



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

软件工程

Lab 2 : MVC Web编程与云平台部署



实验目标

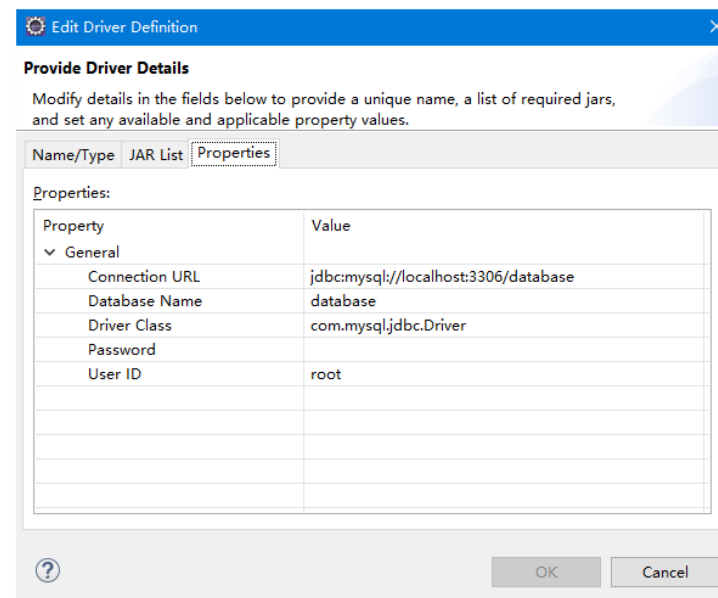
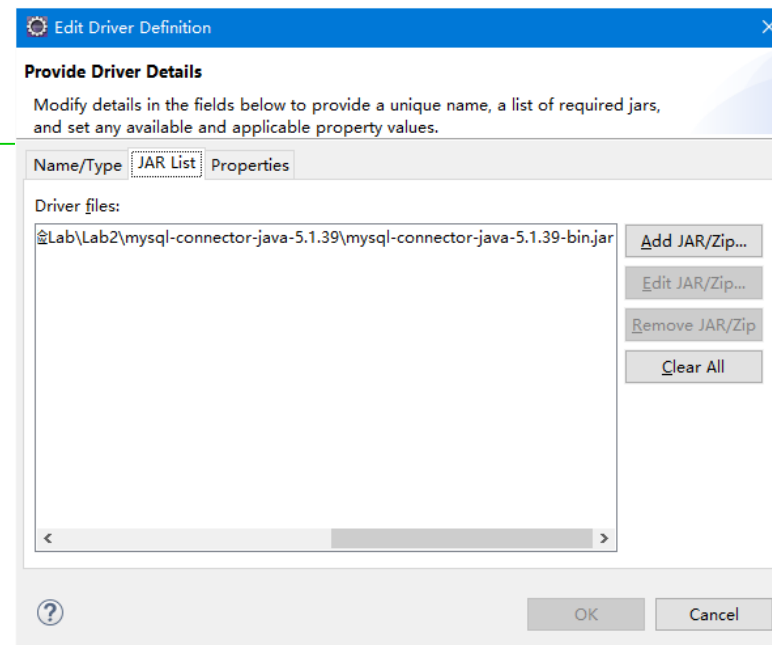
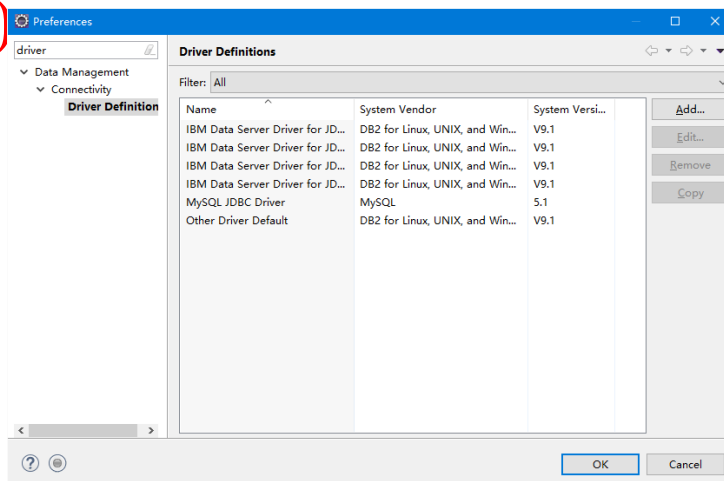
- 掌握MVC架构下SaaS开发的基本流程；
- 开发环境：JAVA+Struts 2+Eclipse+Tomcat+MySQL+SAE(或其他PaaS云平台)
- 在Eclipse中配置Struts开发环境，开发一个小型SaaS，在Web页面中实现对数据库中的数据进行CRUD操作；
 - 在Struts基础上，也可以补充使用某些前端框架(如Bootstrap, jQuery, node.js等)或数据库访问框架(如Hibernate等)，但这并非本实验的目标。
- 部署在自选的公共云计算平台上并对外发布，模拟用户访问。
 - 可以选择SAE这样的PaaS云平台，也可以使用基本的IaaS云平台。
- 本实验个人独自完成

开发要求

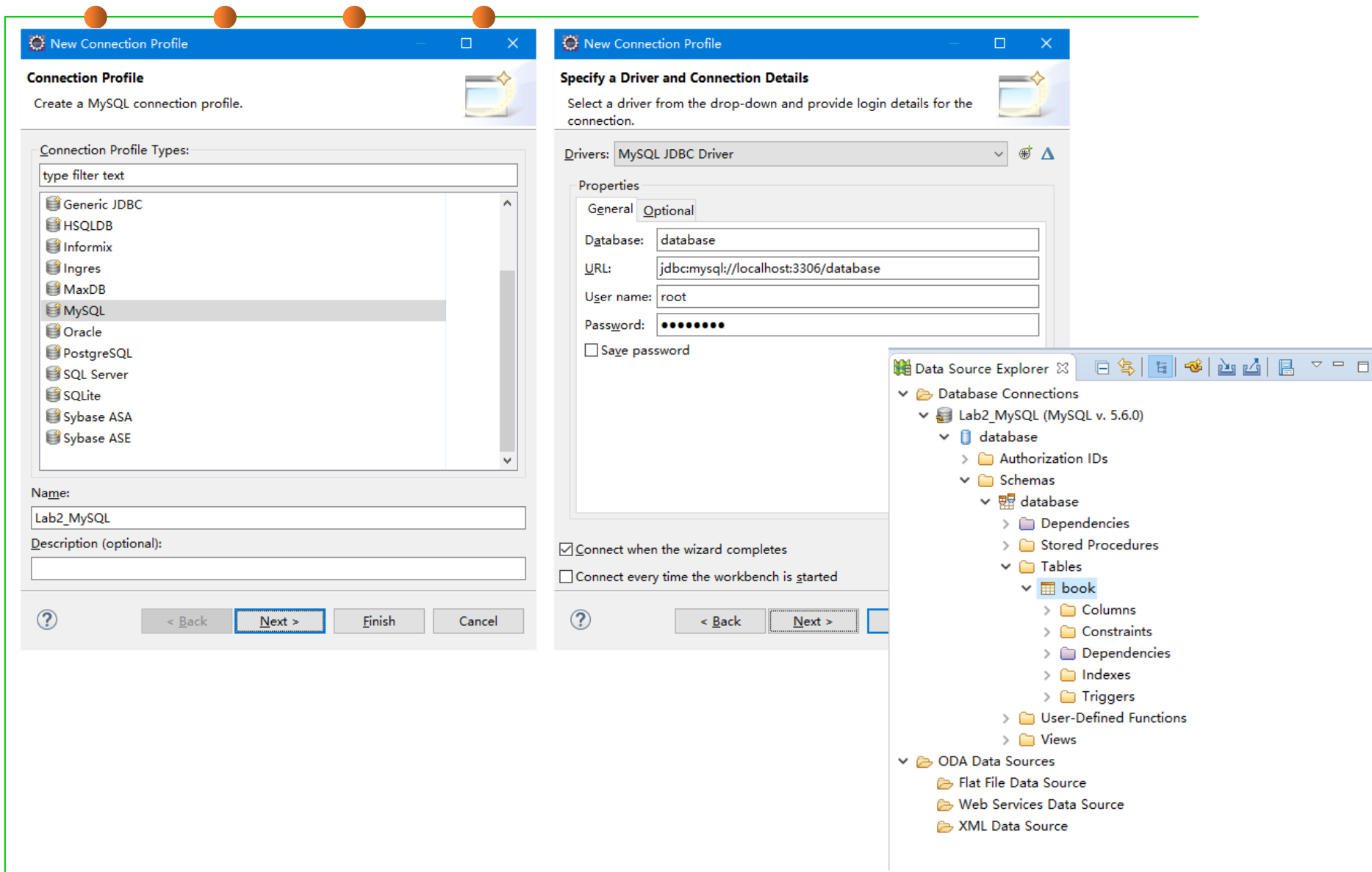
- 使用MySQL建立一个“图书数据库BookDB”，包含两张表：
 - Book {ISBN (PK), Title, AuthorID (FK), Publisher, PublishDate, Price}
 - Author {AuthorID (PK), Name, Age, Country}
- 手工输入足够多的若干测试数据；
- 功能需求：
 - 输入作者名字，查询该作者所著的全部图书的题目；
 - 当用户点击某本图书的题目时，展示图书详细信息和作者详细信息；
 - 当用户点击“删除”按钮时，将对应行的图书从数据表中删除；
 - (选做) 用户可新增一本图书，若该书作者不在库中，还需增加新作者；
 - (选做) 用户可更新一本图书的作者、出版社、出版日期、价格。
- 性能需求：
 - 页面的美观性
 - 功能执行的效率

