1、部署数据挖掘集群

数据挖掘包含两大部分：数据挖掘执行引擎、数据挖掘服务引擎

数据挖掘执行引擎：

- 负责接收Smartbi 发送执行请求。
- 通过解析执行定义，生成spark 计算任务或python计算任务，分别发送给spark集群或python集群。
- 本身并不承担计算任务，只负责计算任务的调度跟分发。

数据挖掘服务引擎：

- 提供模型预测服务给第三放系统调用

> **文档环境**
>
> 集群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
10.10.35.64	10-10-35-64	**数据挖掘-1**，**Zookeeper-1**，Python-1	/data
10.10.35.65	10-10-35-65	**数据挖掘-2**，Spark-1，Hadoop-1	/data
10.10.35.66	10-10-35-66	Spark-2，**Zookeeper-2**，Hadoop-2	/data
10.10.35.67	10-10-35-67	Spark-3，**Zookeeper-3**，Hadoop-3，Python-2	/data
10.10.204.250	10-10-204-250	Smartbi-Proxy	/data

> 如果Python计算任务较多，建议Python节点单独部署

1、系统环境准备

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
执行引擎	8899，4040，7777,，[30000-65535]
服务引擎	8900
Zookeeper	2181，2888，3888

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：8900，8899，4040，7777，[30000-65535]

```
firewall-cmd --zone=public --add-port=8899/tcp --permanent
firewall-cmd --zone=public --add-port=8900/tcp --permanent
firewall-cmd --zone=public --add-port=4040/tcp --permanent
firewall-cmd --zone=public --add-port=7777/tcp --permanent
firewall-cmd --zone=public --add-port=30000-65535/tcp --permanent
firewall-cmd --zone=public --add-port=2181/tcp --permanent
firewall-cmd --zone=public --add-port=2888/tcp --permanent
firewall-cmd --zone=public --add-port=3888/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
```

1.2 取消打开文件限制

修改/etc/security/limits.conf文件在文件的末尾加入以下内容：

```
vi /etc/security/limits.conf
```

在文件的末尾加入以下内容：

```
* soft nofile 65536
* hard nofile 65536
* soft nproc 131072
* hard nproc 131072
```

1.3 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
```

1.4 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
```

文件末尾添加(根据实际环境信息设置)：

```
10.10.35.64 10-10-35-64
10.10.35.65 10-10-35-65
10.10.35.66 10-10-35-66
10.10.35.67 10-10-35-67
```

注意！

部署smartbi服务器的/etc/hosts，需要添加所有数据挖掘组件的主机和IP地址映射

2、部署数据挖掘-执行引擎(高可用)

数据挖掘执行引擎-高可用 节点说明

数据挖掘执行引擎需要依赖zookeeper，故而文档环境部署zookeeper集群。

主机名	角色
10-10-35-64	执行引擎(主)，Zookeeper-1
10-10-35-65	执行引擎(备)
10-10-35-66	Zookeeper-2
10-10-35-67	Zookeeper-3

2.1 部署Zookeeper集群

> **注意！**
>
> Zookeeper的版本必须要高于3.5.5版本

1、登陆zookeeper-1节点执行操作。

① 上传zookeeper安装包到服务器，并解压到指定目录：

```
tar -zxvf zookeeper-3.5.9.tar.gz -C /data/
```

② 创建zookeeper数据目录、日志目录

```
cd /data/zookeeper-3.5.9
mkdir {data,log}
```

③ 修改zookeeper配置文件

```
cd /data/zookeeper-3.5.9/conf
mv zoo_sample.cfg zoo.cfg       #
vi zoo.cfg                      #
```

zookeeper配置文件参考：

```
tickTime=60000
initLimit=300
syncLimit=5
#
dataDir=/data/zookeeper-3.5.9/data
dataLogDir=/data/zookeeper-3.5.9/log
clientPort=2181
#20
autopurge.snapRetainCount=20
#48
autopurge.purgeInterval=48
#zookeeper
server.1=10-10-35-64:2888:3888
server.2=10-10-35-66:2888:3888
server.3=10-10-35-67:2888:3888
```

④ 将Zookeeper安装包分发到其他节点

假设当前的系统用户为root命令如下：

```
scp -r /data/zookeeper-3.5.9  root@10-10-35-66:/data/
scp -r /data/zookeeper-3.5.9  root@10-10-35-67:/data/
```

2、创建myid文件，并写入ID，集群中每个节点mysqid不能相同

```
echo 1 > /data/zookeeper-3.5.9/data/myid        #zookeeper-1
echo 2 > /data/zookeeper-3.5.9/data/myid        #zookeeper-2
echo 3 > /data/zookeeper-3.5.9/data/myid        #zookeeper-3
```

3、启动Zookeeper集群

所有节点启动Zookeeper服务

```
cd /data/zookeeper-3.5.9/bin
./zkServer.sh start
```

4、查看每个节点Zookeeper状态

```
cd /data/zookeeper-3.5.9/bin
./zkServer.sh status
```

其中有一个leader节点，两个follower节点

zookeeper集群部署完成。

2.2 部署执行引擎(主节点)

1、解压数据挖掘安装包到指定的目录

```
tar -zxvf SmartbiMiningEngine-V10.0.64186.21183.tar.gz -C /data
```

