# PCM演示系统部署说明文档

作者：叶东

时间：2017年3月28日

1、PCM项目简介

common-core：为基础层，包含工具类，错误编码等信息。

业务处理层分为:

业务层全部基于common-core

pcm-serivce：内部系统业务处理

pcm-serivce-core：核心业务处理，主要处理库存及价格的变动。

pcm-serivce-outer：外部系统业务处理

Controller层分为：

pcm-core：用于库存价格变动，由PCM内部使用；基于pcm-service-core

pcm-inner：用于其他内部系统进行数据修改 如：拍照系统修改图片；基于pcm-service

pcm-inner-sdc：用于其他内部系统进行信息获取 如：拍照系统，搜索系统，OMS；基于pcm-service

pcm-import：用于商品、库存、供应商等信息导入；基于pcm-service

pcm-outer-sdc：用于外部系统查询 如：导入终端，供应商平台；基于pcm-service-outer

pcm-syn：用于PCM数据下发；基于pcm-service

pcm-admin：用于OPS信息修改；基于pcm-service

pcm-admin-sdc：用于OPS信息查询。基于pcm-service

整体构造

common-core

pcm-service-outer

pcm-service

pcm-service-core

pcm-outer-sdc

pcm-admin-sdc

pcm-syn

pcm-inner-sdc

pcm-inner

pcm-admin

pcm-import

pcm-core

2、PCM部署顺序

pcm部署时需要 搜索、OMS、EDI、MQ、监控平台、富基、门店ERP、SAPERP部署完毕或提供相应的信息下发地址。

**pcm-syn**->**pcm-core**->**pcm-outer-sdc、pcm-inner、pcm-inner-sdc、pcm-import、pcm-admin** 因项目之间相互调用，需同时部署->**pcm-admin-sdc**

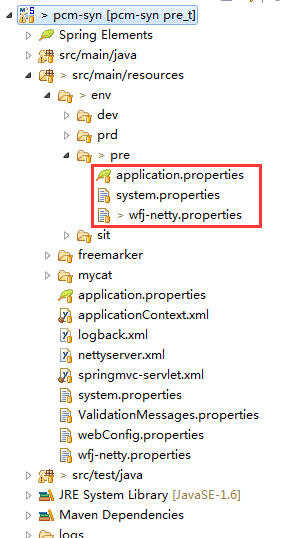
3、配置文件修改详情

配置文件修改，以pre配置为例

对上述8个项目的**pre\_t** 分支上的代码的配置文件演示环境配置，以pre配置为例：

3.1、pcm-syn:

需修改下图所示的三个文件



pcm-syn修改详情：

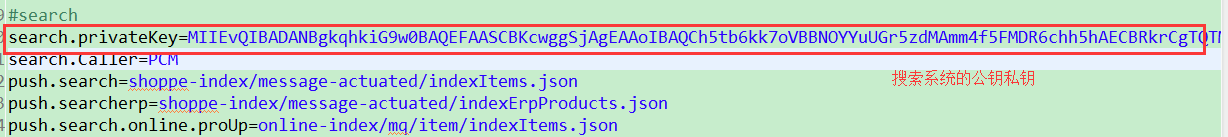
application.properties 包含10个地方需要修改：数据库相关配置、REDIS配置、OMS地址、

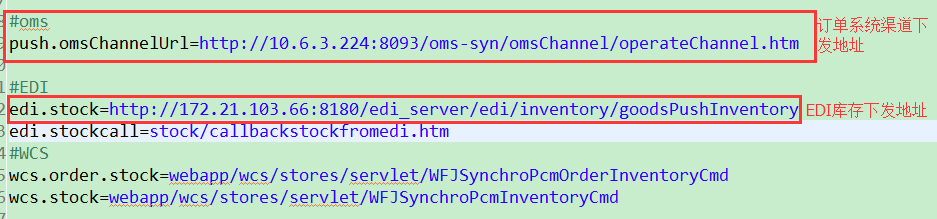
PCM-SYN地址（即当前本项目地址）、MQ地址、EDI地址、门店系统地址、搜索系统地址及公钥私钥、支付系统地址、和导入终端地址。