在Eclipse中配置MySQL

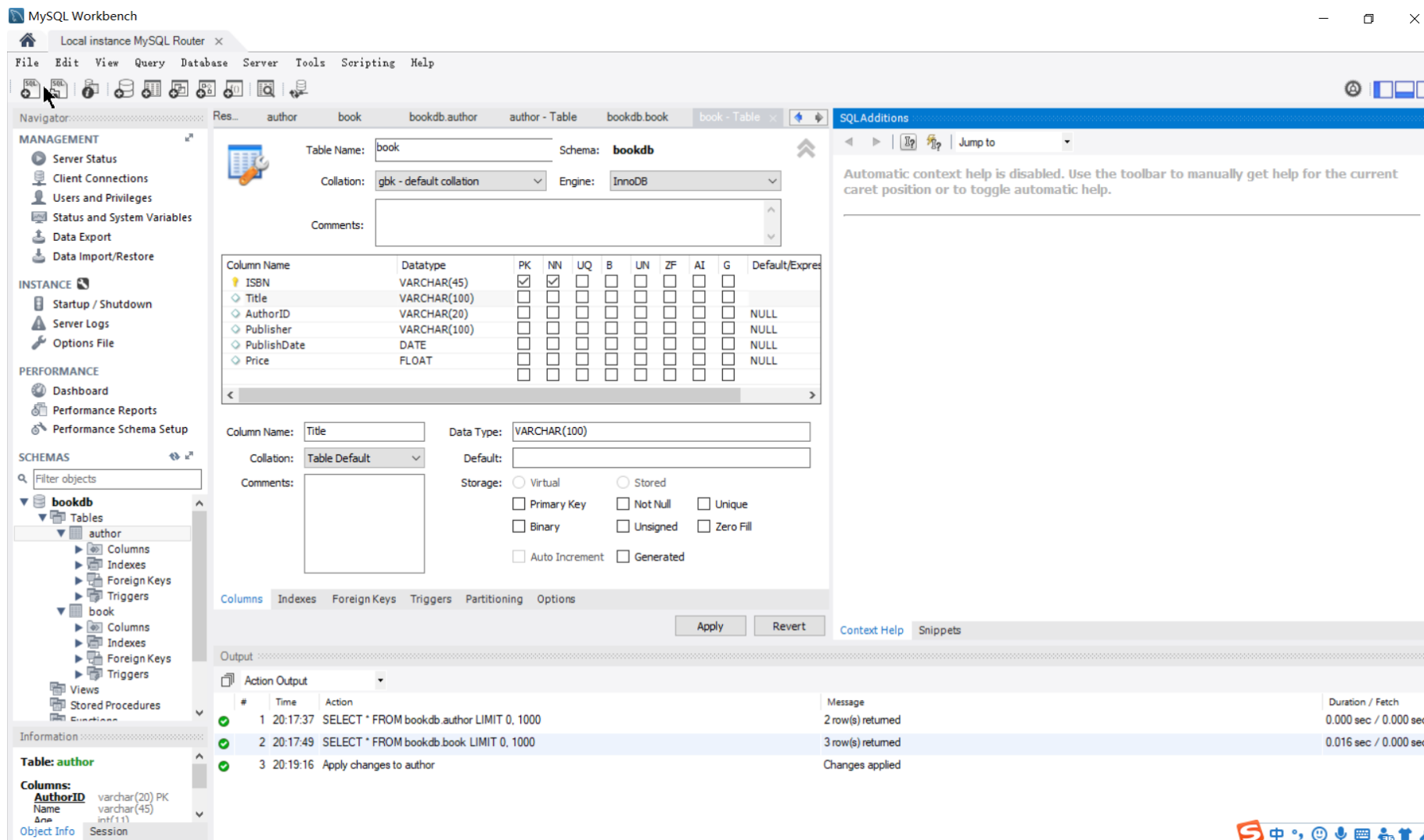
- 自行安装MySQL数据库
<https://dev.mysql.com/downloads/windows/installer/5.7.html> 建议下载MSI Installer版本，如果熟悉命令行方式操作数据库，可下载ZIP版本。
- 下载相应的DB Driver
<http://dev.mysql.com/downloads/connector>
- 在Eclipse中配置MySQL数据连接/或通过MySQL自带Workbench工具管理数据库(推荐此方法)



