**Jenkins配置  
版本号：Ver2.0**

# 版本说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本号** | **日期** | **內容** | **作者** | **审核人** |
| **Ver1.0** | 2016-01-20 | 初稿 | 邓世伟 |  |
| **Ver2.0** | 2016-03-23 | 增加Multiple SCMs 插件 | 邓世伟 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 目录结构

[版本说明： 2](#_Toc17043)

[目录结构 3](#_Toc14864)

[1. Mac下配置jenkins 4](#_Toc10206)

[1.1 下载及安装 4](#_Toc23122)

[1.2 访问jenkins 4](#_Toc9551)

[1.3 端口修改 4](#_Toc23419)

[1.4 配置用户 5](#_Toc30975)

[1.5 下载插件 6](#_Toc4107)

[1.6 配置插件属性 6](#_Toc11535)

[1.7 发布Android构建 7](#_Toc23735)

[1.8 发布IOS构建 10](#_Toc29206)

[2. Windows下配置jenkins 14](#_Toc23461)

[2.1 下载jenkins及安装 14](#_Toc7646)

[2.2 访问jenkins 14](#_Toc9909)

[2.3 配置用户 15](#_Toc31007)

[2.4 安装插件 16](#_Toc26818)

[2.5 系统设置 17](#_Toc7205)

[2.6 发布Android构建 18](#_Toc5700)

[3. 分布式部署 20](#_Toc22910)

[3.1 介绍 20](#_Toc28609)

[3.2 部署 20](#_Toc28439)

[3.3 服务器 21](#_Toc17635)

[4. Multiple SCMs plugin插件 23](#_Toc14727)

[4.1 介绍 23](#_Toc25909)

[4.2 下载安装 23](#_Toc16972)

[4.3 配置使用 24](#_Toc21459)

# Mac下配置jenkins

## 下载及安装

1. 在官网上直接下载MAC的dmg安装包，安装完毕后即可，官网地址：

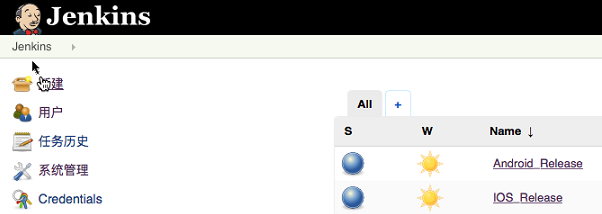
[http://jenkins-ci.org/](http://jenkins-ci.org/" \t "http://www.xuanyusong.com/archives/_blank)

1. 执行 brew install jenkins 安装，你会看到安装的一些信息。等安装好以后,在执行命令启动服务：

java -jar /usr/local/opt/jenkins/libexec/jenkins.war

## 访问jenkins

在浏览器里面打开 http://localhost:8080/ 局域网内别人可以打开http://”ip”:8080/



## 端口修改

默认端口为8080，如果需要修改端口号的话，可以执行下面步骤。

1. 端口号修改，9999为端口号，执行命令：

sudo defaults write /Library/Preferences/org.jenkins-ci httpPort 9999

1. 查询端口号，执行命令：

defaults read /Library/Preferences/org.jenkins-ci

1. 修改了端口号重新启用服务，执行命令：

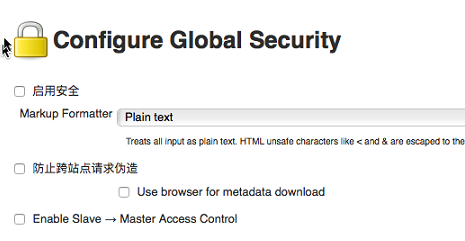
java -jar /usr/local/opt/jenkins/libexec/jenkins.war --httpPort=9999

