

# Struts, Spring, Hibernate 框架在 OA 开发中的应用

刘斌<sup>1</sup>, 王最<sup>2</sup>

(1. 云南大学 软件学院, 云南 昆明 650091; 2. 云南大学 工程技术研究院, 云南 昆明 650091)

**摘要:**随着软件开发技术的发展与演变,为了加速软件开发进程、提高软件生产率、改变软件质量、便于项目维护,文中基于 MVC 模式提出 J2EE 技术平台四层体系结构,分别为表现层、控制层、服务层、持久层。研究和整合当前最流行的轻量级 Struts, Spring, Hibernate (SSH) 框架技术, Struts 为构建 Web 应用程序提供了一个稳定、成熟的 MVC 框架, Hibernate 是目前最流行的持久层解决方案, Spring 为 J2EE 应用程序开发提供集成的框架。最后重点以机构管理模块为例说明 SSH 框架在开发办公室自动化(OA)系统中的应用,为开发管理信息系统提供了较高的理论指导和实际意义。

**关键词:**办公自动化; J2EE; 模型-视图-控制器; Struts; Hibernate; Spring

**中图分类号:** TP311

**文献标识码:** A

**文章编号:** 1673-629X(2010)01-0151-04

## Application of Office Automation Based on SSH Framework

LIU Bin<sup>1</sup>, WANG Zui<sup>2</sup>

(1. School of Software, Yunnan University, Kunming 650091, China;

2. Research Institute of Engineering and Technology, Yunnan University, Kunming 650091, China)

**Abstract:** With the development and evolution of software development technology, in order to accelerate the process of software development, to improve software productivity, to change the software quality and to ease of maintenance projects, MVC based on J2EE technology platform, makes four-storey structure, namely the view layer, control layer, service layer, persistence layer. Research and integration of the most popular lightweight Struts, Spring, Hibernate (SSH) framework technology, Struts provides the stable and mature MVC framework for building web applications, Hibernate is the most popular persistence layer solution, Spring provides an integrated framework for J2EE application development framework. Finally take the management module as an example in order to explain the application of SSH framework in the development of office automation framework (OA) system, it is a high theoretical guidance and practical significance for the development of management information system.

**Key words:** office automation; J2EE; MVC; Struts; Hibernate; Spring

## 0 引言

随着我国信息化建设进程的加快、信息技术的高速发展、Internet 的普及和环保意识的增强, Web OA 系统受到前所未有的关注, 大型企业、政府机构、高等院校等正致力跨时空实现协同工作和无纸化的新型办公方式, 这将节省大量的办公费用, 解放众多的人力资源, 大幅度地提高办公效率, 并为高层领导的决策分析提供充分的依据。

## 1 Struts 框架

MVC 是 Model-View-Controller 的简称, 即模型

-视图-控制器。基于 MVC 模式软件开发, 可以降低系统各部分之间的耦合程度, 有利于开发人员的分工、快速开发, 增强系统的可维护性及可扩展性<sup>[1]</sup>。

MVC 设计模式分离了程序的表现形式、控制和数据, 具有设计清晰、易于扩展、运用可分布的特点, 可适用于多用户的、可扩展的、可维护的、具有很高交互性的系统<sup>[2]</sup>。MVC 模式的广泛应用, 催生了 MVC 框架<sup>[3]</sup>。Struts 是基于 J2EE 平台所有 MVC 框架中最早出现、应用最广的框架, 同时也是开发 Web 应用程序的开源框架, Struts 把 JavaBean, Servlet, Action, JSP 和信息资源整合到一个统一的框架中, 开发人员利用其进行开发时不用再自己编写实现全套的 MVC 模式, 极大地提高了开发效率, 节省了时间, 充分解放程序员的劳动力, 同时也便于项目的维护。

Struts 的体系结构如图 1 所示。

Model 主要负责业务逻辑, 包括数据库的访问与

收稿日期: 2009-05-14; 修回日期: 2009-08-10

基金项目: 中国移动科技应用项目(HT090204)

