**Jenkins项目自动化构建部署(Windows环境)**

====================================================================  
【个人网站】：<http://www.2b2b92b.com>  
【网站源码】：<http://git.oschina.net/zhoubang85/zb>  
【技术论坛】：<http://www.2b2b92b.cn>  
【开源中国】：<https://gitee.com/zhoubang85>  
  
【支付-微信\_支付宝\_银联】技术QQ群：470414533  
【联系QQ】：842324724  
【联系Email】：842324724@qq.com  
====================================================================

前言

此文档教程是根据项目 <https://gitee.com/zhoubang85/zb-pay-dubbo> 进行讲解的，也是对zb-pay-dubbo这个项目如何使用Jenkins进行自动化构建的步骤讲解，对于初学者来说，也算是一个实际项目的构建案例吧。

文档将会从Jenkins的下载、安装已经环境配置、相关插件的安装(比如maven插件)、项目构建以及其他配置等内容进行介绍说明。

【补充】: zb-pay-dubbo是一个分布式服务项目，项目的最大特点是：以【商城下单支付】作为当前分布式项目的核心业务功能，最终的目的是通过该项目完整实现并解决【分布式事务】的问题，后期也许会加入其他技术点，不限于分布式事务。

常见的分布式事务解决方案：

1、可靠消息最终一致性（异步确保型）

2、最大努力通知（定期校对）

3、TCC（两阶段型、补偿型）

以上3种分布式事务的解决方案，将会在项目中通过代码完整的展现。

关于此项目更多的详细功能介绍，请查看 <https://gitee.com/zhoubang85/zb-pay-dubbo> 肯定会对你有所帮助。

通过这个项目，你将学会分布式事务的解决方案，也就是上面提到的3点方案，不同的方案在不同的业务场景下，有它自己的发挥作用。

Jenkins下载以及安装

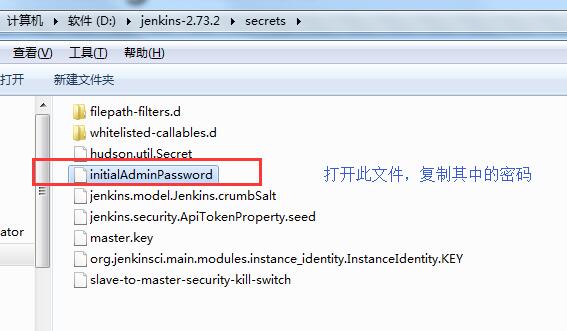
本教程是基于Windows环境讲解的。我本地使用的Jenkins的版本为2.73.2

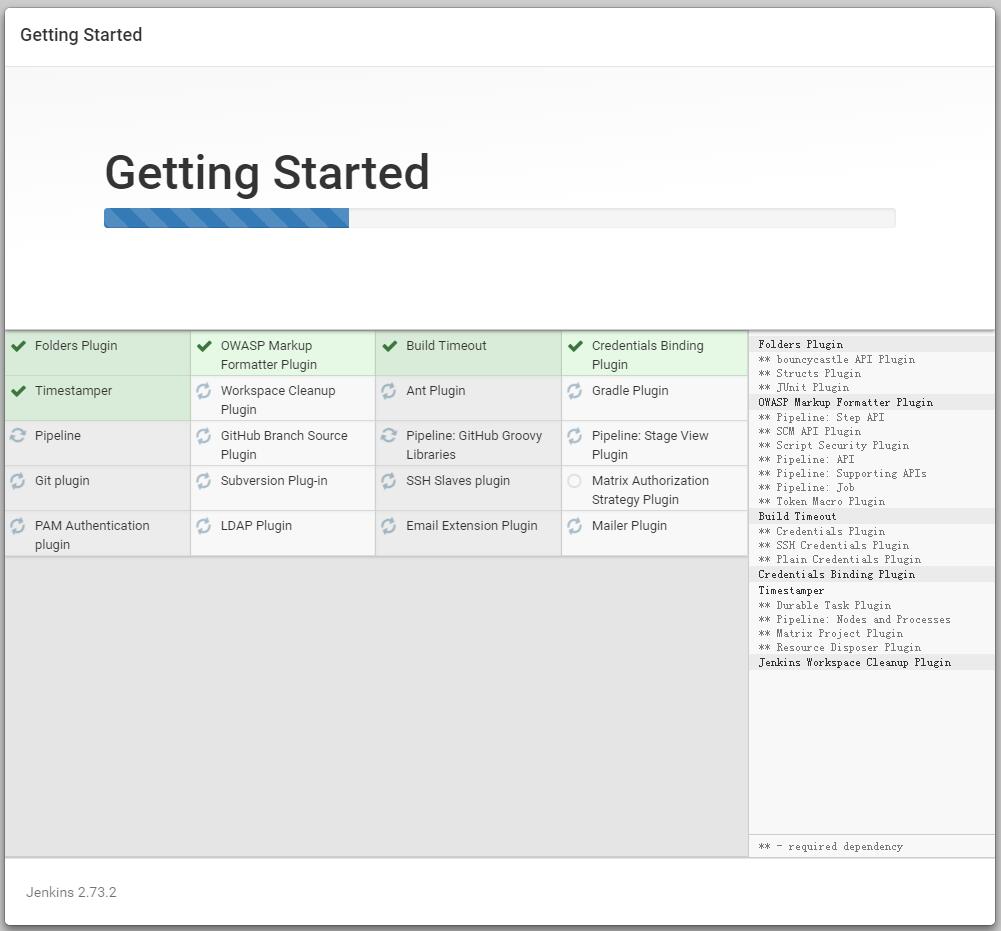
1、请先下载Jenkins软件， Jenkins最新官方下载地址 <http://mirrors.jenkins-ci.org/windows-stable/latest>

2、将下载的压缩包进行解压到本机指定目录，我这里是解压到了D:\jenkins-2.73.2

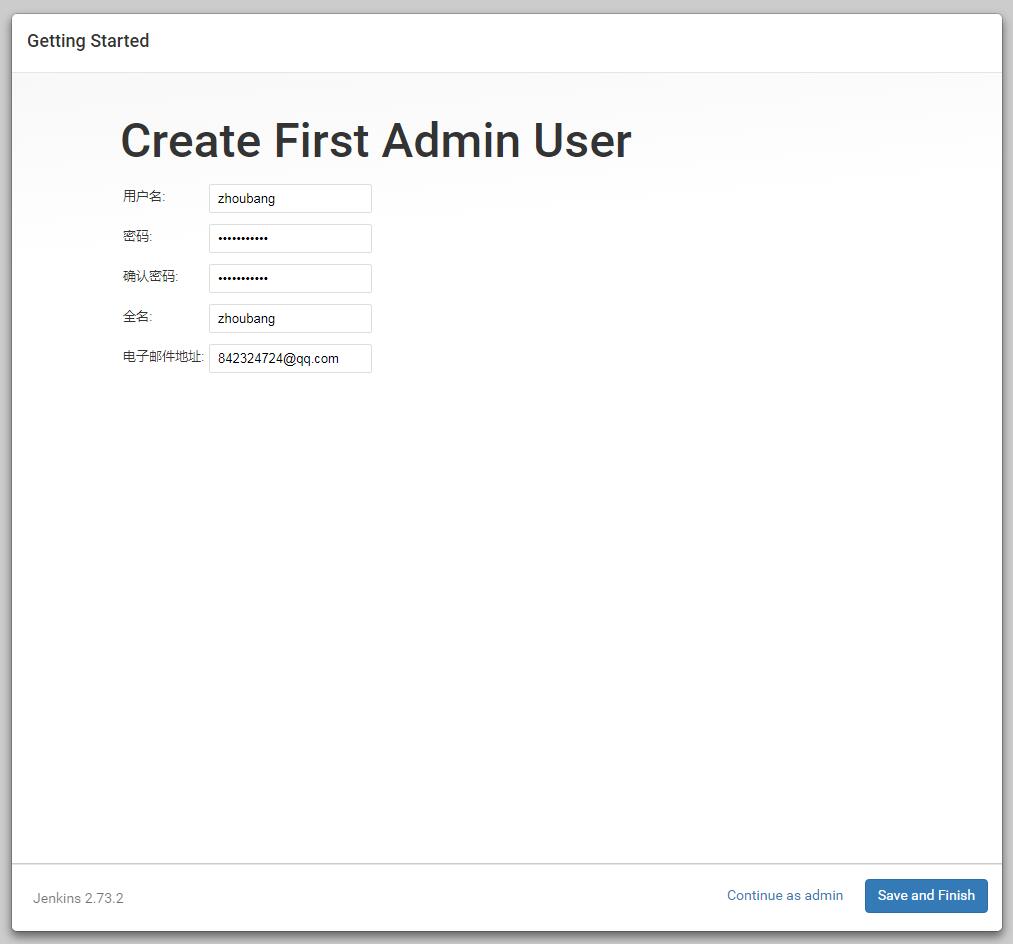
3、安装好之后，Jenkins会默认以“8080”为端口，并自动启动浏览器打开了Jenkins的安装配置界面。

4、一开始的时候，Jenkins会让你输入一个密码，这个密码是Jenkins自动为你生成的，密码所在的文件页面上会有具体文件的目录说明。之后应该是打开如下图的文件，复制其中的密码填入浏览器文本框中即可。