1. 在浏览器里访问，执行命令：

<http://localhost:9999/>

## 配置用户

1. 系统管理-->Configure Global Security



1. 勾选启用安全，选择jenkins专有用户数据库，选择登录用户可以做任何事

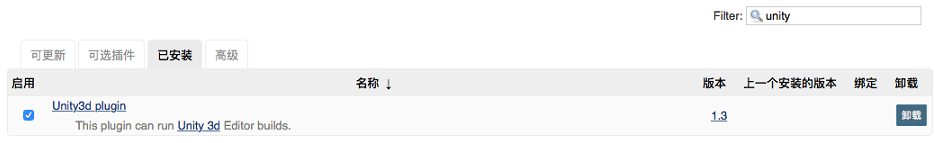


## 下载插件

1. 系统管理-->管理插件-->可选插件。输入需要安装的名称进行安装即可

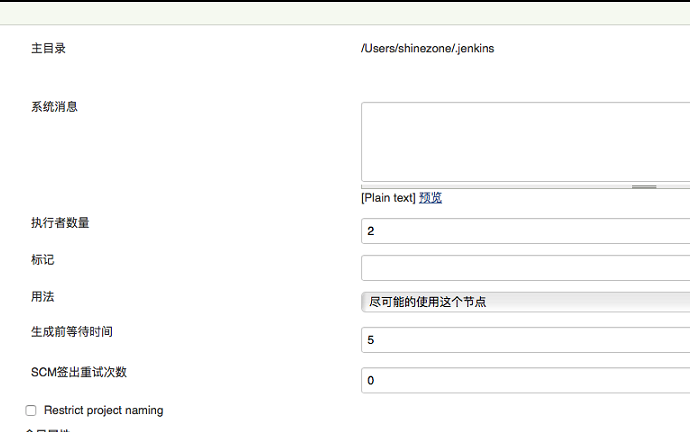


1. 需要安装的插件：Unity3d plugin，安装完成



## 配置插件属性

1. 系统管理-->系统设置

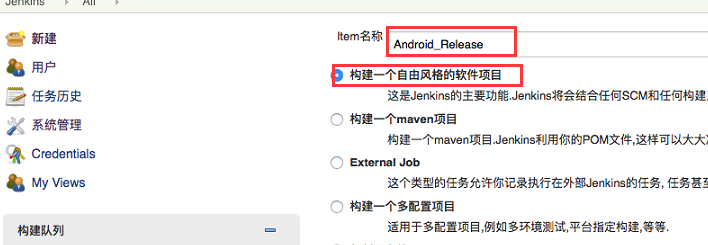


1. Unity3d plugin插件配置，配置名称及本地安装unity应用的路径

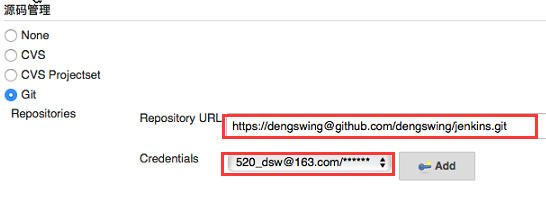


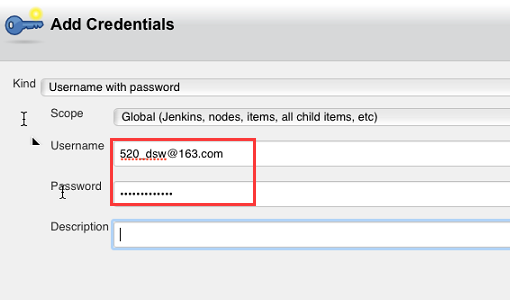
## 发布Android构建

1. 新建，输入名称，选择构建一个自由风格人软件项目

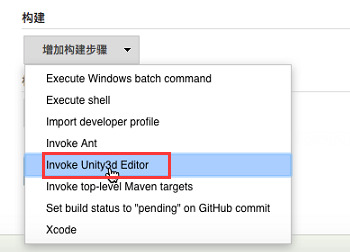


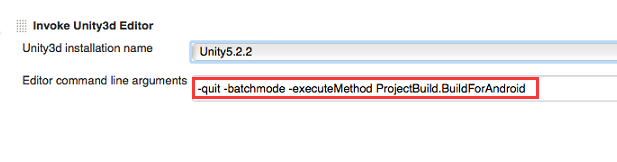
1. 源码管理，采用git方式，输入git地址及安全证书，安全证书这里使用的是账户的方式进行验证