在Eclipse中添加MySQL数据连接

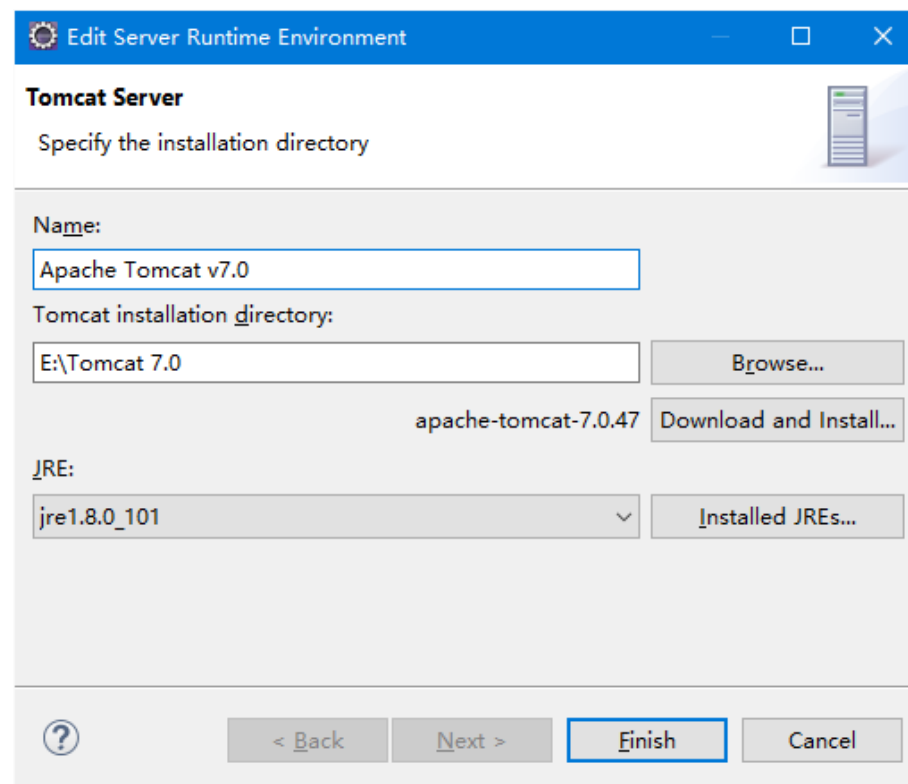
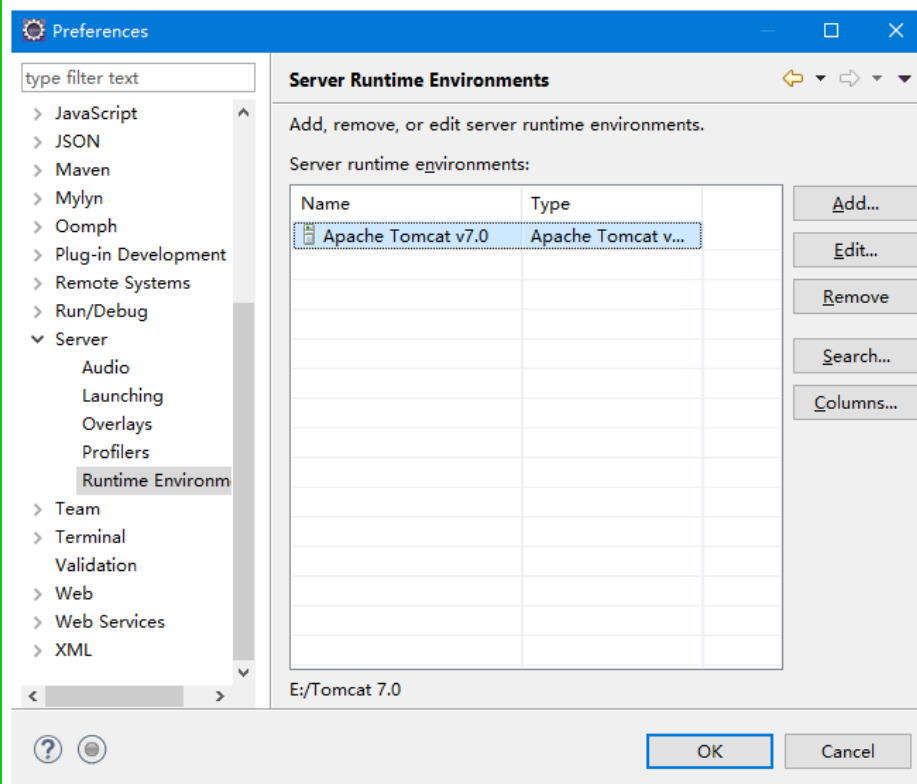


使用MySQL Workbench 管理数据库

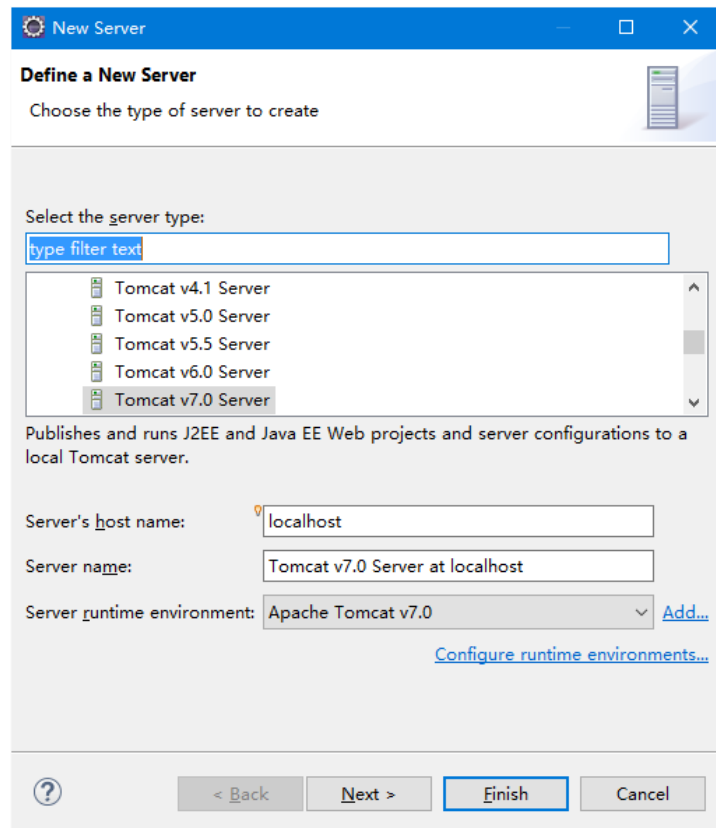


在Eclipse中配置应用服务器(Tomcat)

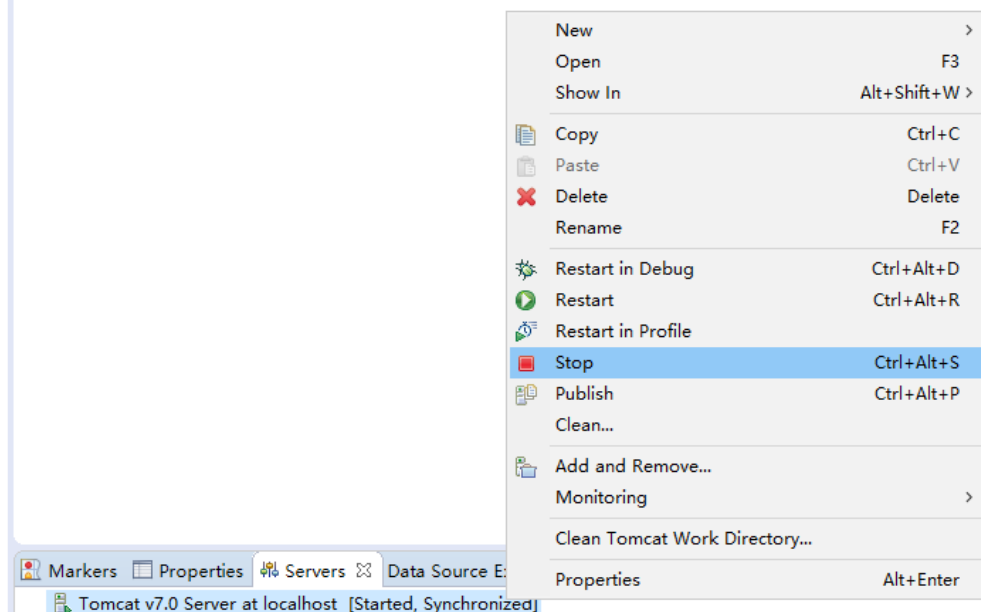
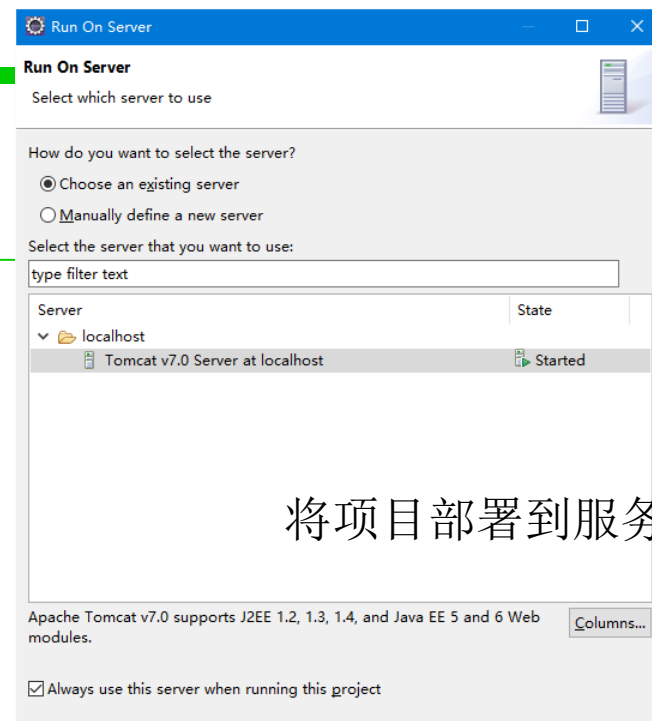
- 从<http://tomcat.apache.org>下载tomcat并在本地安装;
- 在Eclipse中配置Tomcat;