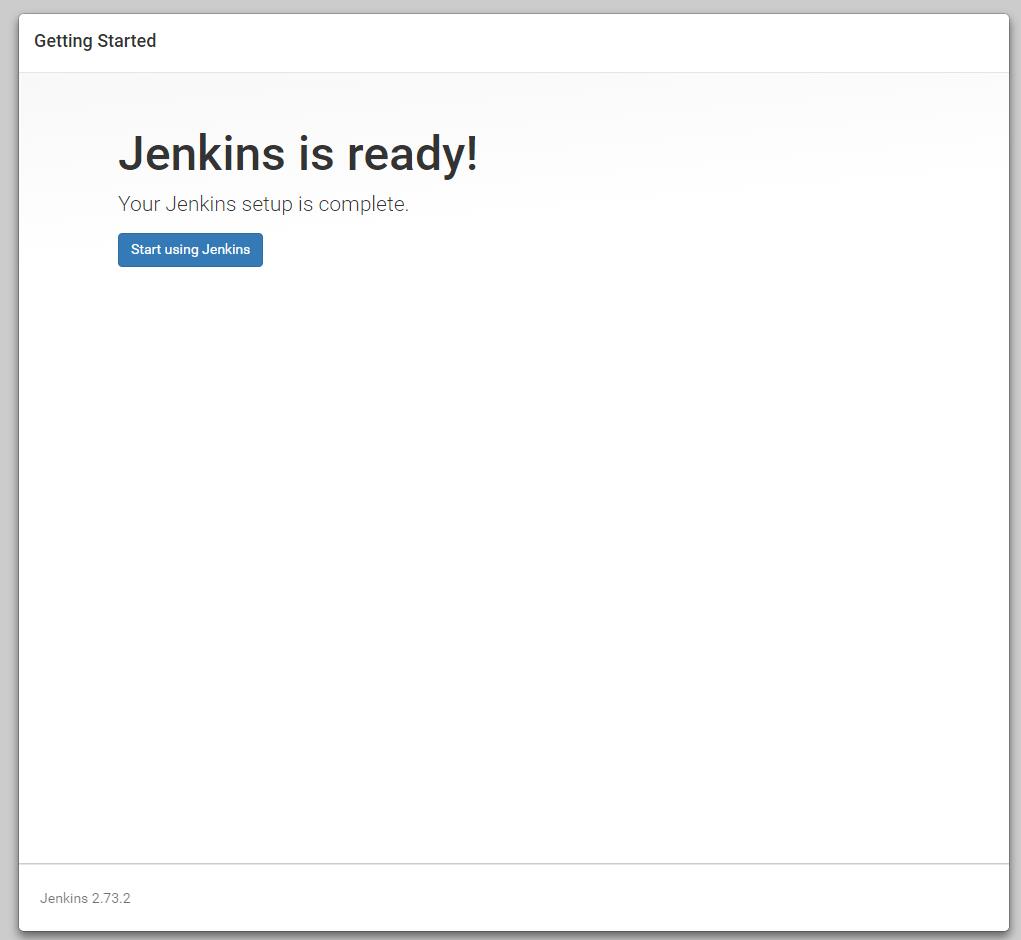


5、一开始Jenkins需要安装一些自己需要的插件，等待它全部安装成功后再继续往下操作，如下图：

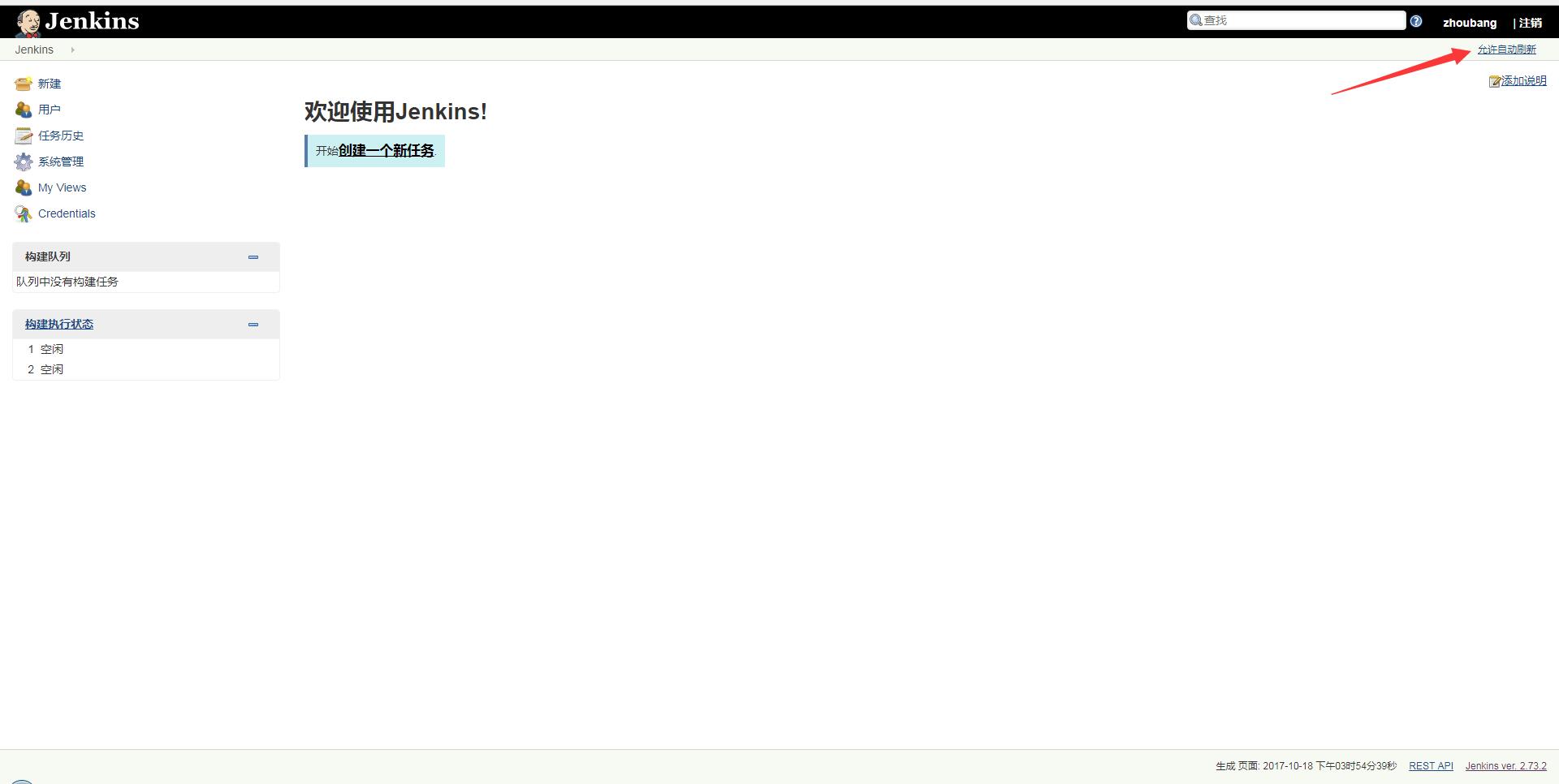
6、当所有插件安装成功之后，会让你输入一个账户和密码，这里你输入自己的即可，界面如图：



