

# Apache Roller 部署到 MyEclipse

Author: 寒武纪

Publish Date: 2009-11-20

Blogger Home: <http://www.blogjava.net/nighty>

都说 roller 在国外是二次开发博客系统的首选，但是下载源代码之后发现它是基于 netbeans 项目结构开发的。平时用习惯了 Eclipse，所以想搬到 MyEclipse 上面，但没有想到居然那么不容易，折腾了整整一天，才最终跑起来。网上的参考信息太少，大概都是那二三篇的转贴，全部结合起来就差不多可以解决，借此总结一下，希望用到的朋友有参考作用！

## 1. 准备环境

MyEclipse 6.0  
JDK 1.6.0\_05  
Tomcat 5.5.27 (建议用 5.5 以上版本)  
Roller 4.0.1

## 2. 下载 Roller

从 Roller 的主页 <http://roller.apache.org/> 下载最新的 roller 版本，目前是 4.0.1，同时下载源代码包和官笔发布程序包的 zip，分别为 apache-roller-src-4.0.1.zip 和 apache-roller-4.0.1.zip。

下载发布程序的原因是源代码包里有一些内容和发布包不一致，导致源代码在 Eclipse 中编译后在运行时会报错，因为对源代码不熟悉，所以还无法定位到错误的原因所在。另外一些配置文件只有在它官方的发布包中才有，一会要用到它。

## 3. 搭建项目环境

在 MyEclipse 中新建一个 Web 项目，命名为 roller，J2EE 环境选择 1.4。  
解压缩 roller 的二个包，为了说明的方便，我们把 apache-roller-src-4.0.1.zip 的解压缩后目录叫做 roller-src，把 apache-roller-4.0.1.zip 的解压缩后目录叫做 roller-bin。

把 roller-src/apps/weblogger/src/java 目录下的 org 整个目录的源文件复制到项目中的 src 里面；

向 lib 中添加 jar 文件。二种方法，可以从 roller-src 中的 tools 目录中逐个目录地把所有的 jar 文件拷贝出来，为了简单起见，还是从它的发布程序中直接拿更方便，从 roller-bin 中的 WEB-INF/lib 下面的所有 jar 都复制出来，放到我们的项目中的 lib 中去；

roller 默认是使用 MySQL 数据库，但是 lib 里面缺少 MySQL 的 JDBC 驱动，所以还要自己从网上下载驱动的 jar，添加到 WEB-INF/lib 中，如果你是使用的其它数据库，同样需要手动下载它对应的驱动程序；

从 roller-bin 中复制除了 WEB-INF 目录以外的所有文件，粘贴到我们项目

中的 WebRoot 下面，这些是页面文件和一些资源文件；

复制 roller-bin 中的 WEB-INF 目录下除了 classes 外的所有文件到我们项目中的 WEB-INF 目录下，覆盖掉默认的 web.xml，你还可以发现它的一些 jsp 文件是放在 WEB-INF 下面的，起到很好的保护作用，roller 是基于 struts2 开发的；

从 roller-bin 中的 WEB-INF/classes 目录中，复制除了 org 目录外的所有文件，到我们项目中的 src 里面，这些是系统的配置文件，包括了国际化资源、struts 配置、log 配置、数据库脚本等等。

roller 还用到了邮件功能，需要把 activation.jar，mail.jar 二个 j2ee 的 jar 复制到 tomcat 的 common/lib 目录下，如果没有这一步，可能启动初始化会失败！导致应用无法正常运行。

