# 2、部署Spark3.1

Spark 分布式计算平台，主要承担实验引擎发送过来的计算任务，其中Worker实例可以横向扩展。

> **前置条件**
>
> 需要使用Smartbi提供的Spark3.1安装包部署
>
> 数据挖掘数据量2000万以下时，无需单独部署spark节点，只需要提升数据挖掘服务器配置即可

> **文档环境**
>
> 单机部署数据挖掘组件环境如下：

| 服务器IP | 主机名 | 组件实例 | 部署目录 |
| --- | --- | --- | --- |
| 10.10.204.248 | 10-10-204-248 | 数据挖掘 | /data |
| 10.10.204.249 | 10-10-204-249 | Spark，Hadoop | /data |
| 10.10.204.250 | 10-10-204-250 | Python | /data |

## 1、系统环境准备

> **温馨提示**
>
> 配置防火墙，selinux相关操作，需要管理员权限。

### 1.1 防火墙配置

为了便于安装，建议在安装前关闭防火墙。使用过程中，为了系统安全可以选择启用防火墙，但必须启用服务相关端口。

**1.关闭防火墙**

临时关闭防火墙（立即生效）

```
systemctl stop firewalld
```

永久关闭防火墙（重启后生效）

```
systemctl disable firewalld
```

查看防火墙状态

```
systemctl status firewalld
```

**2.开启防火墙**

相关服务及端口对照表：

| 服务名 | 需要开放端口 |
| --- | --- |
| Spark | 8080，8081，7077，[30000-65535] |

如果确实需要打开防火墙安装，需要给防火墙放开以下需要使用到的端口
开启端口：8080，8081，7077，[30000-65535]

```
firewall-cmd --zone=public --add-port=8080/tcp --permanent
firewall-cmd --zone=public --add-port=8081/tcp --permanent
firewall-cmd --zone=public --add-port=7077/tcp --permanent
firewall-cmd --zone=public --add-port=30000-65535/tcp --permanent
```

配置完以后重新加载firewalld，使配置生效

```
firewall-cmd --reload
```

查看防火墙的配置信息

```
firewall-cmd --list-all
```

**3.关闭selinux**

临时关闭selinux，立即生效，不需要重启服务器。

```
setenforce 0
```

永久关闭selinux，修改完配置后需要重启服务器才能生效

```
sed -i 's/=enforcing/=disabled/g'  /etc/selinux/config
```

# 2、Spark单节点安装

> **温馨提示**
>
> 配置主机名映射，需要管理员权限。

## 2.1 配置主机名映射

将数据挖掘组件中的服务器主机名映射到hosts文件中

```
vi /etc/hosts
```

文件末尾添(根据实际环境信息设置)：

```
10.10.204.248 10-10-204-248
10.10.204.249 10-10-204-249
10.10.204.250 10-10-204-250
```

## 2.2 配置系统免密登录

登陆服务器，生成密钥

```
ssh-keygen
```

输入ssh-keygen后，连续按三次回车，不用输入其它信息。

分别复制248节点的公钥文件到集群中所有的服务器上：

```
ssh-copy-id -i ~/.ssh/id_rsa.pub 10-10-204-248
ssh-copy-id -i ~/.ssh/id_rsa.pub 10-10-204-249
ssh-copy-id -i ~/.ssh/id_rsa.pub 10-10-204-250
```

测试是否设置成功

示例:

```
ssh 10-10-204-249
```

如果不用输入密码，表示配置成功

## 2.3 安装JAVA环境

解压jdk到指定目录：

```
tar -zxvf jdk-8u181-linux-x64.tar.gz -C /data
```

添加环境变量

```
vi /etc/profile
```

在文件末尾添加下面内容：

```
export JAVA_HOME=/data/jdk1.8.0_181
export JAVA_BIN=$JAVA_HOME/bin
export CLASSPATH=:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
export PATH=$PATH:$JAVA_BIN
```