作者简介: 刘斌(1981-), 男, 陕西渭南人, 硕士研究生, 助理讲师, 主要研究方向为 Web 开发、信息系统分析与集成方面的研究。

操作,通常使用 JavaBean, EJB, Hibernate 来实现业务逻辑。View 主要负责向客户端提供显示功能,是 Model 的外在表现形式,用户通过视图来了解模型状态。视图主要是由 JSP 页面, ActionForm 类, Struts 标签库, 标准标签库构成。控制器相当于分发器,主要负责拦截请求,并将请求分发到相应的业务层,同时将业务层返回的处理结果传递给 View 层呈现给请求者,即将模型与视图匹配在一起共同完成用户请求,主要由 ActionServlet 类, Action 类, Struts-config 配置文件构成。

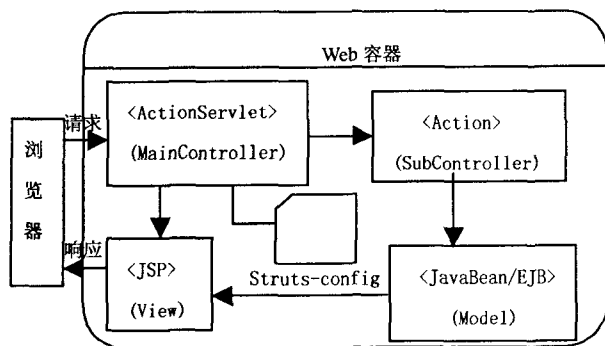


图 1 Struts 体系结构

当启动一个采用 Struts 框架开发的 Web 应用程序时, ActionServlet 就会被加载并初始化。然后 ActionServlet 读取 Struts 配置文件中的信息, 并根据文件中的各模块的配置来初始化相应的配置对象<sup>[4]</sup>。客户端进行请求时, ActionServlet 拦截用户请求并读取配置文件中的信息, 查找匹配该请求的 <action> 子元素, 同时将用户所要提交的表单数据封存到 ActionForm 类对象中并进行表单验证。验证成功后, ActionServlet 将控制权转交给对应的 Action 类, 在 Action 类中进行业务逻辑处理, 包括对数据库的操作和访问, 与此同时, 返回一个 ActionForward 对象, 将控制权交于 ActionServlet, ActionServlet 将根据 ActionForward 对象返回的不同参数把请求转发给指定的 JSP 页面或者执行相应的 Action 类。

## 2 Hibernate 框架

Hibernate 是一种基于 Java 平台开放源代码的对象关系映射(ORM, Object/Relational Mapping) 框架。它对 JDBC 进行了轻量级的封装, 帮助开发者建立面向对象语言中的对象与关系型数据的之间的相互映射。它可以使人们脱离具体的数据库细节, 简化 JDBC 编程, 这样就可以实现程序的面向对象化和数据库的移植<sup>[5]</sup>。并且开发者在设计阶段可以不必关心具体的数据库情况, 使得 Java 程序员可以随心所欲地使用对象思维来操作数据库<sup>[6]</sup>。

Hibernate 技术框架本质上是一个提供数据库服

务的中间件<sup>[7]</sup>, Hibernate 的体系结构如图 2 所示。

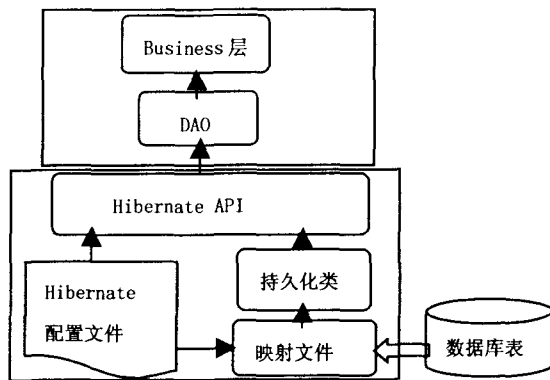


图 2 Hibernate 体系结构

当启动 Hibernate 框架时, Hibernate 的配置文件对数据库进行配置同时加载映射文件, 通过映射文件把持久化类对象和数据库表之间建立一个映射关系。该映射文件描述了数据库表中的信息, 也描述了对应的持久化类对象的信息, 同时, Hibernate 内部封装了通过 JDBC 访问数据库的操作, 向上层应用提供了面向对象的数据访问 API。

