

基于 Struts2 技术的社交网络服务平台^①

陈默^{1,2}, 杨小平¹, 李瑶², 李震宇²

¹(中国人民大学 信息学院, 北京 100872)

²(北京联合大学 商务学院, 北京 100025)

摘要: 以应用 Struts2 技术为核心, 提出一种社交网络服务平台的设计方案. 通过对用例模型的设计、对基于 MVC 设计模式的 Struts2 技术的应用、对用例实现算法的研究等过程, 该实施方案将平台结构进行清晰的逻辑划分; 可将三层逻辑结构进行有效的分离; 可为基于主流框架技术的平台设计提供一个借鉴的实例; 可为平台商业决策的制定提供一个技术性的分析环境.

关键词: Struts2; 社交网络服务; MVC; 用户行为序列

Social Networking Service Platform Based on Struts2 Technology

CHEN Mo^{1,2}, YANG Xiao-Ping¹, LI Yao², LI Zhen-Yu²

¹(School of Information, Renmin University of China, Beijing 100872, China)

²(School of Business, Beijing Union University, Beijing 100025, China)

Abstract: This paper will put forward a design scheme of social networking service platform based on Struts2 technology. Through the design of use-case model, the application of Struts2 technology based on the design mode of MVC, and the research of use-case implementation algorithms, the design scheme proposed in this paper can make clear logical division to the platform structure and effective separation for three-tier logical structure. It does not only provide a reference example for the design of the networking application platform based on the mainstream framework technology, but also a technical and analytical condition for formulating the business decisions of social networking service platform.

Key words: Struts2; social networking service; model view controller; user behavior sequence

1 引言

人类社交活动激发了网络技术的迅猛发展, 并对基于 Web 应用的各种服务产生了一定的影响. 社交网络服务(Social Networking Service, SNS)不仅表现了社会、网络和服务之间的紧密关系, 而且也体现了人类社交关系所具有的高度自组织特性和分布式特点^[1]. 基于对社交网络服务平台的用例分析, 以平台组件中的用例设计过程为例, 本文将提出一种基于 Struts2 技术的社交网络服务平台设计方案.

2 社交网络服务平台的设计背景

社交网络服务可引导用户建立社交网络关系, 体

现了以用户需求为核心、以注重个性化和交互性为特点的 Web2.0 应用思想^[2]. 中国互联网数据中心近期发布的统计数据表明, 从用户访问总量、单位时间内受访比例、用户访问量动态变化、社交网络行业日均有访客访问的站点数量和日均访问行业站点的用户数量等维度观察, 我国网络用户也正在充分利用社交网络服务进行着沟通与交流.

不同于一般的门户网站、操作型网站和业务处理型网站, 社交网络服务平台有着自身特殊的应用背景. 除了在社交网络中进行沟通与交流以外, 各类用户所需的应用服务也正逐步加入到该平台中, 社交网络服务现已成为用户分享各类应用服务和体验的需求平台.

① 基金项目: 国家自然科学基金(71271209); 北京市自然科学基金(4132067); 教育部人文社会科学青年基金(11YJC630268);

北京市教委科技计划面上项目(KM201311417011)

收稿时间: 2013-06-09; 收到修改稿时间: 2013-07-11

绑定费用均为支付费用的两个实现用例。

3.2 企业用户角色用例的设计

在上述设计的社交网络服务平台用例模型中,企业用户角色具有查看平台信息、搜索平台信息、发布平台信息、注册企业信息、修改企业信息和管理企业邮件等权限,体现了社交网络服务平台面向企业用户的应用需求。

其中,可查看和搜索的平台信息包含个人发布的需求信息、企业发布的供应信息、个人发表的评论、个人发布的日志、个人共享的相册和已成功注册的企业信息等;<<extend>>表示支付费用与发布平台信息两个用例之间存在扩展关系,若企业发布平台信息的次数已用完,平台将提示企业用户应先支付费用,再发布信息;<<include>>表示发布平台信息和申请发布平台信息两个用例之间存在使用关系,即企业发布平台信息时,应提交申请发布平台信息表单,待网站管理员审核通过后,才能发布成功;支付发布平台信息次数和手机绑定费用均为支付费用的两个实现用例。

3.3 网站管理员角色用例的设计

在上述设计的社交网络服务平台的用例模型中,网站管理员角色具有查看平台信息、搜索平台信息、审核平台信息、管理平台信息、采集用户信息、分析用户行为序列和注销用户等权限,体现了社交网络服务平台面向网站管理员的应用需求。