让配置生效

```
source /etc/profile
```

验证安装

```
java -version
```

## 2.4 安装Spark

> **温馨提示**
>
> 1、安装部署、启动spark等，可以使用普通用户权限进行操作。
>
> 2、部署过程中用普通用户操作，则后续的所有运维操作等，都需要用普通用户来执行。如果切换其他用户操作，可能会因为权限问题导致服务启动失败。
>
> 3、spark的端口配置如果小于1024，也需要管理员权限才能启动服务。

解压Spark到指定目录

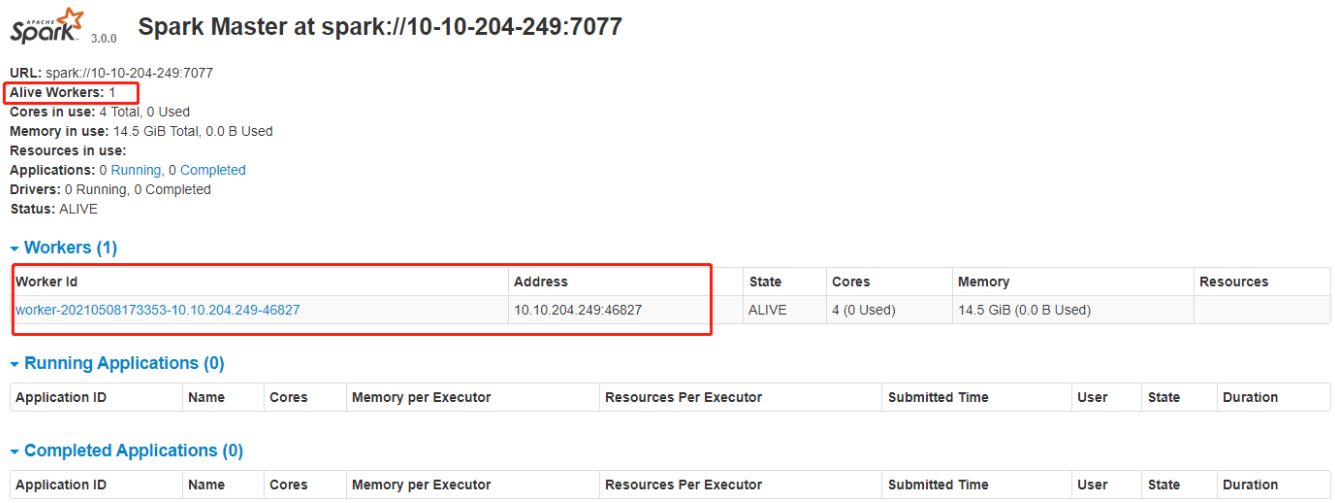
```
tar -zxvf spark-3.1.3-bin-hadoop3.2.tgz -C /data
```

启动Spark

```
cd /data/spark-3.1.3-bin-hadoop3.2/sbin
 ./start-all.sh
```

## 2.5 检查Spark

在浏览器中输入：http://master节点的ip:8080，查看集群状态




在spark节点提交任务测试进入/data/spark-3.1.3-bin-hadoop3.2/bin目录，执行以下命令(注意将”节点IP”替换对应的IP或主机名)