到这里，所有的文件就已经全部搬到 MyEclipse 上面了，接下来要修改一些内容使用我们的项目能正常部署。

#### 4. 修改部分内容

在我们项目的 src 中新建一个名称为 roller-custom.properties 的文件，roller 采用这个扩展配置现覆盖系统的默认配置。网上有些说要把这个文件放到 tomcat 的 common/classes 中，其实不用的，只要放在 classpath 中能找到就行了，所以我们把它放在我们的 src 目录下面，方便修改，在该文件中添加如下内容：

```
installation.type=auto
database.configurationType=jdbc
database.jdbc.driverClass=com.mysql.jdbc.Driver
database.jdbc.connectionURL=jdbc:mysql://localhost:3306/rollerdb?autoReconnect=true&useUnicode=true&characterEncoding=utf-8&mysqlEncoding=utf8
database.jdbc.username=scott
database.jdbc.password=tiger
mail.configurationType=properties
mail.hostname=smtp-server.example.com
mail.username=scott
mail.password=tiger
```

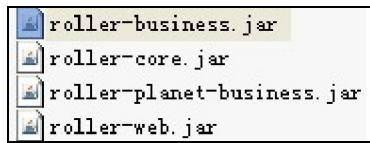
注意 database.jdbc.connectionURL 这个配置后面需要带上上面编码的参数，不然安装的过程可能会出现一些乱码。修改 url 和 username，password 这些，如果你有能正常使用的邮箱服务器，也填上。

创建 MySQL 数据库，名称为 rollerdb，并指定编码为 UTF-8。如果你使用的是 root 用户，则不用授权，如果不是，则需要为用户授权，下面是一个示例：

```
1. mysql> create database rollerdb default charset utf8 collate utf8_general_ci;
2.
3. 文档install-guide 上 create database rollerdb ; 建立的数据库默认是 lat1 字符
4. 编码，中文会有问题所以使用 utf8 编码
5.
6. 给用户授权
7.
8. mysql> grant all on rollerdb.* to scott@'%' identified by 'tiger';
9. mysql> grant all on rollerdb.* to scott@localhost identified by 'tiger';
```

启动 roller 的时候它会查询数据库是否已经初始完成，如果没有，它会自动进入安装页面，引导用户进行安装，做得非常人性化。

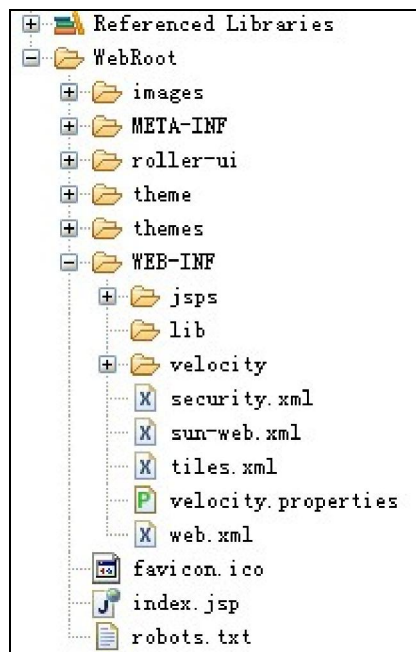
修改部分源代码。org.apache.roller.weblogger 包里面的 pojos 包是有问题的！刚开始的时候老是无法启动，报错提示的是 JPA 的一个异常，根本不知道如何解决，后来在 CSDN 上找到一篇文章，说是 pojos 这个包的源码有问题，可能跟发布包中的版本不一致，所以需要变通一点的处理办法。