其中,可查看和搜索的平台信息包含个人发布的日志、个人共享的相册等;可审核的平台信息包含提出发布申请的个人需求信息和企业供应信息、提出注册申请的个人和企业信息;可管理的平台信息包含个人发布的需求信息、企业发布的供应信息和个人发表的评论;采集的用户信息包括个人、企业及其好友的信息;<<extend>>分别表示预测主题传播趋势与管理平台信息两个用例之间、构成社交关系网与采集用户信息两个用例之间、分析用户行为序列与产生网站管理决策两个用例之间存在扩展关系,<<include>>分别表示分析用户行为序列与预处理用户行为数据两个用例之间、预处理用户行为数据与采集用户行为数据两个用例之间存在使用关系。

4 社交网络服务平台的架构模式

本文所设计的社交网络服务平台将采用浏览器/服务器(Browser/Server, B/S)的三层架构模式和模型、

视图、控制器(Model、View、Controller, MVC)设计模式^[5],该模式的应用不仅能够对社交网络服务平台进行清晰地逻辑划分^[6],还可增强平台功能的可扩展性,并提高平台管理和维护的效率,体现平台业务逻辑、表示逻辑和控制逻辑的有效分离^[7]。

该架构模式的表示层将以页面的形式展示平台信息和体现平台所具有的应用功能,基于角色权限,普通用户、企业用户和网站管理员可向应用服务层发送服务请求,并等待请求服务的结果;应用服务层将完成平台的业务逻辑功能,依据表示层发送的服务申请,该层将与数据服务层进行业务数据的存取操作,并向表示层返回处理结果;数据服务层将完成业务数据的定义、组织、存储、操作和管理等任务,将完成数据库的运行管理、保护、维护和网络通信等任务,并利用存储过程对应用服务层提交的数据进行业务处理,返回其结果,如图 2 所示。

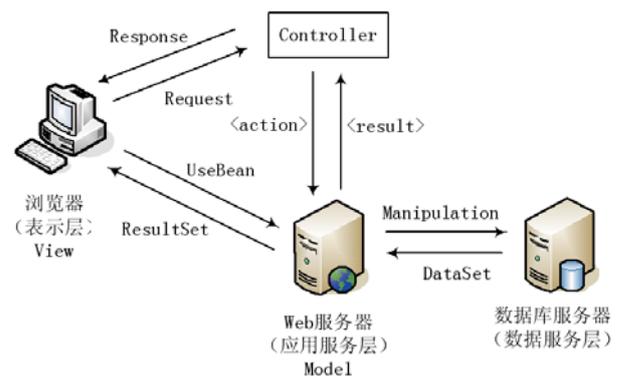


图 2 社交网络服务平台架构模式工作流程

在表示层与应用服务层之间的交互设计中,本文将采用实现了 MVC 设计模式的 Struts2 技术,该技术改进了 Struts 1.X 的标签库^[8],降低了框架组件间的耦合性^[9],并为后续的模式测试工作带来了极大的方便,也为设计与开发社交网络服务平台带来了更好的体验。在社交网络服务平台业务处理过程中,本文利用 Struts2 技术,可将提交的服务请求转交给不同的 Action 控制器进行处理,待进行业务处理的类和控制器的映射关系均可在 struts.xml 配置文件中设置,其应用方法如下所示。

```
<action name="action 名称"
        class="包名称.类名称">
    <result name="业务处理状态 1">
```

```

    转向的 JSP 页面 1</result>
<result name="业务处理状态 2">
    转向的 JSP 页面 2</result>
    .....
<result name="业务处理状态 N-1">
    转向的 JSP 页面 N-1</result>
<result name="业务处理状态 N">
    转向的 JSP 页面 N</result>
</action>

```

5 社交网络服务平台的用例实现

基于 Struts2 技术的社交网络服务平台的物理构件将分为通用组件、前台组件和后台组件。其中，通用组件为登录组件，即普通用户、企业用户和网站管理员均可使用的组件；前台组件面向的是普通用户和企业