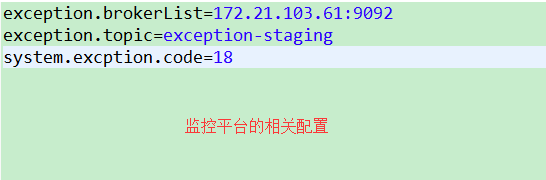








system.properties文件 修改监控平台的相关配置即可

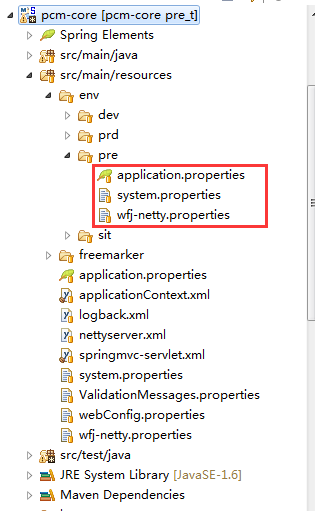


wfj-netty.properties文件有两个地方需要修改



3.2、pcm- core

需修改下图所示的三个文件

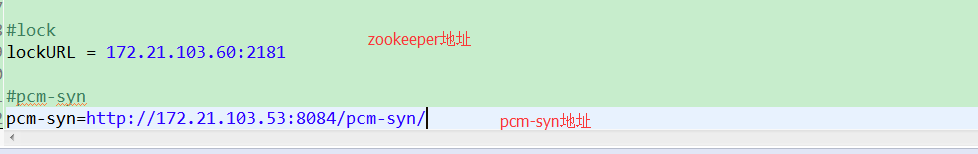


pcm-core修改详情：

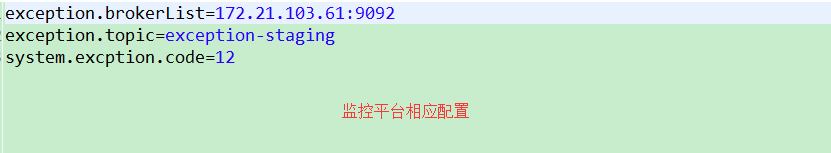
application.properties文件 包含4个地方需要修改：数据库相关配置、REDIS配置、zookeeper地址、pcm-syn地址。