7、之后会出现一个成功页面，表明Jenkins已经准备就绪了，如图所示：



8、点击下方按钮，进入Jenkins主界面，如图：



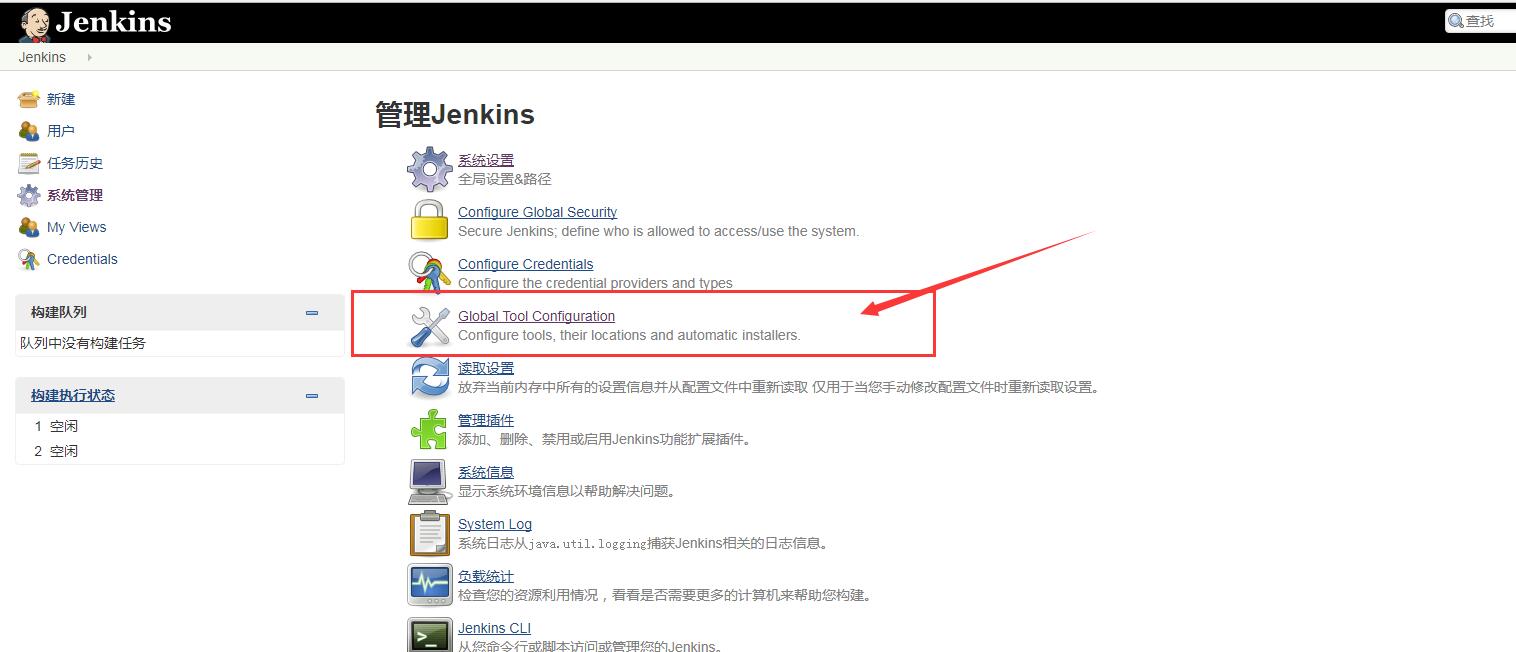
环境配置

1、**配置JDK环境。**

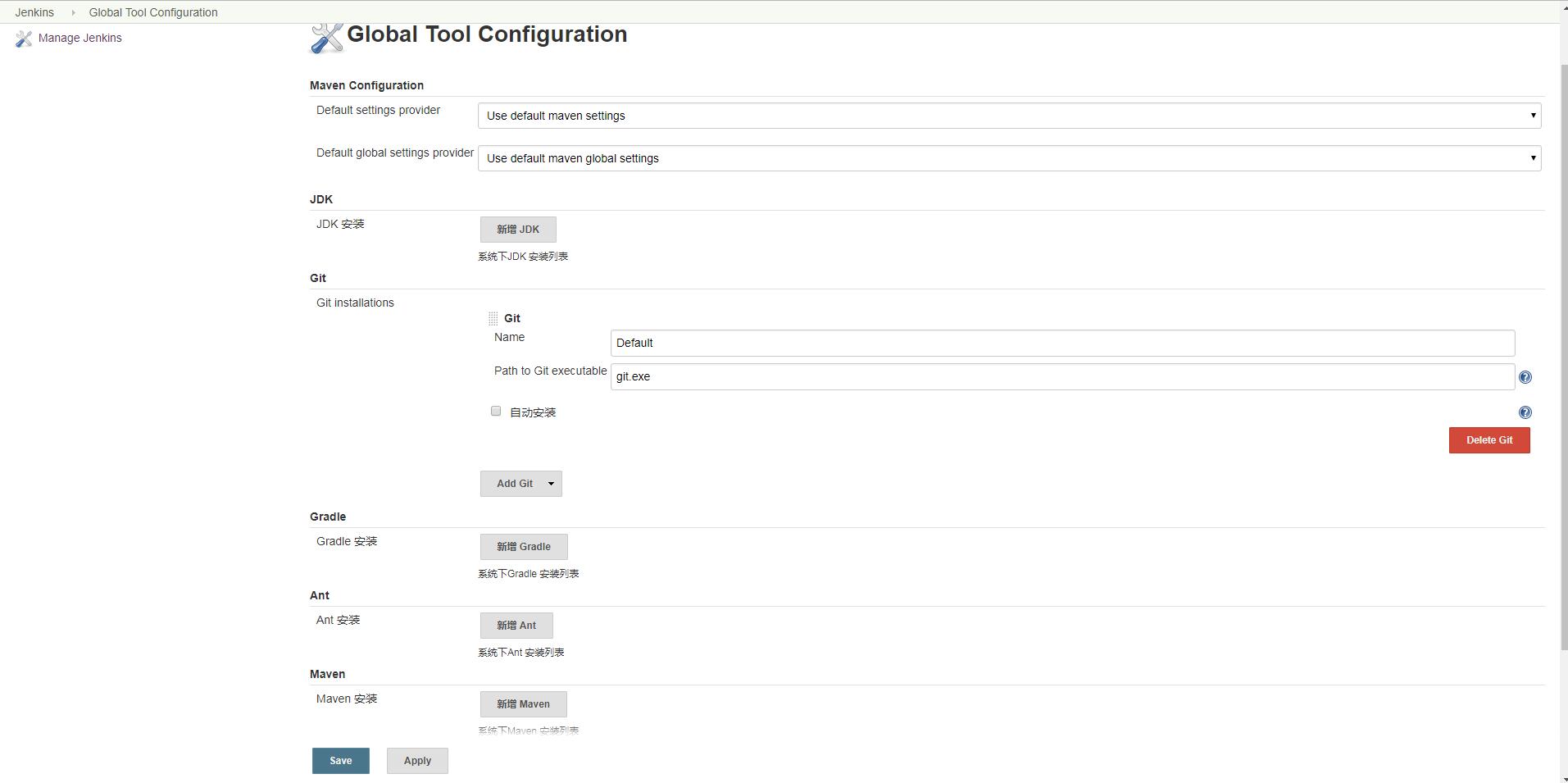
1.1首先点击左侧的【系统管理】，进入环境配置页面，如图：



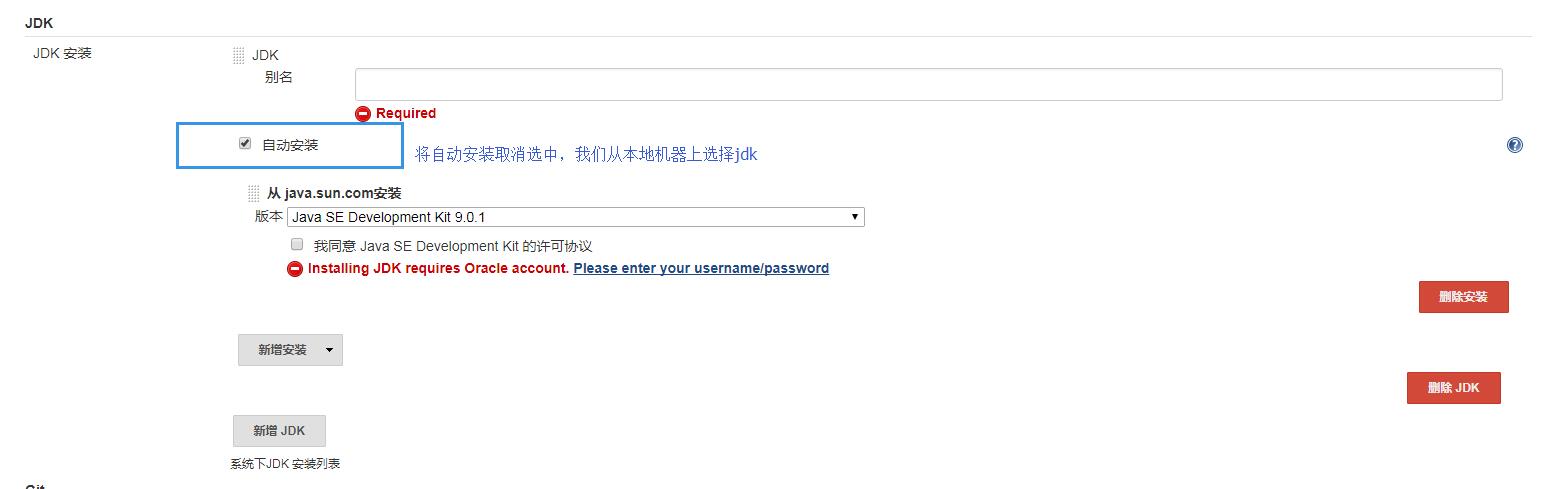
1.2进来之后，会看到如下图所示的管理页面，我们点击红色区域链接：



1.3打开页面之后，进入到了Jenkins真正的管理配置页面，如下图：



1.4 在当前页面找到JDK配置的区域，如下图：



JDK肯定是需要使用我们本地的，故先将【自动安装】取消选中，就会出现JDK配置的内容。

1.5取消选中之后如下图：



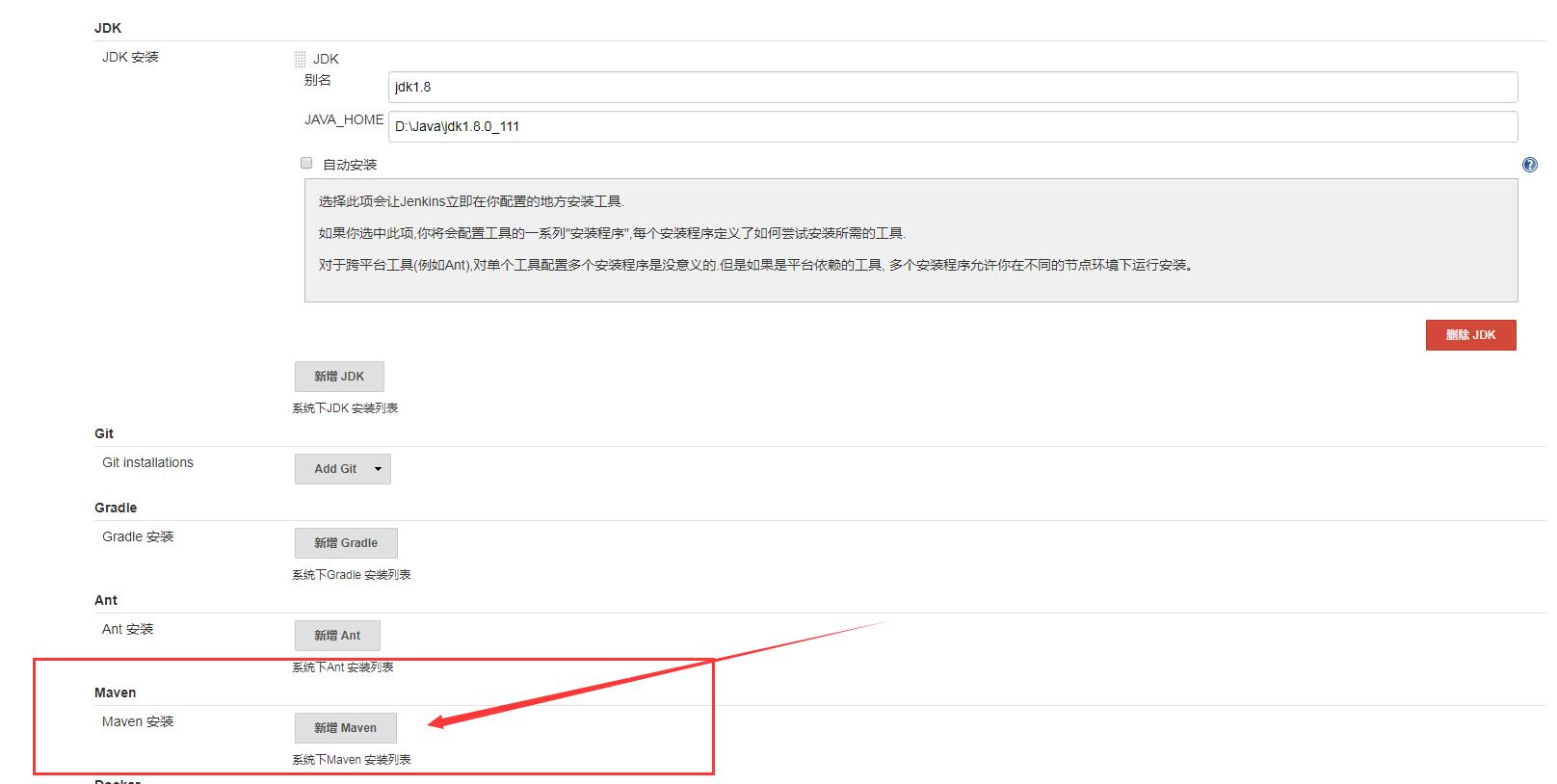
这里需要配置2个地方，红色区域内。我本机的JDK是1.8版本的，别名的话我就命名为“jdk1.8”了，这个随意命名，不作要求。

然后下面的JAVA\_HOME输入你本地的JDK安装目录即可，这个就不多说了。

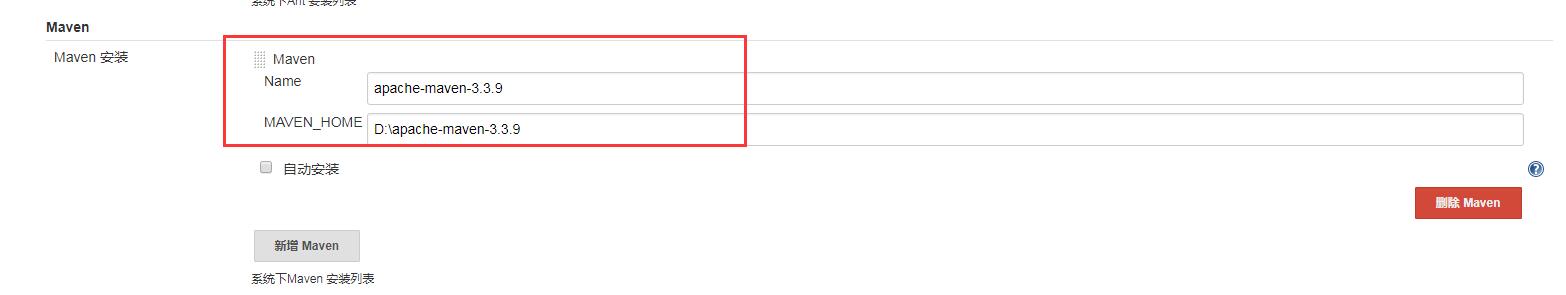
如果想实时保存的话，页面最下面会有2个按钮，点击相应的按钮进行保存操作即可。

2、**配置Maven环境**

由于目前企业的项目都是基于maven来开发的，而Jenkins默认没有安装此插件，故我们需要手动安装maven插件，其实很简单。

同样是在JDK环境配置所在的页面里面，找到Maven相关配置区域，点击【新增Maven】，然后同样取消【自动安装】选中的状态，就会出现Maven配置的内容了。

输入自己本地的Maven安装目录以及名称。Maven的配置操作与JDK类似，这里不多说了。如下图：



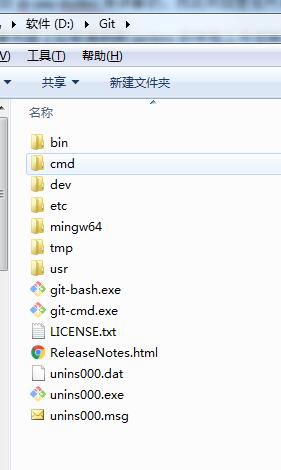
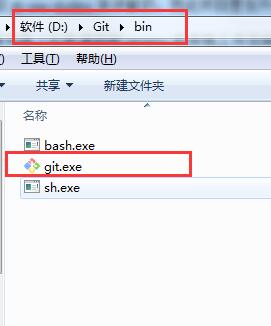
3、Git安装

由于此文档是基于项目zb-pay-dubbo来讲解的，而此项目是在开源中国的Git上托管的，故，若想让Jenkins自动化构建，就需要安装Git插件，就能从开源中国的Git服务器上拉取源码到Jenkins的本地工作空间了。

同样的，Git和JDK以及Maven处于同一页面，找到对应的Git区域，配置Git即可。



【注意】：这里需要提前在本地安装好Git软件，具体安装请百度搜索，这里不多作说明。我本地的Git之前已经安装过了，安装好之后如下图所示(我本地的Git版本为 2.11.0)：

