第1章 介绍与安装

作者：ThinkGem

更新日期：2014-4-8

Modify:Jeffen

Date:2015-9-18

# 平台简介

**Pactera-jeesite Kickstart videos share drive url:**

http://pan.baidu.com/s/1o6y2VNk

OpenSourceCommunity,

@thinkgem Jeesite: <https://github.com/thinkgem/jeesit>

@阿里巴巴Druid <https://github.com/alibaba/druid>

JeeSite是基于多个优秀的开源项目，高度整合封装而成的高效，高性能，强安全性的开源Java EE快速开发平台。

JeeSite本身是以Spring Framework为核心容器，Spring MVC为模型视图控制器，MyBatis为数据访问层， Apache Shiro为权限授权层，Ehcahe对常用数据进行缓存，Activit为工作流引擎。

JeeSite主要定位于企业信息化领域，已内置企业信息化系统的基础功能和高效的代码生成工具，包括：系统权限组件、数据权限组件、数据字典组件、核心工具组件、视图操作组件、工作流组件、代码生成等。前端界面风格采用了结构简单、性能优良、页面美观大气的Twitter Bootstrap页面展示框架。采用分层设计、双重验证、提交数据安全编码、密码加密、访问验证、数据权限验证。使用Maven做项目管理，提高项目的易开发性、扩展性。

JeeSite目前包括以下三大模块，系统管理（SYS）模块、内容管理（CMS）模块、在线办公（OA）模块、代码生成（GEN）模块。 系统管理模块 ，包括企业组织架构（用户管理、机构管理、区域管理）、菜单管理、角色权限管理、字典管理等功能； 内容管理模块 ，包括内容管理（文章、链接），栏目管理、站点管理、公共留言、文件管理、前端网站展示等功能； 在线办公模块 ，提供简单的请假流程实例。

JeeSite 提供了常用工具进行封装，包括日志工具、缓存工具、服务器端验证、数据字典、当前组织机构数据（用户、机构、区域）以及其它常用小工具等。另外还提供一个强大的在线 代码生成 工具，此工具提供简单的单表、一对多、树结构功能的生成，如果对外观要求不是很高，生成的功能就可以用了。如果你使用了JeeSite基础框架，就可以很高效的快速开发出，优秀的信息管理系统。

# 内置功能

1. 用户管理：用户是系统操作者，该功能主要完成系统用户配置。
2. 机构管理：配置系统组织机构（公司、部门、小组），树结构展现，可随意调整上下级。
3. 区域管理：系统城市区域模型，如：国家、省市、地市、区县的维护。
4. 菜单管理：配置系统菜单，操作权限，按钮权限标识等。
5. 角色管理：角色菜单权限分配、设置角色按机构进行数据范围权限划分。
6. 字典管理：对系统中经常使用的一些较为固定的数据进行维护，如：是否、男女、类别、级别等。
7. 操作日志：系统正常操作日志记录和查询；系统异常信息日志记录和查询。
8. 连接池监视：监视当期系统数据库连接池状态，可进行分析SQL找出系统性能瓶颈。
9. 工作流引擎：实现业务工单流转、在线流程设计器。
10. 语言管理：引入国际化多国语言，基于语言数据表，支持和扩展：简体中文/英文

# 开发工具

1. Eclipse IDE：采用Maven项目管理，模块化。
2. 在线演示：演示系统常用组件的使用代码，内嵌地图案例等。
3. 代码生成：通过界面方式简单配置，自动生成相应代码，目前包括三种生成方式（增删改查）：单表、一对多、树结构。生成后的代码如果不需要注意美观程度，生成后即可用。