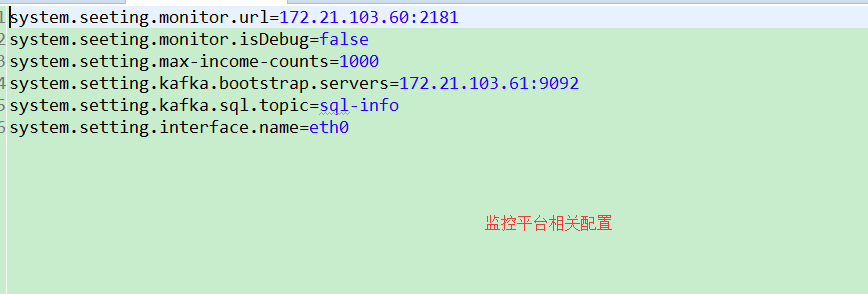




system.properties文件 修改监控平台的相关配置即可

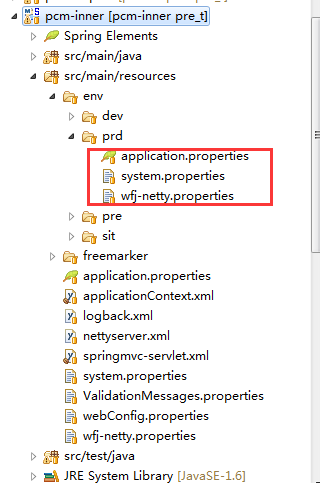


wfj-netty.properties文件修改监控平台的相关配置即可



3.3、pcm-inner

需修改下图所示的三个文件

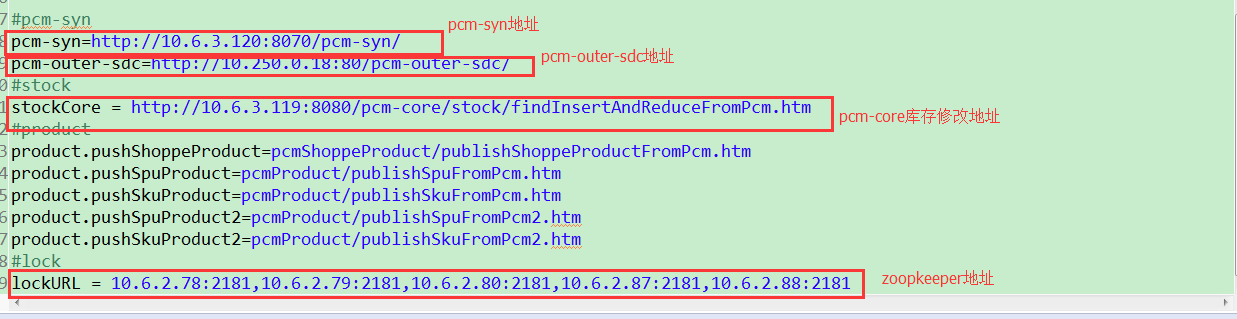


pcm-inner修改详情：

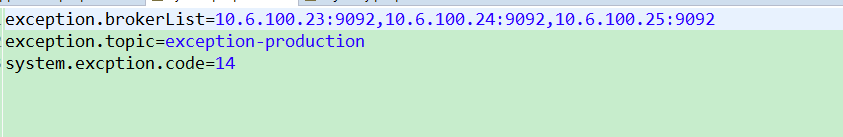
application.properties文件 包含6个地方需要修改：数据库相关配置、REDIS配置、zookeeper地址、pcm-syn地址、pcm-core地址、pcm-outer-sdc地址。



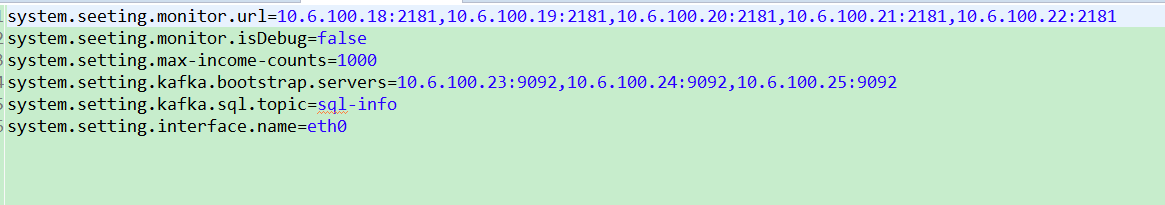




system.properties文件 修改监控平台的相关配置即可

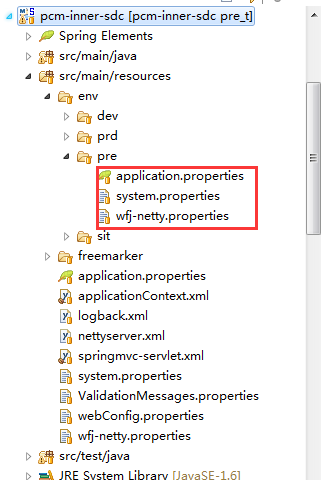


wfj-netty.properties文件修改监控平台的相关配置即可



3.4、pcm-inner-sdc

需修改下图所示的三个文件

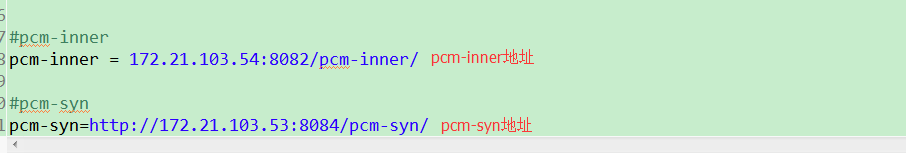


pcm-inner-sdc修改详情：

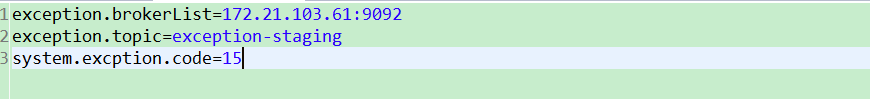
application.properties文件 包含4个地方需要修改：数据库相关配置、REDIS配置、pcm-inner地址、pcm-syn地址。