1. 构建，增加构建步骤，选择Invoke unity3d Editor方式，输入-quit -batchmode -executeMethod ProjectBuild.BuildForAndroid (注如果需要写路径加上：-projectPath)执行命令，ProjectBuild.BuildForAndroid是unity打包发布android的一个静态方法





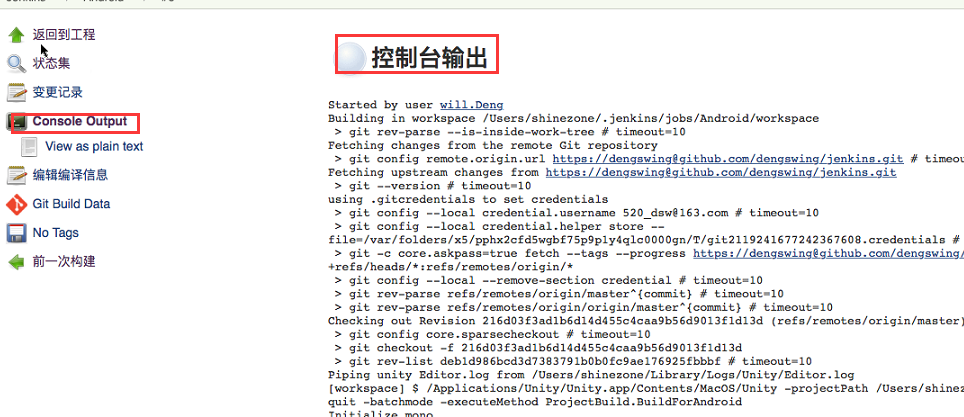
1. ProjectBuild.cs的代码



1. 完成以上几步保存，可以开始构建发布项目，点击刚才构建的项目-->立即构建,开始构建（构建时候记得关闭unity否则构建会失败）



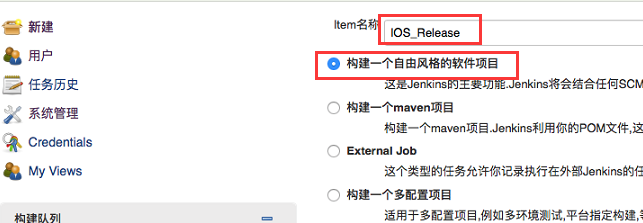
1. 开始构建，可以查看构建信息，点击项目-->点击左侧队列-->点击Console Ouptput查看输出