# 设计思想

## 分层设计

（数据访问层，业务逻辑层，展示层）层次清楚，低耦合，各层必须通过接口才能接入并进行参数校验（如：在展示层不可直接操作数据库），保证数据操作的安全。

## 双重验证

用户表单提交双验证：包括服务器端验证及客户端验证，防止用户通过浏览器恶意修改（如不可写文本域、隐藏变量篡改、上传非法文件等）而跳过客户端验证操作数据库。

## 安全编码

用户表单提交所有数据，在服务器端都进行安全编码，防止用户提交非法脚本及SQL注入获取敏感数据等，确保数据安全。

## 密码加密

登录用户密码进行SHA1散列加密，此加密方法是不可逆的。保证密文泄露后的安全问题。

## 访问验证

系统对所有管理端链接都进行用户身份权限验证，防止用户直接通过URL进行未授权页面。

## 数据验证

对指定数据集权限进行过滤，七种数据权限可供选择（所有权限，公司及子公司，本公司，部门及子部门，本部门，本人数据，跨机构数据）

## 快速编码

提供基本功能模块的源代码生成器，提高开发效率及质量。

# 技术选型

## 后端

\* 核心框架：Spring Framework 4.0

\* 安全框架：Apache Shiro 1.2

\* 视图框架：Spring MVC 4.0

\* 服务端验证：Hibernate Validator 5.1

\* 布局框架：SiteMesh 2.4

\* 工作流引擎：Activiti 5.15

\* 任务调度：Spring Task 4.0

\* 持久层框架：MyBatis 3.2

\* 数据库连接池：Alibaba Druid 1.0

\* 缓存框架：Ehcache 2.6、Redis

\* 日志管理：SLF4J 1.7、Log4j

\* 工具类：Apache Commons、Jackson 2.2、Xstream 1.4、Dozer 5.3、POI 3.9

## 前端

\* JS框架：JQuery 1.9。

\* CSS框架：Twitter Bootstrap 2.3.1。

\* 客户端验证：JQuery Validation Plugin 1.11。

\* 富文本：CKEcitor

\* 文件管理：CKFinder

\* 动态页签：Jerichotab

\* 手机端框架：Jingle

\* 数据表格：jqGrid/datatables/bootstrapTheme

\* 对话框：jQuery jBox

\* 下拉选择框：jQuery Select2

\* 树结构控件：jQuery zTree

\* 日期控件： My97DatePicker

## 平台

\* 服务器中间件：在Java EE 5规范（Servlet 2.5、JSP 2.1）下开发，支持应用服务器中间件

有Tomcat 6、Jboss 7、WebLogic 10、WebSphere 8。

\* 数据库支持：目前仅提供Oracle数据库的支持，但不限于数据库，平台留有其它数据库支持接口，

可方便更改为其它数据库，如：SqlServer 2008、MySql 5.5、H2等

\* 开发环境：Java EE、Eclipse、Maven、Git/svn/csv、SVN

\* 操作系统OS：Windows Server 32/64位、Redhat Linux、HP-UX等

# 必会基础知识

1. 软件工程基础、Java语言基础、JSP内置对象、EL表达式
2. Maven：包管理构建工具
3. Spring Framework：类的依赖、自动注入、事务处理
4. Spring MVC：URL映射、参数传递
5. Apache Shiro：安全拦截方式，应用方法，控制按钮
6. SiteMesh：布局框架的用法
7. JSP标准标签库（JSTL）：if、choose、forEach、set
8. Spring MVC表单标签库：form、input、textarea、select、checkbox
9. 文件存放规范，命名规范。

# 学习索引

1. Spring Framework ：

* [跟我学Spring3](http://www.iteye.com/blogs/subjects/spring3)

1. Spring MVC

* [跟开涛学Spring MVC](http://www.iteye.com/blogs/subjects/kaitao-springmvc)
* [JSP语法](http://blog.csdn.net/redarmy_chen/article/details/8197928)、[Spring Form](http://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/html/spring-form.tld.html)、[JSTL](http://blog.csdn.net/redarmy_chen/article/details/8252541)、[EL](http://blog.csdn.net/redarmy_chen/article/details/8231827)
* [Bean Validation](http://www.ibm.com/developerworks/cn/java/j-lo-beanvalid/)

1. Apache Shiro

* [官方文档](http://shiro.apache.org/documentation.html)
* [快速入门](http://wenku.baidu.com/view/138839c28bd63186bcebbc51)

1. SiteMesh 2.4

* [官方文档](http://wiki.sitemesh.org/display/sitemesh/Home)
* [入门实例](http://flyer2010.iteye.com/blog/1296596)、[标签使用](http://flyer2010.iteye.com/blog/1319636)、[装饰文件](http://flyer2010.iteye.com/blog/1319664)

1. Bootstrap 2.3

* [Bootstrap官网](http://twitter.github.io/bootstrap/)
* [Bootstrap中文网](http://v2.bootcss.com/getting-started.html)

1. jQuery

* jQuery API：[官方文档](http://api.jquery.com/)  [中文文档](http://www.css88.com/jqapi-1.9/)
* jQuery Validation：[官方文档](http://jqueryvalidation.org/documentation/)  [使用范例](http://www.cnblogs.com/easyinsc/archive/2009/02/27/1407826.html)
* jQuery zTree: [官方文档](http://www.ztree.me/v3/main.php)
* jQruey jBox：[官方文档](http://www.kudystudio.com/jbox/jbox-demo.html)

1. Activiti

* [官方网站](http://www.activiti.org/)
* [中文文档](http://www.mossle.com/docs/activiti/)
* [咖啡兔Activiti](http://www.kafeitu.me/activiti.html)

1. Ehcache

* [官方文档](http://ehcache.org/documentation)
* [EhCache分布式缓存](http://www.docin.com/p-22141155.html)

1. 其它资料

* [Alibaba Druid](http://code.alibabatech.com/wiki/display/Druid/Home)
* [FreeMarker](http://freemarker.org/docs/ref.html)

# 安装部署(请移步参看新的安装手册：2.1.父子项目继承和聚合)

1. 运行Maven目录下的settings.bat文件，用来设置maven仓库路径，并按提示操作（设置PATH系统变量、配置Eclipse）。
2. 执行jeesite/bin/eclipse.bat 生成工程文件并下载jar依赖包（如果需要修改默认项目名，请打开pom.xml修改第7行artifactId，然后再执行eclipse.bat文件）。
3. 将 jeesite 工程导入到eclipse，选中maven工程，按F5刷新。
4. 设置数据源：src/main/resources/jeesite.properties
5. 导入数据表并初始化数据：运行db/init-db.bat文件。（导入时如果出现“drop”失败提示信息，请忽略。）
6. 新建Server（Tomcat），注意选择以下两个选项：
7. 配置server.xml的Connector项，增加URIEncoding="UTF-8"
8. 部署到Tomcat，设置Auto Reload为Disabled。
9. 访问工程：http://127.0.0.1:8080/ jeesite 用户名：thinkgem 密码：admin

# 常见问题

## 'mvn' 不是内部或外部命令

原因如下

* 1. PATH未配置或配置了多个不一致的Maven地址，如用户/系统变量。
  2. M2\_HOME系统/用户变量地址不正确，可删除M2\_HOME变量。
  3. 检查mvn运行是否正常，cmd执行：mvn -v