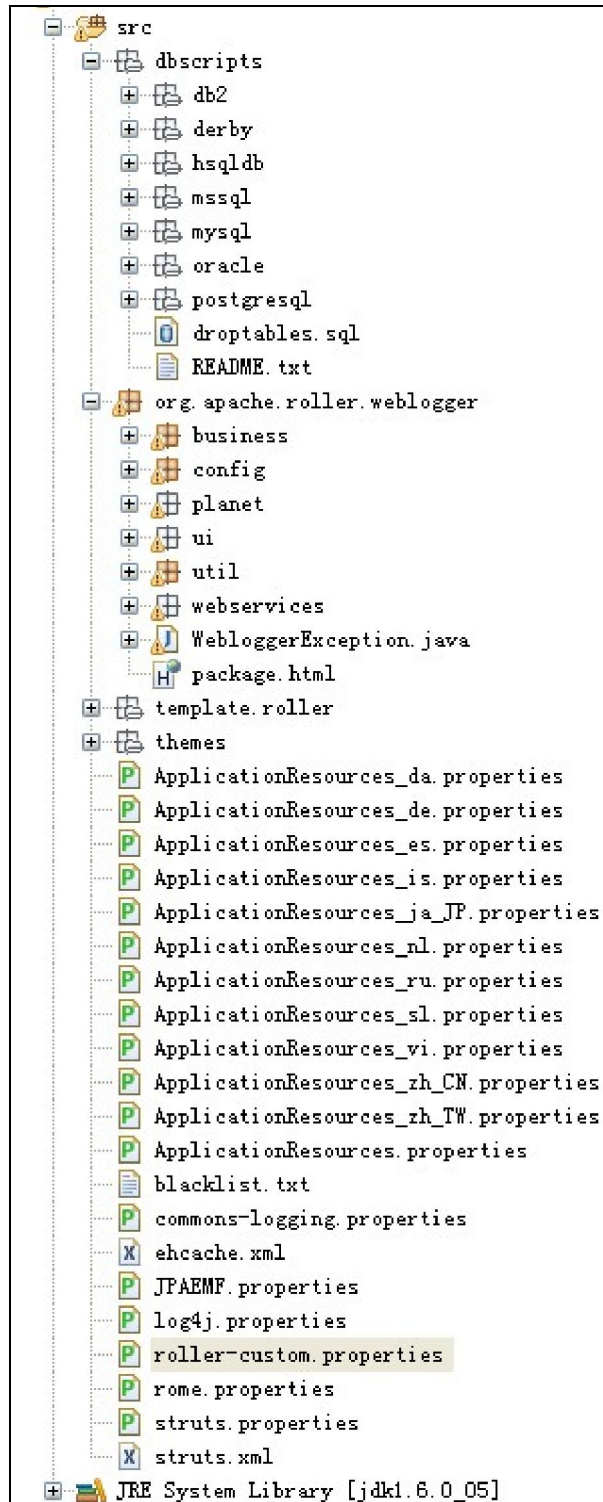
在我们的项目 WEB-INF/lib 中，找到 roller-business.jar 和 roller-web.jar（如上图所示），这二个 jar 其实就是我们 src 中的所有 java 文件的编译打包，为了不重复，先把 roller-web.jar 删除掉，然后用 winrar 打开 roller-business.jar，删除 pojos 以外的其它四个包和那个 WebloggerException.class。这样，其它代码可以正常编译！

下面是最终的基本结构截图(仅参考)：

WeRoot 目录截图



Src 目录截图：



## 5. 其它补充说明

按照上面的步骤，项目结构就基本上正常搭建了，真是费心费力！NetBeans 下面没有试过，兴许会比这个简单许多！反正都习惯 Eclipse 了，就都移过，后面再花些时间研读一下它的源代码，网上说是非常好的学习例子。

Roller 的系统全局配置文件是放在 `org.apache.roller.weblogger.org.apache.roller.weblogger.config` 包中的 `roller.properties` 中的。在这里能找到各项系

统的配置，我们自己手动创建的 `roller-custom.properties` 是用来覆盖一些常用属性的。

`src` 下面的 `dbscripts` 是所有数据库的 SQL 脚本，`apache-roller-src-4.0.1.zip` 中提供的脚本是不完整的！所以需要使用正式发布包的脚本，它带了数据库的初始脚本，由系统自动引导创建，还有许多版本升级的补充脚本。

默认的 `log` 配置是用文件形式，保存在 `tomcat` 的 `logs` 目录中，名称为 `roller.log` 的，如果你启动的过程遇到问题，请到这里去查看启动日志，分析一下异常的原因。

`roller-custom.properties` 中配置的邮箱主机和用户名密码等，如果无法正常连接，它初始化时会报错，但是不会影响正常启动，前提是你必须把 `activation.jar`，`mail.jar` 这二个文件放到 `tomcat` 里面。