## 3 Spring 框架

Spring 是一种开源框架, 它的诞生是为了解决复杂的企业应用<sup>[8]</sup>。Spring 框架以 IoC(控制反转)和 AOP(面向切面编程)两种先进技术为基础, Spring 提供的控制反转(Inversion of Control, IoC)和面向方面编程(Asspect-Oriented Programming, AOP)插件式架构降低了应用组件之间的依赖性<sup>[9]</sup>。框架的主要优势之一就是分层架构, 分层架构允许用户选择使用那一个模块。

Spring 体系结构如图 3 所示。

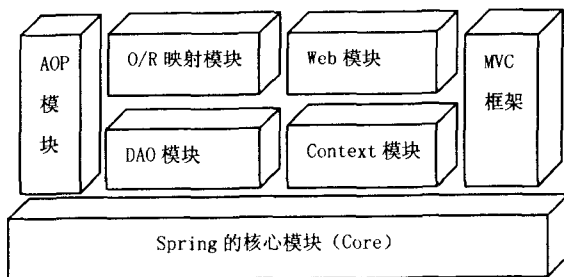


图 3 Spring 体系结构

Spring 由 7 个模块组成。Spring 的核心模块是 Spring 的核心容器, 它的主要组件是 BeanFactory, 负责对 JavaBean 的配置、管理、创建。Spring AOP 模块直接将面向切面的编程功能集成到了 Spring 框架中。可以很容易地使 Spring 框架管理的任何对象支持 AOP, 并为对象提供了事务管理服务。Spring DAO 层提供了 JDBC 的抽象层, 简化了数据库厂商的异常错

误,大幅度地减少代码的编写,并提供了对声明式事务和编程式事务的支持。Spring 的上下文模块继承 BeanFactory 类,并且添加了国际化、事件处理、资源装载、数据校验、企业级服务。Spring ORM 模块集成了现有的 ORM 框架的支持,包括 Hibernate, iBatis。Spring Web 模块建立在 Spring 上下文的基础上,它提供了 Servlet 监听器的 Context 和 Web 应用的上下文。Spring MVC 是一个全功能的构建 Web 应用程序 MVC 框架。

## 4 整合 SSH 框架

SSH 是 Struts - Spring - Hibernate 的缩写,即表示层 - 业务层 - 持久层。SSH 架构是目前最为流行的轻量级开发架构,很多金融、电信项目,政府网站均基于该架构作为业务支撑架构。采用 SSH 框架的优点就是使用不同的三种框架,每层使用它们三种框架中的最优秀部分。Struts 是表示层框架,优点主要集中在 Struts 的页面导航功能以其松耦合形式成功的把 View 和 Model 层进行了分离,通过一个配置文件,即可把握系统各部分之间的联系,使系统脉络更加清晰,便于后期的维护。Spring 框架是一个业务层框架,是一个整合的框架,能够很好地粘合表示层与持久层,整合以后, Spring 将充当了整个应用的工厂。Hibernate 使用 O/R Mapping 技术把模型和关系数据库表进行了映射,通过持久化类和表的映射关系实现了数据持久化层。

整合 Struts, Spring, Hibernate 框架主要分为两步:整合 Spring 和 Struts,利用 Struts 框架的扩展点 PlugIn 来创建 Spring 容器, Spring 框架提供了 PlugIn 接口的实现类 ContextLoaderPlugIn,通过这个实现类, Struts 框架启动时将自动创建 Spring 容器。并采用 Spring 框架中提供了 DelegatingActionProxy 类,用来代理 Struts 框架中的 Action,同时把 Struts 的 Action 处于 Spring 框架的管理中, Struts 框架配置文件中配置如下代码:

```
< plug - in className = "org. springframework.
web. struts. ContextLoaderPlugIn" >
```

```
< set - property property = "contextConfigLocation"
value = "/WEB = INF/applicationContext. xml" /> < /
plug - in >
```

整合 Spring 与 Hibernate,采用 Spring 的 IoC 提供的 LocalSessionFactoryBean 类创建 SessionFactoryBean, Spring 框架启动时将会加载 Hibernate 的配置文件,创建 SesionFactory 实例,将 SessionFactoryBean 注入任何 DAO 类,完成实际的数据库访问与操作。

```
< bean id = " sessionFactory " class = " org.
```

```
springframework. orm. hibernate3. LocalSession-
FactoryBean" >
```

```
< property name = "configLocation" value = "class-
path:hibernate. cfg. xml" />
```

```
< /bean >
```