## 运行eclipse.bat 找不到文件路径或乱码

一般原因是路径中包含空格或中文。

## 导入到eclipse下找不到jar包

maven未配置，查看m2\_repo仓库路径是否正确。

## 运行init-db.bat提示ORA-xxx

根据错误码排除错误，一般是数据库url不对，用户名或密码错误。

## 部署时出现某某被锁定

一般原因：Tomcat服务中的两个选项未选中。

## 字典列表，添加键值，出现乱码

server.xml未配置url编码为utf-8。

## Ini-db.bat插入数据时，执行错误日志出现

可通过删除pom.xml里L980的建库sql只保留插数据sql，再执行一遍init-db.bat单插xls data即可success；【日志】：

## Maven Tomcat reload/hot deploy热部署

java build path>default output folder: jeesite/target/classes

设定热部署后，修订源码无需重启：

1. Jsp
2. Xml
3. Java方法体内容

例外：

1. Java文件新增或修订方法名
2. Shiro授权annotation，如：

《QarEventController.java》

@RequiresPermissions("test:qarEvent:io")

## Maven 远程仓库下载

mvn help:system

## 无法生成maven project

Maven生成的java项目转化为支持 Eclipse IDE的项目

mvn eclipse:eclipse

Here’s a guide to show you how to convert the Maven generated Java project to Eclipse supported style project.

1. mvn eclipse:clean
2. mvn eclipse:eclipse -Dwtpversion=2.0

It’s really easy to do it. Navigate to your Java project folder, where pom.xml file is placed. And issue this command : mvn eclipse:eclipse

**Note**  
If this is your first time to run this command, it may take some time to download all required dependency to convert your project to Eclipse style project.

1. Verify Java Project

After that, you will notice two new files are created – “.classpath” and “.project“. Both files are created for Eclipse IDE.

File : “.classpath”, by m2eclipse

...

**<classpathentry** kind="con" path="org.maven.ide.eclipse.MAVEN2\_CLASSPATH\_CONTAINER"**>**

**<attributes>**

**<attribute** name="org.eclipse.jst.component.dependency" value="/WEB-INF/lib"**/>**

**</attributes>**

**</classpathentry>**

...

File : “.project”

<projectDescription>

<name>henrycublog</name>

<comment>NO\_M2ECLIPSE\_SUPPORT: Project files created with the maven-eclipse-plugin are not supported in M2Eclipse.</comment>

<projects/>

<buildSpec>

<buildCommand>

<name>org.eclipse.jdt.core.javabuilder</name>

</buildCommand>

</buildSpec>

<natures>

<nature>org.eclipse.jdt.core.javanature</nature>

</natures>

</projectDescription>

## Unrecognized Windows Sockets error: 0: JVM\_Bind

此异常的原因是服务器端口被占用 所以解决办法是：

一：更改服务器的端口号；

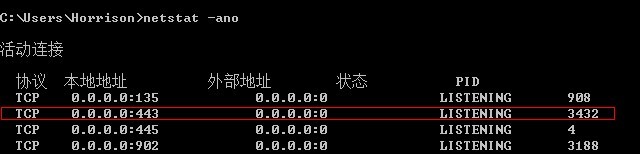
二：关闭占用当前端口的进程

下边介绍第二种方法

1.首先进入命令行 查看端口是否被占用

使用命令： netstat -ano

我的服务器的端口是443

此端口已被PID为3432的进程占用  


2.查看该PID为 3432 的进程

使用命令：tasklist|findstr "3432"

由于在解决问题时，我把PID为3432的进程已关掉，所以演示时，我用2784来做

http://www.blogjava.net/images/blogjava_net/huozhicheng/%E6%9F%A5%E7%9C%8B%E8%BF%9B%E7%A8%8B.png