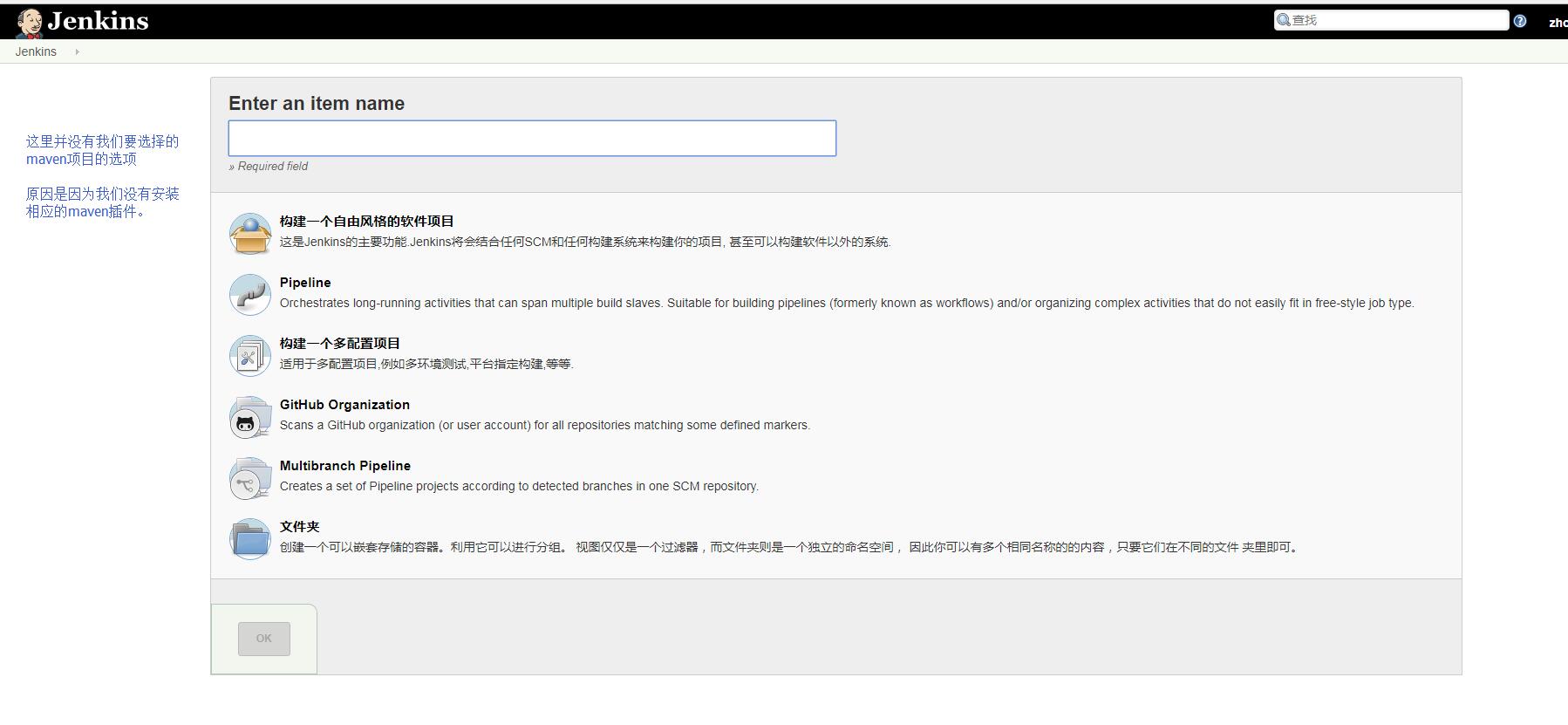
 

插件安装

基本的环境已经配置好了，还有什么没有配置好呢？往下看…

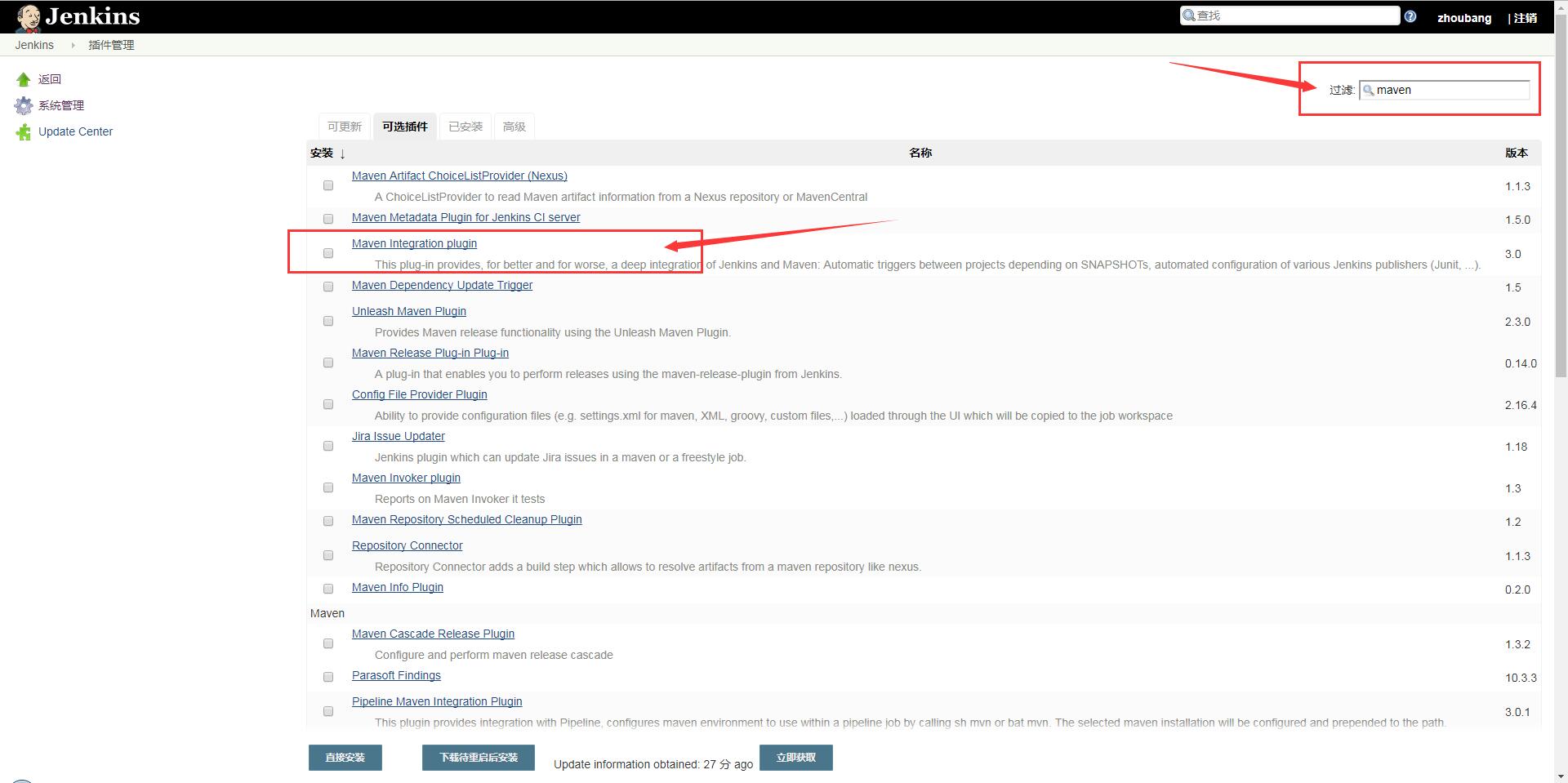
1、回到Jenkins的首页，点击创建一个新任务

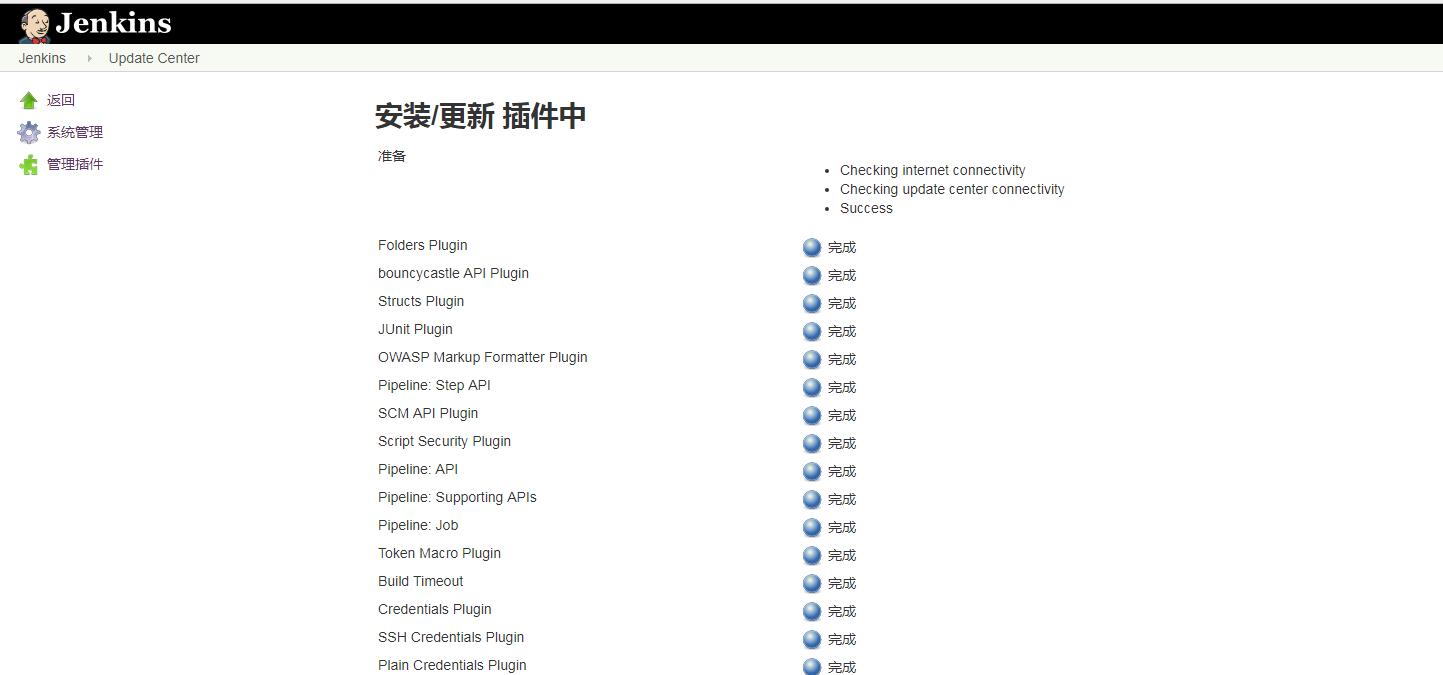


2、创建新任务的页面如下图所示，由于我们的项目是maven项目，最终的页面中应该会有一个【构建一个Maven项目】的选项，而当前页面没有这个的选项，是因为我们还没有安装maven的部署打包的插件，那就需要先安装好maven的插件，往下看…

3、回到Jenkins首页，点击左侧的【系统管理】，然后再点击【管理插件】



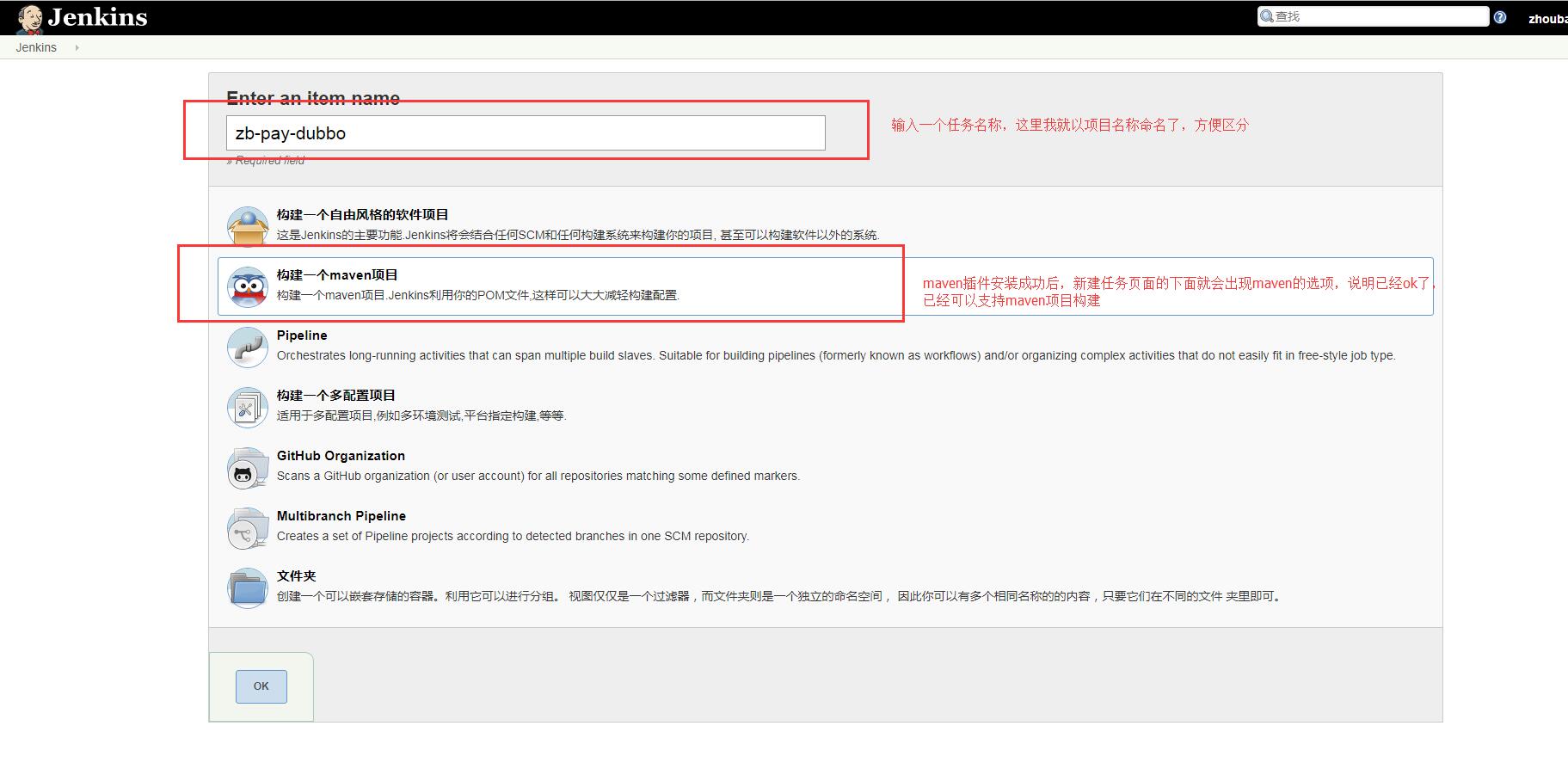
4、进入插件管理的页面后，首先将选项卡切换到【可选插件】，之后在右上角文本框中输入Maven Integration plugin (这个名称在不同版本的Jenkins中可能会不一样，那就输入一个maven进行搜索，然后找到对应的插件即可)，筛选出这个插件，然后选择此插件，点击下方的【直接安装】

