# 数据库表

## Oracle

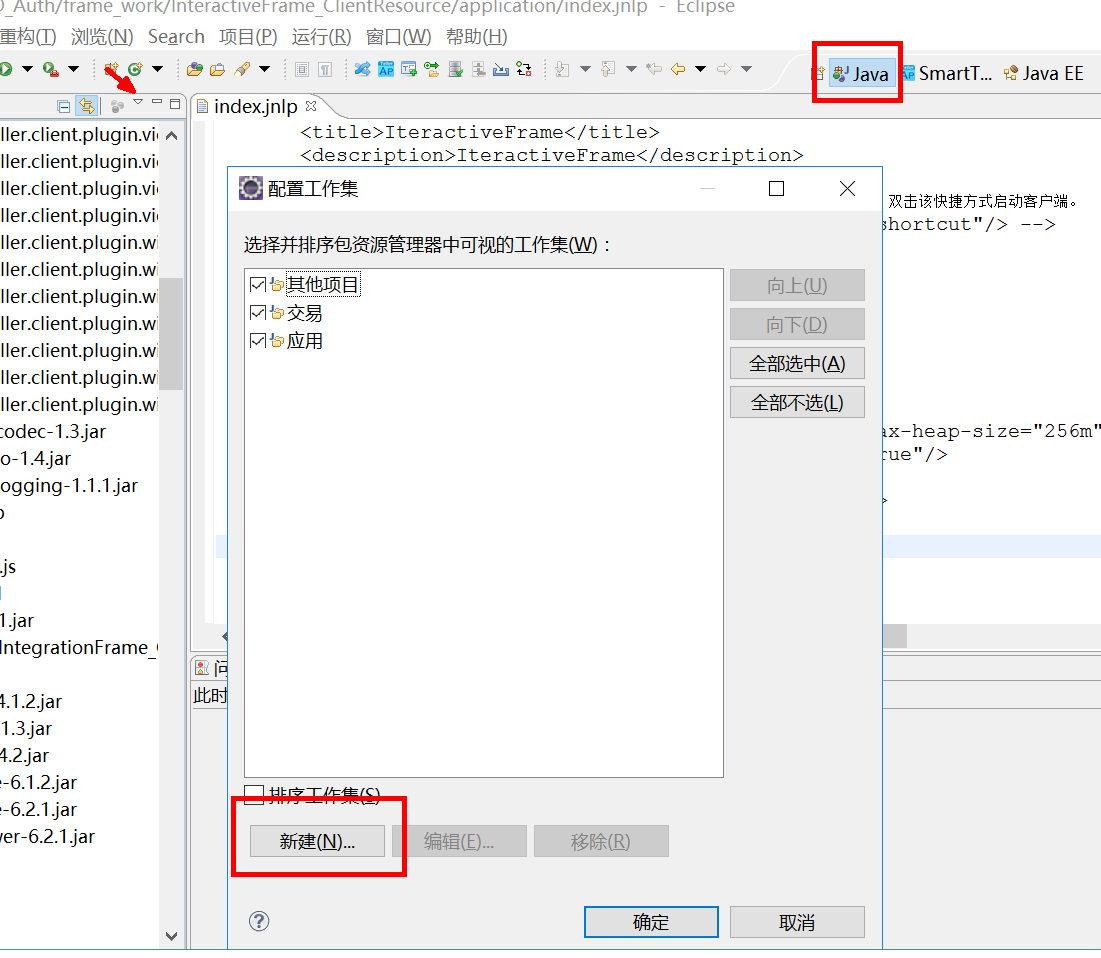
Sys oracle

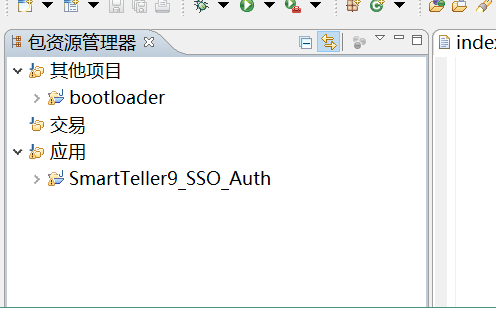
System oracle oracle数据库主用户名密码

|  |
| --- |
| CIF\_CLIENT 查看用户  fw\_sequences 序列表 |

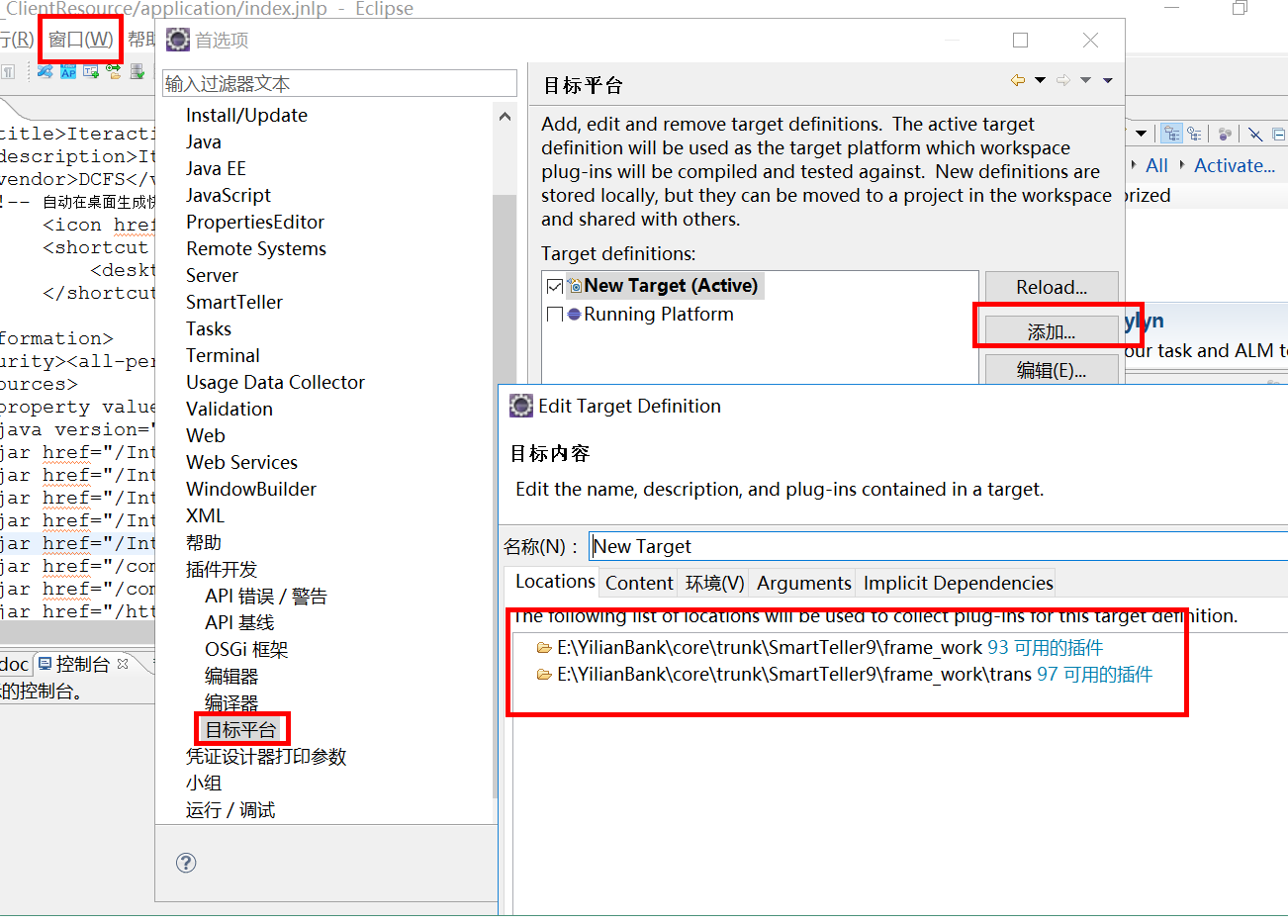
# Teller搭建步骤

## 设置工作集导入项目和交易



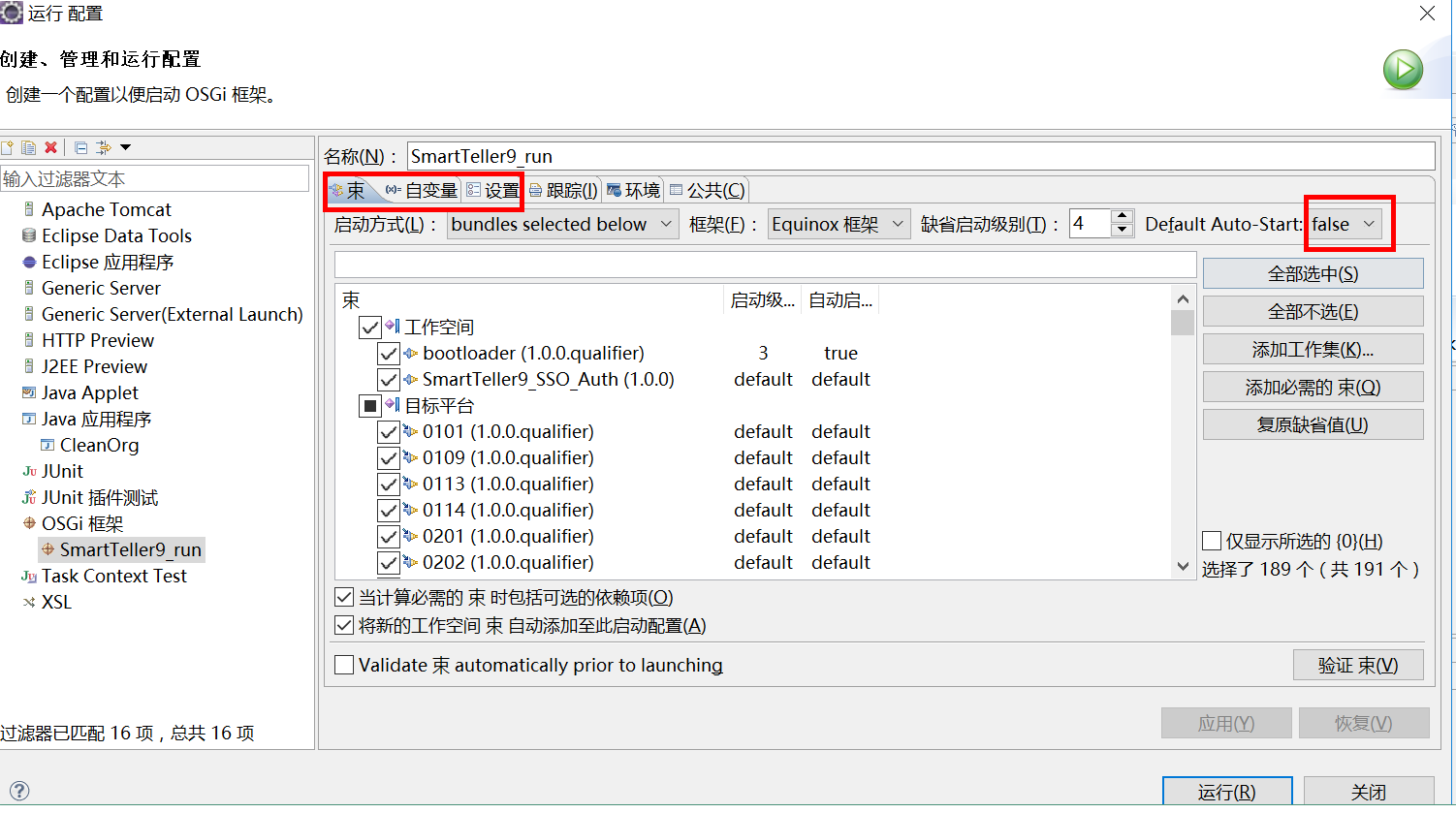


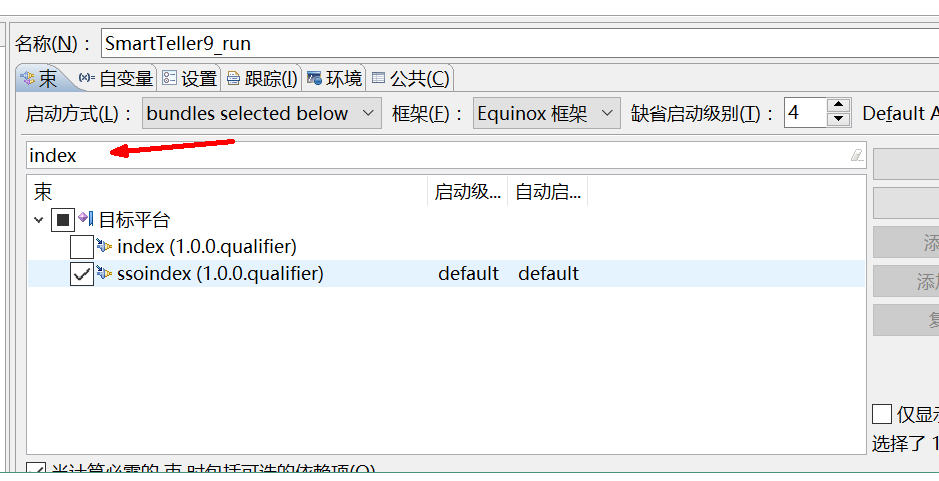
## 配置插件



## 3.配置OSGI

修改前面3项





