
< NodeBrotherNo > 0 < /NodeBrotherNo > // 节点的兄弟间排行
< NodeLayerNo > 0 < /NodeLayerNo > // 节点所在的层次
< /NodeItem >
...
< /BinaryTree >
< /Model >
```

4.4 系统实现

4.4.1 数据库的连接

创建数据库连接 sun.jdbc.odbc.JdbcOdbcDriver, 数据库服

务器使用 SQL SERVER 2000。

```
String sDBDriver = " sun.jdbc.odbc. JdbcOdbcDriver" ;
//使用 JDBC-ODBC 桥
String sConnStr = " jdbc: odbc: DBname" ; //要打开的数据库
String strSQL = " Select * From Tablename" ; //要执行的 SQL 语句
Connection conn = null;
Statement stmt = null;
ResultSet rs = null;
try
{ Class.forName( sDBDriver) ; //加载 JDBC-ODBC 驱动
//利用 JDBC 驱动程序连接数据库
conn = DriverManager. getConnection ( sConnStr," user", " pass-
word") ;
stmt = conn. createStatement( ) ; //创建一个 Statement 对象
rs = stmt. executeQuery( strSQL) ; //执行 SQL 语句
... //对记录集的处理
stmt.close() ; //关闭 Statement
conn.close() ; //关闭连接接口
} catch( Exception e) //异常处理
{ out.println( e. getMessage() ) ; }
```

4. 4. 2 运用层次分析法进行数据处理

```
//将判断矩阵 每一列正规化
for( int column = 0; column < currentNode. branch; column ++ )
{ sum[ column] = 0;
for( int row = 0; row < currentNode. branch; row ++ )
sum[ column] + = rweight[ row][ column] ;
for( int row = 0; row < currentNode. branch; row ++ )
weight[ row][ column] = rweight[ row][ column] /sum[ column] ;
}
//每一列经正规化的判断矩阵按行相加
for( int row = 0; row < currentNode. branch; row ++ )
{ sum1[ row] = 0;
for( int column = 0; column < currentNode. branch; column ++ )
sum1[ row] + = weight[ row][ column] ;
sum1[ currentNode. branch] + = sum1[ row] ;
}
//对向量正规化
for( int row = 0; row < currentNode. branch; row ++ )
sum[ row] = sum1[ row] /sum1[ currentNode. branch] ;
//求特征向量(即相对于父节点的权重) 和判断矩阵最大特征根
```

(上接第 204 页) 数据访问的权限进行限定,在每个成员的本地子系统内均有一个根据其权限分配的数据库,该数据库包含系统数据库的一部分数据,供该成员使用。系统管理员负责对所有成员的管理,负责分配各成员本地子系统数据,并对系统数据库进行维护管理。成员的登录窗口如图 6 所示。



图 5 NGDS 原型系统的主页窗口



图 6 NGDS 原型系统的登录与权限管理

4 结论

本文从绿色设计技术的现实要求出发,提出了 NGDS 系统的设计目标、总体结构及核心子系统、平台框架结构、网络工作模式、分布式数据库技术的应用、原型系统的体系框架结构等核心内容。通过该项研究,初步建立了基于网络环境下的绿色设计的技术和方法体系,不仅能适应绿色设计技术水平提升的需要,而且进一步拓展了网络技术的实际应用。

```
λmax
for( int row = 0; row < currentNode. branch; row ++ )
{ sum1[ row] = 0;
for( int column = 0; column < currentNode. branch; column ++ )
sum1[ row] + = rweight[ row][ column] * sum[ column] ;
max + = sum1[ row] /( 3* sum[ row] ) ;
}
```

5 结束语

开发基于 Web 构架的网站质量评估工具应用了 MVC 模式,本工具建立在 Browser /Server/ DataBase 三层构架之上,采用跨平台的 JSP, Servlet, JDBC 技术开发的网站质量评估工具,数据存储采用了 XML,移植性好,使用方便。它还可以解决评估数据处理周期长、信息反馈慢等问题。工具应用了层次分析法和模糊综合评价相结合的方法对软件质量进行评估,可使评估结果更为客观、准确。我们用它进行了软件商品化程度及网站质量的评估,结果比较满意。

参考文献:

[1] hou Yongle, Wan Jiangping, Guo Heqing. Research on Software Characteristics Evaluation and Its Tool[C] . Lan Hua eds. Proceedings of 2003 International Conference on Management Science & Engineering, Georgia, USA, August 15-17, 2003. Harbin Institute of Technology Press, 2003. 234-238.

[2] 理解 JSP 的两种体系结构[EB/OL] . http://tech. china. com/zh_cn/ netschool/ homepage/jsp/3088/20010326/153168. html.

[3] HansBergsten. JavaServer PagesTM[M] . O ’REILLY & Associates, Inc 2001. 37-38.

作者简介:

万江平(1964-), 男, 副教授, 博士, 研究方向为管理决策、软件质量管理和因特网应用; 卓永乐(1978-), 男, 硕士研究生, 研究方向为软件工程、计算机应用; 徐晓(1968-), 男, 讲师, 博士后, 研究方向为人工智能、电讯工程等。

参考文献:

[1] osy W, CHEN D. A System Methodology of Material Selection with Environmental Considerations[J] . IEEE, 1994, 252-257.

[2] Hentschel C , Selige G, Zussman E . Recycling Process Planning for Discarded Complex Products: A Predictive and Reactive Approach [C] . Erlangen, Germany: Proc. CIRP 2 Int. Seminar on Life Cycle Engineering, 1994. 195- 209.

[3] 刘瑞素, 郭兰申, 王阳, 等. 绿色设计和绿色制造是机械行业的发展趋势[J] . 江苏机械制造与自动化, 2001, (4) : 20- 23.

[4] 乌兰木其, 邓家祹. 现代产品设计方法及其演进[J] . 机械工程学报, 2000, 36 (5) .

[5] 刘振岩, 刘慧敏, 王欢. ASP. NET 数据库开发[M] . 北京: 清华大学出版社, 2004.

[6] 信息管理系列. 分布式数据库[M] . 北京: 中国人民大学出版社, 2001.

作者简介:

陈志伟(1961-), 男, 江苏常州人, 副教授, 博士研究生, 主要从事 CAD/CAM 技术和绿色设计与制造技术的研究; 陈志锋(1971-), 男, 江苏无锡人, 副教授, 硕士, 主要研究方向为计算机网络及信息安全; 陈伟(1965-), 男, 江苏常州人, 高级工程师, 硕士, 主要从事 CAD/CAM 技术开发。