可以看出 PID为2784的进程是 ConnectifyService.exe

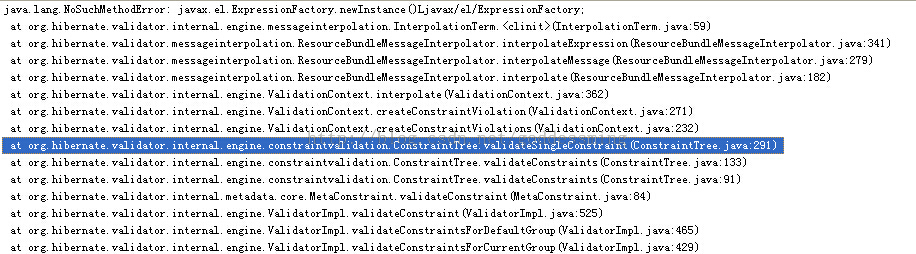
第一种解决办法是：把ConnectifyService.exe进程关掉

第二种解决办法是：直接通过任务管理器查看PID为3432的进程，然后关掉

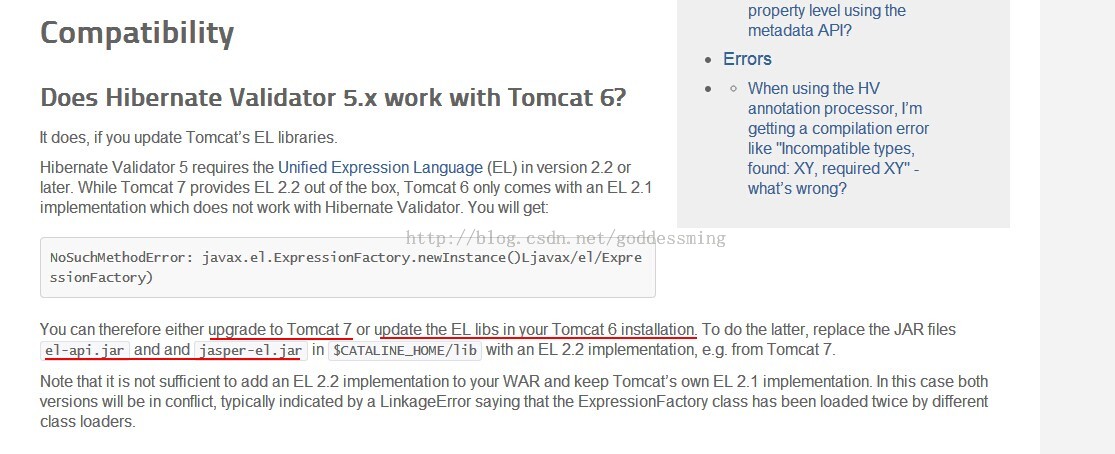
windows任务管理器->查看->选择列  选中PID 然后查看任务管理器，关掉PID为3432的进程

## javax.el.ExpressionFactory.newInstance()Ljavax/el/ExpressionFactory;

报错情况：在SpringMVC与Hibernate的环境下，当使用了Hibernate Validator验证后，修改更新实体时就报错，报错为：java.lang.NoSuchMethodError: javax.el.ExpressionFactory.newInstance()Ljavax/el/ExpressionFactory;



原因：查了Hibernate文档才知道，我使用的是Hibernate-validator-5.1.1.Final.jar，官网是这么说的（网址：http://hibernate.org/validator/faq/）



意思是：

HIbernate的5.0以上的验证是否兼容tomcat6？

Hibernate Validator 5需要统一表达式语言(EL)在版本2.2或更高版本。当Tomcat 7提供了开箱即用的EL 2.2,Tomcat 6只提供了一个EL 2.1实现不使用Hibernate验证器。您将得到:NoSuchMethodError:javax.el.ExpressionFactory.newInstance()Ljavax / el / ExpressionFactory)  
因此你可以升级到Tomcat 7或更新安装Tomcat 6中的EL的jar包。做后者,替换JAR el-api.jar和jasper-el.jar。 在tomcat安装目录的bin目录下。

## [maven build NG: maven-surefire-plugin:2.7.1:test (default-test) on project](http://blog.csdn.net/y353027520dx/article/details/45025707)

maven报错：org.apache.maven.plugins:maven-surefire-plugin:2.7.1:test (default-test) on project

* plan1：追加set pom plugin，

<!-- test artifact plugin -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.7.1</version>

<configuration>

<skipTests>true</skipTests>

</configuration>

</plugin>

* Plan2: install jar,

http://repo1.maven.org/maven2/org/apache/maven/plugins/maven-surefire-plugin/2.7.1/

mvn install:install-file -Dfile=D:\Administrator\Downloads\maven-surefire-plugin-2.7.1.jar -DgroupId=org.apache.maven.plugins -DartifactId=maven-surefire-plugin -Dversion=3.1.0.RELEASE -Dpackaging=jar

## ERROR: JDWP Transport dt\_socket failed to initialize, TRANSPORT\_INIT(510)异常

<http://blog.csdn.net/cicada688/article/details/8549031>

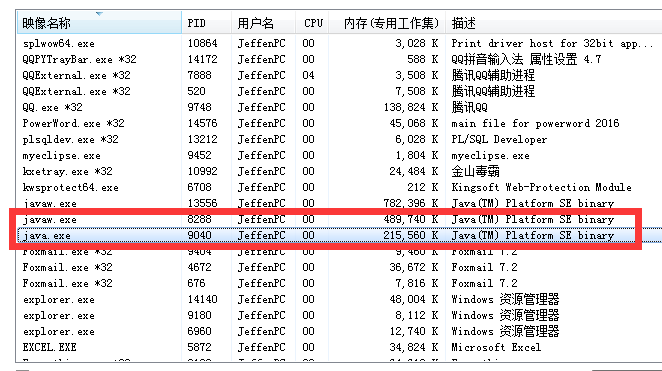
RROR: transport error 202: bind failed

ERROR: JDWP Transport dt\_socket failed to initialize, TRANSPORT\_INIT(510)

JDWP exit error AGENT\_ERROR\_TRANSPORT\_INIT(197): No transports initialized [../../../src/share/back/debugInit.c:690]

FATAL ERROR in native method: JDWP No transports initialized, jvmtiError=AGENT\_ERROR\_TRANSPORT\_INIT(197)

netstat -na|grep 8888



# Eclipse+Maven构建Web项目

环境准备:  
JDK: 1.6  
Eclipse: Indigo 3.7.2 - http://www.eclipse.org/downloads/packages/release/indigo/sr2  
Maven: apache-maven-3.1.0 - http://maven.apache.org/download.cgi