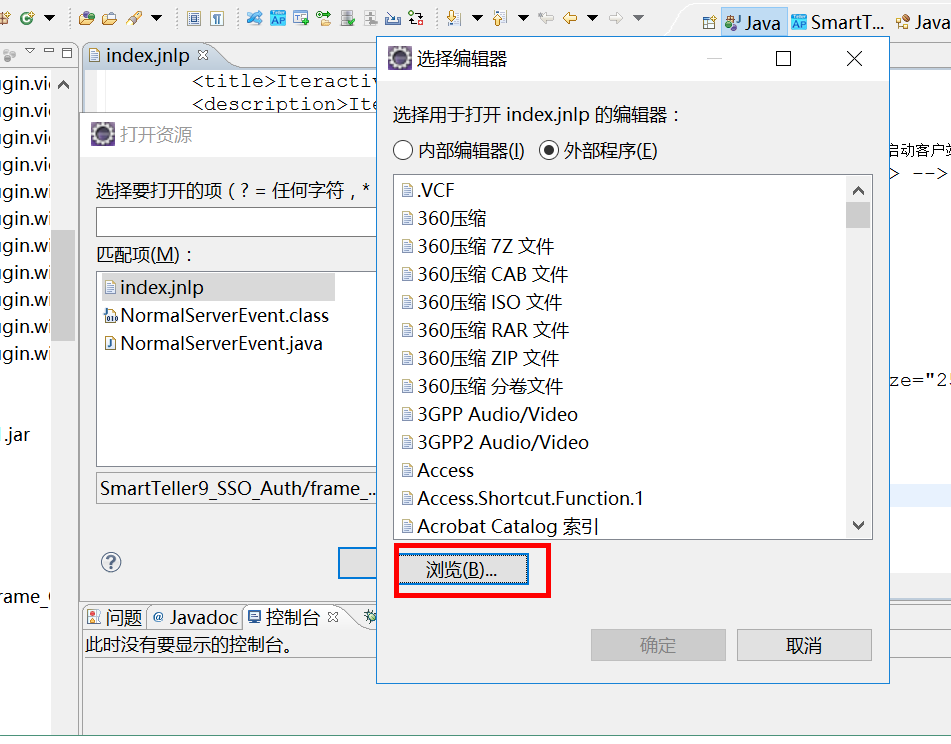
搜索并设置参数 (index 不勾选) (ssol 不勾选) (boo 3 true) org 第二项为 true

Im\_openfire不勾选

## 柜面vm配置

|  |
| --- |
| -Declipse.ignoreApp=true  -Dosgi.noShutdown=true  -Dosgi.osgi.service.http.port=8080  -Dorg.apache.commons.logging.Log=org.apache.commons.logging.impl.Log4JLogger  -Dlog4j.configuration=file:/E:/YilianBank/Teller\_workspace/.metadata