在Project中管理Tomcat

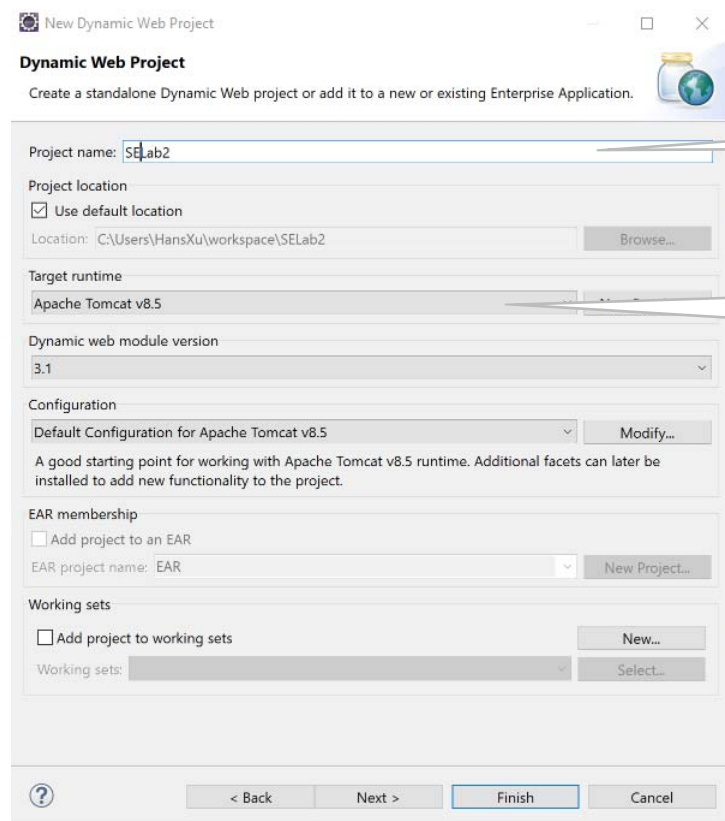


在Servers视图中管理Server



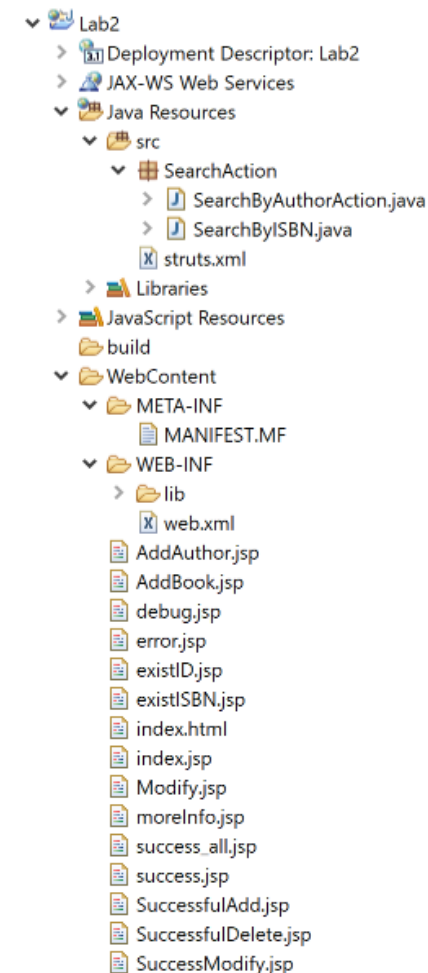
在Eclipse中建立Web项目

- Eclipse→File→New→Project→web→Dynamic Web Project



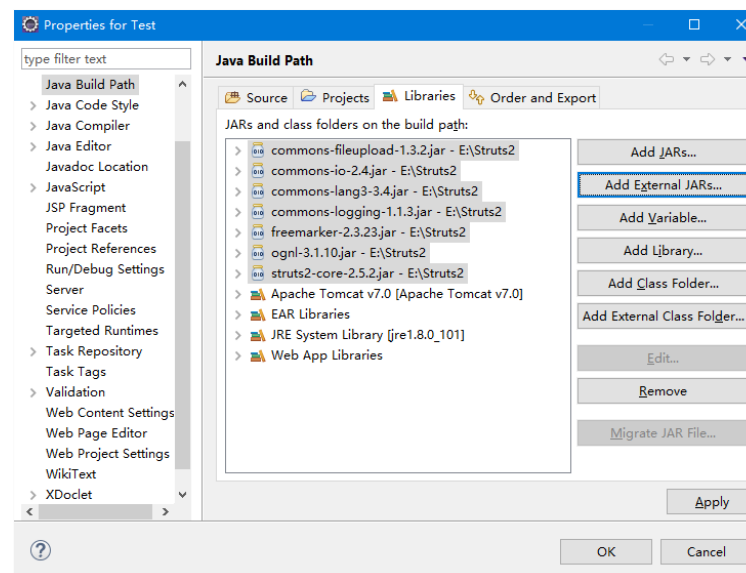
项目名

选择配置好的
Tomcat服务器



在Eclipse中配置Struts2

- 手工从<http://struts.apache.org/download.cgi>下载Struts2的核心支持库 (jar文件)
 - 建议下载2.3.34版本(Full Distribution), 里面包含了struts2的支持库、源代码、文档和示例
 - 压缩包中的lib目录下为Struts2的jar库全集; 实验过程中,可以使用apps目录中struts2-blank.war文件中web-inf→lib目录下的jar库即可(struts2最小库集合)
- 在Project的Build Path中增加Struts2的jar库