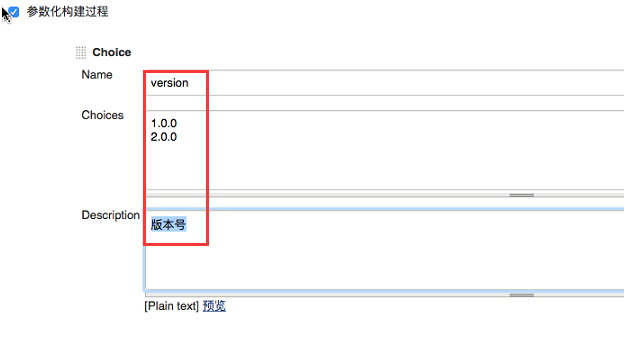
1. 构建结束，控制台输出，显示Finished:SUCCESS

## 发布IOS构建

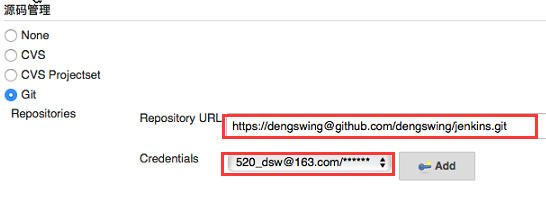
1. 新建，输入名称，选择构建一个自由风格人软件项目

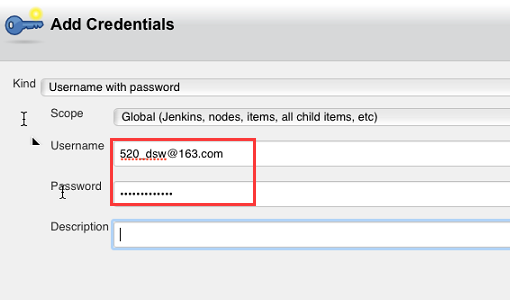


1. 勾选参数化构建-->点击增加参数-->选择Choice

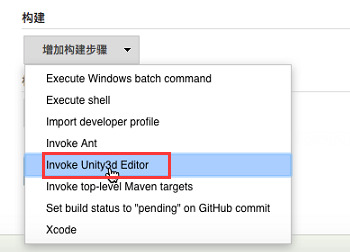


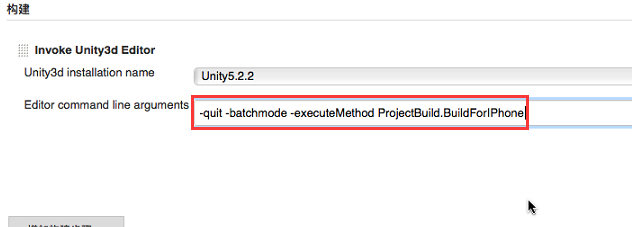
1. 源码管理，采用git方式，输入git地址及安全证书



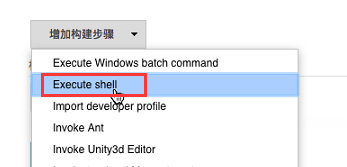


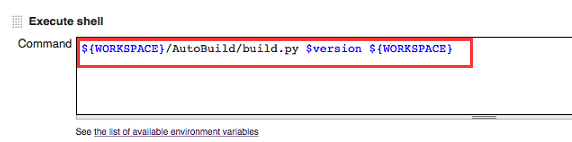
1. 构建，采用Invoke unity3d Editor方式，输入-quit -batchmode -executeMethod ProjectBuild.BuildForIPhone (注如果需要写路径加上：-projectPath)执行命令，ProjectBuild.BuildForIPhone是unity打包发布Ios的一个静态方法





1. 构建，采用Execute shell方式，输入${WORKSPACE}/AutoBuild/build.py $version ${WORKSPACE}执行命令，build.py来打包发布\*.ipa

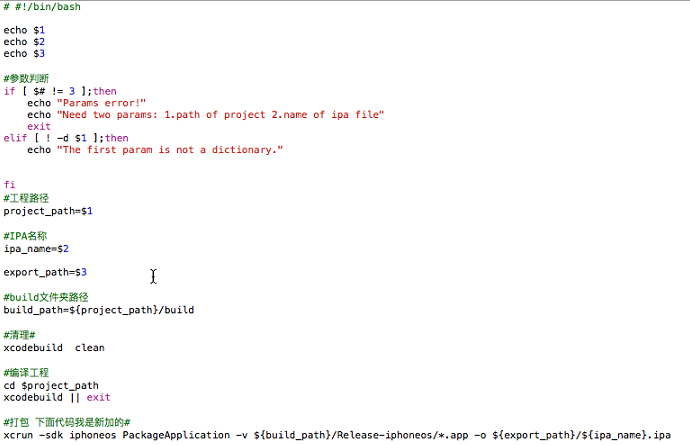




1. Build.py的代码



1. Buildios.sh的代码



1. 完成以上几步保存，可以开始构建发布项目，点击立即构建开始构建-->选择版本-->开始构建，（构建时候记得关闭unity否则构建会失败）



1. 开始构建，可以查看构建信息，点击项目-->点击左侧队列-->点击Console Ouptput查看输出



1. 构建结束，控制台输出，显示Finished:SUCCESS

# Windows下配置jenkins

## 下载jenkins及安装

[http://jenkins-ci.org/](http://jenkins-ci.org/" \t "http://www.xuanyusong.com/archives/_blank) 在官网上直接下载安装包,安装完毕后即可在本机完成搭建jenkins的工作



