Hudson平台搭建及使用

1>Hudson简介

Hudson 是一种革命性的开放源码 CI (持续集成)服务器，它从以前的 CI 服务器吸取了许多经验教训。Hudson 最吸引人的特性之一是它很容易配置：很难找到更容易设置的 CI 服务器，也很难找到开箱即用特性如此丰富的 CI 服务器。Hudson 容易使用的第二个原因是它具有强大的插件框架，所以很容易添加特性。例如，一个 Hudson 插件可以随时间的推移跟踪 FindBugs 和代码覆盖。它还可以报告测试结果的趋势（来自 JUnit 或 TestNG）以及构建结果和对应的执行时间。  
hudson主要用于：持续、自动地构建/测试软件项目.监控

Hudson 需要运行 java5或以上版本，容器Servlet2.4或以上的版本。

2>Hudson平台搭建

下载最新版本，（它打包为一个 WAR 文件）。hudson官方网址: <http://hudson-ci.org/>

将hudson.war包文件直接部署到servlet容器中，然后启动容器，在浏览器中输入：<http://localhost:8080/hudson/> 就会出现hudson的主界面。

3>Hudson系统配置

进入hudson的主页面，点左边的系统管理🡪系统配置

主目录，这里包含hudson要执行的任务所在工作区与相关配置信息。

系统消息：填写一些说明信息

执行者数量：能同时执行的任务数量，

生成前等待时间：一个计划中的构建在开始之前需要等待选项中设置的秒数

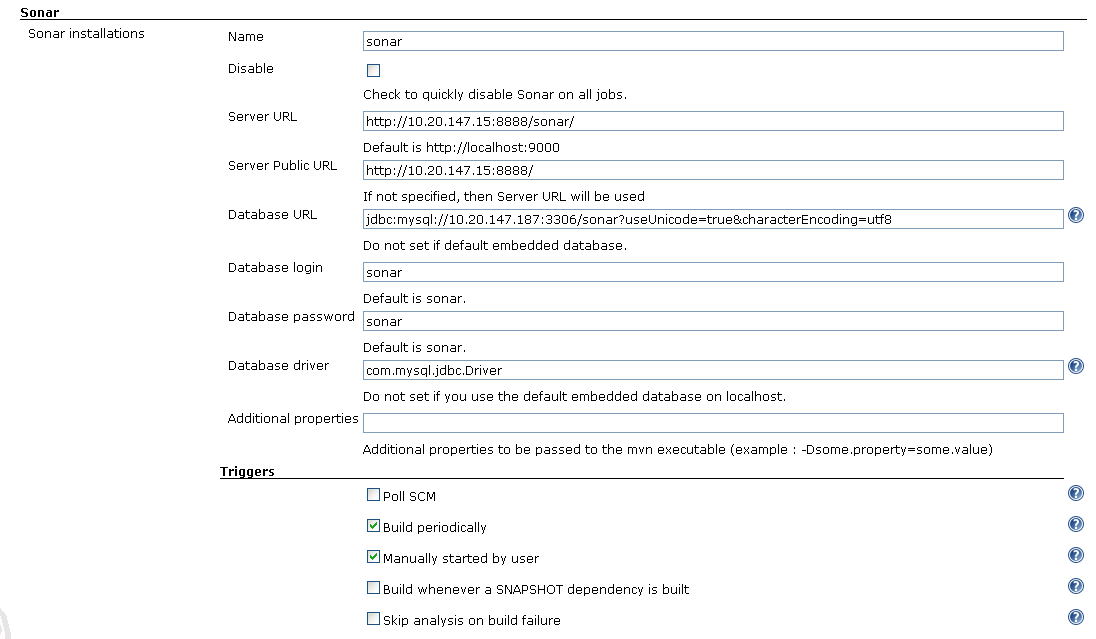
SCM签出重试次数：如果从版本库签出代码失败,Hudson会按照这个指定的次数进行重试之后再放弃.

启用安全:设置hudson的登录规则，默认登录为匿名登录，

Global properties:设置一些全局变量，类似于windows的环境变量

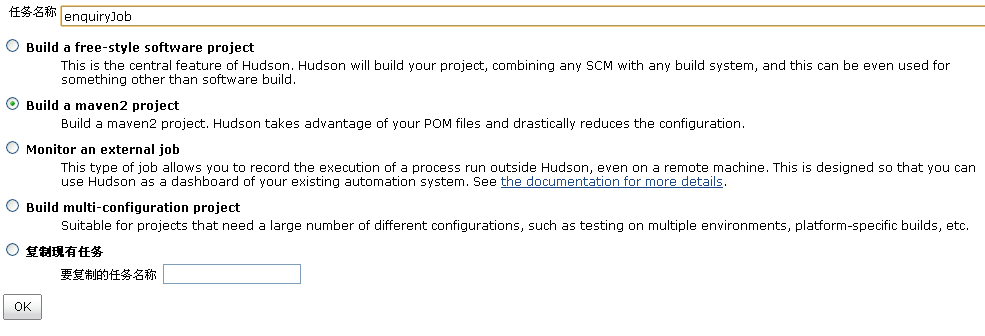
JDK installations:设置JDK的安装路径  
Shell executable:设置window shell命令  
Ant installation:设置ant 的安装路径  
mave installation设置mave的安装路径  
cvs executable:设置cvsnt执行进程的路径(cvs.exe)  
.cvspass file:设置cvsnt管理员文件的路径(passwd文件)  
e-mail notification:设置当发生错误时发送的邮箱地址,多个邮箱地址以空格分开  
hudson url:就是hudson的默认地址