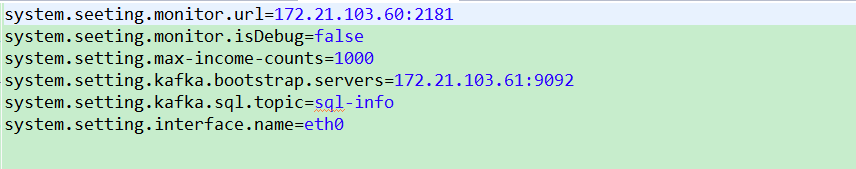




system.properties文件 修改监控平台的相关配置即可

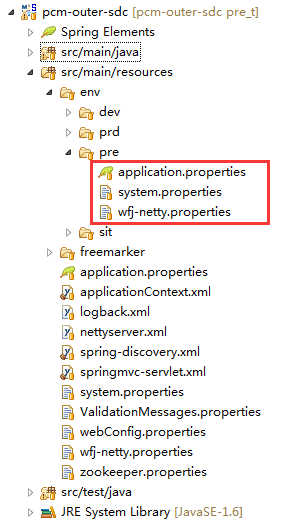


wfj-netty.properties文件修改监控平台的相关配置即可



3.5、pcm-outer-sdc

需修改下图所示的三个文件

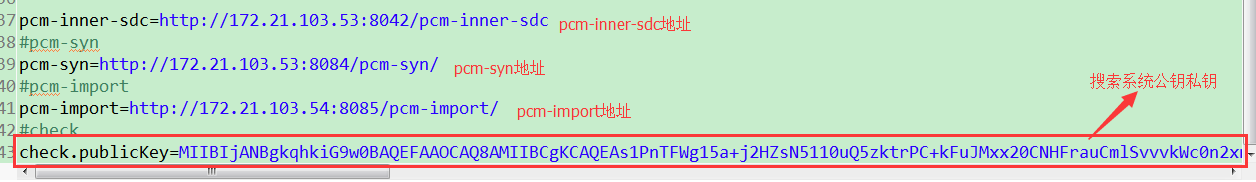


pcm-outer-sdc修改详情：

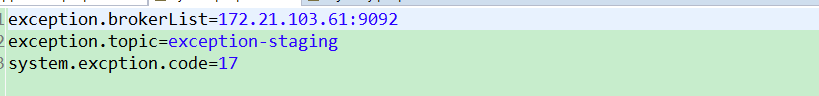
application.properties文件 包含6个地方需要修改：数据库相关配置、REDIS配置、pcm-inner地址、pcm-syn地址、pcm-import地址、搜索系统公钥私钥。



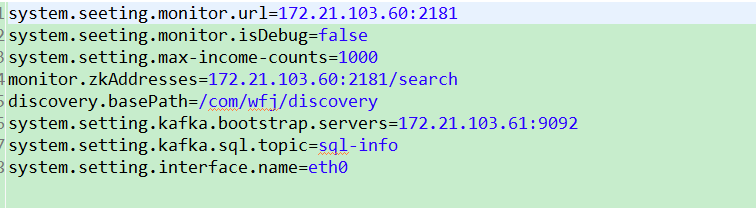




system.properties文件 修改监控平台的相关配置即可

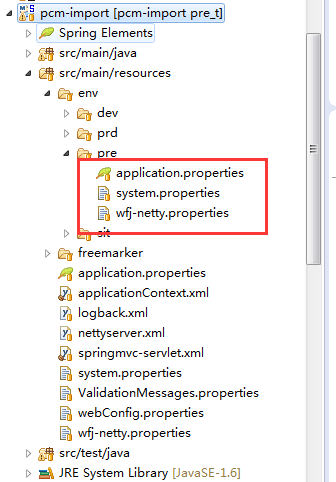


wfj-netty.properties文件修改监控平台的相关配置即可



3.6、pcm-import

需修改下图所示的三个文件

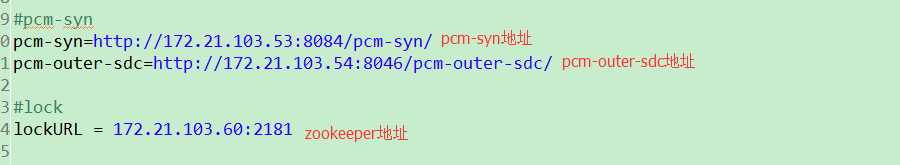


pcm-import修改详情：

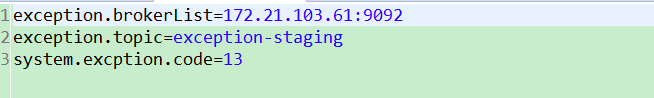
application.properties文件 包含5个地方需要修改：数据库相关配置、REDIS配置、zookeeper地址、pcm-syn地址、pcm-outer-sdc地址。



