

## 1) 项目介绍

本文所涉及的代码都在 FireflowExample 项目中，该项目模拟了某商场送货流程。您可以用 Eclipse 打开该 Project。

该 Example 所使用的程序框架是 JSF(MyFaces1.5+facelet1.1)+Spring+hibernate。

## 2) 建立 Example 的数据库表 (deprecated)

注：自 20090217 日之后，Example 采用了 hsqldb 内存数据库，不用执行这一创建数据库表的操作。

Example 提供的缺省的数据库脚本是 Oracle 系统的，在目录 “FireflowExample \dbscript \oracle” 中。

### ➤ MySQL

如果你使用 MySQL 数据库，则可以用 Fire-Workflow-Engine-All-In-One\dist\dbscript 下面的 schema-export-4-mysql.sql 创建工作流引擎表结构，

示例程序业务表结构你可以根据 hibernate 的映射文件 “FireflowExample \src \org \fireflow \example\data\TradeInfo.hbm.xml” 和 “FireflowExample \src \org \fireflow \example \workflow\_ext \MyTaskInstance.hbm.xml” 自行创建。

### ➤ SQLSERVER 或者其他数据库

如果你使用 SQL Server 或者其他数据库，则所有的表结构需要你自己根据 hibernate 映射文件创建。(以后会完善)

工作流引擎的映射文件在引擎源代码的如下目录中 “org\fireflow\engine\persistence\hibernate”。

示例程序业务表结构创建方法和 MySQL 一样。

特别注意：在 src\applicationContext.xml 文件中配置了大字段(Blob)的抽取器，因为大字段的存取是与数据库相关的，所以如果你不用 Oracle 做为数据库，该文件中相关的配置也要修改一下。

## 3) 发布运行

本示例程序在 Tomcat 5.5.x 版本下测试通过。

只要将程序包扔到 webapps 目录即可，无需做任何配置。

## 4) 操作手册

### a. 登录系统进行操作

通过如下地址登录系统 <http://IP:Port/FireflowExample>。登录界面如下，界面上列出了系统的测试用户，你可以通过这些测试用户登录系统进行操作。

**Fireflow**  
企业级开源工作流

Example

用户名：

密码：

保存

测试用户列表			
角色名称	用户名	密码	中文名
收银岗	Cashier1	123456	收银员1
	Cashier2	123456	收银员2
仓管岗	WarehouseKeeper1	123456	仓管员1
	WarehouseKeeper2	123456	仓管员2
送货岗	Deliveryman1	123456	送货员1
	Deliveryman2	123456	送货员2
	Deliveryman3	123456	送货员3
管理岗	manager	123456	管理员

### b. 收银岗操作

#### ➤ 表单录入

收银岗是整个商场业务流程的首个环节，收银员完成收银工作后，流程随即流转到备货环节。在左边菜单树点击“收银”，打开录入界面如下图：

Fireflow

企业级开源工作流

Example

文档

业务菜单

- |-- 收银
- |-- 我的待办工单
- |-- 我的已办工单
- |-- 流程进度查询

流程管理菜单

- |-- 查看数据库中部署的流程
- |-- 发布流程到数据库中

收银

流水号: 20090217-202340

货品名称: TCL 电视机

单价: 1399

数量: \*

总价: 0

客户姓名: \*

手机:

电话/传真:

Save

录入“收银”表单，点击“Save”；系统保存业务数据，同时，创建流程实例并将其流转到“仓库备货”环节。

### ➤ 已办工单查询

我们通过菜单栏中的“我的已办工单”可以看到当前收银员已经办理的业务，如下图：

Fireflow

企业级开源工作流

Example

文档

业务菜单

- |-- 收银
- |-- 我的待办工单
- |-- 我的已办工单
- |-- 流程进度查询

流程管理菜单

- |-- 查看数据库中部署的流程
- |-- 发布流程到数据库中

查询

流水号	货物名称	数量	业务名称	环节状态	处理时间	处理人
20090217-201638	TCL 电视机	2	付款	已处理	2009-2-17	Cashier2
20090217-201701	万和 热水器	3	付款	已处理	2009-2-17	Cashier2
20090217-201727	海尔 洗衣机	1	付款	已处理	2009-2-17	Cashier2

在本案例中，收银岗的操作员显然没有待办工单，所以通过菜单“我的待办工单”查询不到数据。

## c. 备货、送货

### ➤ 查询待办工单

通过备货或者送货岗位的用户登录系统，这两个岗位上的操作员理论上可以通过“我的待办工单”作为操作所有业务的入口。点击菜单栏上的“我的待办工单”，然后点击页面上的“查询”，即可查询出当前用户的待办工单信息。如下图：