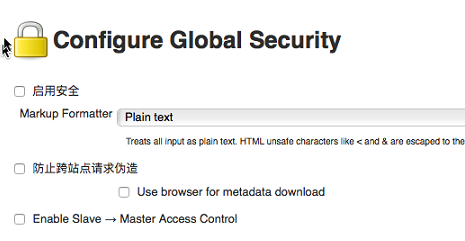
## 访问jenkins

在浏览器里面打开 http://localhost:8080/ 局域网内别人可以打开http://”你的ip”:8080/



## 配置用户

1. 系统管理-->Configure Global Security



1. 勾选启用安全，选择jenkins专有用户数据库，选择登录用户可以做任何事



## 安装插件

1. 系统管理-->管理插件-->可选插件。输入需要安装的名称进行安装即可



1. 需要的插件包括如下：
   1. [GitHub plugin](http://wiki.jenkins-ci.org/display/JENKINS/Github+Plugin)
   2. [Git plugin](http://wiki.jenkins-ci.org/display/JENKINS/Git+Plugin)
   3. [Unity3d plugin](http://wiki.jenkins-ci.org/display/JENKINS/Unity3dBuilder+Plugin)
   4. [Xcode integration](https://wiki.jenkins-ci.org/display/JENKINS/Xcode+Plugin)