/.plugins/org.eclipse.pde.core/SmartTeller9\_run/log4j.properties |

## 4.启动界面



Shift + ctrl+R 搜索index选择1.6 的javaws进行启动

# Teller部署的常见问题

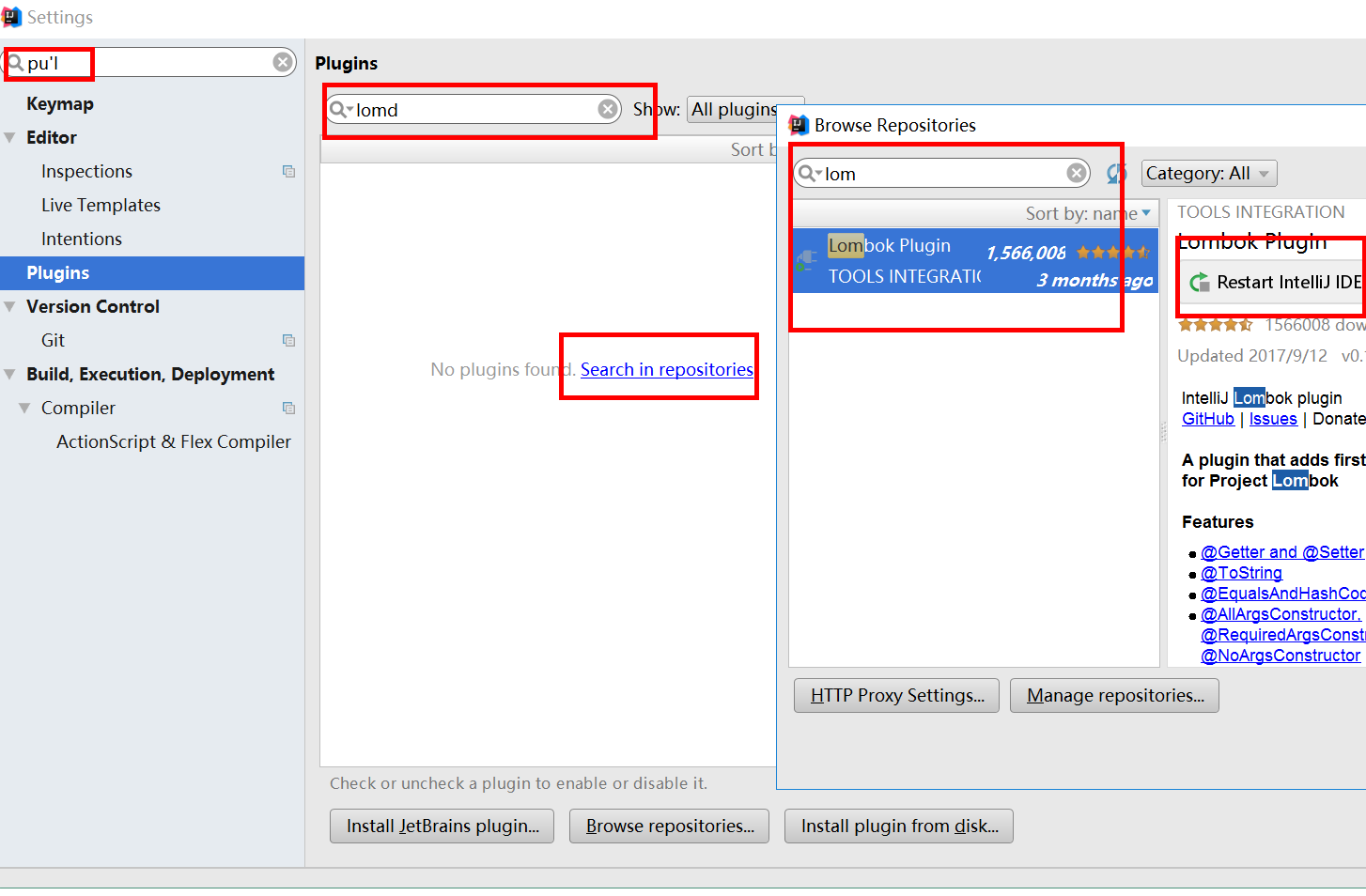
1. 配置目录中的xml不能有多份，xml是按照目录扫描的
2. 若报表没有更新成功，那么可以删除相应obj缓存文件
3. 集群部署时，密钥不一致时，目前没发现程序哪有问题，跟XMPPserver没有关系，所以一般都是加密机的问题
4. 做业务交易时，测试键盘读卡能正常读取，然而业务读卡报空指针异常，要么是代码程序写错，要么是网络不通或者是验证账号的程序没有启动，因为读卡之后会去远程验证卡号是否正确

