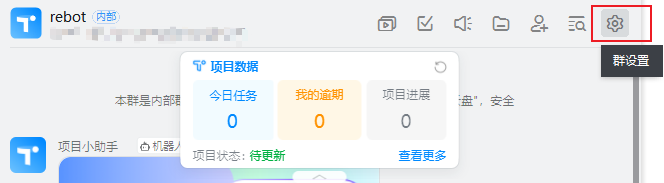
# jenkins集成钉钉通知消息

### 1 钉钉添加机器人

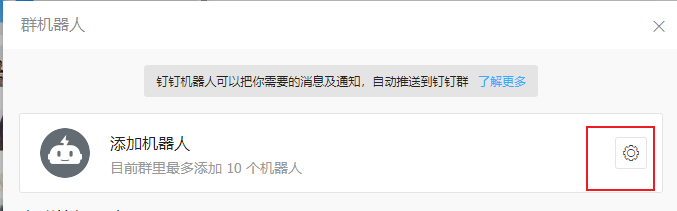
在群设置中找到“智能群助手”选项，点击进入对应的界面



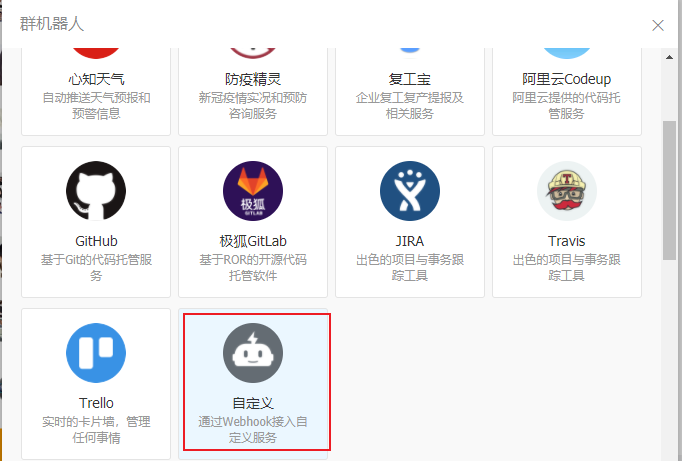


进入智能群助手页面后，选择添加机器人类型，如下图所示



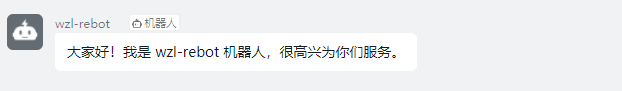


这里选择“自定义”机器人，根据引导页输入机器人名称（如“自动化测试”）后，完成对该群机器人的添加







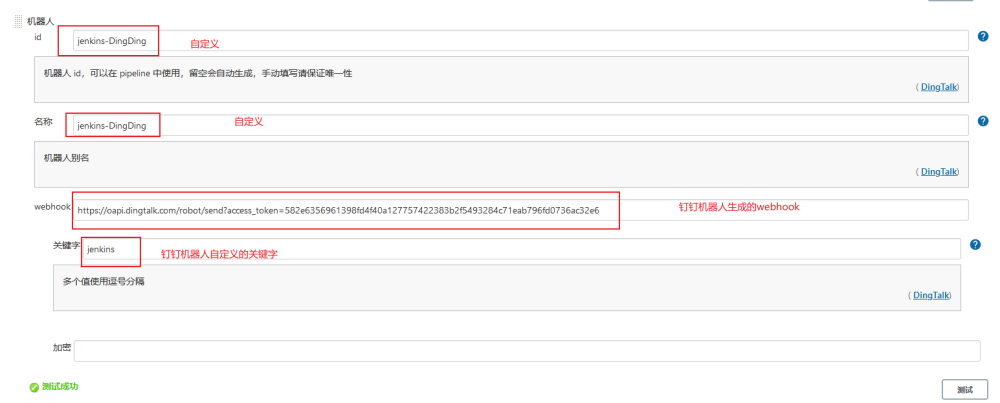


### 2 Jenkins配置

#### 2.1 安装插件

进入“系统管理->管理插件->可选插件”，在可选插件页面右上角“过滤”框中输入dingding

#### 2.2 系统配置



#### 2.3 Jenkins项目关联钉钉通知

##### 方式一 流水线方式

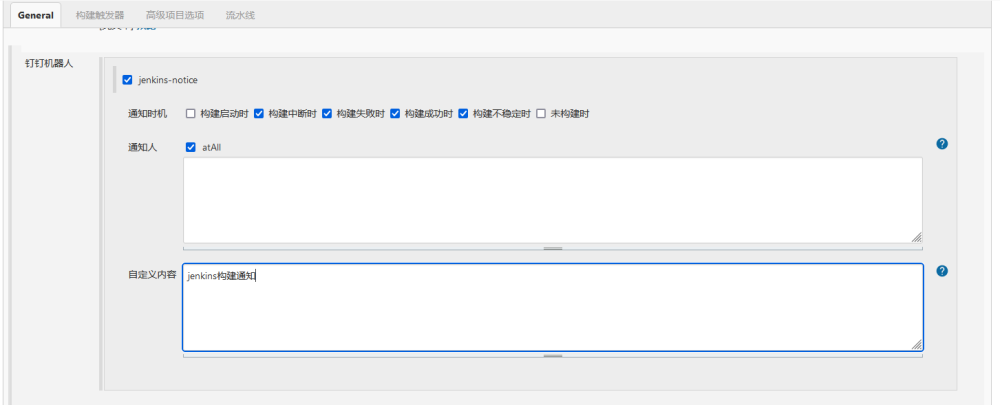
//env  
String RobotID = "jenkins-DingDing"  
  
//func  
def getChangeString() {  
 def changeString = ""  
 def MAX\_MSG\_LEN = 20  
 def changeLogSets = currentBuild.changeSets  
 for (int i = 0; i < changeLogSets.size(); i++) {  
 def entries = changeLogSets[i].items  
 for (int j = 0; j < entries.length; j++) {  
 def entry = entries[j]  
 truncatedMsg = entry.msg.take(MAX\_MSG\_LEN)  
 commitTime = new Date(entry.timestamp).format("yyyy-MM-dd HH:mm:ss")  
 changeString += " - ${truncatedMsg} [${entry.author} ${commitTime}]\n"  
 }  
 }  
 if (!changeString) {  
 changeString = " - No new changes"  
 }  
 //return (changeString)  
 return changeString  
}  
  
//func  
def DingdingReq(RobotID, Status) {  
 wrap([$class: 'BuildUser']) {  
 //def changeString = GetChangeString()  
 dingtalk (  
 robot: RobotID,  
 type: 'MARKDOWN',  
 at: [], //@具体某个人，可以多个。@手机号  
 atAll: true, //@所有人  
 title: '你有新的消息，请注意查收',  
 text: [  
 "### 构建信息",  