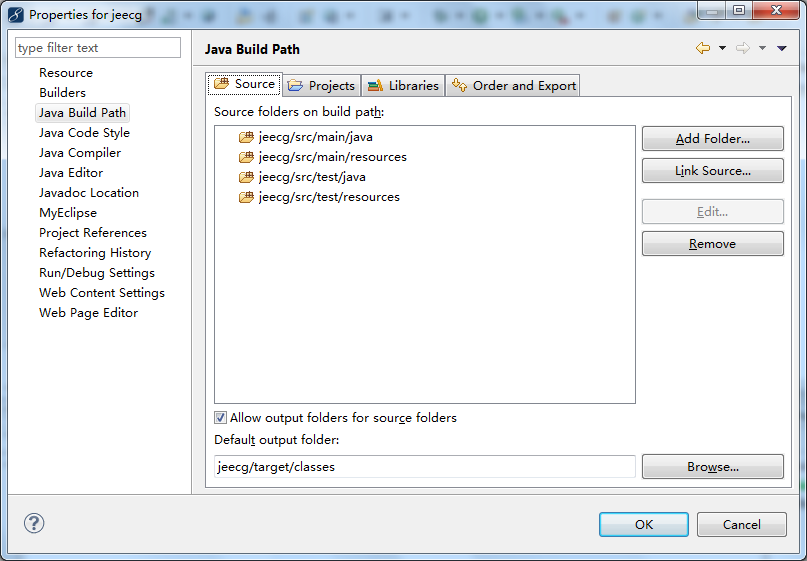
1. Maven  
1.1 Maven安装  
     解压[下载](http://www.2cto.com/soft)的Maven zip包,放置于F:\Tool\apache-maven-3.1.0  
     设置环境变量 MAVEN\_HOME = F:\Tool\apache-maven-3.1.0  
     追加用户变量：在path中追加%MAVEN\_HOME%\bin  
     打开cmd，运行mvn -version查看maven是否安装正确

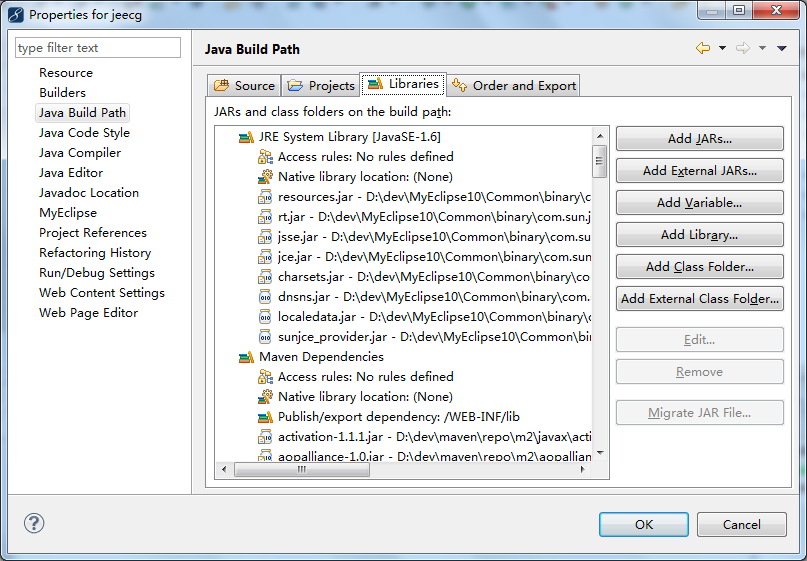
1.2 Maven配置：  
     修改maven 仓库存放位置：F:\Tool\apache-maven-3.1.0\conf.settings.xml  
     <localRepository>F:/Tool/apache-maven-3.1.0/repo</localRepository>

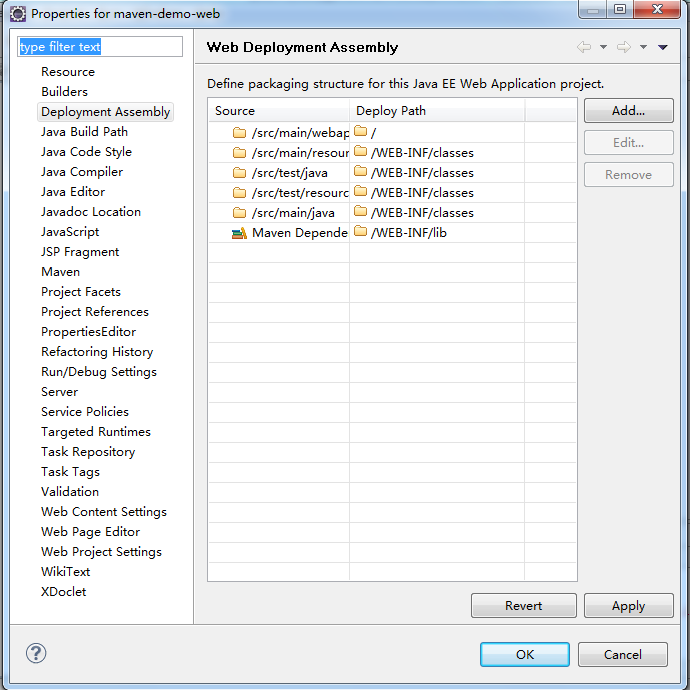
1.3 运行Maven  
     打开cmd，运行 mvn help:system  
     maven 就会从远程仓库开始下载一大堆的东西

2. Maven Eclipse插件安装  
2.1 插件安装  
     Eclipse3.7以上的版本都可以通过Eclipse Marketplace安装.  
     打开Marketplace，搜索Maven,找到Maven Integration for Eclipse进行安装  
2.2 配置Maven in Eclipse  
     Window  -->  Preferences --> Maven  
2.2.1 点击Installation --> 点击Add 添加本地的MavenF:\Tool\apache-maven-3.1.0  
2.2.2 点击User Setting 确保Maven setting的文件的位置为F:\Tool\apache-maven-3.1.0\conf\settings.xml  
     注意：如果配置不改，使用Maven默认的设置,在创建web工程时会出现Jar引用混乱的错误