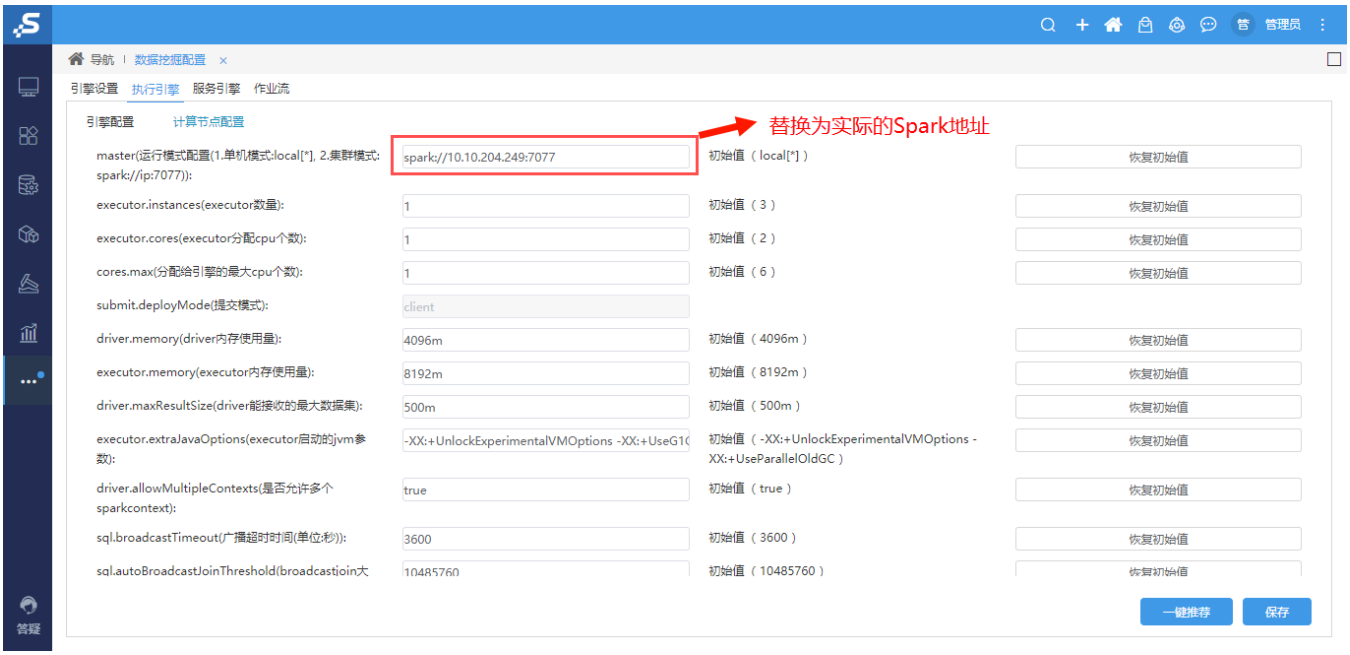
```
./spark-submit --class org.apache.spark.examples.SparkPi --master spark://IP:7077 /data/spark-3.1.3-bin-hadoop3.2/examples/jars/spark-examples_2.12-3.1.3.jar 100
```

```
21/05/08 18:12:24 INFO TaskSetManager: Finished task 95.0 in stage 0.0 (TID 95) in 108 ms on 10.10.204.249 (executor
21/05/08 18:12:24 INFO TaskSetManager: Finished task 98.0 in stage 0.0 (TID 98) in 58 ms on 10.10.204.249 (executor 0
21/05/08 18:12:24 INFO TaskSetManager: Finished task 99.0 in stage 0.0 (TID 99) in 53 ms on 10.10.204.249 (executor 0
21/05/08 18:12:24 INFO DAGScheduler: ResultStage 0 (reduce at SparkPi.scala:38) finished in 8.223 s
21/05/08 18:12:24 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
21/05/08 18:12:24 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for this job
21/05/08 18:12:24 INFO TaskSchedulerImpl: Killing all running tasks in stage 0: Stage finished
21/05/08 18:12:24 INFO DAGScheduler: Job 0 finished: reduce at SparkPi.scala:38, took 8.343117 s
Pi is roughly 3.1418171141817113
21/05/08 18:12:24 INFO SparkUI: Stopped Spark web UI at http://10-10-204-249:4040
21/05/08 18:12:24 INFO StandaloneSchedulerBackend: Shutting down all executors
21/05/08 18:12:24 INFO CoarseGrainedSchedulerBackend$DriverEndpoint: Asking each executor to shut down
21/05/08 18:12:24 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
21/05/08 18:12:24 INFO MemoryStore: MemoryStore cleared
21/05/08 18:12:24 INFO BlockManager: BlockManager stopped
21/05/08 18:12:24 INFO BlockManagerMaster: BlockManagerMaster stopped
21/05/08 18:12:24 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
21/05/08 18:12:24 INFO SparkContext: Successfully stopped SparkContext
21/05/08 18:12:24 INFO ShutdownHookManager: Shutdown hook called
21/05/08 18:12:24 INFO ShutdownHookManager: Deleting directory /tmp/spark-2663bbbd-1aec-45ac-b366-c7f1fcbaa33e
```