## 5 基于 SSH 框架实现 Web OA 系统

### 5.1 系统模型

经过做客户的访谈,并进行需求获取与分析, Web OA 的功能需求如图 4 所示。

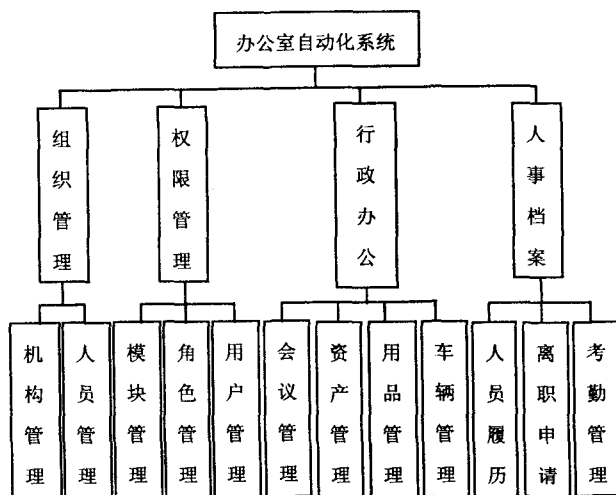


图4 功能需求

### 5.2 体系架构

J2EE 三层结构的划分当然是经典的,但是在实际项目中,往往会对三层体系结构做一些扩展来满足项目的需要。一个最常用的扩展就是将三层体系扩展为四层体系,即表示层 (View)、控制层 (Controller)、服务层 (DAO)、数据持久层 (Data Persistence)。它其实是将三层架构中的模型层进行了拆分,服务层用于操作持久层进行数据操作,数据持久层表示数据库的持久内存存储对象<sup>[10]</sup>。

J2EE 四层体系结构如图 5 所示。

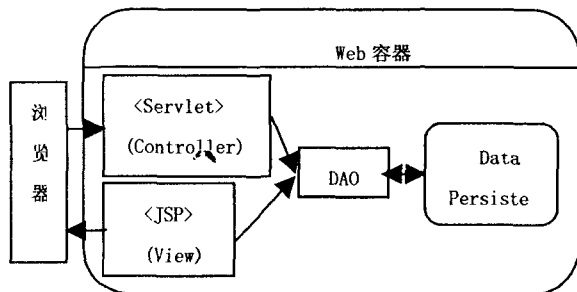


图5 J2EE 四层体系结构

文中整合当今框架的流行技术, Struts 负责表示层、Spring 负责业务逻辑层、Hibernate 负责持久层的数据库的操作。

### 5.3 系统实现

依据模块之间的依赖关系和关联性,机构管理是最基础的模块,其它模块均依赖于机构管理模块,所以下面以机构的管理模块为例介绍整个系统的实现。

#### 5.3.1 数据持久层

在本例中使用完全的面向对象的方法,即先是创建实体类(即持久化类),基于 ANT 和 Xdoclet 自动创建配置文件、映射文件、数据库表。持久化类是其实例被 Hibernate 持久化到数据库中类。持久化对象是一个普通的 JavaBean,包含若干属性,以及对应的 Getter 和 Setter 方法,每一个持久化类对应于数据库中的一张数据库表。在机构管理模块中创建持久化类 Orgnization 并在源代码中添加注释,基于 ANT 与 Xdoclet 生成配置文件 Orgnization.hbm.xml 及配置文件 hibernate.cfg.xml,当启动项目时,会根据配置文件及映射文件自动创建数据库表 Orgnization。

#### 5.3.2 服务层(DAO)

数据访问采用 DAO(数据访问对象)模式<sup>[11]</sup>,DAO 是数据库访问接口,向客户隐藏数据库的具体实现,降低了客户端、数据库、数据库访问的耦合度,增强了系统的重用性,有利于项目开发时功能的划分和工作分配,在项目维护时也更容易定位和解决问题。定义接口类 OrgnizationDao 和接口实现类 OrgnizationDaoImpl,在接口类 OrgnizationDao 中声明要实现的操作函数添加机构(addOrg)、删除机构(delOrg)、更新机构(updateOrg)、查找机构(findOrg),接着在接口实现类 OrgnizationDaoImpl 中依次实现该接口中的方法。紧接着在 applicationContext.xml 中配置该组件的 Bean 对象,并为该对象注入 SessionFactory 对象。

```
<bean id="OrgnizationDao" class="com.skdynu.
oa.OrgnizationDaoImpl"><property name="session-
Factory" ref="sessionFactory"/></bean>
```