3. Maven创建Web工程  
3.1 新建Maven Project  
     File -> New -> Other -> Maven -> Maven Project  
3.2 选择项目路径 Usedefault Workspace location  
3.3 选择项目类型 在搜索框中输入web, 选择maven-archetype-webapp  
3.4 输入Group ID和 Artifact ID，以及Package  
     Group ID 项目的唯一标识. Artifact ID是子项目名称. Package是默认建一个包，不写也可以  
3.5 把项目变成Dynamic Web项目  
     右键项目,选择Project Facets，点击Convert to faceted from...  
     勾选Dynamic Web Module, [Java](http://www.2cto.com/kf/ware/Java/), 也可勾选其他需要的[组件](http://www.2cto.com/kf/all/zujian/),如Javascript.  
     JDK的设置需要与Dynamic Web Module的版本一致. 3.0为Java7的,2.5/2.3为JDK1.6的.因为本机是jdk1.6的因此需要选择Dynamic Web Module的Version为2.5  
     上面步骤设置完成后，点击OK，Properties窗口会关闭，重新右键项目,在左侧列表中会出现一个Deployment Assembly  
3.6 配置项目  
     需要确保项目中包含添加src/main/java,src/test/java,src/test/resources三个Source Folder, 如果不存在需要创建。注意是Source Folder不是普通的Folder  
3.7 更改class路径

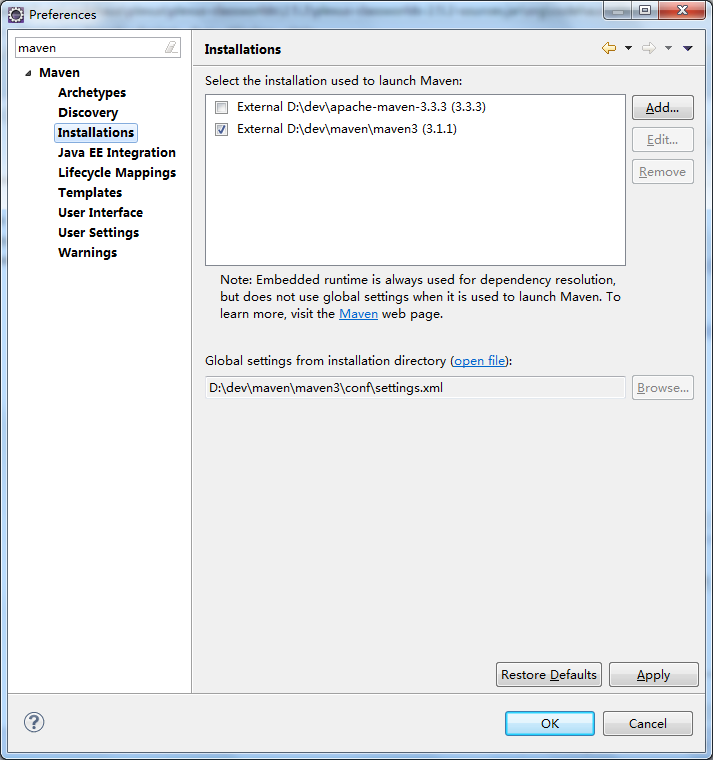
  
     右键项目, Java Build Path -> Source 下面应该有4个文件夹, src/main/java，src/main /resources，src/test/java ，src/test/resources  
     选上Allow output folders for source folders  
     双击每个文件夹的Output folder，查看target的路径  
      src/main/java，src/main/resources，选择target/classes;  
      src/test/java ，src/test/resources, 选择target/test-classes;  
     更改JDK版本：在Libraries双击JRE System Library，确认你需要的JDK版本. 本机目前是要1.6版本

  
3.8 设置部署程序集(Web Deployment Assembly)  
3.8.1 删除其中的WebContent，新增src/main/webapp  
3.8.2 设置将Maven的jar包发布到lib下。  
      Add -> JavaBuild Path Entries -> Maven Dependencies -> Finish  
      <http://img.my.csdn.net/uploads/201211/23/1353639542_8233.png>