用户，用户登录平台后，可访问社交网络服务子系统和 C2B 子系统；后台组件面向的是网站管理员，网站管理员登录平台后，可访问管理后台。

5.1 用户发布平台信息用例的实现

用户所发布的平台信息是社交网络服务中最重要的资源之一，通过该资源，用户可了解到网络应用服务的推荐体验，可发表对该资源的个性评论，可模糊检索发布相似资源的用户，并申请添加其为好友；将该资源作为联系的纽带，可搭建起企业与企业之间双向的供需关系；将该资源作为共同的话题，可搭建用户之间的社交关系。因此，用户发布平台信息是社交网络前台组件中的一个重要用例，本节将主要阐述如何通过基于 Struts2 技术的页面设计、业务逻辑处理和逻辑设计过程，完成该用例的原型实现，如图 3 所示。



图 3 用户发布信息用例的平台实现页面

(1) 页面设计过程

在用户发布平台信息用例的实现过程中，本用例需要设计用户编辑待发布平台信息的页面、允许用户选择支付发布平台信息次数方式的页面和已发布平台信息待审核的页面。通过使用 `<%@taglib prefix="s" uri="/struts-tags"%>` 编译指令，可导入 Struts2 提供的 UI 标签库，在页面设计或源码中加入 `<s:form action="publishInfo">` 标签后，可继续在 Struts2 表单中

加入页面体设计所需的 `<s:textfield>`、`<s:textarea>`、`<s:submit>` 和 `<s:reset>` 等 UI 标签。在前台组件的 Struts2 表单中，当用户编辑待发布的平台信息后，可单击重置按钮重填信息，也可单击提交按钮发布信息。

当用户还未单击提交按钮发布信息时，本用例将使用 Ajax 技术实现浏览器端校验过程，过滤用户编辑的信息。基于 DWR 和 Dojo 框架，将调用 JavaScript 中封装的实例方法，以显示页面信息过滤效果，在配

置文件中, 本用例将添加如下所示的部分校验规则.

```
<validator type="requiredstring">
<!--必须编辑的平台信息-->
<!--必须编辑的主题信息-->
<param name="fieldName">
mytheme</param>
<param name="trim">true</param>
<message>必须输入发布信息的主题
</message>
</validator>
<validator type="regex">
<!--利用正则表达式校验已编辑的平台信息-->
<param name="fieldName">
mytheme</param>
<!--发布信息的主题长度控制在
5 至 30 个字符之间-->
<param name="expression">
<![CDATA[(\w{5,30})]]></param>
<message>主题长度必须为 5 至 30 个
字符之间</message>
</validator>
```

(2) 业务逻辑处理过程

在用户发布平台信息用例的实现过程中, 本用例将使用 JavaBean 技术实现业务逻辑的处理过程. 在设计实例方法的过程中, 首先应对用户拥有的发布平台信息次数进行判断, 若用户帐号信息表中发布信息次数的属性值为 0, 则应将“recharge”状态传给控制逻辑, 并在页面中向用户提供两种选择, 是支付发布平台信息次数费用, 还是支付手机绑定费用; 若用户帐号信息表中发布信息次数的属性值大于 0, 则应将待发布的平台信息提交至后台组件, 并将“toExamine”状态传给控制逻辑, 并在页面中向用户提供说明, 管理员正在审核已提交的平台信息, 在某个时间点即可发布, 业务逻辑处理过程的描述如下所示.

```
public interface Action { //定义 Struts2 行为接口
public static String RECHARGE="reCharge";
//封装支付费用标志
public static String TOEXAMINE="toExamine";
//封装审核信息标志
public String execute()throws Exception;
//封装执行业务逻辑处理方法
```

```
}
//定义 Struts2 行为类
public class PubInfoAction extends ActionSupport
implements Action{
private TreeSet tree;//封装有序集合的对象
public TreeSet getTree() { //获取有序集合的对象
return tree;
}
public void setTree(TreeSet tree) {
//设置有序集合的对象
this.tree = tree;
}
//实现用户发布平台信息的业务逻辑处理方法
public String execute() throws Exception {
//以 tree 封装的用户为构造参数, 创建模型对象.
CountCheck countcheck
= new CountCheck(tree);
//调用模型对象中的方法, 判断用户拥有的
发布平台信息次数.
if(countcheck.judge_count()==0) {
/*若用户发布信息次数的属性值为 0, 则
在 request 范围中添加属性 judge, 并将其属性值设置
为 false. */
ServletActionContext.getRequest().
setAttribute("judge", "false");
return "reCharge";//返回支付费用标志
}else if(countcheck.judge_count()>0){
/*若用户发布信息次数的属性值大于 0,
则在 request 范围中添加属性 judge, 并将其属性值设置
为 true. */
ServletActionContext.getRequest().
setAttribute("judge", "true");
countcheck.submit_info();
//将用户待发布的信息提交至后台
return "toExamine";//返回审核信息标志
}
}
```