# 搭建配置

## 核心VM配置

|  |
| --- |
| -Xmx1g  -Xms1g  -Xmn256m  -XX:PermSize=128m  -Xss256k  -javaagent:E:\YilianBank\maven\_jar\org\aspectj\aspectjweaver\1.8.2\aspectjweaver-1.8.2.jar  -javaagent:E:\YilianBank\maven\_jar\com\dcits\galaxy\galaxy-instrument\2.0.0\galaxy-instrument-2.0.0.jar  -Ddubbo.properties.file=galaxy.properties |

## IDEA的get和set注解插件安装



## Teller9数据库转换

注：遇到sql执行异常时候解决方法

启动teller时候会把jar包解压到org..的文件夹下

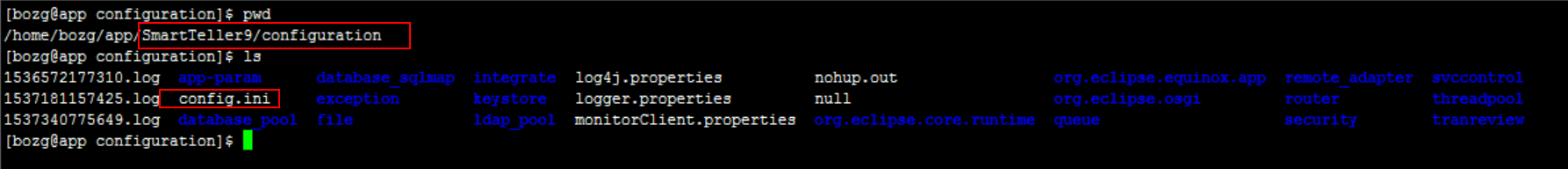
Grep –lr 内容 目标文件夹下路径 搜索该文件

telnet ip 1242 连接到teller控制台

ss 在控制台查看jar启动项

通过id 定位到对应的jar

若找不到端口可以看下面文件中的osgi.console属性



### 1.修改一下trans下jiar包中的plugin 把其中的oracle改成mysql

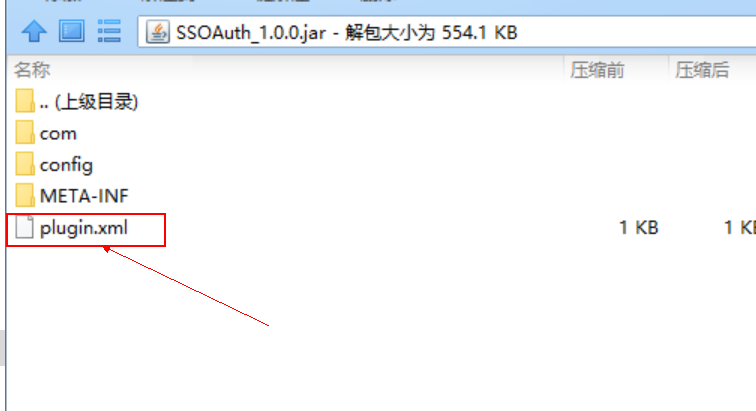
AUTH\_1.0.0.jar

BaseApp\_1.0.0.jar

SSOAuth\_1.0.0.jar

SSOBaseApp\_1.0.0.jar**(SSO版，需修改)**

TransApp\_1.0.0.jar



### 2.修改数据库地址

修改im\_openfire\_3.8.1.jar 中openfireHome\conf\openfire.xml

修改database\_pool/pool.xml 文件中如下

其中mysql为isdefault="true" oracle为isdefault="false"

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <dataSources>  <dataSource id="DBCP" isdefault="true" className="org.apache.commons.dbcp.BasicDataSource">  <property name="driverClassName" value="com.mysql.jdbc.Driver" />  <property name="url" value="jdbc:mysql://21.131.11.126:3306/teller9?useUnicode=true&amp;characterEncoding=utf8" />  <property name="username" value="root" />  <property name="password" value="123456" />  <property name="maxActive" value="50" type="INT"/>  <property name="maxIdle" value="30" type="INT"/>  <property name="minIdle" value="5" type="INT"/>  <property name="initialSize" value="5" type="INT"/>  <property name="maxWait" value="-1" type="LONG"/>  <property name="poolPreparedStatements" value="true" type="BOOLEAN"/>  <property name="maxOpenPreparedStatements" value="1000" type="INT"/>  <property name="validationQuery" value="SELECT 1"/>  <property name="testOnBorrow" value="true" type="BOOLEAN"/>  <property name="testOnReturn" value="false" type="BOOLEAN"/>  <property name="testWhileIdle" value="true" type="BOOLEAN"/>  <property name="connectionProperties" value="config.decrypt=false"/>  </dataSource>  <!--<dataSource id="other\_source" className="org.apache.commons.dbcp.BasicDataSource">  <property name="driverClassName" value="com.mysql.jdbc.Driver" />  <property name="url" value="jdbc:mysql://10.3.2.34:4001/ensemble?useUnicode=true&amp;characterEncoding=utf8" />  <property name="username" value="ensemble" />  <property name="password" value="ensemble\_level" />  <property name="maxActive" value="50" type="INT"/>  <property name="maxIdle" value="30" type="INT"/>  <property name="minIdle" value="5" type="INT"/>  <property name="initialSize" value="5" type="INT"/>  <property name="maxWait" value="-1" type="LONG"/>  <property name="poolPreparedStatements" value="true" type="BOOLEAN"/>  <property name="maxOpenPreparedStatements" value="1000" type="INT"/>  <property name="validationQuery" value="SELECT 1"/>  <property name="testOnBorrow" value="true" type="BOOLEAN"/>  <property name="testOnReturn" value="false" type="BOOLEAN"/>  <property name="testWhileIdle" value="true" type="BOOLEAN"/>  <property name="connectionProperties" value="config.decrypt=false"/>  </dataSource>-->  <dataSource id="other\_source" className="org.apache.commons.dbcp.BasicDataSource">  <property name="driverClassName" value="com.mysql.jdbc.Driver" />  <property name="url" value="jdbc:mysql://21.131.11.126:3306/ensemble?useUnicode=true&amp;characterEncoding=utf8" />  <property name="username" value="root" />  <property name="password" value="123456" />  <property name="maxActive" value="50" type="INT"/>  <property name="maxIdle" value="30" type="INT"/>  <property name="minIdle" value="5" type="INT"/>  <property name="initialSize" value="5" type="INT"/>  <property name="maxWait" value="-1" type="LONG"/>  <property name="poolPreparedStatements" value="true" type="BOOLEAN"/>  <property name="maxOpenPreparedStatements" value="1000" type="INT"/>  <property name="validationQuery" value="SELECT 1"/>  <property name="testOnBorrow" value="true" type="BOOLEAN"/>  <property name="testOnReturn" value="false" type="BOOLEAN"/>  <property name="testWhileIdle" value="true" type="BOOLEAN"/>  <property name="connectionProperties" value="config.decrypt=false"/>  </dataSource>  </dataSources> |