运行得出圆周率Pi的近似值3.14即部署成功。

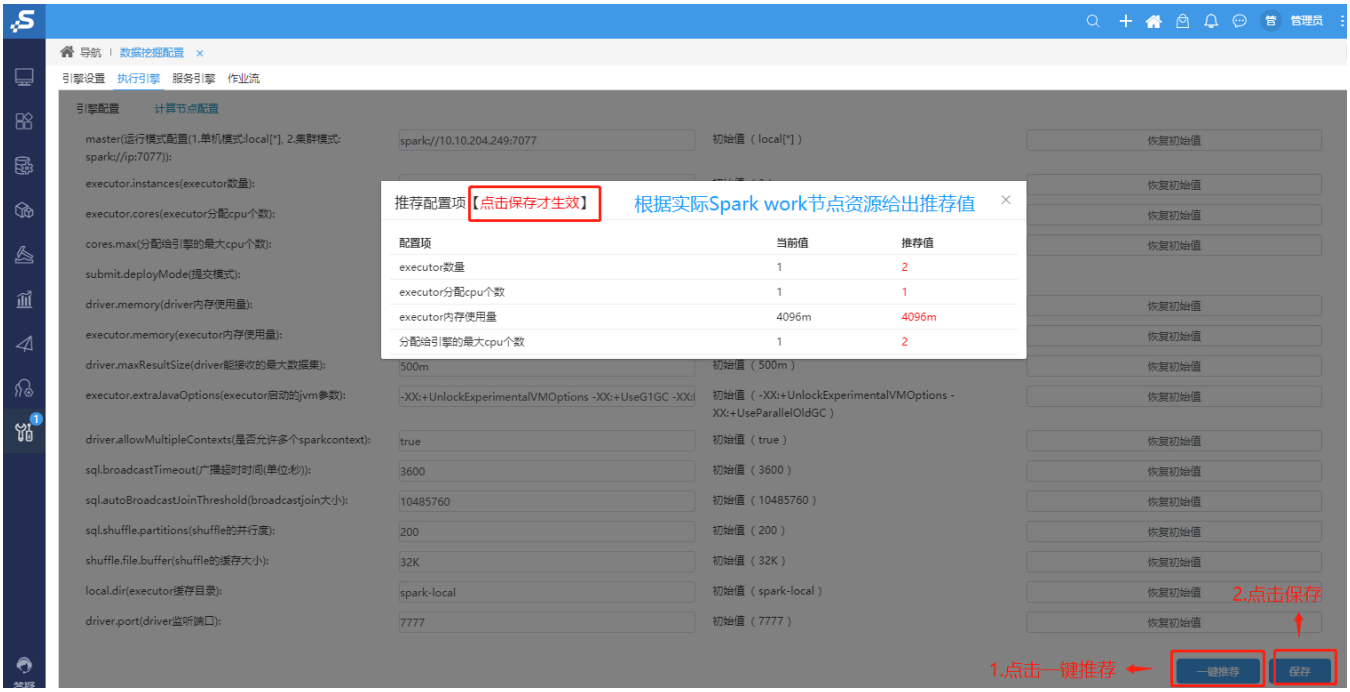
## 2.6 Smartbi连接Spark


配置spark计算节点，打开**系统运维－数据挖掘配置－执行引擎—计算节点配置**，参考下图设置，修改完成后点击保存




配置Spark节点资源，点击一键推荐，系统会根据Spark work节点的服务器资源，生成推荐的配置(如果使用推荐值，记得点击保存，否则配置不生效)：

注意：如果Spark节点服务器还部署了其他应用，spark节点资源建议手动配置。




配置完成后可参考： 测试数据挖掘及其组件 运行数据挖掘实验

## 2.7 运维操作

启动/停止spark服务

```
cd /data/spark-3.1.3-bin-hadoop3.2/sbin
./start-all.sh    #spark
./stop-all.sh     #spark
```

查看日志
Spark的日志路径：/data/spark-3.1.3-bin-hadoop3.2/logs
安装部署或者使用中有问题，可能需要根据日志来分析解决。