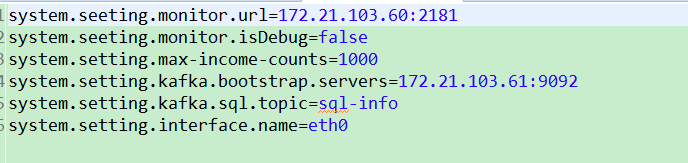




system.properties文件 修改监控平台的相关配置即可

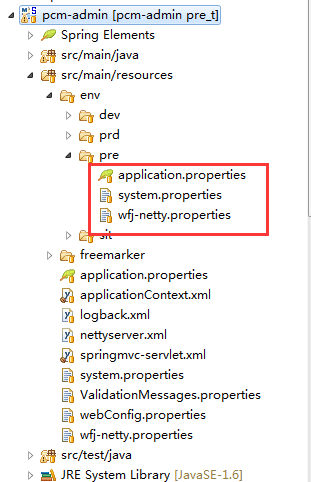


wfj-netty.properties文件修改监控平台的相关配置即可



3.7、pcm-admin

需修改下图所示的三个文件

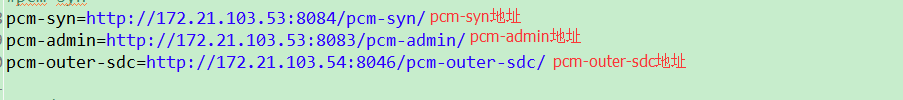


pcm-admin修改详情：

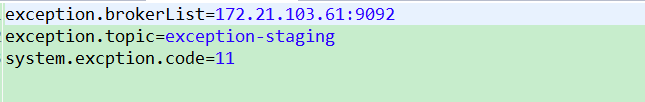
application.properties文件 包含5个地方需要修改：数据库相关配置、REDIS配置、pcm-admin地址、pcm-syn地址、pcm-outer-sdc地址。



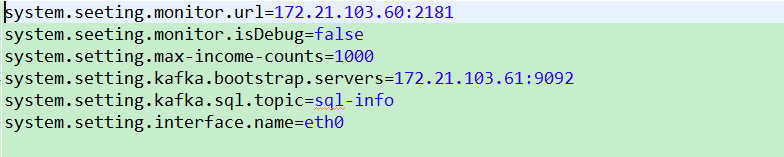




system.properties文件 修改监控平台的相关配置即可

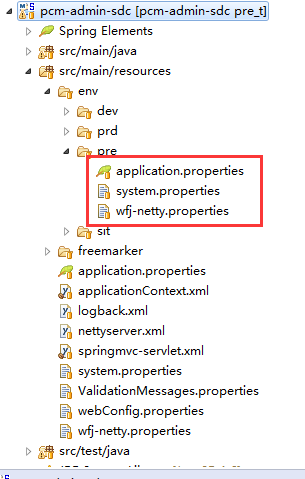


wfj-netty.properties文件修改监控平台的相关配置即可



3.7、pcm-admin-sdc

需修改下图所示的三个文件

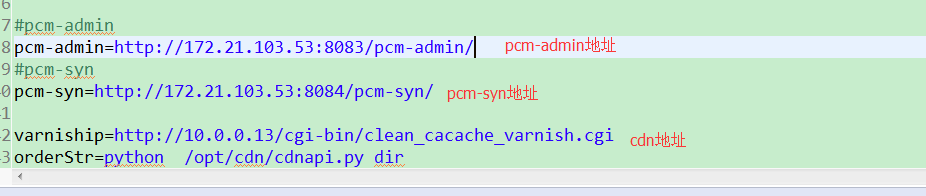


pcm-admin-sdc修改详情：

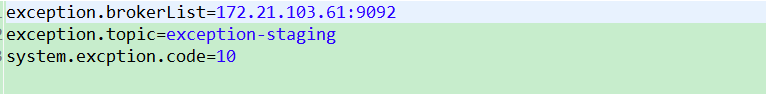
application.properties文件 包含6个地方需要修改：数据库相关配置、REDIS配置、zookeeper地址、pcm-syn地址、pcm-outer-sdc地址。