Sonar配置,如图

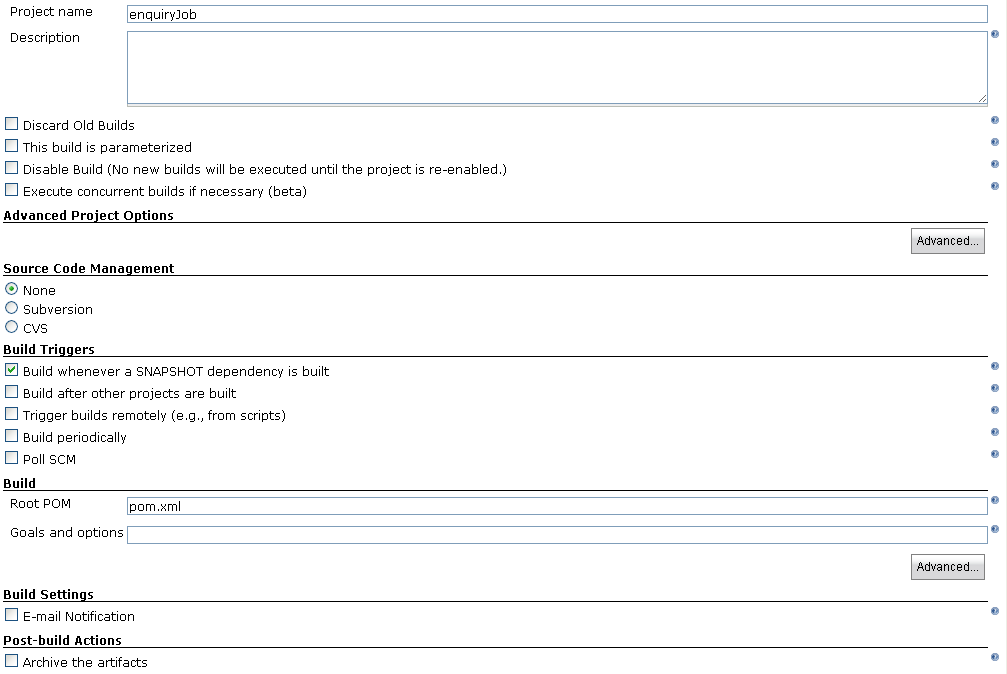


4>新建任务

点击右边的新建任务，会出现下图所示页面



这时，需要为新的构建任务指定一个名称。（这里输入的任务名称为：maven-introduce）这里有几种的任务类型可供您选择，鉴于初步介绍，先选择**Build a free-style software project** ，这个更加普遍化，可以更灵活的配置，一般二方库的部署建议选择该种类型，对于其他的类型,经常使用的是**Copy existing job**;这主要为了能在复制现有的任务基础上新建一次构建任务。如果是maven部署方式的应用可以选择build a maven2 product工程。点击OK按钮，进入下一界面。如下图：



图中有很多项目构建的设置

Project name：项目名称

Description：项目说明

Discard Old Builds：是控制着您想要在Hudson所在的磁盘把构建记录存储的有效期 (诸如控制台输出、编译产成品等等)。Hudson为此提供了两个标准：1。时间驱动。在Hudson中您可以判断如果达到一定时限来删除一条记录(例 如，七天前)。2。数量驱动。在Hudson中您可以确保它拥有N份构建。如果又有新的构建开始，最早的那份(记录)就将被删除。Hudson也可以让您 建立的个别构建定义为'永远保持这个记录'，以便防止某些重要的构建被自动丢弃。)这里输入的是5，也就是hudson确保5份构建。