同时为该 DAO 对象配置事务组件,使得对 OrgnizationDaoImpl 的访问在 Spring 的事务监管下。

#### 5.3.3 控制层

在增加、修改、删除一个机构对象时,需要一个表单来保存各个字段的数据,因此需要建立一个表单类 OrgnizationActionForm.java。该类是一个普通的 JavaBean,定义了与数据库表中的字段相对应的属性,并为每一个变量添加 Getter 和 Setter 函数。最后在 struts-config.xml 中添加一个 <form-beans>,命名为 orgForm,指向上面新建的表单类 OrgActionForm.java。

```
<form-bean name="orgForm" type="com.sk-
dynu.oa.forms.OrgnizationActionForm">
```

```
</form-bean>
```

对于机构管理模块,添加如下映射请求配置:

```
<action path="/org" type="org.springframework-
work.web.struts.DelegatingActionProxy" name =
"orgForm" scope="request" parameter="method">
<forward name="index" path="/org/index.jsp"
></forward>
<forward name="add" path="/org/add.jsp"/>
<forward name="del" path="/org/del-sucessful.
jsp"/>
<forward name="update" path="/org/update-
sucessful.jsp"/>
<forward name="find" path="/org/find.jsp">
</forward>
</action>
```

上面配置的 <action> 元素都使用了 org.springframework.web.struts.DelegatingActionProxy 来委托给 Spring 进行管理,因此需要在 applicationContext.xml 中添加请求代理配置组件。并在 applicationContext.xml 中添加请求代理。

```
<bean name="/org" class="com.skdynu.oa.ac-
tions.OrgAction" scope="prototype">
```

```
<property name="orgDao" ref="Orgnization-
Dao"/></bean>
```

已经在 Struts-config.xml 中配置了请求地址,并在 application.xml 中配置了请求代理,请求会通过 class 属性找到 Action 类 OrgAction,该类才是真正处理用户请求的地方。

OrgAction 继承自 Struts 的基类 DispatchAction,在该类中要实现以下几种页面请求函数。处理添加机构页请求 addOrg()、处理删除机构页请求 delOrg()、处理更新机构页请求 updateOrg()、处理查找机构页请求 findOrg()。

#### 5.3.4 表示层

通过开发 JSP 页面,将页面的链接地址、表单请求指向 <action>,由 Struts 和 Spring 来将请求转发到 OrgAction.java 的方法中,实现整个页面的流转。

## 6 结束语

框架技术的开源性,调动了全世界所有的编程人员参与到了 Java 技术的进程中,进而产生了众多优秀的框架,Struts 框架为构建 Web 应用程序提供了最为稳定的 MVC 框架,Hibernate 框架在持久层中解决方案中性能卓越,Spring 框架由于其 IoC 和 AOP 的突出

(下转第 162 页)