当用户填写完待发布的平台信息后, 发起提交请求时, Web 服务器在收到客户端请求之前, 应设计具有链条功用的过滤器. 通过实现 Filter 接口, 本用例设计

了中文编码转换的过滤器,以提高模型的中文识别和代码重用性能,其过程描述如下所示.

```
private String chinese_encoding;
//封装编码类型对象
//实现初始化设置编码类型方法
public void init(FilterConfig fc)
    throws ServletException {
    this.chinese_encoding
        =fc.getInitParameter("encoding");
}
//实现请求与响应中文信息的过滤功能
public void doFilter(ServletRequest request,
    ServletResponse response,FilterChain chain)
    throws IOException,ServletException {
    request.setCharacterEncoding
        (this.chinese_encoding);
    response.setCharacterEncoding
        (this.chinese_encoding);
    chain.doFilter(request,response);
}
}
```

在 MyEclipse 企业版开发平台中, Web Deployment Descriptor 的 Design 可配置中文编码转换的过滤器. 在 web.xml 的 Filters 子节点上添加 Filter, 并设置 Filter-Name 为 ChineseFilter, Filter-Class 为 myfilter.ChineseFilter, 在 Init-Params 中, 设置 Param-Name 为 encoding, Param-Value 为 GBK; 在 web.xml 的 Filters 子节点上添加 Filter Mappings, 设置 Filter-Name 为 ChineseFilter, URL-Pattern 为 /*, 表示需要过滤社交网络服务平台中的所有资源, web.xml 文件中过滤器配置描述如下所示.

```
<filter>
    <filter-name>ChineseFilter</filter-name>
    <filter-class>myfilter.ChineseFilter
</filter-class>
    <init-param>
        <param-name>encoding</param-name>
        <param-value>GBK</param-value>
    </init-param>
</filter>
<filter-mapping>
```

```
<filter-name>ChineseFilter</filter-name>
```

```
<url-pattern>/*</url-pattern>
```

```
</filter-mapping>
```

(3) 控制逻辑设计过程

当控制器 Controller 接收到来自用户发布平台信息的服务请求后, 配置文件 struts.xml 将其分发给 publishInfo 相对应的模型, 接收来自服务请求表单中待发布的平台信息, 具体描述如下所示.

```
<action name="publishInfo" class="bll.PliAction">
    <result name="reCharge">
        /pli_recharge.jsp</result>
    <result name="toExamine">
        /pli_toexamine.jsp</result>
</action>
```

依据业务逻辑处理过程中的实现描述, 通过调用 PliAction 类中的业务逻辑处理方法, 并在方法中调用数据库中已设计的存储过程, 将对服务请求进行处理, 最终依据返回的“reCharge”或“toExamine”业务处理状态, 将 HTTP 请求转交给相应的“pli_recharge”或“pli_toexamine”JSP 页面, 并发送响应到浏览器端.

--判断用户拥有的发布平台信息次数存储过程

```
CREATE PROCEDURE judge_count
```

```
--创建存储过程
```

```
@username varchar(10) as inputparam
```

```
--将发布平台信息的用户帐户作为输入参数
```

```
@flag int as output
```

```
--将用户拥有的发布平台信息次数作为输出参数
```

```
AS
```

```
BEGIN
```

```
--从数据表中选择用户所拥有的
```

发布平台信息次数

```
SELECT count into @flag FROM <数据表名>
```

```
WHERE username=@username
```

```
--将用户所拥有的发布平台信息次数返回到模型
```

实例方法调用存储过程的位置

```
return @flag
```

```
END
```

5.2 网站管理员分析用户行为序列用例的实现

用户行为是社交网络服务中最重要的信息资源之一, 通过对用户行为的采集、预处理和分析等过程, 网站管理员可获得具有一定规律的用户行为序列, 该序列可为

预测社交热点主题的传播趋势、进一步分析网络用户间的社交类型、产生网站商业管理决策等提供一个实际的信息集合. 因此, 网站管理员分析用户行为序列是社交

网络后台组件中的一个重要用例, 本节将主要阐述如何通过基于 Struts2 技术的页面设计、业务逻辑处理和控制逻辑设计过程, 完成该用例的原型实现, 如图 4 所示.