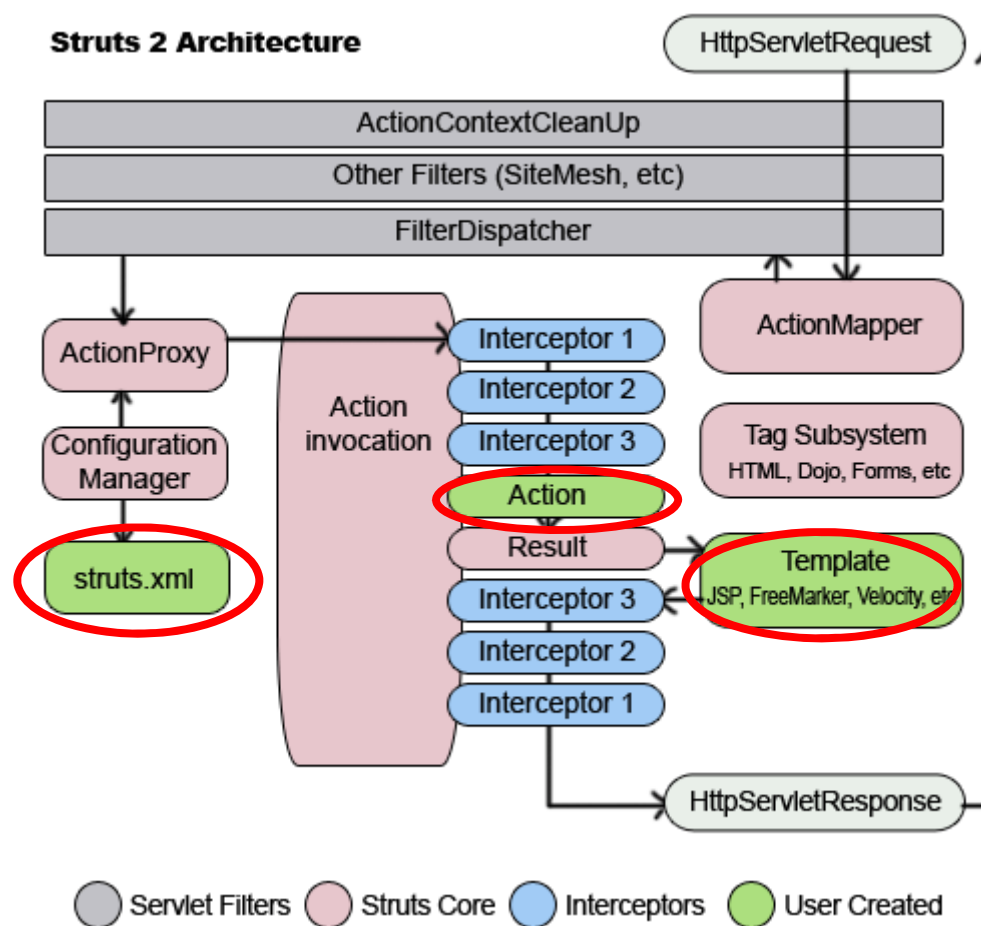
在Eclipse中配置Struts2

- 如果使用Maven管理项目，在pom.xml中增加dependency，无需手工下载和配置：

```
<dependency>
  <groupId>org.apache.struts</groupId>
  <artifactId>struts2-core</artifactId>
  <version>2.5.13</version>
</dependency>
```

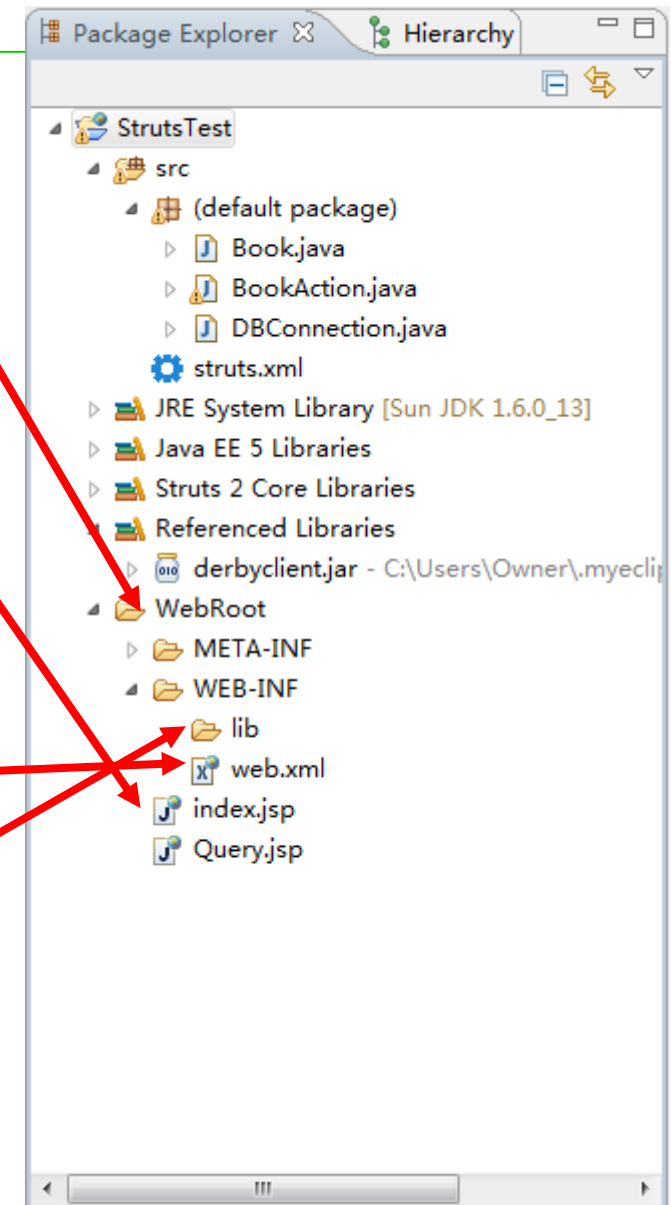
- 某些Eclipse版本中集成了对Struts2的支持，无需配置。
- 如果你的Eclipse的版本与Struts2的版本之间存在兼容性问题，请自行选择Struts2的其他版本。

Struts的构成要素



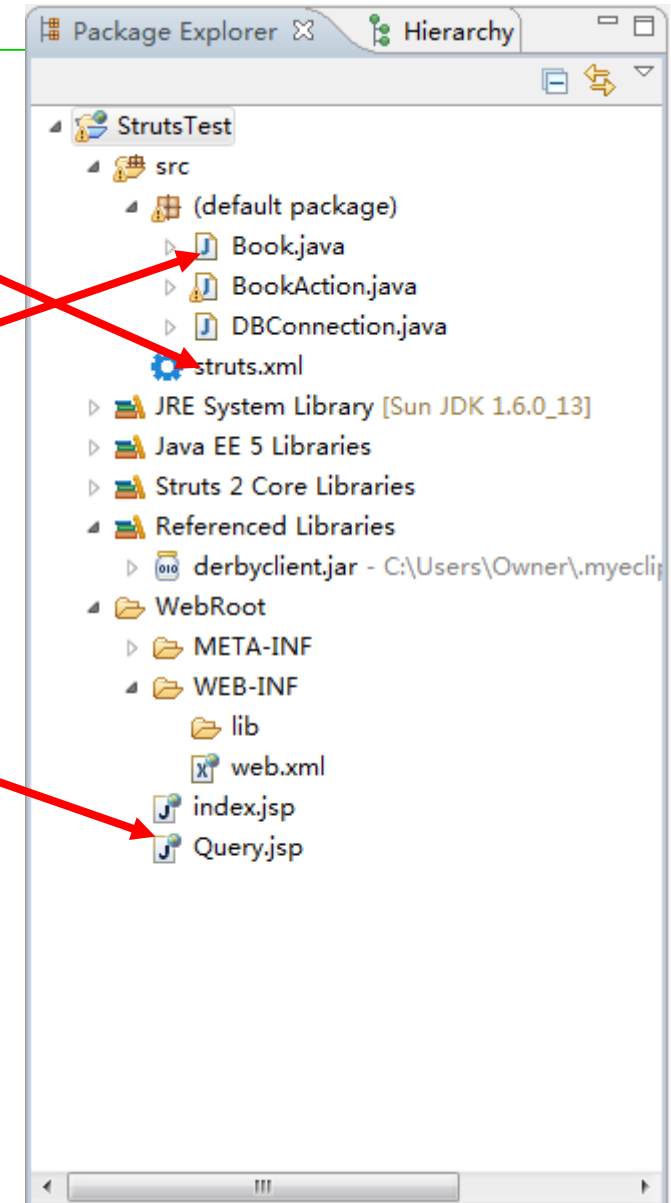
一个Web项目的基本构成要素

- **WebRoot**目录为Web应用的名称，可自行定义，如**BookApp**等；
- **WebRoot**目录下放置各JSP、HTML或其他网页文件，可以按照目录管理各文件。如目录结构为 **WebRoot\A\B\C.JSP**，则访问的URL为：**http://ip地址:端口号\WebRoot\A\B\C.JSP**
- **WEB-INF**根目录下面放置**Web.xml**文件：**SaaS**的入口配置文件，即使不用**Struts**的**SaaS**应用也需配置该文件；
- **WEB-INF**目录下的**lib**目录中放置各种**Jar**文件，如**JDBC**驱动、**struts**所需的**Jar**包等。