## 系统设置

1. 系统管理--系统设置



1. JDK的设置,设置本地安装的sdk目录



1. Git的设置,设置本地安装的git目录，记得是cmd目录下面的git.exe

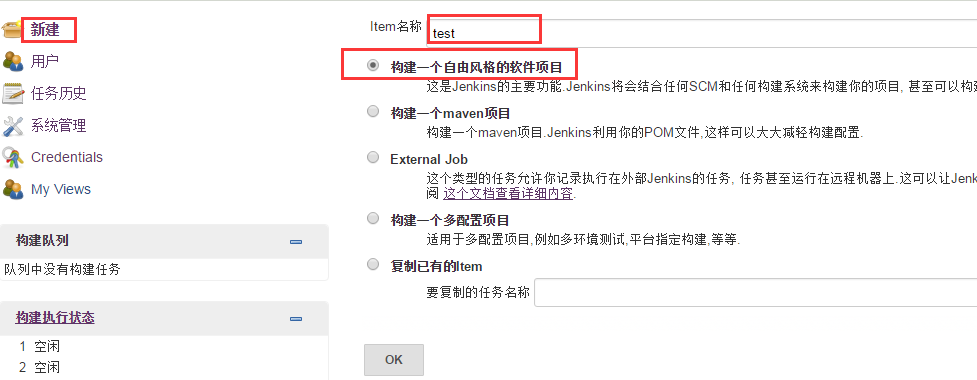


1. Unity3d的设置,设置安装的unity目录

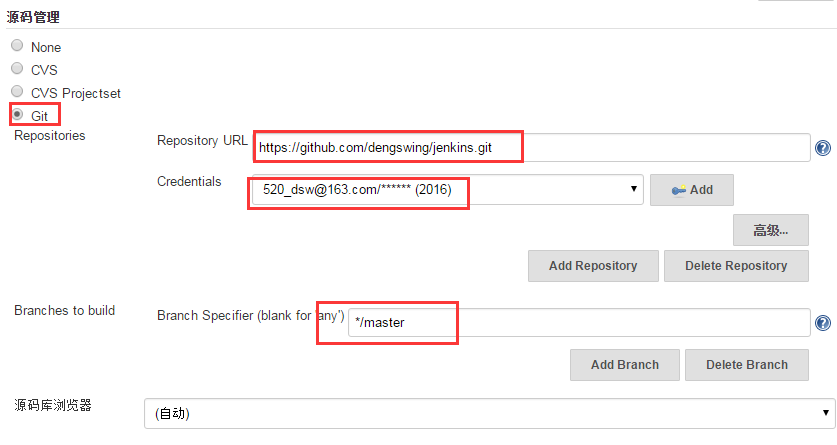


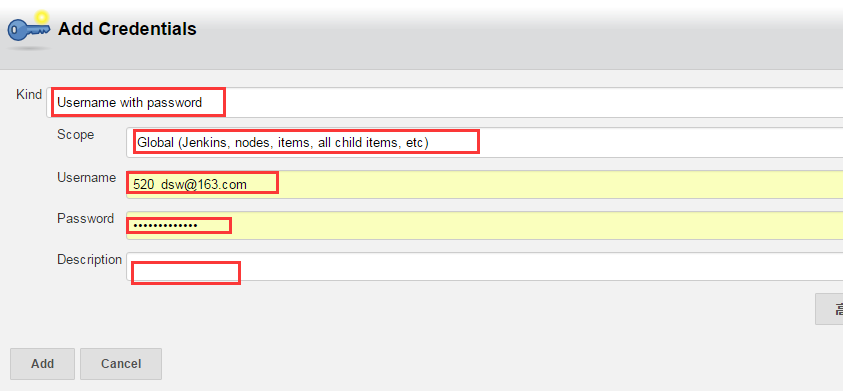
## 发布Android构建

1. 创建一个自动化配置，如下图：

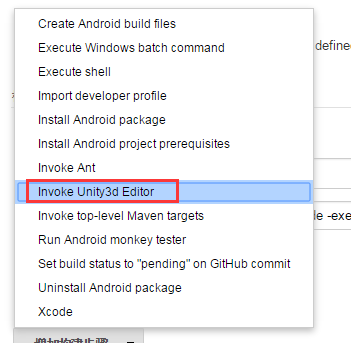


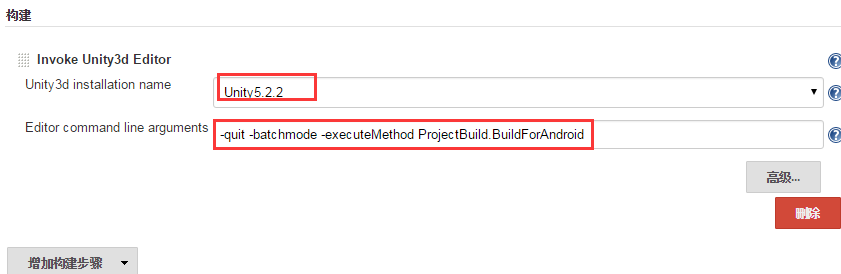
1. 红框选择完后,配置源码管理，这里使用的是git来管理，如下图：



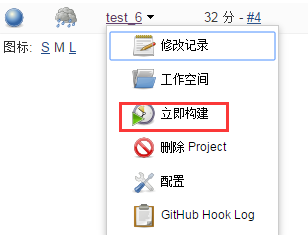


1. 构建,这里使用unity3d Editor,如下图：

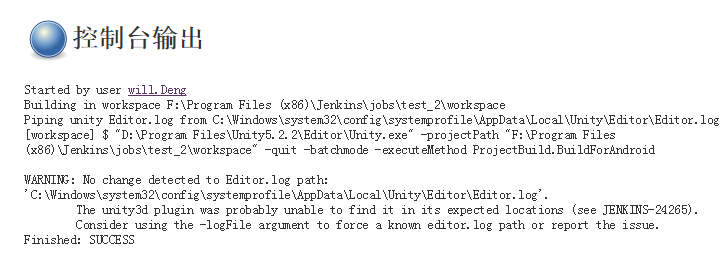




1. 完成以上几步保存，可以开始构建发布项目，点击立即构建开始构建，（构建时候记得关闭unity否则构建会失败）



1. 开始构建，可以查看构建信息，点击项目-->点击左侧队列-->点击Console Ouptput查看输出



1. 构建结束，控制台输出，显示Finished:SUCCESS

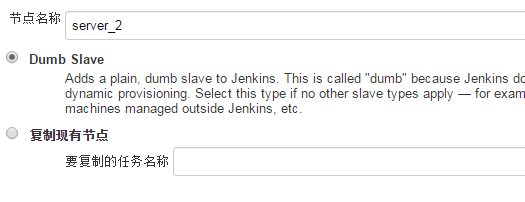
# 分布式部署

## 介绍

Jenkins的分布式构建，在Jenkins的配置中叫做节点，分布式构建能够让同一套代码或项目在不同的环境(如：Windows和Linux系统)中编译、部署等

## 部署

1. 系统管理-->管理节点-->新建节点(左边)，如下图



1. 配置基本信息，如下图



## 服务器

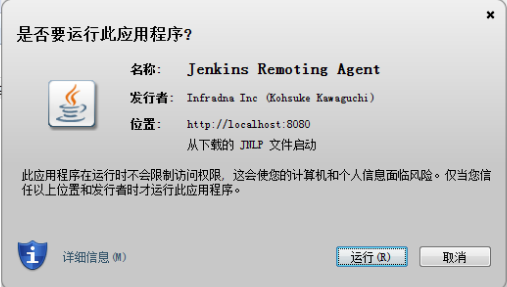