5、maven插件安装的过程如下图

插件安装成功后，点击【返回首页】，至此，Jenkins所需的Maven打包构建的插件已经安装完毕。

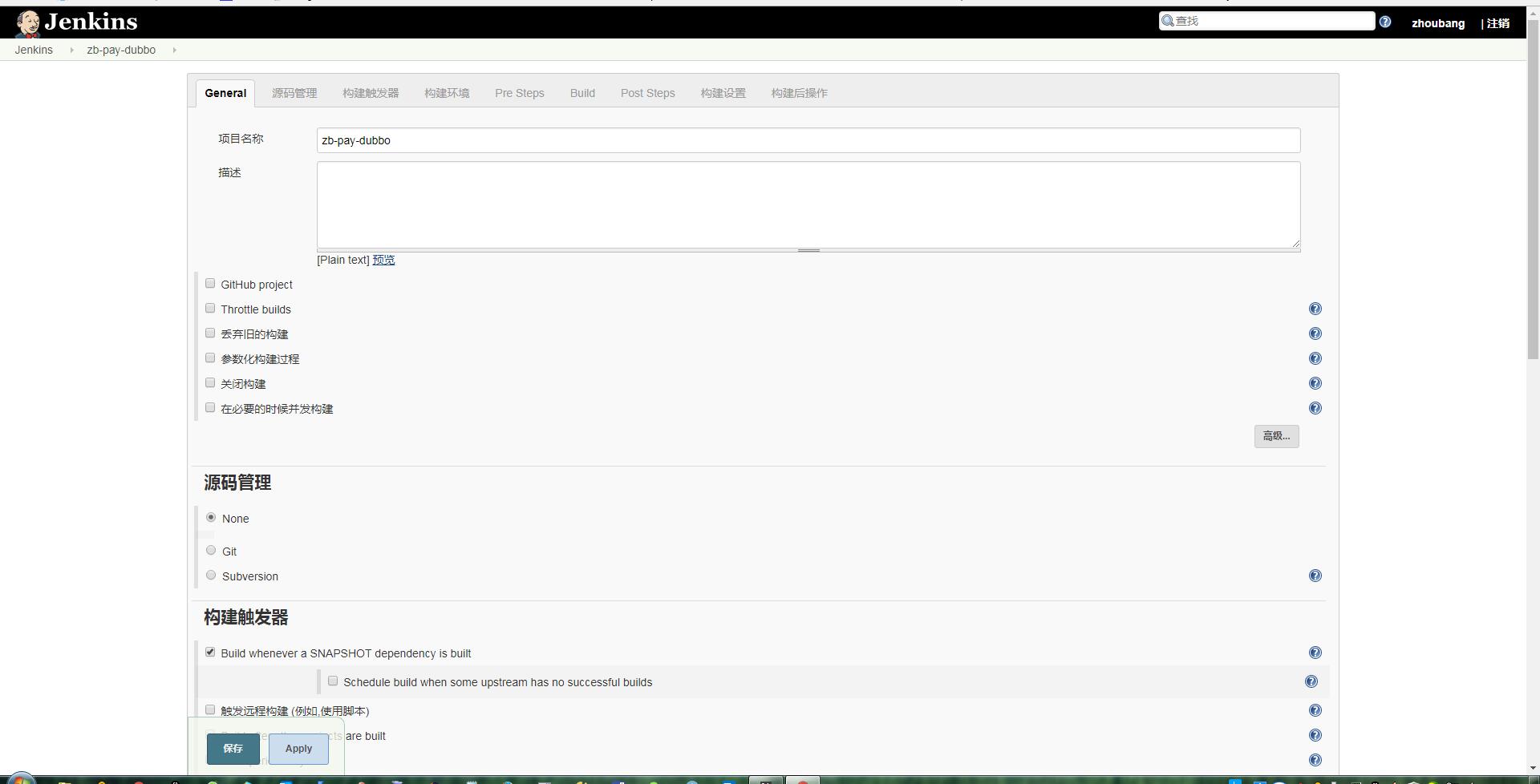
项目构建

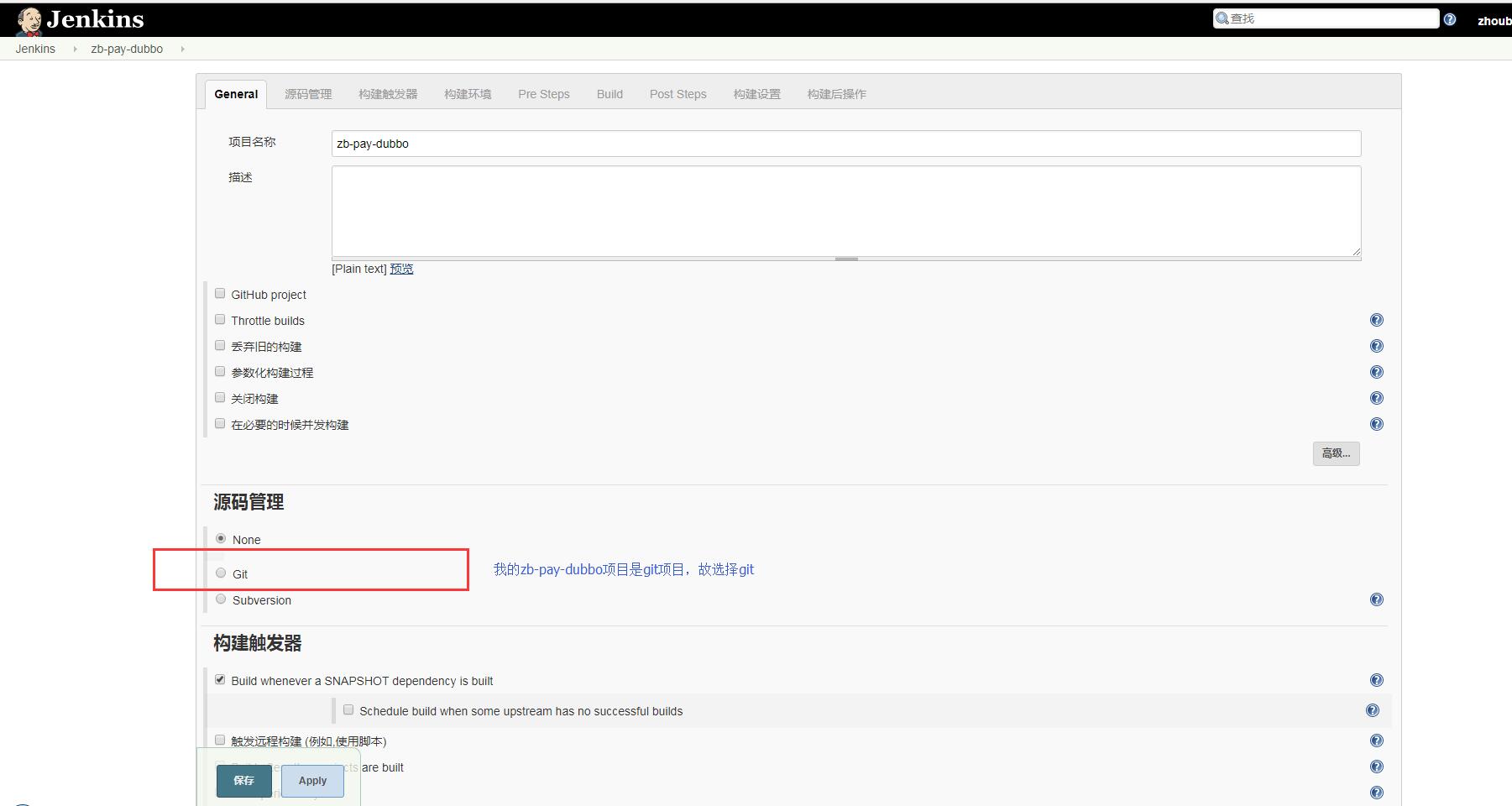
1、回到Jenkins的首页，点击【新建任务】进入创建任务页面，如下图：



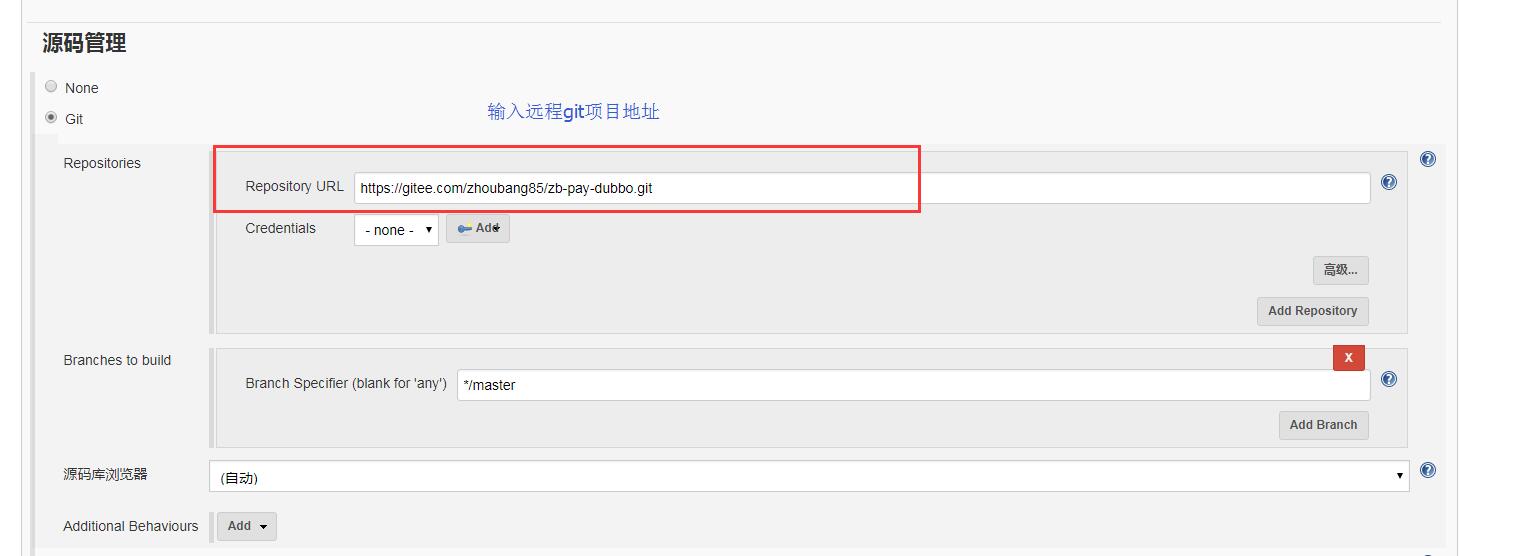
输入任务名称，这里我就用项目名称作为任务名称了，方便区分。建议以项目名称作为任务名称。