system.properties文件 修改监控平台的相关配置即可



wfj-netty.properties文件修改监控平台的相关配置即可



4.部署方法

1.基础层instal打包

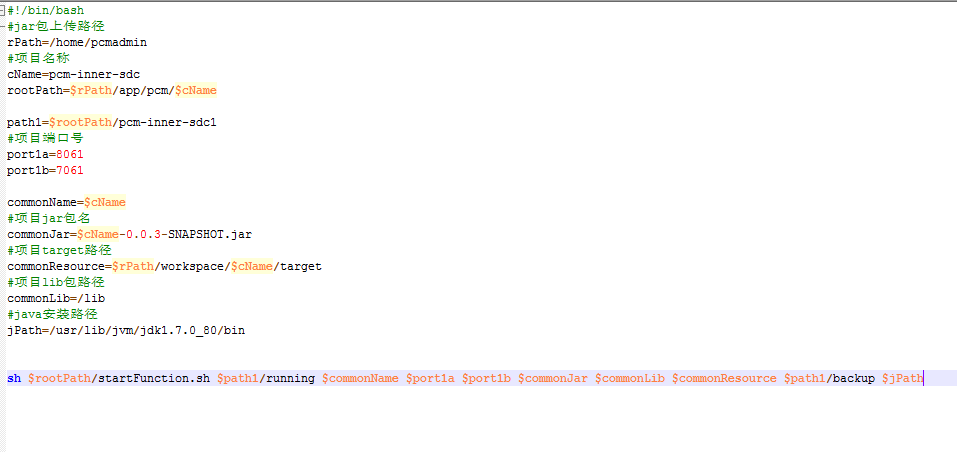
2.业务层instal打包

3. Controller层打包

打包命令：clean install -P pre -Dmaven.test.skip=true （其中pre 为配置文件分支及resources/env目录下的四个文件中的配置文件,使用哪个配置文件填哪个文件夹名称）

启动脚本

**startup.sh文件（详见附录1）**



**startFunction.sh文件（详见附录2）**



打完包后如需启动项目只需修改startup.sh文件中的 **项目名称，java安装目录，项目jar包及lib路径，项目端口号以及 startFunction.sh文件中项目启动时使用的内存大小。**修改完成后执行startup.sh文件即可启动。

**附录1**

**startup.sh文件**

**#!/bin/bash**

**#jar包上传路径**

**rPath=/home/pcmadmin**

**#项目名称**

**cName=pcm-inner-sdc**

**rootPath=$rPath/app/pcm/$cName**

**path1=$rootPath/pcm-inner-sdc1**

**#项目端口号**

**port1a=8061**

**port1b=7061**

**commonName=$cName**

**#项目jar包名**

**commonJar=$cName-0.0.3-SNAPSHOT.jar**

**#项目target路径**

**commonResource=$rPath/workspace/$cName/target**

**#项目lib包路径**

**commonLib=/lib**

**#java安装路径**

**jPath=/usr/lib/jvm/jdk1.7.0\_80/bin**

**sh $rootPath/startFunction.sh $path1/running $commonName $port1a $port1b $commonJar $commonLib $commonResource $path1/backup $jPath**

附录2

**startFunction.sh文件**

**#!/bin/bash**

**#创建项目运行路径**

**path=$1$6**

**if [ -d "$path" ]**

**then**

**echo "yes";**

**else**

**mkdir -p -m 777 "$path"**

**fi**

**#创建项目备份路径**

**path1=$8**

**if [ -d "$path1" ]**

**then**

**echo "yes";**

**else**

**mkdir -p -m 777 "$path1"**

**fi**

**#将项目打包备份**

**tarName=$8/$2\_`date +%y%m%d%H%M`.tar**

**tar cvzf $tarName $1/$5 $1/lib/\***

**#关闭运行的系统**

**echo 关闭 $2 系统开始....**

**kill -9 `cat $1/main.pid`**

**echo 关闭 $2 系统成功...**

**echo 拷贝编译jar包到应用目录 ...**

**rm -f $1/\*.jar**

**rm -f $1/lib/\*.jar**

**cp -f $7/$5 $1/**

**cp -f $7/lib/\*.jar $1/lib/**

**echo 拷贝完成....**

**echo 开始运行服务...**

**#后台启动进程**

**cd $1**

**BUILD\_ID=$9/java -Xmx2048m -Xms2048m -Xmn800m -Xss512k -XX:MaxPermSize=256m -XX:MaxTenuringThreshold=0 -XX:+UseConcMarkSweepGC -jar $5 start $3 $4 >$1/$2.log 2>&1 &**

**echo $! > $1/main.pid**

**echo 启动完毕.....**