下载并安装节点服务器



1. 将slave-agent.jnlp文件复制到远程服务器的远程工作目录如：E:\jenkinsServer下
2. 双击运行slave-agent.jnlp，如果如法运行，在cmd命令中输入:

javaws E:\jenkinsServer\slave-agent.jnlp

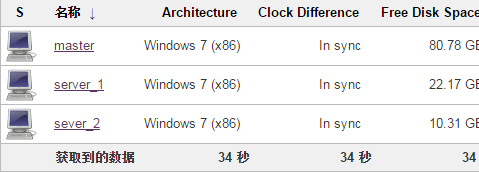
1. 运行过程如下图



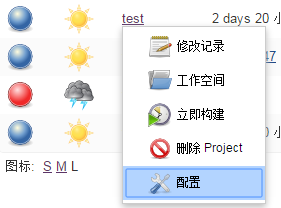
1. 点击运行如下图：

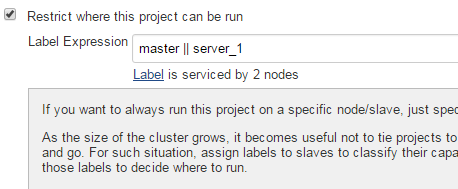


1. 将这个节点加入服务,点击窗口的file菜单，点击Install as a service,等待窗口关闭即完成



1. 绑定相关服务器，点击节点-->配置，如下图





# Multiple SCMs plugin插件

## 介绍

该插件是能管理多个版本库的资源，比如同一个项目，使用到的源码在不同的git库上，那么编译的时候就需要拉取不同git库中的源码。该插件能达到的效果如下：

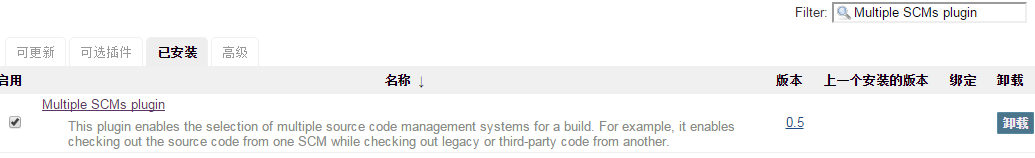
1. 同时监测多个版本库，其中有一个或者多个版本库有新的提交，就能自动触发新的构建
2. 同时监测的多个版本库中，可以是svn，可以是git等常用代码管理工具，也可以混搭

## 下载安装

1. 系统管理-->管理插件-->可选插件。输入需要安装的名称进行安装即可

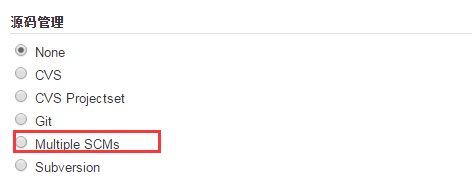


1. 需要安装的插件：Multiple SCMs plugin，安装完成

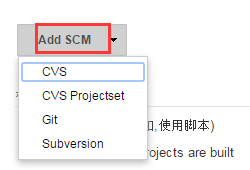


## 配置使用

1. 新建项目-->源码管理，可以看到我们刚刚安装的插件



1. 选择Multiple SCMs 插件，里面有git及svn等版本控制，选择需要的，这里可以多选及混搭



1. 下面我选择git来配置