 "> - 应用名称：\*\*${env.JOB\_NAME}\*\*",  
 "> - 构建结果：<font color=#008000>${Status}</font>",  
 //"> - 当前版本：\*\*${env.BUILD\_NUMBER}\*\*", //与构建编号雷同  
 "> - 构建发起：\*\*${env.BUILD\_USER}\*\*",  
 "> - 构建编号：<font color=#008000>${BUILD\_ID}</font>",  
 //"> - 构建状态：<font color=#008000>${currentBuild.result} ✅</font>", //与构建结果雷同  
 "> - 持续时间：\*\*${currentBuild.durationString}\*\*",  
 "> - 项目地址：[点击查看详情](${BUILD\_URL})",  
 "> - 构建日志：[点击查看详情](${env.BUILD\_URL}console)",  
 "### 更新记录:",  
 "${changeString}"  
 ]  
 )  
 }  
}  
  
pipeline {  
 agent any  
 stages {  
 stage('link'){  
 steps {  
 echo '测试 MARKDOWN 消息...'  
 }  
 post {  
 success {  
 script {  
 DingdingReq(RobotID, "构建成功 ✅")  
 }  
 }  
   
 failure {  
 script {  
 DingdingReq(RobotID, "构建失败 ❌")  
 }   
 }  
   
 aborted{  
 script {  
 DingdingReq(RobotID, "构建暂停或中断 ❌")  
 }   
 }  
  
 unstable{  
 script {  
 DingdingReq(RobotID, "不稳定异常 ❌")  
 }   
 }  
 }  
 }  
 }  
}

[使用Jenkins扩展共享库进行钉钉消息推送](https://www.bilibili.com/read/cv11849765/)

**效果如下**



##### 方式二 勾选方式（对流水线任务不生效，待进一步验证）



### 3 参考文档

<https://cloud.tencent.com/developer/article/2028121>

[使用Jenkins扩展共享库进行钉钉消息推送](https://www.bilibili.com/read/cv11849765/) 可参考（主要参考）

[Jenkins Pipeline 配置钉钉消息通知](https://blog.csdn.net/IT_ZRS/article/details/125674756)

[钉钉机器人插件在 pipeline 中使用](https://jenkinsci.github.io/dingtalk-plugin/guide/pipeline.html)  钉钉开放文档（官方）

[基于Jenkins共享库实践之自定义通知器](https://blog.csdn.net/weixin_38320674/article/details/111189377) 可参考