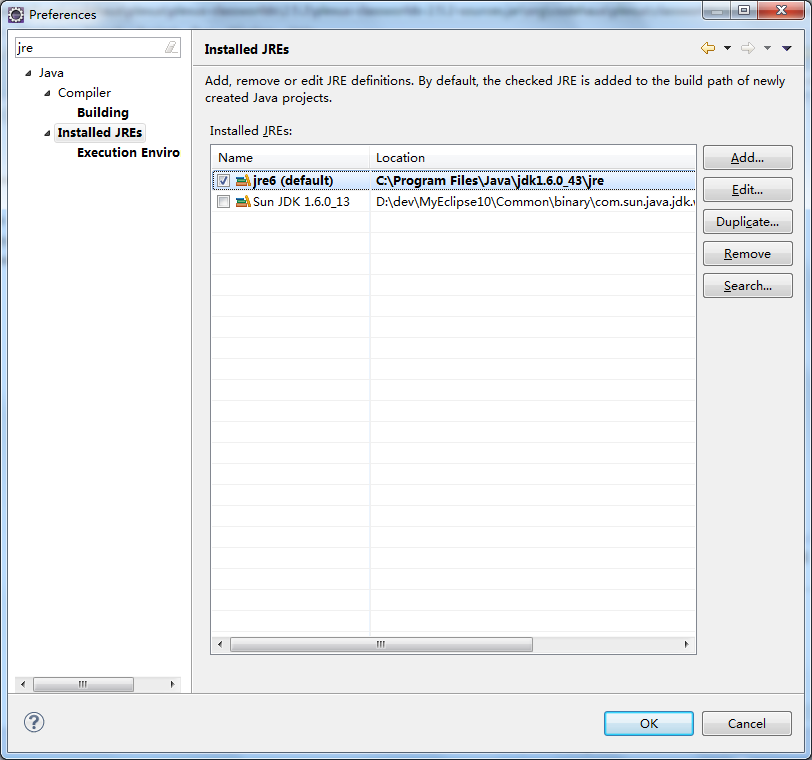
  
3.8.3 发布  
      Run As ->Maven install

NG issue: Exception in thread "main" java.lang.UnsupportedClassVersionError: org/apache/maven/cli/MavenCli : Unsupported major.minor version 51.0

Solution: ① change the maven3.3.3 to 3.1.1



②change jre url to Jdk/jre, for the tools.jar

  
      Run on server

Run eclipse.bat NG issue：《.project》 ：NO\_M2ECLIPSE\_SUPPORT: Project files created with the maven-eclipse-plugin are not supported in M2Eclipse. - See more at: http://blog.teamextension.com/m2eclipse-m2e-support-for-maven-eclipse-plugin-projects-497#sthash.5EmQUBiG.dpuf

# 修改myeclipse的workspace所在的位置

首次启动Eclipse/MyEclipse时, 会弹出"Workspace Launcher"对话框, 提示设置Workspace路径. 设定好路径后, 倘若勾选了"Use this as the default and do not ask again", 那么以后再启动时就不会有提示, 直接进入默认工作空间.   
有3中方法可以更改workspace的路径设置：  
1. 启动Eclipse/MyEclipse后, 打开"Window -> Preferences -> General -> Workspace", 点Workspace页上的"Startup and Shutdown", 然后勾选"Startup and Shutdown"页中的"Prompt for workspace on startup";  
2. 用记事本打开"\eclipse\configuration\.settings\org.eclipse.ui.ide.prefs", 将"SHOW\_WORKSPACE\_SELECTION\_DIALOG"的值改为"true";  
P.S.: "RECENT\_WORKSPACES"的值表示设置过的workspace绝对路径. 第一个路径是当前设定的路径, 向后依次之前曾设置过的. 各路径之间用"\n"分隔.  
3. 删掉"\eclipse\configuration\.settings\org.eclipse.ui.ide.prefs".  
执行上述操作后, 再次启动, 又会弹出"Workspace Launcher"对话框, 可以重新设置了.

# Myeclips调优优化

## 内存调整：myeclipse.ini

-vmargs

-Xmx1024m

-XX:MaxPermSize=512m

-XX:ReservedCodeCacheSize=128m

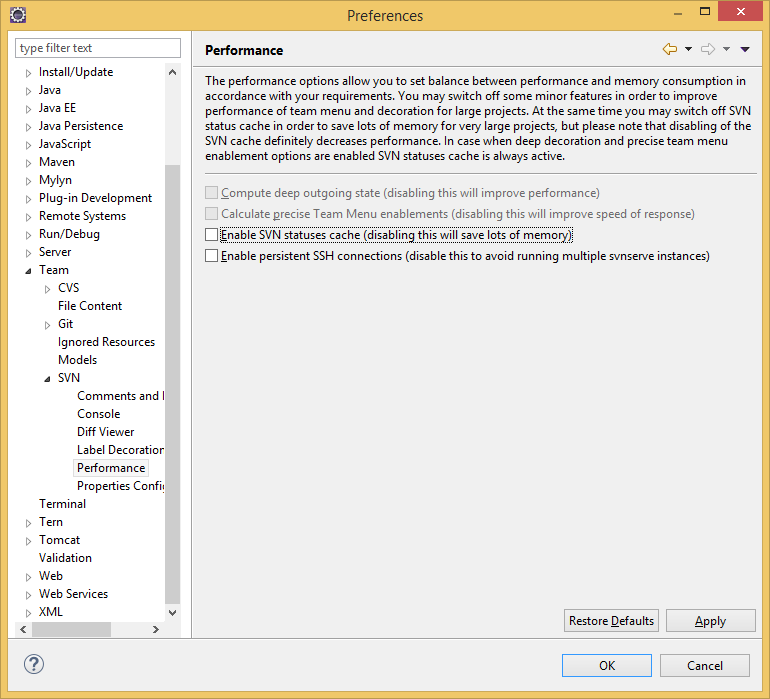
-Dosgi.nls.warnings=ignore

## [svn](http://blog.csdn.net/y353027520dx/article/details/45025707) performance

via. <http://stackoverflow.com/questions/11086474/eclipse-update-svn-cache-hangs-and-locks-up>

Eclipse: “Update SVN cache” hangs and locks up

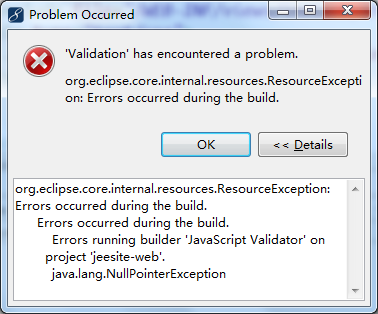
* Windows > Preferences > Team > SVN > Performance
* Disable: Computing deep outgoing state, Cache, persistent SSH