在调试的过程中发现 `org.apache.roller.weblogger.business.WebloggerImpl` 类的创建方法中有加载 `roller-version.properties`，但是在我们的 `src` 中没有找到这个文件，原来是在 `roller-business.jar` 中包含了。好像另一个类也包含这一句启动的初始代码，如果找不到这个类，会抛出空指针 `Exception`，同样导致整个应用启动失败。所以刚开始的时候生硬地在 `WebloggerImpl` 类中 115 行加载了一段异常处理代码如下：

```
Properties props = new Properties();
try {
    props.load(getClass().getResourceAsStream("/roller-version.properties"));
} catch (IOException e) {
    log.error("roller-version.properties not found", e);
}
catch (NullPointerException npe) { // 这里是我加入的代码
    log.error("roller-version.properties not found", npe);
}
```

当然，如果你已经包含了 `roller-version.properties` 文件，这个异常是不会存在的！

Roller 的文档非常齐全，图文并茂，多多阅读，就可以了解它的使用方法。IBM 开发者论坛就是基于它进行开发的，的确非常强大，想了解 `struts2`、`JPA`、`Spring`、`Acegi`、`Velocity` 等主流技术的都可以从这里找到示例，不过要多花些时间，从其它方面，比如结构划分、代码规范、`theme` 切换等，都可以学习到很多东西。

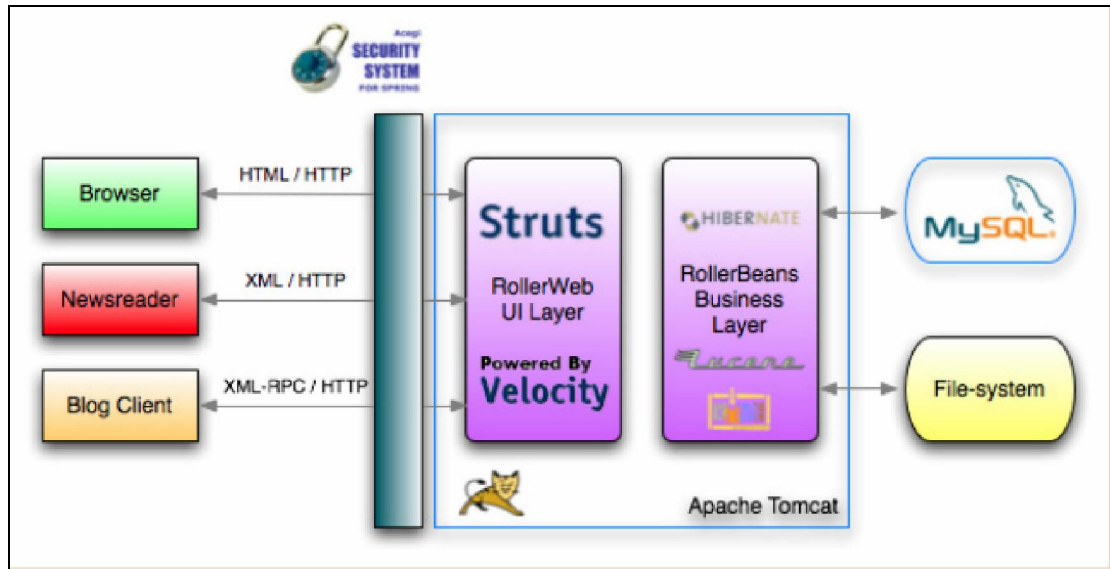
## Roller Architecture (摘自网络)

### Roller Web : Web and UI Layer

- \* Editor UI via Struts and JSP , blog and feed rendering via Velocity
- \* Feed parsing via ROME, Blogger API via Apache XML-RPC

### Roller Beans : Business and Persistence Layer

- \* Hibernate/JPA for DBMS , Lucene for search



多多学习，愿和大家共同分享！^\_^