This build is parameterized：设置编译时参数

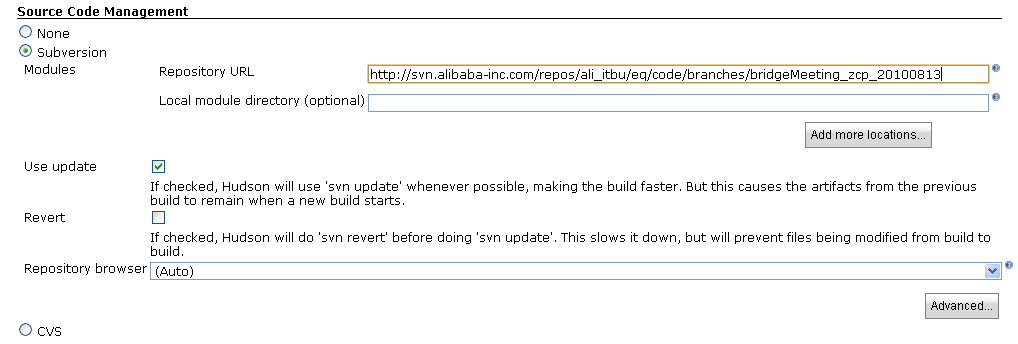
Disable Build:选择将会禁用此构建任务

|  |  |
| --- | --- |
|  |  |
| Advanced Project Options：高级项目设置， | |

Source Code Management：源代码管理 , 在默认情况下是这三个选项：Subversion,CVS,None

如果您安装了一个与SCM相关的插件，并重新启动Hudson，那么在这个清单上也将出现一些新的选择。

图为svn的配置



Repository <URL:为本任务需要构建的源代码分支>

Use update ：选中的话。每次构建时。会更新下代码分支，再执行构建

Build Triggers

Build whenever a SNAPSHOT dependency is built：当构建成功后，项目jar包会发到maven二方库上去。

Build after other projects are built : 此选项支持一条装配(流水作业)线——作业依赖: 一个作业依赖于另一个作业的输出的情况 —— 就是可以支持连续编译多个任务，设置这个选项后，在其他任务编译之后，可以立即自动启动这个任务，例如，我们通常在项目中会有二方库的部署，我们设置，先部署二方库，然后在编译我们的应用，使得依赖于二方库的应用能够正常编译成功。

Trigger builds remotely (e.g., from scripts)：远程触发脚本

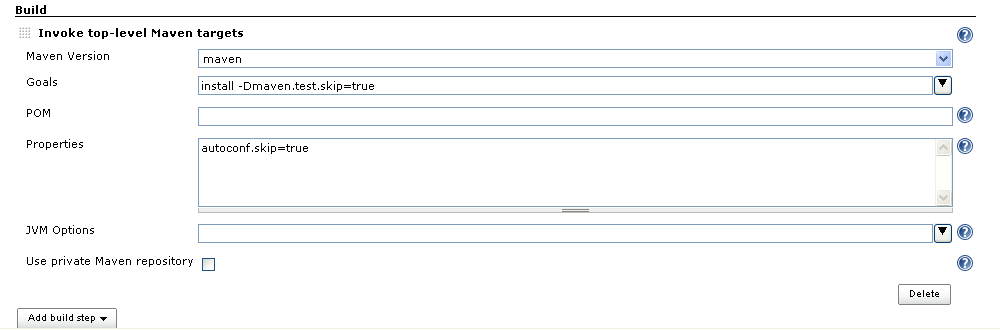
Build periodically：此选项 (也是使用定时作业表达式)仅仅通知Hudson按指定的频率对项目进行构建，而不管SCM是否有变化。如果想在这个作业中运行一些测试用例的话，它就很有帮助。

Poll SCM:定期更新仓库代码，也是使用定时作业表达式

<http://blog.csdn.net/zhuxinhua/archive/2010/08/30/5849265.aspx>



可以填入mvn的配置文件，以及编译命令。



如果建的是freeStyle项目,需要覆盖本地maven.properties的时候，需要设置Properties:autoconf.skip=true

E-mail notification : 当您选择此选项，您可以输入一个或多个电子邮件地址[多个可用空格分隔]，当Hudson完成了执行作业后，将会给它们发送通知。事件触发时将产生一份 Email，包括构建失败、构建不稳定等。



图为构建后的一些相关操作，

目前只选择了一个sonar

点击save就完成了一个任务的新建

点击左边的立即构建，就可以开始新任务的构建了。

5🡪作业说明



上面的build History 下面的#1有下滚动条，说明正在构建中，点击它会出现下图



点变更集会显示本次构建更新了哪些代码。

点命令行输出，可以查看构建时的日志，和我们查看build项目时的jbosslog一样。



作业状态

     在上面展示的截图中，您将注意到有两个图标描述当前作业的状态。Hudson使用两个概念来介绍一个作业的总体状况：

作业状态:  图中分级符号概述了一个作业新近一次构建会产生的四种可能的状态：

Successful: 完成构建，且被认为是稳定的。

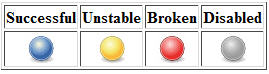
Unstable:    完成构建，但被认为不稳定。

Failed:  构建失败。

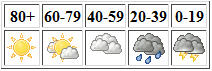
Disabled: 作业已禁用。

作业稳定性: 当一个作业中构建已完成并生成了一个未发布的目标构件，如果您准备暗中评估此次构建的稳定性，Hudson会基于一些后处理器任务为构建发布一个稳健指数(从0-100 )，这些任务一般以插件的方式实现。它们可能包括单元测试(JUnit)、覆盖率(Cobertura )和静态代码分析(FindBugs) 。分数越高，表明构建越稳定。图28中分级符号概述了稳定性的评分范围。任何构建作业的状态(总分100)低于80分就是不稳定的。

作业状态图



作业稳定性图



6>插件安装

系统管理🡪管理插件🡪可选插件🡪选中需要安装的插件点击右下角的安装即可