图 4 网站管理员分析用户行为序列用例的平台实现页面

(1) 页面设计过程

在网站管理员分析用户行为序列用例的实现过程中, 本用例需要设计管理员采集用户行为数据、预处理用户行为数据、分析用户行为序列和产生网站管理决策的页面, 通过导入 Struts2 提供的 UI 标签库, 在页面设计或源码中加入 <s:form action="action 名称"> 标签后, 可继续在 Struts2 表单中加入页面体设计所需的其它 UI 标签. 在此用例的实现过程中, 将以模块引导的方式使管理员能够完成社交网络服务中用户行为分析的过程^[10], 并应用分页技术以显示用户行为数据和用户行为序列.

(2) 业务逻辑处理过程

在网站管理员分析用户行为序列用例的实现过程中, 首先要对用户行为的数据进行采集, 并将采集的结果存储在用户行为数据库中, 如表 1 所示; 其次要对已采集的信息进行清洗、用户识别、会话识别和事务识别等预处理操作, 并将结果存储在用户行为数据库中, 如表 2 所示; 再次要对已预处理的信息进行序列模式挖掘, 并将结果存储在用户行为序列数据库中, 如后续算法所示; 最后, 根据已挖掘出的用户行为频繁序列, 结合序列模式支持度参数的调整, 以产生网

站管理决策的描述, 如表 3 所示.

表 1 采集的用户行为数据

User ID	Time	Event ID	Event	User IP
DYJD 204534	2013-07-04 0:27:34	07	发表评论	10.200. 0.24
CNHC 2013341	2013-07-04 5:12:44	03	发布信息	61.105. 43.99
GLNB 652678	2013-07-04 15:33:57	01	搜索好友	211.305. 0.18
CNHC 2013341	2013-07-04 19:07:12	04	在线游戏	61.105. 43.99
GLNB 652678	2013-07-05 14:57:43	05	添加好友	211.305. 0.18
...
FLTW 100272	2013-07-05 15:35:16	03	发布信息	10.266. 0.13

表 2 用户行为序列

User ID	用户行为序列
DYJD204534	S<(07,05),04>
CNHC2013341	S<(03,07),(05,02)>
GLNB652678	S<(09),(05,01)>
AWEH568342	S<(03,07),(04,02)>
...	...
FLTW100272	S<(05,01),(03,02)>

```
public interface Action { //定义 Struts2 行为接口
public static String DATASL="dataSelection";
//封装采集用户行为标志
public static String DATAPP="dataPreprocess";
//封装预处理用户行为标志
public static String DATAM="dataMining";
//封装挖掘用户行为序列标志
public static String DATAS="dataStrategy";
//封装产生决策支持标志
public String execute()throws Exception;
//封装执行业务逻辑处理方法
}
//定义 Struts2 行为类成员
private ResultSet rs;
//封装用户行为序列数据库中的记录集合
public ResultSet getRs() { //获取记录集合的对象
return rs;
}
public void setRs(ResultSet rs) {
//设置记录集合的对象
this.rs = rs;
}
//实现分析用户行为序列的业务逻辑处理方法
public String execute() throws Exception {
double minsupport=0.4;
//设置序列模式中的最小支持度
while(rs.next())
{
String sequence=
rs.getString("Sequence");
```

```
Candidatek=GSPGenerate(sequence);
//生成频繁序列候选集合
For each sequence s in the DataBase Do
if(s contains Candidatek)
Increase the count of Candidatek;
End For
Lk=Candidates in Candidatek
with minsupport;
}
StoredData(Lk);//存储频繁序列集合
return "dataMining";
//返回挖掘用户行为序列标志
}
```

表 3 用户行为决策描述

支持度达到 40%的序列模式
S<(05,07),01>
S<01,05>
S<(04,05)>
...