### 3.修改SmartTeller9\_SSO\_Auth/plugin.xml文件

该文件下sql模板如下，根据项目而定

|  |
| --- |
| <!-- sql模板 -->  <sql url="config/resource/sql/mysql/tellerStage\_pl.xml" />  <sql url="config/resource/sql/mysql/tellerStage\_ps.xml" />  <sql url="config/resource/sql/mysql/tellerStage\_extends.xml" />  <sql url="config/resource/sql/mysql/tellerStage\_sso.xml" />  <sql url="config/resource/sql/mysql/tellerStage\_ensemble.xml" />  <sql url="config/resource/sql/mysql/tellerStage\_pics.xml" /> |

该模板是继承了1中jar包的sql

## Teller集群和单机参数配置

1. 修改Venus/InteractiveFrame\_DataBuffer\_\*.jar\config\xml\bufferdeploy \bufferdeploy.xml文件：

<deploy-type value="SINGLE"/> <!-- 单机：SINGLE 集中：CENTRALIZE 分布式：DISTRIBUTED-->

把红色标注的SINGLE改为CENTRALIZE。

1. 修改Venus/InteractiveFrame\_DataBuffer\_\*.jar\config\xml\factory\ factory.xml文件，将注释掉的配置释放开，memcached和ehcache只能启用一种。

|  |
| --- |
| <!-- ehcache版本对应的实现 -->  <!-- <item autoLoad="false"id="TerracottaStorage" impl="com.dcfs.teller.databuffer.storage.impl.TerracottaStorage" /> --> |

1. 修改SmartTeller/InteractiveFrame\_DataBuffer\_\*.jar\config\xml\bufferdeploy \terracotta-ehcache.xml文件

|  |
| --- |
| <terracottaConfig url="10.126.252.67:9510,10.126.252.61:9510" rejoin="true"/>  根据前面tc-config.xml文件中配置的服务情况，将对应的服务器地址和端口填写到url处，用逗号分隔 |

## 异常错误搜索

MySQLSyntaxErrorException

Exception

# Zookeeper配置

进入到zk控制台 ./zkCli.sh -server 172.168.1.106:6789

# 其他

《设计原本》

《领域特定语言》

《人月神话》

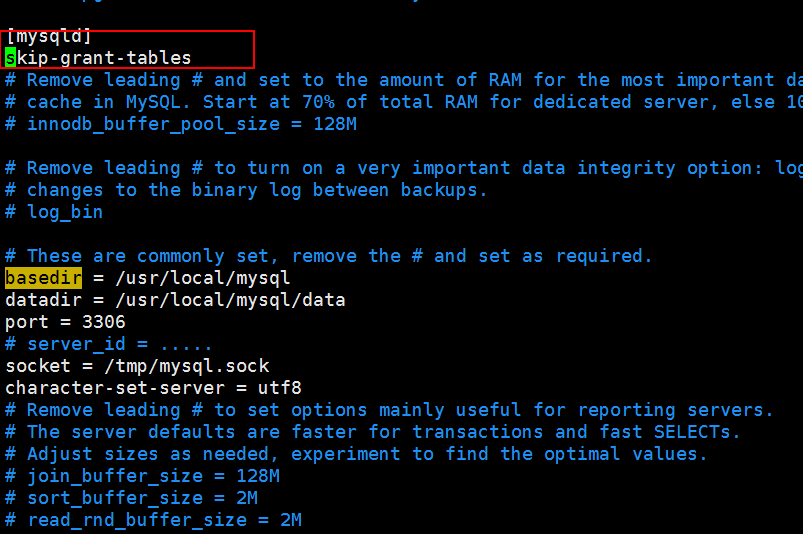
## 微信文件地址

C:\Users\shigyb\Documents\WeChat Files\wxid\_lbhs4zscess\Files

# Mysql操作

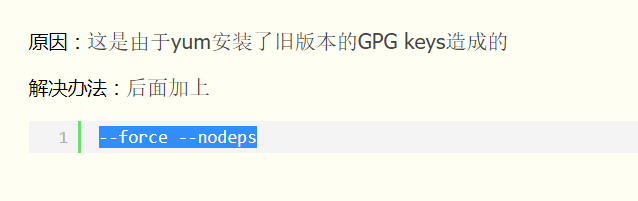
## Mysql无密码登陆

修改my.cnf文件，增加skip-grant-tables



## Mysql安装rmp出现依赖错误

--force --nodeps



这个操作只是跳过一些老版本依赖插件安装，若安装成功不一定能正常使用

## Mysql导入sql文件数据太大时报错

两种方法解决，1是临时的重启后无效，2是永久的重启后也有效

1.进入mysql控制台设置如下后再次导入即可,大小根据文件而定

set global max\_allowed\_packet = 1024\*1024\*512;

2.修改my.conf文件或init.conf文件增加参数,大小根据文件而定

max\_allowed\_packet=512M

## Oracle转Mysql各种对应关系

to\_number(PAR\_VALUE\_ID) 对应 cast(doc\_type as unsigned int)

TO\_CHAR(par\_value) 对应 CONVERT(par\_value,CHAR)

TO\_CHAR (UPDATETIME, 'yyyymmdd') 对应 DATE\_FORMAT(CREATETIME,'%Y%m%d')

SELECT DATE\_FORMAT(NOW(), '%Y%m%d %k:%i:%s' ) A;

date\_format(MEDIUMINFOUPDATETIME,'%Y-%m-%d %T')

date\_format(MEDIUMINFOUPDATETIME,'%Y-%m-%d %m:%i:%s)

Trunk 对应

to\_date(); 对应 str\_to\_date(date,'%Y-%m-%d')

sysdate 对应 sysdate()

|| 字符串连接 concat(,,,…..)

IN 列 对应 IN (列)

Decode(p1,p2,p3,p4) 如果p1=p2 则返回p3 否则返回p4 对应 if(p1=p2,p3,p4)

Nvl(p1,p2) ifnull(p1,p2)

START WITH ORGID = #branch#

CONNECT BY PRIOR ORGID = AUTH\_SUPERIOR

对应

|  |
| --- |
| CREATE FUNCTION `getChildLst`(rootId INT)  RETURNS varchar(1000)  BEGIN  #Routine body goes here...  DECLARE sTemp VARCHAR(1000);  DECLARE sTempChd VARCHAR(1000);    SET sTemp = '$';  SET sTempChd =cast(rootId as CHAR);  WHILE sTempChd is not null DO  SET sTemp = concat(sTemp,',',sTempChd);  SELECT group\_concat(id) INTO sTempChd FROM treeNodes where FIND\_IN\_SET(pid,sTempChd)>0;  END WHILE;  RETURN sTemp;  END; |