一个MVC Struts项目的基本构成要素

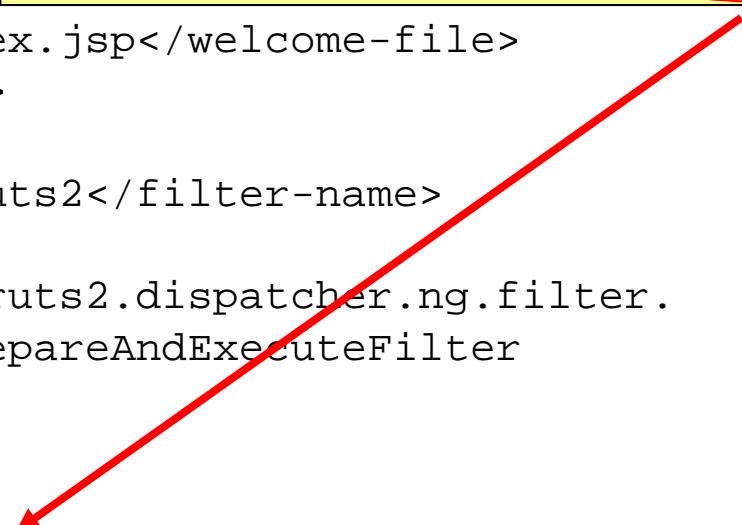
- **Struts.xml**: Struts2的配置文件(编程时放在src根目录下, 发布时放在classes根目录下)
- **Actions**: MVC中的Model和Controller, 是一般的JAVA类;
- 部署到服务器中时, 编译后的class文件, 放置在WEB-INF目录下的classes目录中。注意: 按照package的结构设置目录, 目录层次和名称同package对应。
- **Pages**: MVC中的View, 可使用任何前端页面开发技术(如JSP等);



配置web.xml

- 对一个web project而言，web.xml是其入口配置文件，当用户在浏览器输入某个URL时，满足特定模式的请求会被发送到不同的地方加以处理。

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5"
  <welcome-file-list> http://server:8080/MyApp/HelloWorld
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>
  <filter>
    <filter-name>struts2</filter-name>
    <filter-class>
      org.apache.struts2.dispatcher.ng.filter.
        StrutsPrepareAndExecuteFilter
    </filter-class>
  </filter>
  <filter-mapping>
    <filter-name>struts2</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
</web-app>
```



编写Action (Controller)

- **Action**可以继承自**ActionSupport**类，核心方法是**execute()**，当用户向该**Action**发送请求时，会自动调用。开发者的核心任务之一是重写**execute()**方法。
- 也可以不需要继承，作为一个普通的**JAVA**类。

```
import com.opensymphony.xwork2.ActionSupport;

public class HelloWorld extends ActionSupport {
    private String name;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String execute() {
        name = "Hello, " + name + "!";
        return SUCCESS;
    }
}
```


配置struts.xml

- **struts.xml**是Struts2的配置文件，用于配置Action、Bean、Interceptor等组件
- 作用：映射——将用户通过URL发送来的请求转发给相应的Action类(或者指定该类中相应的方法)，并指定当方法执行之后需要跳转到哪个页面(View)。

```
<struts>
  <include file="struts-default.xml"/>
  <package name="tutorial" extends="struts-default">
    <action name="HelloWorld" class="tutorial.HelloWorld">
      <result>/HelloWorld.jsp</result>
    </action>
  </package>
</struts>
```

配置struts.xml

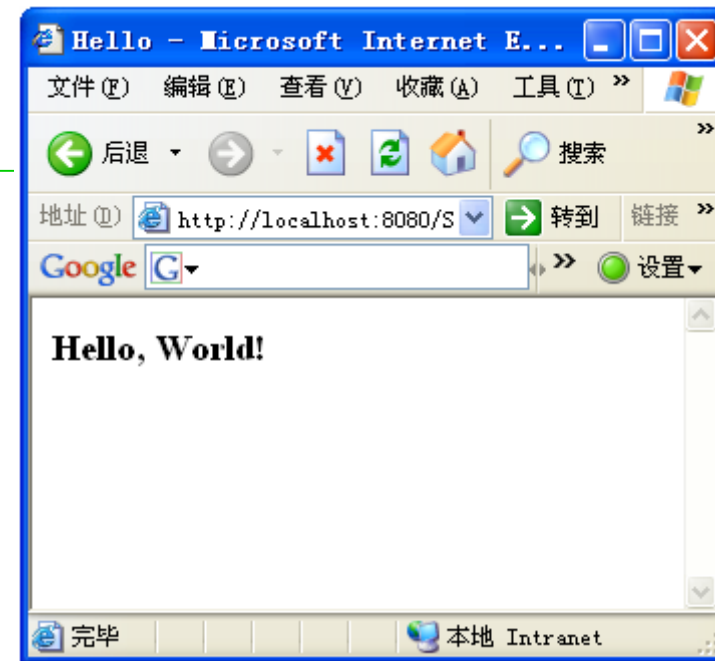
- 也可以在struts.xml中指定method，意即：不再调用缺省的execute()方法，而是调用所指定的方法。

```
<action name="AliasHelloWorld" class="tutorial.HelloWorld"
        method="aliasAction">
    <result>/HelloWorld.jsp</result>
</action>
```

- 这就意味着，你需要在相应的action类中实现所指定的方法。

编写JSP页面(View)

- **JSP页面负责与用户之间的交互：**
 - 根据Action执行的结果，从中获得数据并显示在页面上；
 - 根据用户在页面上所做的操作，将数据收集起来传递给后台的Action；
- **任何前端动态页面的核心部件是form，包含了以下信息：**
 - URL：当form提交时，向后台传递的URL地址，包含了Action名字；
 - 用户数据：用户在页面上填入或修改的数据，通过HTTP session传递到后台；
 - 后台数据：通过HTTP session传递到前端。
- **Struts提供了丰富的标签库(tag)，可以利用这些tag在页面中对数据进行各类动态操作。例如：**
 - iterator：遍历集合或枚举型数据；
 - property：从session中获得某个属性的取值；
 - 各类UI控件：文本框、下拉列表、按钮等。



```
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
  <body>
    <h3>Say "Hello" to: </h3>
    <s: form action="HelloWorld">
      Name: <s:textfield name="name" />
      <s:submit />
    </s:form>
  </body>
</html>
```

```
<%@ taglib prefix="s" uri="/struts-tags" %>
<html>
  <body>
    <h3><s:property value="name" /></h3>
  </body>
</html>
```

访问数据库

- 关系数据库标准操作语句SQL-可参考SQL语句简单教程(CMS)
- Java 访问MySQL示例