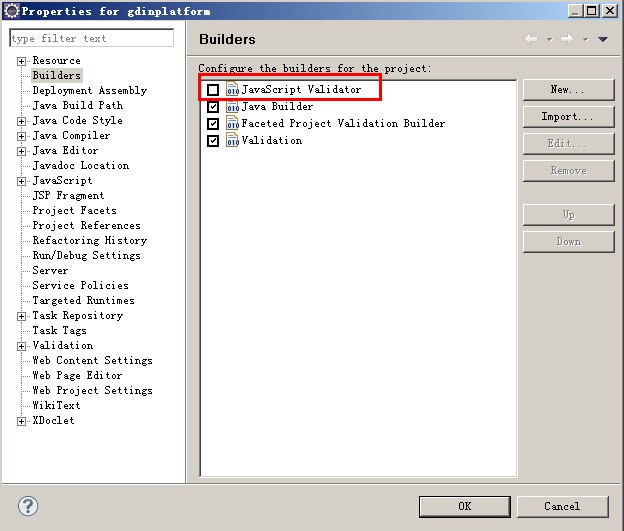
## Unhandled event loop exception GC overhead limit exceeded

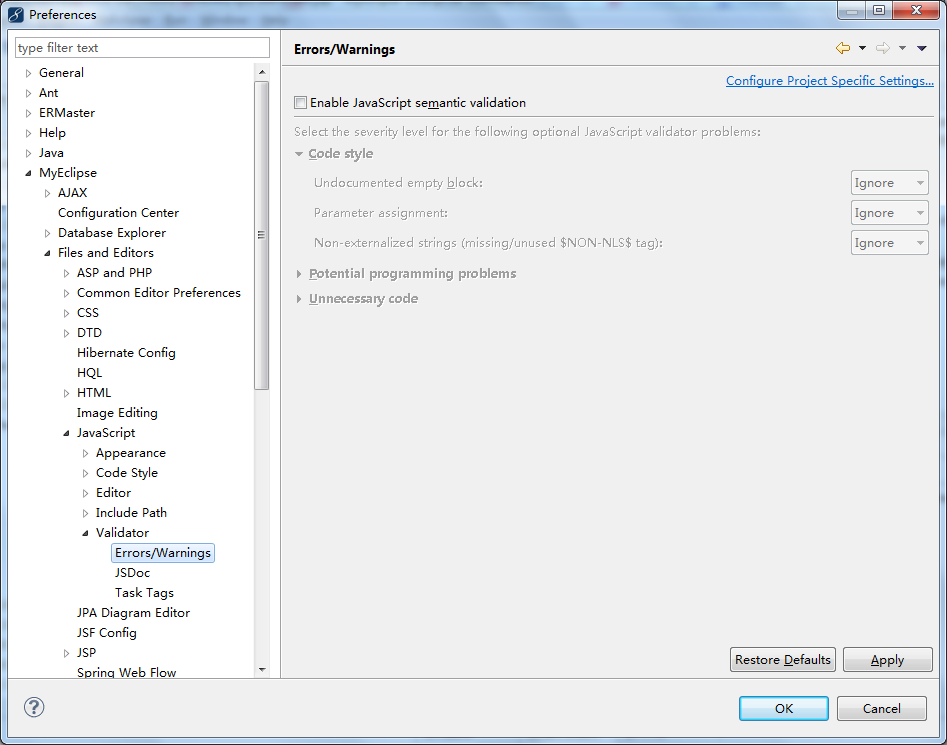
## An internal error occurred during: "Building workspace".GC overhead limit exceeded

## Errors occurred during the build. Errors running builder 'JavaScript Validator' on project 'jeesite-web'. java.lang.NullPointerException



右击项目名–Properties–Builders–将JavaScript Validator前面的勾去掉即可。

[](http://gevin.me/wp-content/uploads/2012/05/eclipse-javascript-validate2.jpg)或preferences首选项：



## Available Memory Is Low

MyEclipse has detected that less than 5% of the 113MB of PS Eden Space (Heap memory) space remains.

It is strongly recommended that you exit and restart MyEclipse with new virtual machine memory

parameters to increase this memory; failure to do so can result in data loss.

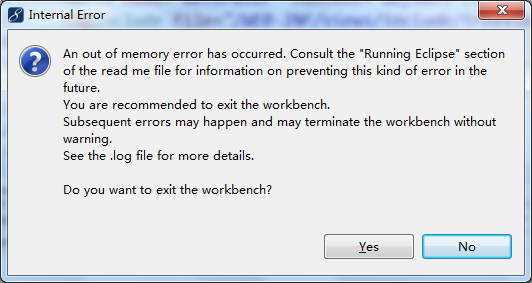
Recommended memory arguments:

-vmargs

-Xmx768m

-XX:MaxPermSize=384m

-XX:ReservedCodeCacheSize=64m



# tomcat优化

内存调整(Windows)：tomcat\bin\Catalina.bat

(catalina.sh for linux)

rem memory 1G

set JAVA\_OPTS=%JAVA\_OPTS% -server -Xms800m -Xmx800m -XX:PermSize=64M -XX:MaxNewSize=256m -XX:MaxPermSize=128m -Djava.awt.headless=true

rem memory 4G

set JAVA\_OPTS=%JAVA\_OPTS% -server -Xms2048m -Xmx3072m -XX:PermSize=256M -XX:MaxNewSize=1024m -XX:MaxPermSize=512m -XX:-UseGCOverheadLimit -Djava.awt.headless=true