## Mysql实现oracle序列

1．创建序列表

|  |
| --- |
| drop table if exists tl9\_sequence;  create table tl9\_sequence  (  name varchar(50) not null comment '序列的名字，唯一',  min\_value bigint not null default 1 comment '最小值',  current\_value bigint not null default 0 comment '当前的值',  increment\_value int not null default 1 comment '步长，默认为1',  max\_value bigint not null default 999999999999999999 comment '最大值,超过最大值则从最小值开始',  primary key (name)  );  alter table tl9\_sequence comment '公共的序列表，用于为非自增且要求唯一的字段记录和获取唯一ID。'; |

1. 创建存储过程

|  |
| --- |
| CREATE FUNCTION func\_nextval (seq\_name varchar(50))  RETURNS integer  BEGIN  declare val integer;  declare min integer;  declare c integer;  declare max integer;  declare incr integer;  set val = 0;  set min = 0;  set c = 0;  set max = 0;  set incr = 0;  select min\_value ,  current\_value ,  increment\_value ,  max\_value into min, c, incr, max  from tl9\_sequence;  SET val=c+incr;  IF min>c OR val>max THEN  SET val=min;  END IF;  return fun\_setval(seq\_name,val);  END  CREATE FUNCTION fun\_setval(seq\_name varchar(50),val integer)  RETURNS integer  BEGIN  update tl9\_sequence  set current\_value = val  where name = seq\_name;  RETURN val;  END |

调用过程函数

SELECT func\_nextval('test\_seq');

# shell

<http://www.cygwin.com/> cygwin工具地址

export

unset

set –x +x 打印脚本日志

sort 排序 | uniq 显示唯一的

join 连接

trap

awk

umask

history 查看执行过的命令

whereis 搜索某文件

export 用于修改或打印环境变量

readonly 使得变量不得修改

env 删除变量

sleep 等待 秒

read 读取输入操作

du –s \* 查看当前磁盘目录和文件的大小

touch 建立文件

eval 将参数作为命令处理

pathfind

touch 更新文件

locate 查看文集名坐在的路径

type 查看命令所在的位置

du -h –s 查看文件大小

diff 比较两个文件的不同

top 显示进程

nohup ./openfire.sh >nohup.out & 后台启动

sh ~/. bash\_profile 重新加载执行该文件

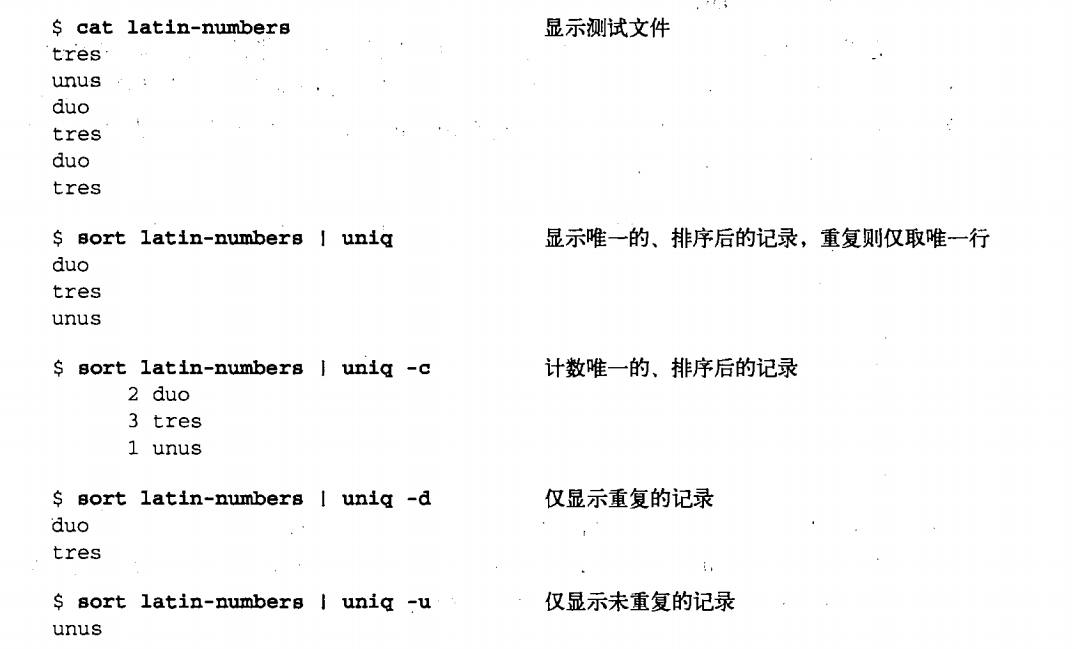
scp 文件名 目的ip : 目的路径 服务器之间文件传送

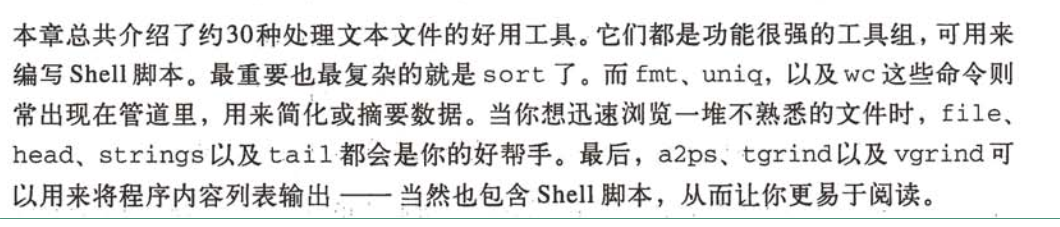
家目录隐藏文件 . bash\_profile 可以增加修改环境变量如下

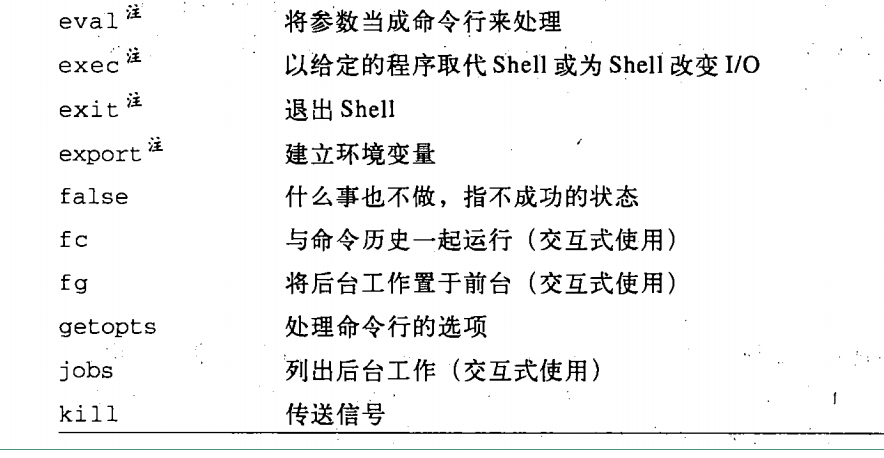
export JAVA\_HOME=/home/dcits/jdk1.7.0\_65