(3) 控制逻辑设计过程

当控制器 Controller 接收到来自网站管理员分析用户行为序列的服务请求后, 配置文件 struts.xml 将其分发给相对应的模型, 具体描述如下所示.

```
<action name="action 名称"
class="bll.UserBehaviorAction">
<result name="dataSelection">
/userBehavior_dataselection.jsp</result>
<result name="dataPreprocess">
/userBehavior_datapreprocess.jsp</result>
<result name="dataMining">
/userBehavior_datamining.jsp</result>
<result name="dataStrategy">
/userBehavior_datastrategy.jsp</result>
</action>
```

依据业务逻辑处理过程中的实现描述, 通过调用 UserBehaviorAction 类中的业务逻辑处理方法, 可完成对用户行为的数据采集、对已采集的数据进行预处理、对已预处理的数据进行序列模式挖掘和网站管理决策的生成等用户行为分析过程. 依据返回的业务处理状

态,可将 HTTP 请求转交给相应的 JSP 页面,并发送响应到浏览器端,以显示用户行为的采集结果、用户行为数据的预处理结果、用户行为序列模式的挖掘结果和网站管理决策的描述。

6 结语

在本文所提出的一种基于 Struts2 技术的社交网络服务平台设计方案中,笔者已对社交网络服务平台的用例模型、架构、设计模式、应用技术以及用例实现过程进行了阐述和分析。该设计方案的提出进一步体现了面向对象程序设计的灵活性和可重用性,已将具有模型层、视图层和控制层的平台结构进行了清晰的逻辑划分,已将业务逻辑、显示逻辑和控制逻辑进行了有效的分离,在满足企业级应用复杂性和安全性需要的同时,也充分体现了基于 Struts2 技术的网络应用服务平台设计思想已成为一种主流的发展态势。

参考文献

- 1 Rémy MW, Michiko Y, Tomoaki W. Social network productivity in the use of SNS. *Journal of Knowledge Management*, 2010, 14(6): 910-912.
- 2 Babushkina YV. Using web 2.0 for the information support of employees. *Scientific and Technical Information Processing*, 2011, 38(1): 38-40.
- 3 李美子,张波.社交网络中的用户信任链形式化模型. *计算机工程*, 2012, 38(23): 60.
- 4 黄建远. SNS 社交网站电子商务经营研究. *现代传播*, 2012, (3): 161.
- 5 蔡骥然,曹海传. B/S 架构下基于 OPC 与 Comet 技术的实时监控系统的实现. *计算机应用*, 2012, 32(S2): 215.
- 6 Hao LC. Application of MVC platform in bank E-CRM. *International Journal of Service, Science and Technology*, 2013, 6(2): 34-36.
- 7 Li YF, Chen ZG. Design and implement of news publishing system based on MVC design pattern. *Advances in Intelligent Systems and Computing*, 2013, 181: 756-757.
- 8 Chen LY, Chen J, Xu B. The implement of AJAX in Struts2 framework. *International Review on Computers and Software*, 2012, 7(6): 3206-3207.
- 9 Tian J, Xu B, Chen L. Design and implementation of BLOB data processing mechanism based on struts2 and hibernate. *International Journal of Advancements in Computing Technology*, 2012, 4(14): 77-78.
- 10 李红,吕本富,申爱华. SNS 网站竞争生存及商业模式创新的关键因素实证研究. *管理评论*, 2012, 24(8): 80.
- 程, 2004, 32(5): 1-4.
- 2 王海欣,邓中亮.位图图象的读取及基本操作. *计算机应用研究*, 2001, 18(2): 17-18.
- 3 耿卫东,陈为. *计算机游戏程序设计*. 第 2 版. 北京:电子工业出版社, 2009.
- 4 孙家广,许隆文. *计算机图形学*. 北京:清华大学出版社, 1998.
- 5 杜晓增,丁宇辰. *计算机图形学基础*. 北京:机械工业出版社, 2004.
- 6 沈荣,廖婷. 图形对象拾取技术在开发 CAD 系统中的应用. *四川文理学院学报*, 2012, 22(5): 72-76.
- 7 郑莉,董渊,何江舟. *C++ 语言程序设计*. 第 4 版. 北京:清华大学出版社, 2010.
- 8 丁宇辰. 圆弧的生成算法研究. *南京工程学院学报(自然科学版)*, 2010, 8(2): 59-62.
- 9 孙崇璇. 圆弧生成方法探究与实现. *云南师范大学学报(自然科学版)*, 2011, 31(4): 64-68.
- 10 生鸿飞,庞爱民. 圆弧生成的 C 程序算法研究. *武汉科技学院学报*, 2007, 20(3): 1-3.
- 11 王志喜,王润云. Bresenham 画圆算法的改进. *计算机工程*, 2004, 30(12): 178-180.
- 12 胡明星. 在 VC 下进行图象处理的方法. *计算机工程*, 2000, 16(10): 91-95.

(上接第 131 页)