## 参考文献:

- [1] Mitola J, Maguire G Q. Cognitive radio: making software radios more personal[J]. IEEE Pers Commun, 1999, 6(4): 13 - 18.
- [2] Mitola J. Cognitive radio: An integrated agent architecture for software defined radio[D]. Stockholm, Sweden: Dept. of Teleinformatics, Royal Institute of Technology (KTH), 2000.
- [3] Pradeep K, Vaidya N H. Protocol design challenges for multi-hop dynamic spectrum access networks[C]//IEEE DySPAN'05. Baltimore: [s. n.], 2005: 645 - 648.
- [4] Mansoor A, Randeep B, Li Li. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks[C]//ACM Mobicom'05. Cologne, Germany: [s. n.], 2005: 58 - 72.
- [5] So Junming, Vaidya N H. A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single transceiver[R]. US: University of Illinois at Urbana-Champaign, 2004.
- [6] Krishnamurthy S, Thoppian M, Venkatesan S, et al. Control channel based MAC-layer configuration, routing and situation awareness for cognitive radio networks[C]//IEEE MILCOM'05. Atlantic City, USA: [s. n.], 2005: 455 - 460.
- [7] Gong Michelle X, Midkiff Scott F. Distributed channel assignment protocols: A cross-layer approach[C]//IEEE WCNC'05. New Orleans, USA: [s. n.], 2005: 2195 - 2200.
- [8] Xin Chunsheng, Xie Bo, Shen Chien-Chung. A novel layered graph model for topology formation and routing in dynamic spectrum access networks[C]//IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks. Baltimore, MD, USA. Piscataway, NJ, USA: IEEE, 2005: 308 - 317.
- [9] Cordeiro C, Challapali K, Birru D, et al. IEEE 802.22: the first worldwide wireless standard based on cognitive radios[C]//IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks. Baltimore, MD, USA. Piscataway, NJ, USA: IEEE, 2005: 328 - 337.
- [10] Xu Chen-guang, Xu Yin-long, Wu Jun-min. On the minimization of the number of forwarding nodes for multicast in wireless ad hoc networks[C]//ICCNMC. Berlin, Germany: Springer-Verlag, 2005: 286 - 294.
- [11] Ruiz P M, Gomez-SKarmeta A F. Heuristic algorithms for minimum bandwidth consumption multicast routing in wireless mesh networks[C]//ADHOC - NOW. Berlin Heidelberg: Springer-Verlag, 2005: 258 - 270.
- [12] Lam P K, Chiu Dah-Ming, Liu J C S. On the access pricing and network scaling issues of wireless mesh networks[J]. IEEE Transactions on Computers, 2007, 56(11): 1456 - 1469.
- [13] Anderegg L, Eidenbenz S. Ad hoc-VCG: A truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents[C]//Ninth Annual International Conference on Mobile Computing and Networking. San Diego, CA, United States: Association for Computing Machinery, 2003: 245 - 259.
- [14] Zhong S, Li L E, Liu Y G, et al. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks - an integrated approach using game theoretical and cryptographic techniques[C]//MobiCom'05. Cologne, Germany: Association for Computing Machinery, 2005: 117 - 131.
- [15] Wang W, Li X-Y, Wang Y. Truthful multicast routing in selfish wireless networks[C]//MobiCom'04. Philadelphia, PA, United States: Association for Computing Machinery, 2004: 245 - 259.

-----

(上接第 154 页)

性能成为粘合 J2EE 各层开发的粘合剂。这三者的结合使得它们的性能更加优越,成为软件开发最为经典的轻量级开发架构。

## 参考文献:

- [1] 胡峰松,熊建新.基于 J2EE 技术的办公自动化系统研究[J].江西科学,2008,26(2):300 - 301.
- [2] 杜威,邹先霞,常会友.基于 MVC 模式的 OA 系统结构设计[J].福建电脑,2007(1):15 - 16.
- [3] 李刚.整合 Struts + Hibernate + Spring 应用开发详解[M].北京:清华大学出版社,2007.
- [4] 王国辉,王毅,伊相群.Java Web 开发技术方案宝典[M].北京:人民邮电出版社,2008.
- [5] Hibernate Reference Documentation[EB/OL]. 2005 - 04. <http://www.hibernate.org/hib-docs/v3/reference/en/html/>.
- [6] 田秀彦.论 J2EE 中 Hibernate + Spring 架构及其在 OA 系统开发中的应用[J].科技信息,2008(1):78 - 79.
- [7] 龚雪冰,何彪.基于 Tapestry + Spring + Hibernate 框架的 Web 应用[J].计算机技术与发展,2007,17(4):131 - 135.
- [8] Walls C, Breidenbach R. Spring in Action[M]. [s. l.]: Manning Publications Co, 2005.
- [9] Johnson R. Spring Framework reference documentation[EB/OL]. 2004. <http://www.springframework.org/documentation>.
- [10] 刘中兵.开发者突击:Java Web 主流框架整合开发:J2EE + Struts + Hibernate + Spring[M].北京:电子工业出版社,2008.
- [11] 贾昆,甘仞初,高慧颖.数据访问对象模式在企业应用集成中的应用[J].计算机工程与设计,2006,27(3):373 - 375.