```
Connection conn = null;
ResultSet rs = null;
Statement stmt = null;
try {
    Class.forName("com.mysql.jdbc.Driver");
    String url = "jdbc:mysql://127.0.0.1:3306/Bookdb"; //数据库ip地址、端口号、数据库名称
    String user = "user"; //用户名
    String psw = "XXXXXX"; //密码
    conn = DriverManager.getConnection(url, user, psw); //建立连接
    stmt = conn.createStatement();
    rs = stmt.executeQuery("select * from book where ISBN = '" + ISBN + "'");
    if (rs.next()) {
        AuthorID = rs.getString("AuthorID");
        ....
    }
}
```

Struts2开发教程

- **Struts 2 CRUD Tutorial**

- <http://www.dzone.com/tutorials/java/struts-2/struts-2-example/struts-2-crud-example-1.html>

- **Struts2官方教程和示例:**

- <http://struts.apache.org/docs/tutorials.html>

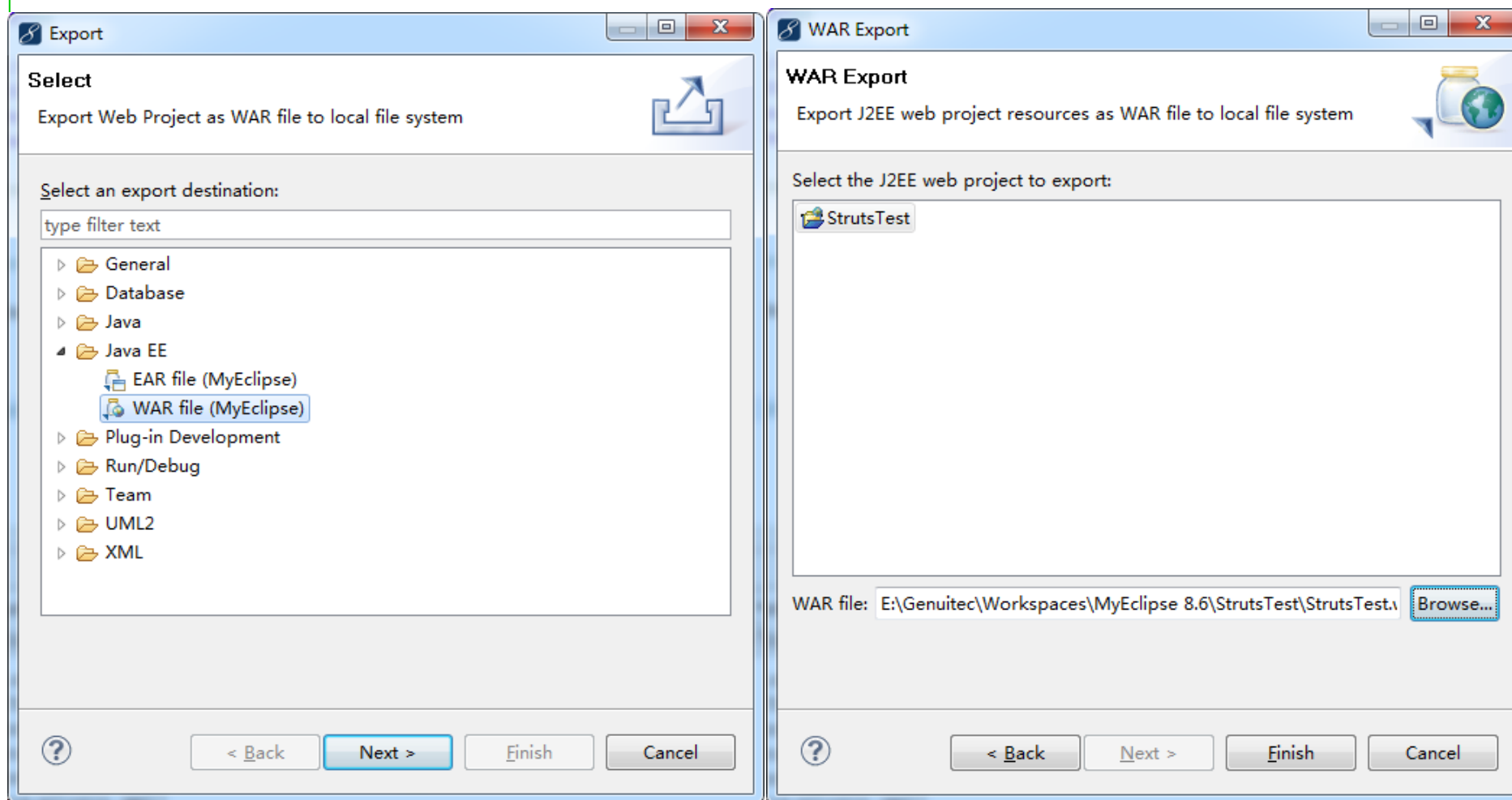
- **非常详细的Step-by-step教程 (中文):**

- <http://www.blogjava.net/max/archive/2006/10/10/74353.html>

- **Struts2 in Action (Struts实战) CMS已经上传**

打包成.war

- 在工程上右键单击，选择“**export**”，选择**WAR file**，保存WAR文件；



将WAR文件部署到Tomcat

- 在Eclipse中配置好Tomcat服务器之后，可选择自动将开发好的SaaS应用部署上去；
- 也可手工将WAR文件复制到tomcat安装目录下的webapps中；
- 开发程序时，应用服务器可以部署在本地，也可以在云端。
- 开发完成后，需将WAR部署到云平台上。

将SaaS部署在云环境中：以SAE为例



- 将本地的SaaS执行环境(Tomcat、MySQL等)迁移到了PaaS云平台上
- 以新浪云为例：<http://www.sinacloud.com>，新浪提供的PaaS云平台；



注意：SAE需要付费！



应用开发、优化及运行的PaaS云计算平台

国内第一家公有云计算平台，支持PHP、Java、Python、NodeJS等语言，提供Web应用开发所需的众多服务，国内最好的PaaS云计算平台

- 弹性扩展，负载均衡智能应对大数据处理请求
- 沙箱技术隔离应用
- 无需运维管理与架构设计
- 防火墙防止攻击
- 99.95%SLA，首批可信云认证
- 免费配额，零成本创业

将SaaS部署在PaaS云环境中

■ 新浪云的收费策略

等级	当前配额					
	普通型	经济型	标准型	高级型	尊享型	VIP定制
年费	免费	10000元	20000元	50000元	150000元	定制
应用数	5	15	30	50	100	定制
Storage 空间限制(单Domain)	10G	20G	100G	1T	100T	定制
Storage Domain 数量	5	10	15	20	25	定制
MySQL存储空间	5G	20G	100G	1T	2T	定制
MySQL 并发连接数	20	50	100	200	500	定制
Memcache 容量	1G	4G	8G	32G	64G	定制
Mail日发送量	100封	2000封	100000封	20000封	100000封	定制
Mail月发送量	1000封	20000封	100000封	300000封	3000000封	定制
HTTP 并发连接数 <small>php</small>	500	800	1000	5000	10000	定制
分钟配额 <small>查看详情</small>	L1	L50	L60	L80	L90	L99
多线路高速接入	✓	✓	✓	✓	✓	✓
域名绑定数量	1个	10个	20个	40个	100个	定制

在SAE上创建应用

- <http://sae.sina.com.cn>

自定义 云市场 ^{NEW}

不懂技术? [一键安装云市场热门应用](#)

*二级域名 (App name)

二级域名 (App name)

applinzi.com

仅允许由数字、字母组成，长度为4到18位。
唯一标识，也是二级域名前缀，创建后不可修改。

建议您创建完应用后在应用设置页面绑定独立域名用于服务。使用独立域名有以下好处:

避免域名的搜索引擎权重受到其他用户的应用影响

避免了DNS劫持与污染问题

可以更灵活的管理应用，增强冗余性

*应用名称

应用名称

应用的中文名称，供显示用。

应用描述

运行环境

PHP 5.3 ^{NEW} PHP 5.6 Python 2.7 Java 1.7 容器 ^{NEW}

配置应用

JVM级别	内存上限	JVM上限	价格/JVM