这时候你会发现，页面中多了一个【构建一个maven项目】的选项，这就是安装了上面的插件之后才会有的，Jenkins默认没有安装。

2、进入构建maven项目的页面后，如下图：

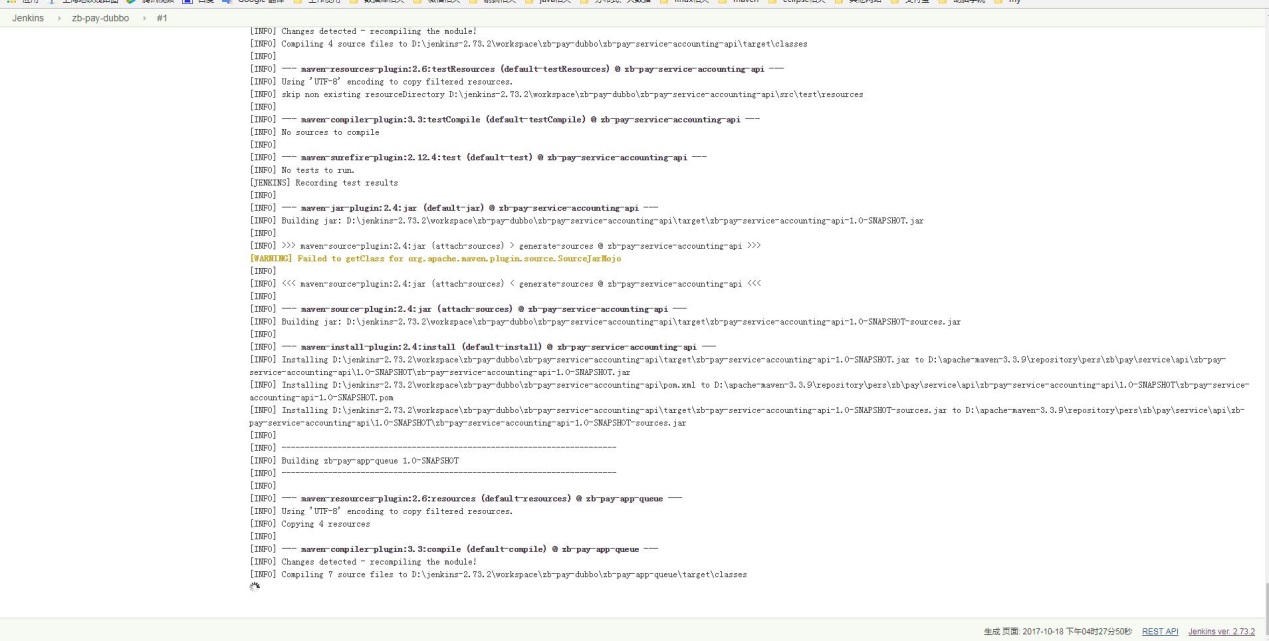
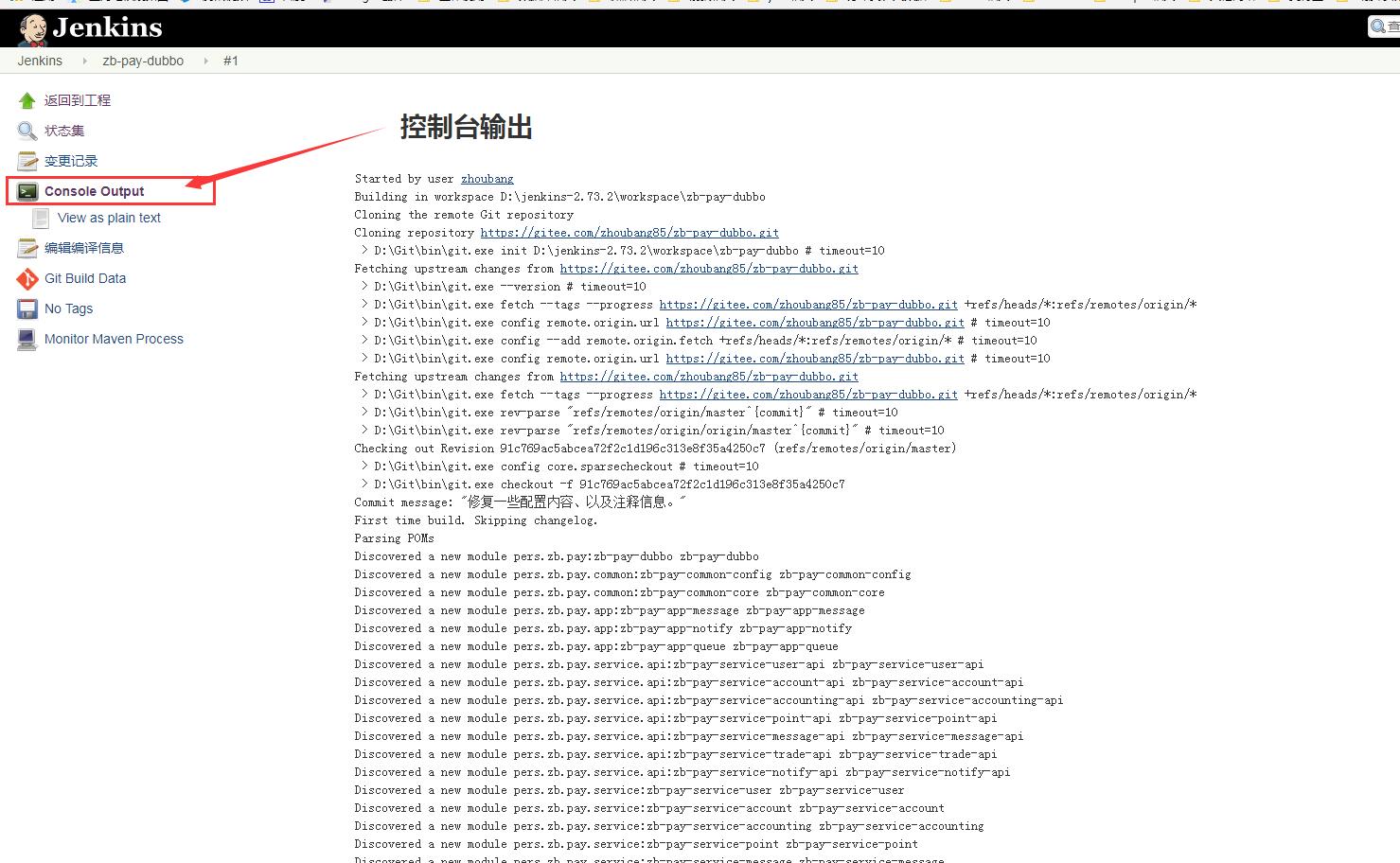
3、由于项目是git项目，故需要在【源码管理】区域选择“Git”

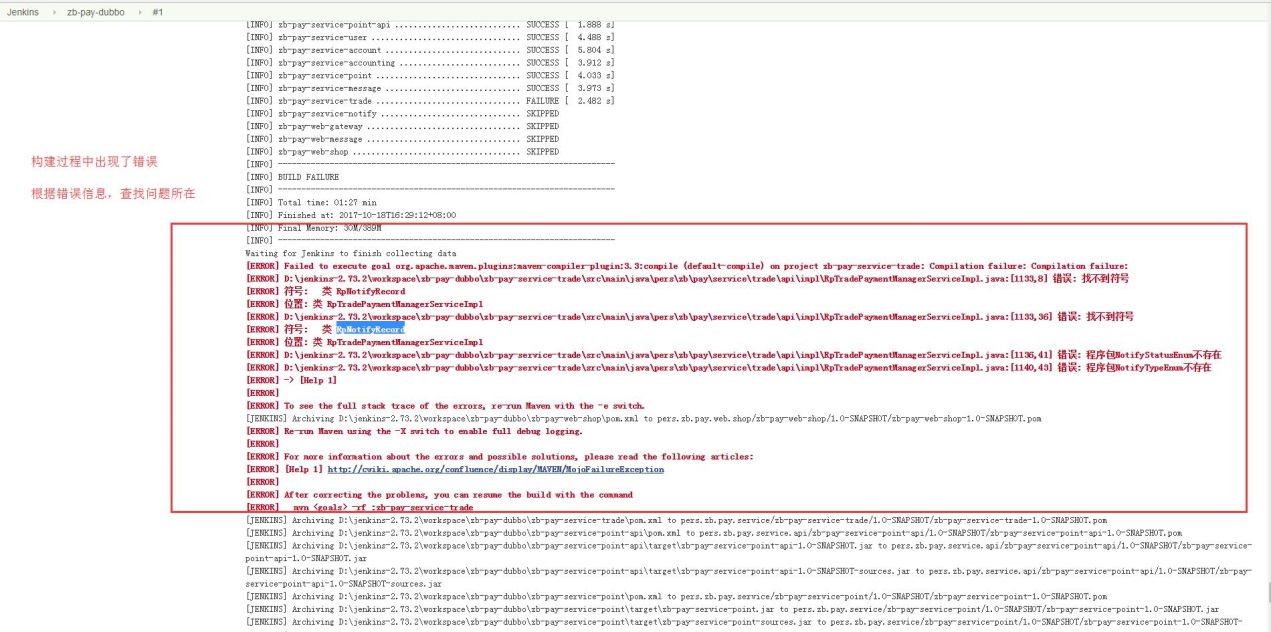
4、在红色区域内，输入项目的git地址，下面的全部默认，不过要说一点，下面的Branch Specitials 的地方，默认是master分支，如果你的项目默认不是master分支，就需要改成对应的分支名称。

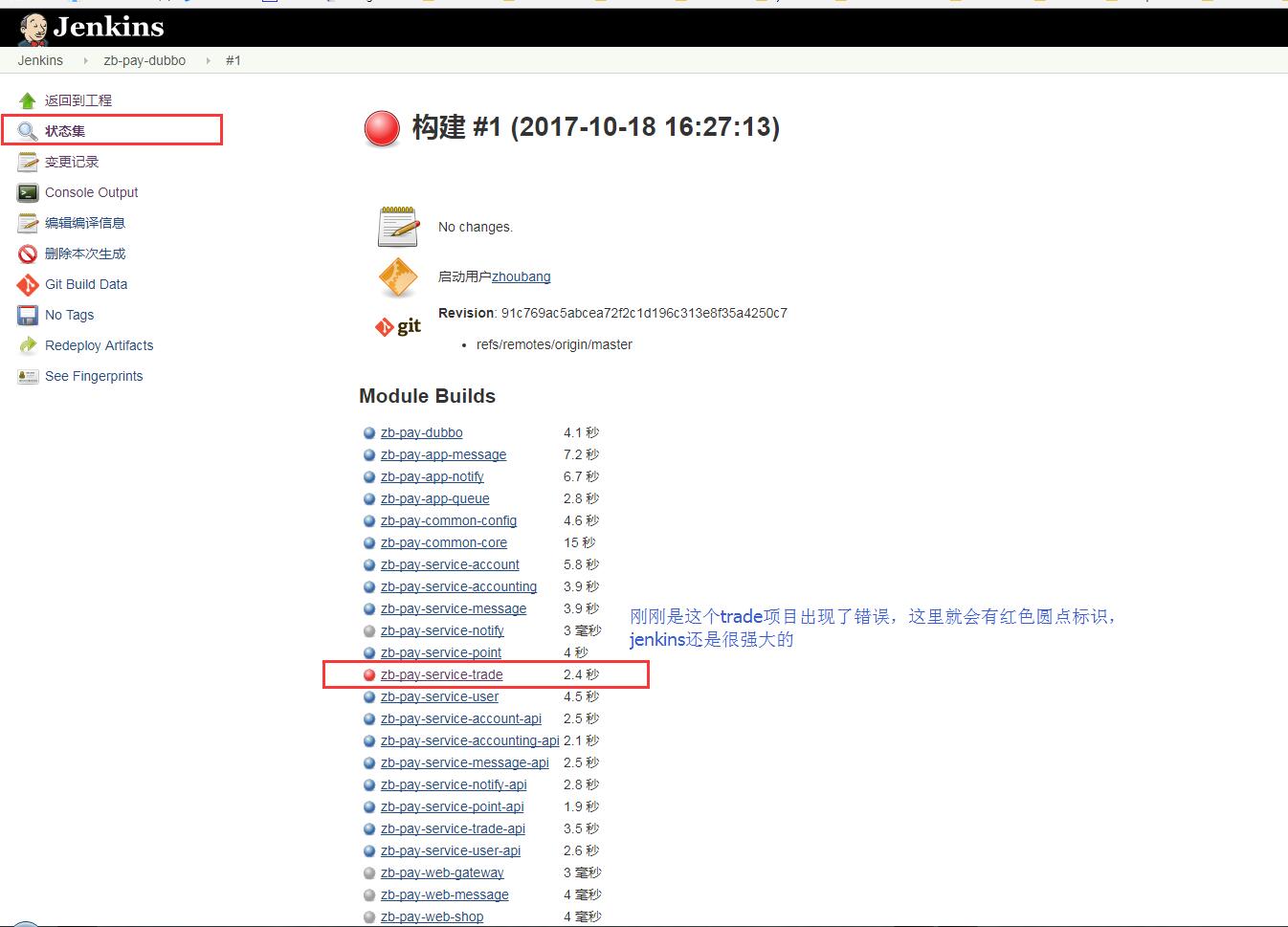


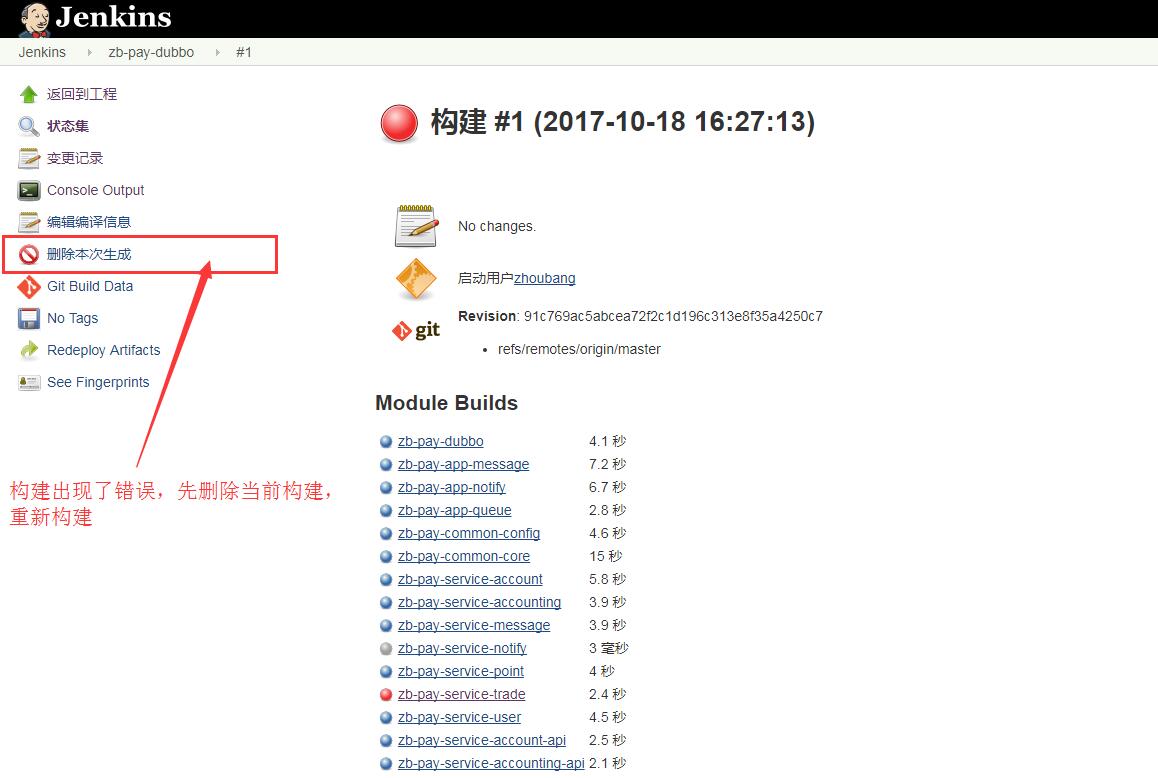
5、上面的git源码地址配置好之后，就可以进行项目构建了。回到首页，点击【立即构建】6、