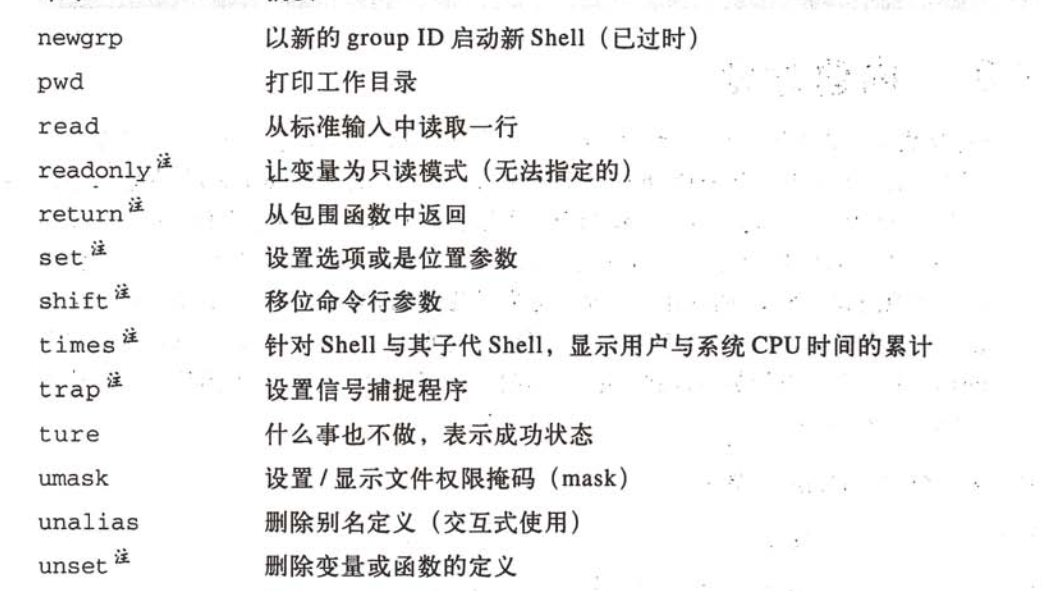
export PATH=/sbin:$JAVA\_HOME/bin:$PATH

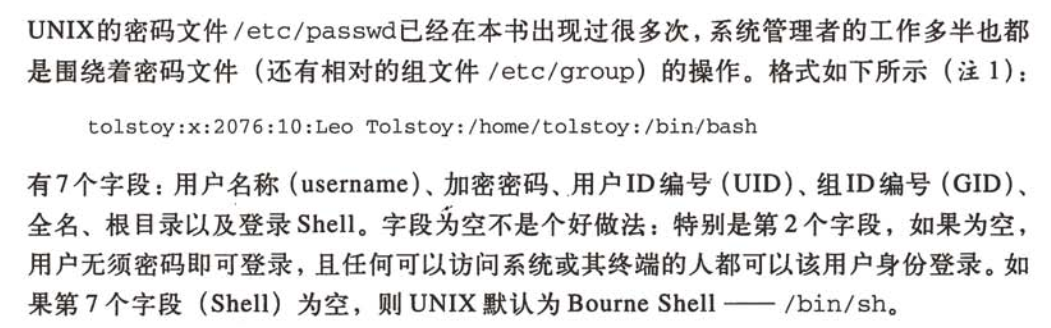
export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar











## 特殊参数

$# ,$\*,$@,

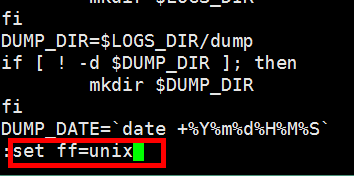
## 给用户设置可操作某目录权限

chown -R mysql:mysql /home/mysql

## 坏的解释器: 没有那个文件或目录

问题:如题

解决:vi 修改如下



回车保存退出

# Java虚拟机JVM

Jdk7u6 源码包下载地址http://jdk.java.net/

48 jdk在linux的编译方式

Java 虚拟机中堆和栈的关系？

-Xoss 设置本地方法栈大小（实际无效）

-Xss 栈容量设置

-XX:PermSize -XX:MaxPermSize 限制方法区大小，从而间接限制常量池的容量

-XX:MaxDirectMemorySize DirectMemory大小指定，若不指定则与java堆最大值-Xmx指定一样

-XX:UseParNewGC 指定收集器（可以一边丢垃圾一边收拾）

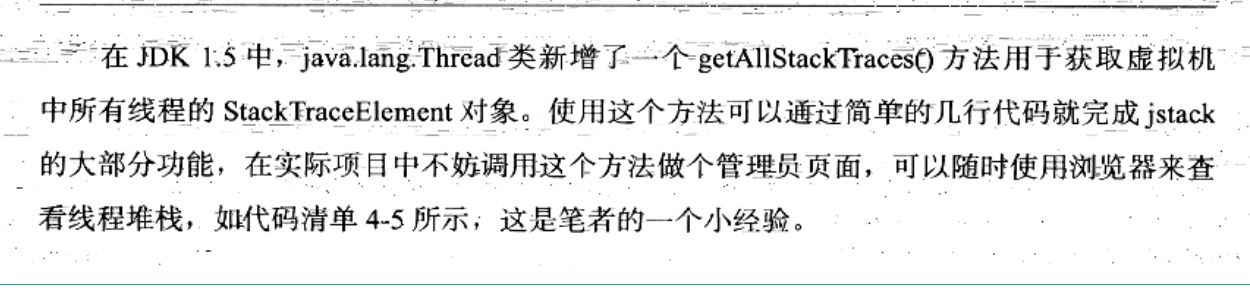
-XX:ParallelGCThreads 在CPU非常多时可以限制垃圾收集的线程数