Fireflow

企业级开源工作流

当前屏

Example

文档

业务菜单

我的待办工单

我的已办工单

流程进度查询

流程管理菜单

查看数据库中部署的流程

发布流程到数据库中

查询

签收选定的记录

处理选定的记录

流水号	货物名称	数量	当前环节	当前状态	选择
20090217-201638	TCL 电视机	2	备货	待签收	<input type="radio"/>
20090217-201701	万和 热水器	3	备货	待签收	<input type="radio"/>
20090217-201727	海尔 洗衣机	1	备货	待签收	<input type="radio"/>

#### ➤ 签收待办工单

选择其中的一条记录，然后点击“签收选定的记录”，对应的工单的状态变成“-待-处-理-”，如下图。

Fireflow

企业级开源工作流

当前屏

Example

文档

业务菜单

我的待办工单

我的已办工单

流程进度查询

流程管理菜单

查看数据库中部署的流程

发布流程到数据库中

查询

签收选定的记录

处理选定的记录

流水号	货物名称	数量	当前环节	当前状态	选择
20090217-201638	TCL 电视机	2	备货	-待-处-理-	<input type="radio"/>
20090217-201701	万和 热水器	3	备货	待签收	<input type="radio"/>
20090217-201727	海尔 洗衣机	1	备货	待签收	<input type="radio"/>

#### ➤ 处理并结束待办工单

选择状态为“-待-处-理-”的工单，点击“处理选定的记录”按钮，进入业务处理界面，如下图。在业务处理界面录入相关的数据保存后（本 Example 没有数据需要录入），点击“结束工单”按钮；系统会将当前工单结束，并流转 to 下一个环节或者结束整个流程实例。

Fireflow

企业级开源工作流

Example

文档

业务菜单

我的待办工单

我的已办工单

流程进度查询

流程管理菜单

查看数据库中部署的流程

发布流程到数据库中

备货操作

流水号：20090217-201638

货品名称：TCL 电视机

数量：2

保存业务操作数据

结束工单

可以通过“我的已办工单”查询到刚才结束的工单的信息。

## d. 流程进度查询

对于任何一单业务都可以通过查询出他当前的办理情况。点击菜单上的“流程进度查询”，在新打开的界面中输入流水号，即可查询出流程实例的办理情况，如下图：

Example	文档			
---------	----	--	--	--

业务菜单

- 我的待办工单
- 我的已办工单
- 流程进度查询

流程管理菜单

- 查看数据库中部署的流程
- 发布流程到数据库中

流水号:

流水号	货物名称	数量	环节名称	环节状态	处理人	处理时间
20090217-201638	TCL 电视机	2	付款	已处理	Cashier2	2009-2-17
20090217-201638	TCL 电视机	2	备货	已处理	WarehouseKeeper1	2009-2-17
20090217-201638	TCL 电视机	2	送货	待签收	Deliveryman3	
20090217-201638	TCL 电视机	2	送货	待签收	Deliveryman2	
20090217-201638	TCL 电视机	2	送货	待签收	Deliveryman1	

## 5) 各角色的流程操作特点

收银员作为流程的第一个岗位，收银完成后，流程即流转 to 备货环节，因此收银员的“我的待办工单”是空的。

除了第一个环节的岗位有“收银”菜单作为业务入口外，其他后续岗位理论上都以“我的待办工单”作为业务入口。因此，要求“待办箱”能够“智能地”打开工单对应业务操作界面，代码“org.fireflow.example.mbeans.MyWorkItemBean.openForm()”向你展示如何办到这一点，如下图

```

1.  /**
2.   * 打开工单对应的业务操作界面
3.   *
4.   * @return
5.   */
6.  public String openForm() {
7.      FacesContext facesContext = FacesContext.getCurrentInstance();
8.      try {
9.          final String workItemId = this.selectedWorkItemId;
10.         IWorkflowSession wflsession = workflowRuntimeContext
11.             .getWorkflowSession();
12.         IWorkItem wi = wflsession.findWorkItemById(workItemId);
13.
14.         if (wi != null && wi.getState() == IWorkItem.STARTED) {
15.             facesContext.getExternalContext().getRequestMap().put(
16.                 "CURRENT_WORKITEM", wi);
17.             String formUri = wi.getTaskInstance().getTask().getEditForm()
18.                 .getUri();
19.             return formUri;
20.         } else {
21.             this.doQueryMyToDoWorkItems();
22.             return "SELF";
23.         }
24.     } catch (EngineException e) {
25.         // TODO Auto-generated catch block
26.         e.printStackTrace();
27.         this.doQueryMyToDoWorkItems();
28.         return "SELF";
29.     }
30. }

```

另外，你也可以关注一下 org.fireflow.example.mbeans.MyWorkItemBean.doQueryMyHaveDoneWorkItems()，该方法想展示如何通过 IWorkflowSession 实现系统没有预先定义的一些功能。