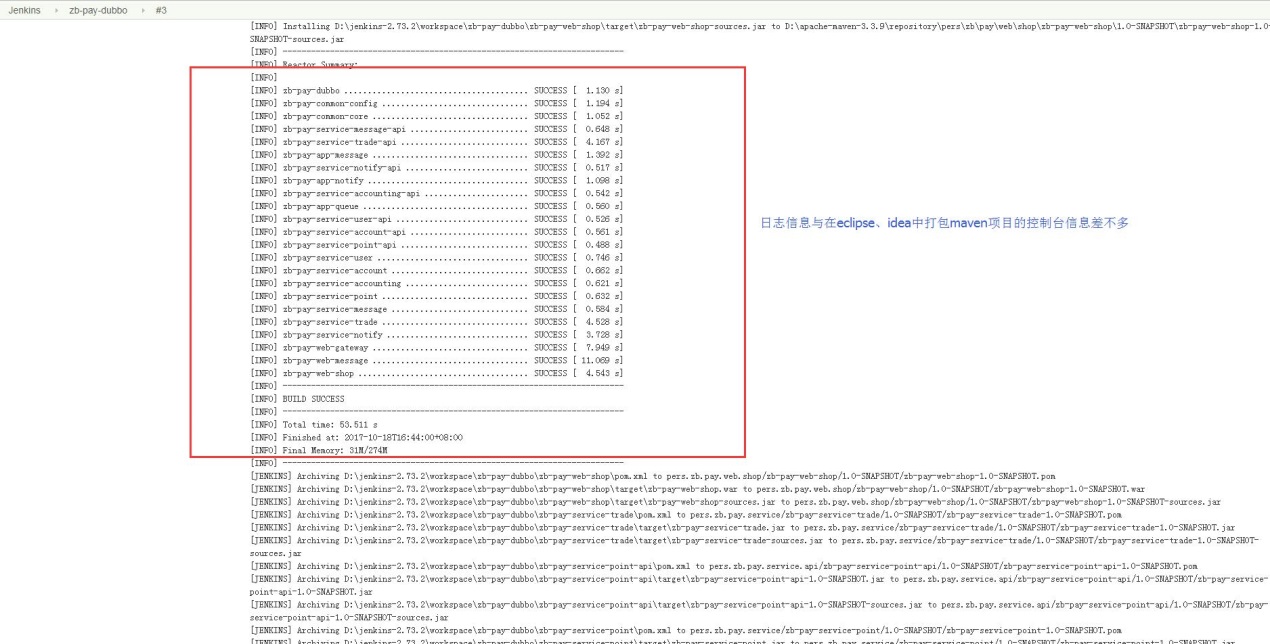
6、这时候在页面的左下角，会出现一个蓝白色的滚动条，说明项目已经开始构建了，我们可以点击这个滚动条进去看看构建的详细日志输出，类似控制台。

7、进入页面之后，点击左侧的【Console Output】查看控制台内容，你会看到项目构建的日志，其实就是maven打包的日志。

8、如果在构建过程中，控制台报错了，这个只能自己去查找问题所在了。比如下图的错误，明显是缺少class的引用，就很容易解决了。

9、如果控制台报错了的话，这时候点击左侧的【状态集】，你会看到红色区域就是构建失败的项目模块，很一目了然吧。

10、既然构建过程中出现了错误，要想重新构建的话，可以先删除刚刚构建所生成的一些资源。点击左侧的【删除本地生成】

11、错误解决之后，重新构建，如果没有什么问题的话，最终的控制台输出如下图

12、回到Jenkins首页，这时候会看到页面上多了一个构建任务的列表展示，就是刚刚我们构建的项目，可以看到一些详情。

13、【补充】jenkins是支持远程服务的发布、部署、启动的（比如将构建好的war发布到远程服务器上的Tomcat目录下）。由于资源有限，我就不演示远程的部署了，这个请自行百度寻找操作步骤吧。

如果只是本地构建打包，这里可以直接跳过。

如果想尝试远程的部署发布，就需要先安装一个插件，和安装maven打包部署插件的步骤一样，输入插件名称Deploy to container Plugin，安装即可。

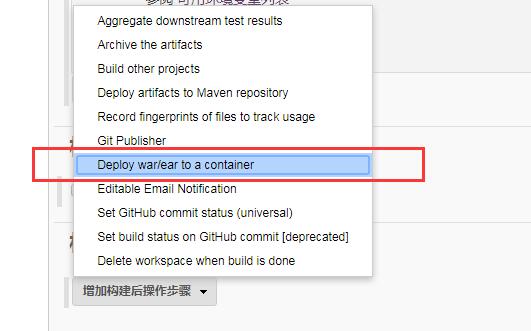


如何查看插件是否安装成功呢？

13.1 回到首页， 找到对应的项目，点击【配置】，如图所示：



13.2、进入项目配置页面，在页面最下方找到【构建后操作】，如下图所示：



如果出现红色区域的选项，说明插件已经安装成功了。

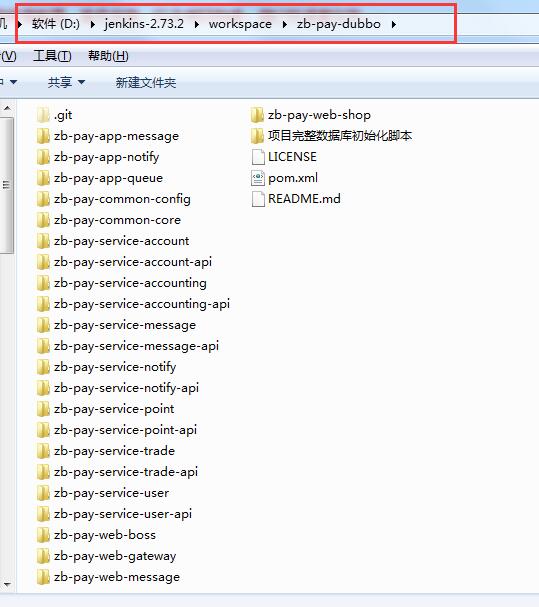
**至此，Jenkins的环境配置、插件安装、以及项目构建，都已经讲解完毕。**

项目构建 ----效果查看

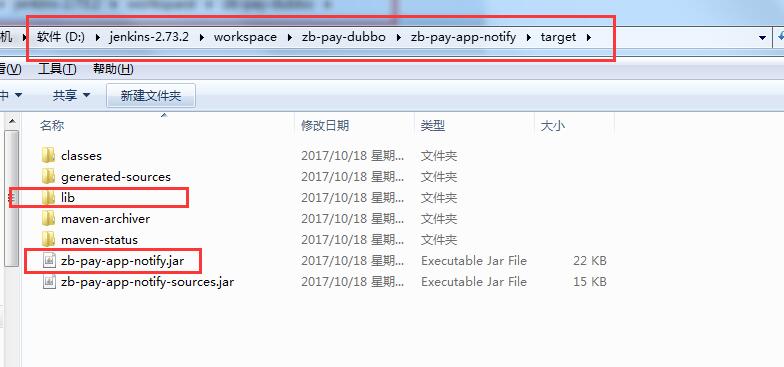
1. 往往看下效果，是最有直观的感受的。那行，我满足一下你们。

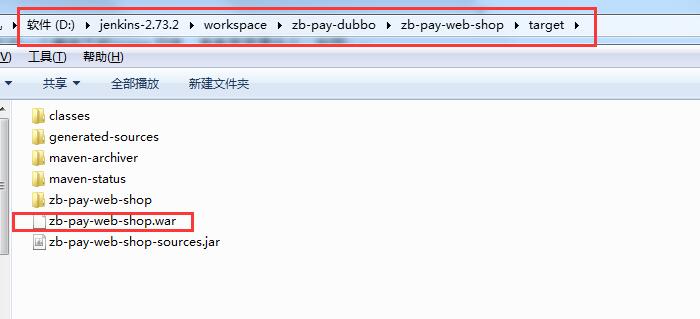
既然是通过Jenkins进行项目构建的，那么，项目的打包后的资源在哪里呢？肯定是在Jenkins的工作空间里面了，也就是workspace。

我本地Jenkins的安装目录是D:\jenkins-2.73.2，进入目录，会看到一个workspace的目录，我们进去看看，最终会看到，里面其实就是我们的zb-pay-dubbo项目的完整源码，当然，每个项目模块下面都有Jenkins为我们打包好的target目录，用过maven的都知道里面存放的是什么。



1. 我们选择2个模块（一个jar应用，一个war应用），打开模块下的target目录，看看里面是什么，如图：





其实就是模块的jar或者war以及所需要的依赖jar（存放在lib目录）。

1. 其实看到这里，一切都明了了。Jenkins帮助我们自动构建打包了项目，生成了jar或者war以及依赖的jar包.

疑惑与解答

有的朋友可能会感觉，这不就是maven打包部署maven项目吗？在eclipse或者IDEA工具上也一样可以打包部署啊，为何还要用Jenkins来弄呢，这不很麻烦吗？

首先，上面说的没错，追根究底，Jenkins就是用maven来打包部署的，那Jenkins的强大之处在于什么呢？那就是“自动化”。

如何体现自动化？

那我问一个问题，当zb-pay-dubbo项目通过jenkins打包成功后，要想启动这个项目，你需要做哪些工作？

**下面就介绍一下项目的服务启动，借此来体现Jenkins的自动化部署的强大。**

**这里只演示在windows环境下的项目启动。远程服务器的部署不演示。**

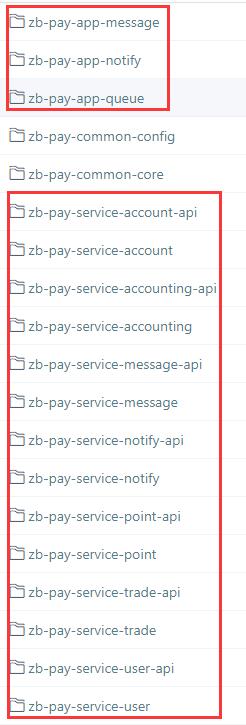
【说明】：由于项目的jar服务目前只有有限几个，那么手动去启动每一个jar应用服务，也是可以的。

然而，当服务模块很多的时候，如果每一个都去手动启动或者关闭服务，就会多很多的工作量的吧，如果由你来部署、启动服务，我想你一定会感觉到劳累的。

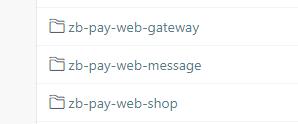
下面的内容只是说明Jenkins的自动化的好处和效果，看看能为我们解决和处理多少麻烦、重复性的工作。

首先看一下项目的功能模块划分，如下：

jar应用（红色区域，以jar的方式启动服务，启动之后就会将服务注册到dubbo，了解dubbo的朋友应该不难理解。）



war应用(需要部署到tomcat容器运行)