-XX: ParallelScavenge 适合后台运算不需要太多任务交互的收集器(设置参数百度)，自适应调节，如果不太了解收集器可以用这个来让虚拟机自己来调节

java -XX:+PrintFlagsFinal 查看虚拟机的默认参数

线程堆栈的管理界面案例

110/131页



## Jvm常用命令

### Jps

显示指定系统内所有的HostSpot虚拟机进程

-q 只输出LVMID,省略主类的名称

-m 输出虚拟机进程启动时传递给main方法的参数

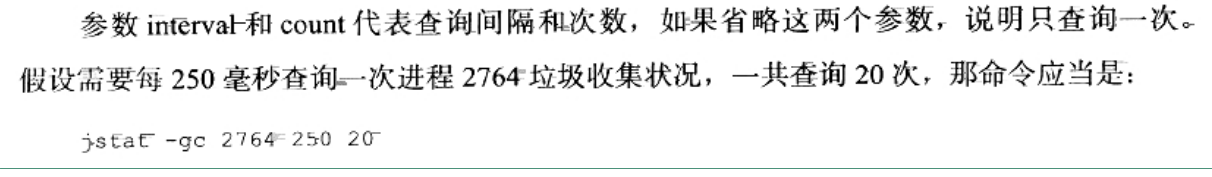
-l 输出主类的全名，如果进程执行的是jar包，输出jar路径

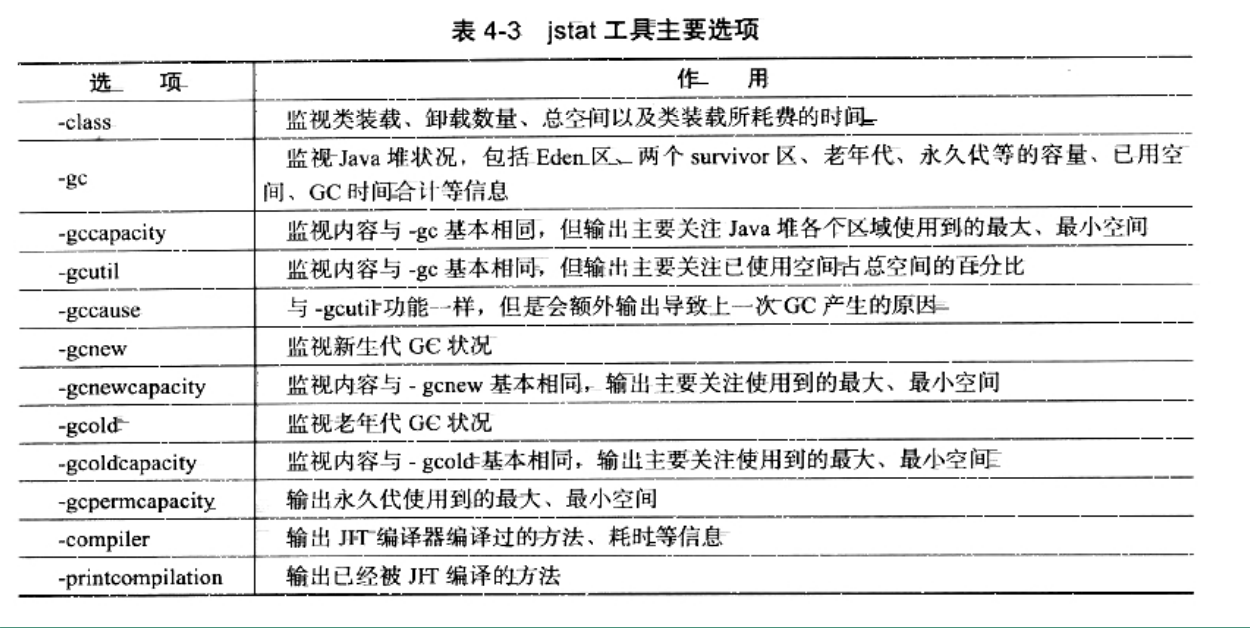
-v 输出虚拟机进程启动时JVM参数

### Jstat

用于收集HotSpot虚拟机各方面的运行数据

Jstat [option vmid [interval [s|ms] count ]]

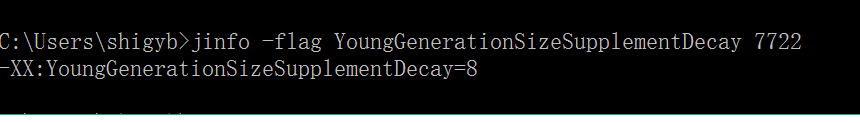




### Jinfo

Configuration Info for java 显示虚拟机配置信息

Jinfo [option] pid

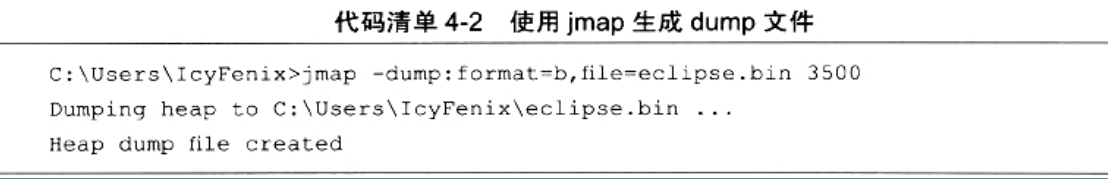


### Jmap

生成虚拟机的内存转储快照heapdump文件

Jmap [option] vmid





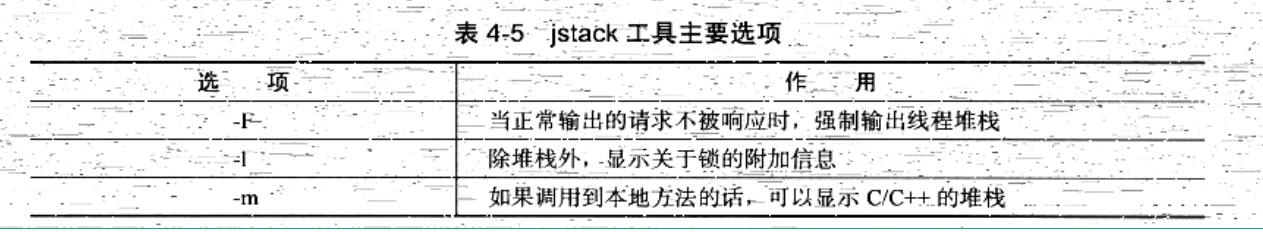
### Jhat

用于分析heapdump文件，它会建立一个HTTP/HTML服务器，让用户可以在浏览器上查看分析结果

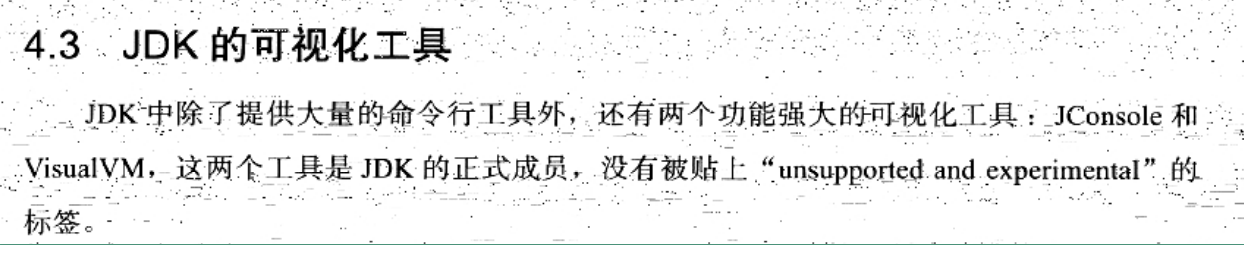
### Jstack

显示虚拟机的线程快照

Jstack [option] vmid



## JDK的可视化工具



配置完jdk环境之后打开cmd输入

Jconsole 打开jdk/bin/下的JConsole程序

Jvisualvm 打开jdk/bin/下的VisualVM 程序

# 性能测试工具

LoadRunner

# 学习地址

|  |
| --- |
| <http://www.cnblogs.com/yiwangzhibujian/p/5832294.html> jetty全指南  http://www.cnblogs.com/xdp-gacl/tag/MyBatis%E5%AD%A6%E4%B9%A0%E6%80%BB%E7%BB%93/ |