<input type="radio"/> A型	256M	4	5云豆/小时
<input checked="" type="radio"/> B型	512M	4	10云豆/小时
<input type="radio"/> C型 (限企业用户, 升级可用)	1024M	16	20云豆/小时

JVM设置

JVM数量 2

配置相关的服务(Database等)



hitruTest

web应用

正常

独享型MySQL

服务状态 未开启

独享型MySQL是分布式，高可靠性的资源独享型MySQL数据库。采用了Container技术对用户进行隔离，保证用户之间的业务不互相影响，支持视图、内存表、存储过程等功能，使用方便，为保证最优性能，MySQL专门对InnoDB引擎进行了优化。

[服务首页](#)

请先选择实例规格进行初始化，初始化后将会生成一个主库和一个从库两个实例。

MySQL版本 5.6

实例规格

微型

小型

标准型

大型

巨大型

磁盘: 20GB

内存: 256M

IOPS: 150

连接数: 50

价格 6云豆/小时/实例

[确认初始化](#)

将.WAR部署至SAE

代码管理

上传代码包

该应用尚未启用CDN服务。您可以通过启用CDN，让您的用户获得更快的访问速度。 [\[启动CDN\]](#)

Git仓库信息

仓库地址 <https://git.sinacloud.com/hitstruttest> [如何使用?](#)

用户名 rainy@hit.edu.cn

密码 您的安全密码 [忘记密码?](#)

Git代码部署说明

在你应用的git代码目录里，添加一个新的git远程仓库 sae
\$ git remote add sae https://git.sinacloud.com/hitstruttest

编辑代码并将代码部署到 'sae' 的版本1。

\$ git add .

\$ git commit -am "make it better"

\$ git push sae master:1

- SAE本身提供了git服务，可以在本地Lab2的git仓库中添加SAE远程仓库，代码修改后可直接推送到SAE。

- 也可以手工上传WAR包。

代码上传

目前只支持java应用war包，请选择war包，且文件大小不能超过100MB。

另外，您已经上传的war包将被覆盖。

上传附件

关闭

关于公共云平台的选择

■ 公共云平台的类型：

- IaaS平台：提供虚拟机和操作系统，在操作系统上手工部署运行环境、在运行环境中安装部署你的程序，并对外提供访问。
- PaaS平台：提供分布式Web应用/业务开发的托管和运行平台，无需自行安装虚拟机和OS，在该平台上直接部署应用，并可使用平台提供的一系列基础服务(如数据库、消息、缓存、安全、检索等)。

■ 可用的公共云平台：

- 新浪SAE: <http://sae.sina.com.cn>
- 百度BAE: <https://cloud.baidu.com/product/bae.html>
- 阿里云: <https://cn.aliyun.com>
- 腾讯云: <https://www.qcloud.com>
- 华为云: <http://www.hwclouds.com>
- Google App Engine: <http://appengine.google.com>

评判标准

- 是否完成图书SaaS中必选功能的开发；
- 是否顺利部署到公共云平台并能对外提供访问；
- 所开发的图书SaaS的质量；
- 是否能够在Eclipse中配置Tomcat/MySQL/Struts；
- 对Eclipse中开发Struts程序的流程是否理解；
- 对Struts2的机理是否了解清楚；

提交方式

- 请遵循实验报告模板撰写。
- 不能修改数据库/表的结构和名字，不能变更规定的功能，但可扩展新功能。
- 提交日期：第6周周一晚(10月9日 23:55)
- 提交两个文件到CMS：
 - 实验报告：命名规则“学号-lab2-report.doc”，实验报告中务必标明你的应用的公共云平台访问入口（URL）
 - 部署文件：命名规则“学号-lab2-deploy.war”

检查方式

- TA根据你所提供的公共云平台URL地址，访问你的应用，并测试各项功能，进行打分。如果不能访问应用，则该部分分数为0。
- TA测试时间：10月10日8:00-20:00，在此期间请务必保持你的应用可访问。
- TA阅读实验报告，进行打分。
- 两项分数综合起来，为本次实验最终得分。



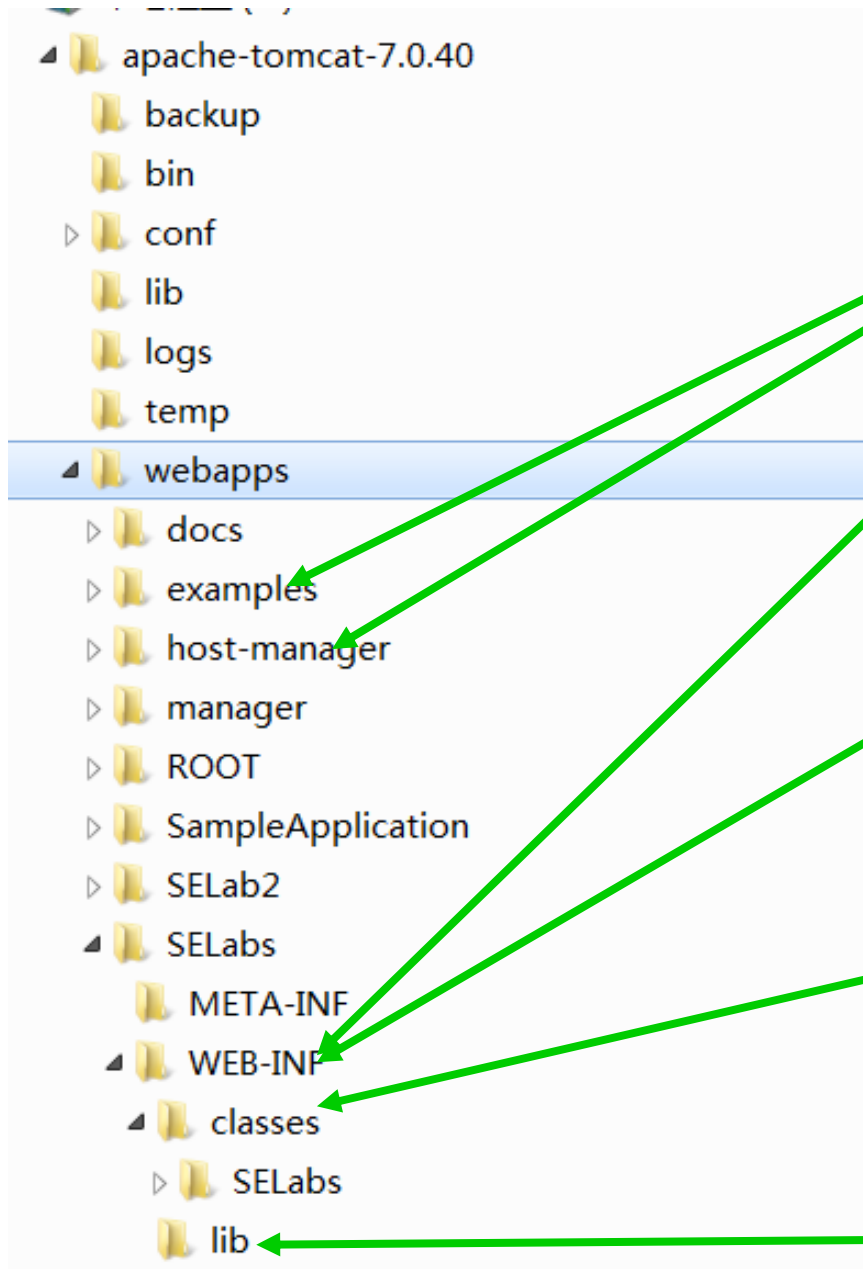
哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

软件工程

结束



Web程序结构



Webapps目录下放置各个网站，每个目录代表一个网站

Web-inf根录下放置web.xml文件，用于配置整个应用程序

Webapps根录下放置各JSP和HTML文件，可以按照目录管理各网页文件。注意地址格式如 SELabs\A\B\C.JSP
URL为: <http://ip:端口\SELabs\A\B\C.JSP>

Web-inf目录下的classes目录下放置编译好的类(class)文件，注意：按照包的结构设置目录(如类a.class属于SE Labs包，则将文件a.class放置在web-inf\classes\SE Labs目录下)

Web-inf目录下的lib目录下放置各种Jar文件，如JDBC驱动、struts所需的Jar包等。