了解了项目的模块划分，就应该知道，要想完整部署启动zb-pay-dubbo项目的话，上面的jar服务和war服务都要启动、部署。

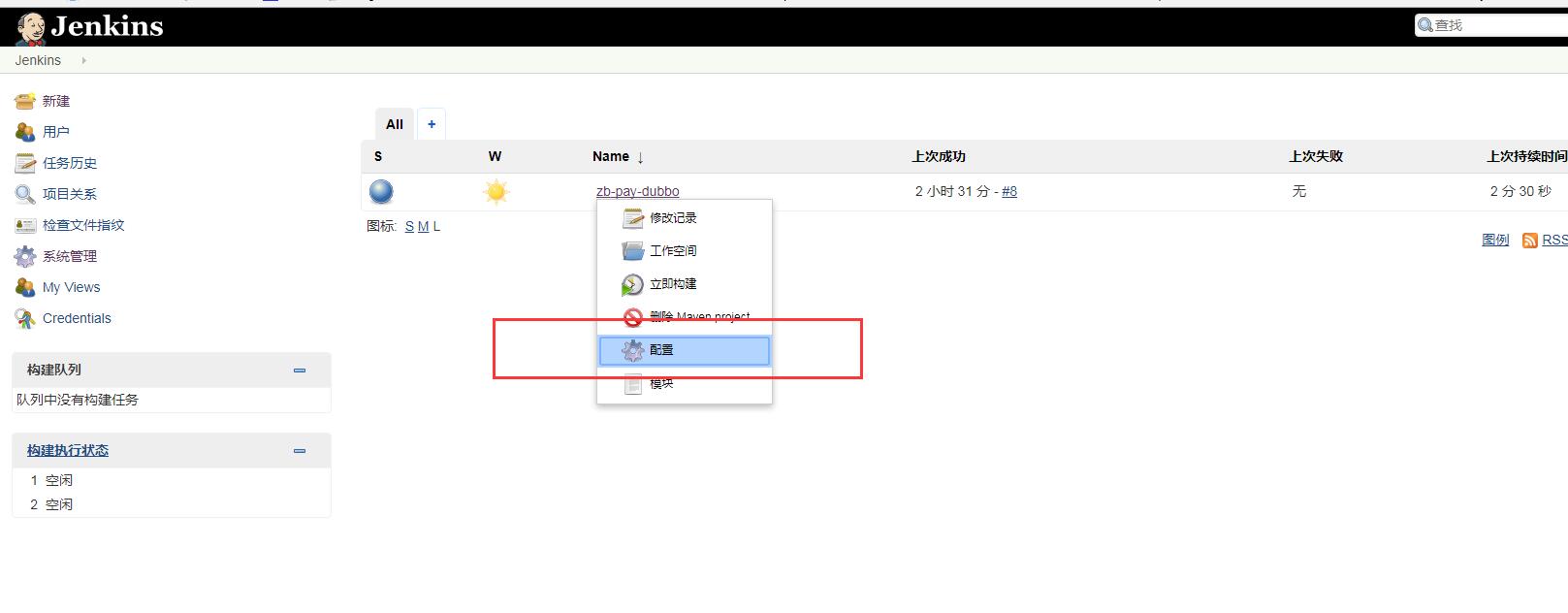
如果按照正常操作的步骤，肯定是启动每一个模块服务，光是启动jar服务就至少需要启动7项（这里还不包括jar以及依赖jar的抽离与整理）。

好的，这里我们想一下，有没有什么方式，可以在项目构建打包完毕之后，给我们自动生成启动服务的脚本呢？我们只需要执行一下脚本，就可以启动所有的service服务了。这就太方便了。

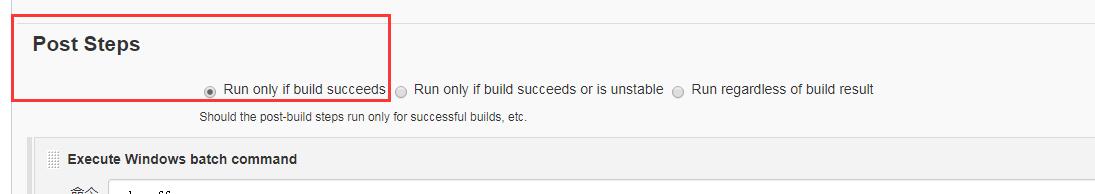
办法当然是有的！

首先，Jenkins支持项目构建成功后，自动执行一些命令（windows下的CMD命令、Linux下的Shell命令等），在哪里配置命令呢？

首先回到Jenkins首页，打开对应的项目的配置，如下图：



找到Post Steps 区域，点击【增加构建后操作步骤】，选择Execute Windows batch command，由于我们是演示windows环境的，故需要选择这个。然后就可以在这个地方配置项目构建成功后需要执行的脚本，如下图：



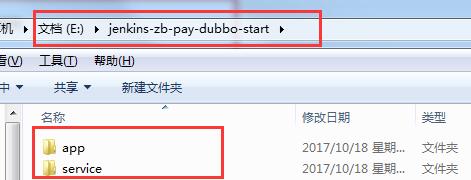
当脚本命令编写好之后，将脚本复制到Execute Windows batch command 的内容区域里，然后保存。这样，当每次构建完项目之后，会自动执行这里的脚本，实现自动化操作。

【说明】：既然想通过脚本的方式，来自动整理service服务相关的jar和依赖包，就需要自己手动编写脚本了，具体怎么写，这个…..只能说根据具体项目或者业务情况，来自己编写了。

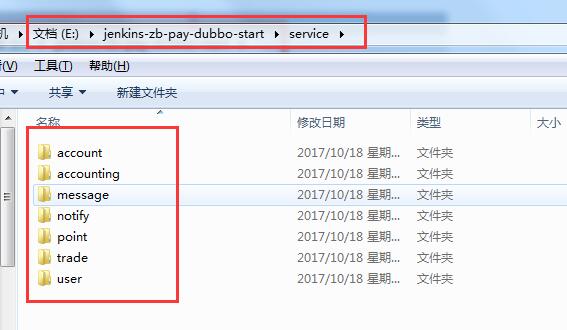
项目构建成功后---执行脚本

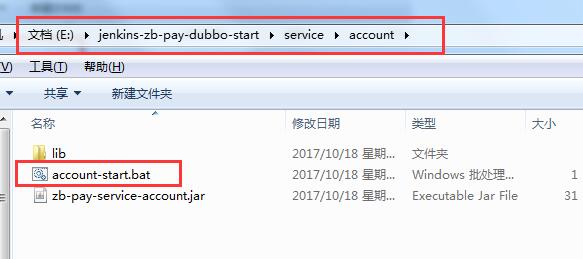
比如我这个项目的话，最终想达到的效果如下：

1、将每一个app、service的模块整理出来，放在独立的目录下(我本地存放的目录是 E:\jenkins-zb-pay-dubbo-start )，再在E:\jenkins-zb-pay-dubbo-start目录下分别建立app、service目录，用于区分服务种类。本地效果截图如下：



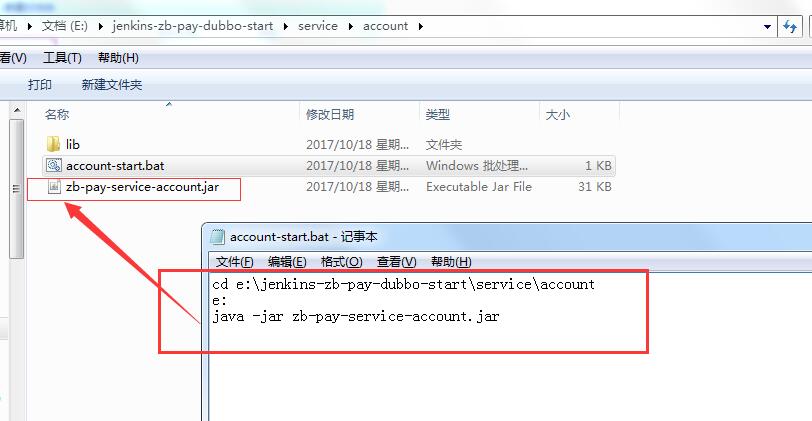
2、然后再在每一个模块下生成对应的启动服务jar的.bat文件。拿service目录来说，如下图：





3、再在每一个模块下生成一个.bat脚本文件，用于启动相对于的jar文件。

比如account-start.bat的脚本文件内容如下：



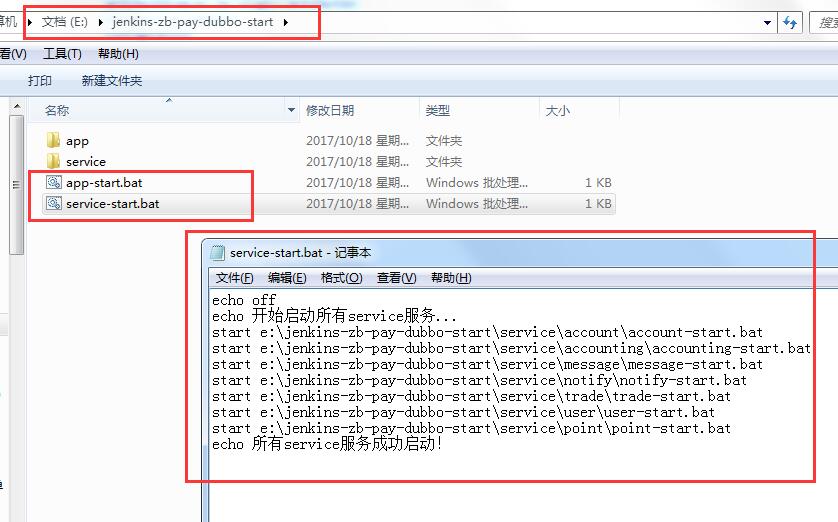
你会发现，其实很简单，就是通过java命令来执行jar文件的。

如果不明白为什么jar可以当作应用启动，建议百度了解一下，或者查看此模块对应的pom.xml中的打包配置，如下图：



你会发现，其实通过java命令去启动jar，其实就是默认执行了红色区域的类中的main方法，之后将当前应用的服务注册到了dubbo上。

4、每一个服务模块下都生成了启动服务的.bat文件，那就在E:\jenkins-zb-pay-dubbo-start目录下再生成2个bat脚本命令app-start.bat、service-start.bat，随便看一下service-start.bat的内容，如下图：



你会发现，其实就是将每个模块下的bat都启动而已。没有什么难理解的。这里的所有.bat文件的命令肯定是要能在windows系统上执行才行。

5、到这里为止，通过编写命令的方式实现上面这样的服务整理、启动，就变得很方便了，只要执行一下app-start.bat、service-start.bat这2个脚本即可实现所有的服务启动了。

其实，如果是自己研究学习的话，命令就简单写一下就好，不要求太完美，能实现上面的想法需求就行了，实际企业工作中，肯定是要用心编写的了。

这里我分享一下我编写的bat脚本吧，内容很多都是复制修改的，没有进行过多的优化。效果是能实现上面的要求的。双击下面的txt文件



如果对bat里面的一些命令不明白的，建议百度查询。

上面的脚本里面的大概的步骤是：

1、定义存放service、app应用的目录，统一管理启动服务（比如上面说的E:\jenkins-zb-pay-dubbo-start）

2、指定zb-pay-dubbo项目所在的jenkins的工作空间目录，方便下面命令引用，类似变量。

3、将jenkins工作空间中构建打包好的项目资源(每个service、app模块下的target目录下的jar以及lib目录)，复制到对应的目录下（比如将D:\jenkins-2.73.2\workspace\zb-pay-dubbo\zb-pay-service-account\target目录下的zb-pay-service-account.jar和lib目录复制到E:\jenkins-zb-pay-dubbo-start\service\account目录下）

4、为每一个模块都创建一个启动服务的bat脚本

5、在service、app应用所在的目录（E:\jenkins-zb-pay-dubbo-start）下创建service、app应用的统一启动的脚本bat。

6、启动统一的脚本，实现所有服务的启动。

最终就实现了：当构建完毕项目后，自动执行脚本，实现service、app应用的自动化整理、启动，方便管理。

【经验补充】：

1、由于Jenkins不支持.bat文件里面启动其他.bat，不会弹出命令窗口，最终每个服务的进程都会关闭。所以上面也就只演示在E:\jenkins-zb-pay-dubbo-start下生成统一的启动服务的脚本，需要手动去执行才可以。

2、如果是构建部署在linux环境下，就可以执行shell命令。服务不会关闭。这是linux的shell脚本与window的bat脚本运行的区别。

总结

花了几个小时的时间来写这篇教程，在结束之际，心里还是很高兴的，毕竟这是自己的一份劳动成果，也为需要的朋友带来了很好的教程。

我个人一向的做法就是：免费开源分享。

只有经历过那种无助、头疼的时刻，才深刻体会到，有一份优秀的教程文档是多么的好，能给需要的朋友带来很多的帮助，节省很多精力和时间。这也是人生的一大风景点吧。

该教程结合zb-pay-dubbo项目进行了讲解，其实也是对该项目的打包部署以及模块划分进行了讲解，一箭双雕吧~

如果上面有什么问题或者不明白的，可以联系我、咨询我，我有时间的话会给你们解答。

QQ：842324724

Email：[842324724@qq.com](mailto:842324724@qq.com)

[支付-微信\_支付宝\_银联]QQ技术交流群 ：470414533

也欢迎各位经常光临：

我的网站：<http://www.2b2b92b.com>

技术论坛：<http://www.2b2b92b.cn>

我的开源中国： <https://gitee.com/zhoubang85>

zb-pay-dubbo项目地址： <https://gitee.com/zhoubang85/zb-pay-dubbo>