2、启动数据挖掘执行引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./experiment-daemon.sh start
```

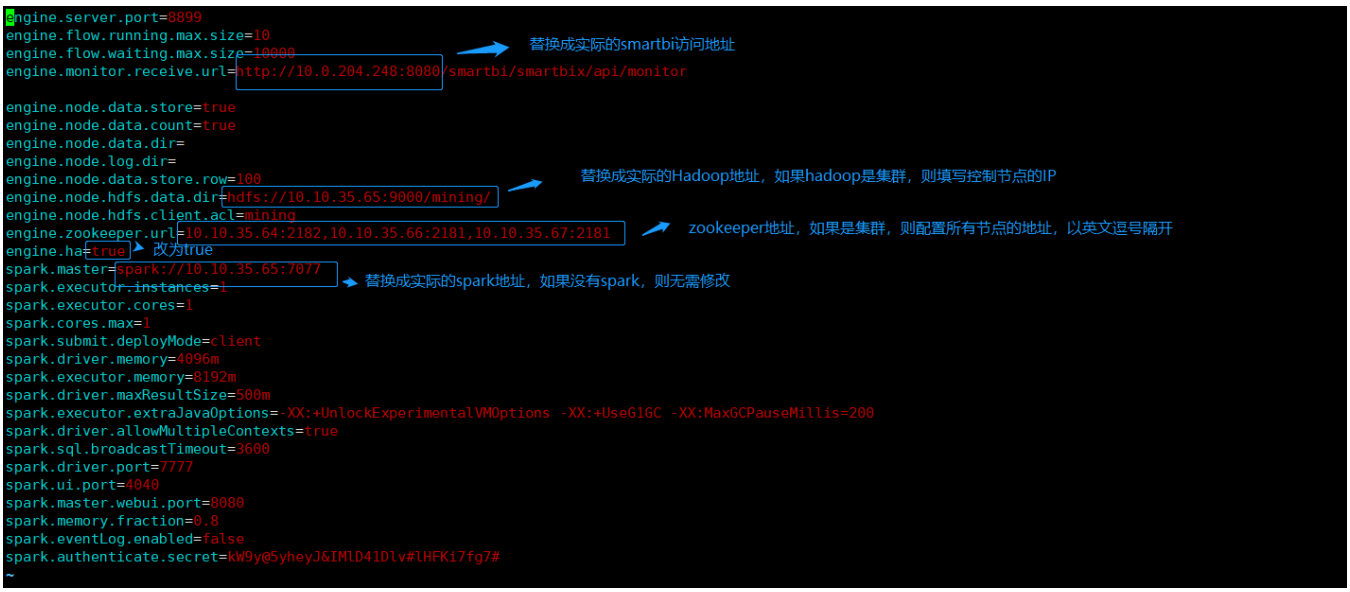
> **注意事项**
>
> 首次启动执行引擎只是为了生成相关配置文件

3、修改执行引擎配置文件

进入配置文件目录，注意：下面的配置文件是执行引擎首次启动时生成的。

```
cd /data/smartbi-mining-engine-bin/conf
vi experiment-application.properties
```

experiment-application.properties配置文件具体修改如下图：



4、停止执行引擎服务

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./experiment-daemon.sh stop
```

5、将数据挖掘安装包分发到其他节点

假设当前的系统用户为root命令如下：

```
scp -r /data/smartbi-mining-engine-bin  root@10-10-35-65:/data/
```

6、启动数据挖掘执行引擎集群

分别登陆两个节点，执行脚本启动执行引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./experiment-daemon.sh start
```

2.3 运维操作

1、启动/重启/查看执行引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./experiment-daemon.sh restart   #
./experiment-daemon.sh stop      #
./experiment-daemon.sh status    #
```

2、测试执行引擎

参考 测试数据挖掘集群

3、部署数据挖掘服务引擎(负载均衡)

数据挖掘执行引擎-高可用 节点说明

数据挖掘服务引擎需要依赖zookeeper，故而文档环境部署zookeeper集群，服务引擎需要依赖Smartbi-Proxy代理，故而需要部署Smartbi-Proxy环境。

主机名	角色
10-10-35-64	执行引擎(主)，Zookeeper-1
10-10-35-65	执行引擎(备)
10-10-35-66	Zookeeper-2
10-10-35-67	Zookeeper-3
10-10-204-250	Smartbi-Proxy

服务引擎使用相同zookeeper集群，无需重复部署。

3.1 部署Smartbi-Proxy

登陆10-10-204-250节点部署Smartbi-Proxy。

1、Tomcat安装包解压到/opt目录

```
tar -zxvf apache-tomcat-8.5.57.tar.gz -C /data
```

2、修改Tomcat启动参数

进入Tomcat下的bin目录

```
cd /data/apache-tomcat-8.5.57/bin
```

创建Tomcat启动参数文件：setenv.sh

```
vi setenv.sh
```

具体参数如下(根据实际部署替换配置中的路径)：

```
export JAVA_HOME="/data/jdk1.8.0_181"
export JRE_HOME="/data/jdk1.8.0_181/jre"
export CATALINA_HOME="/data/apache-tomcat-8.5.57"
export JAVA_OPTS="-Dfile.encoding=UTF-8 -Duser.region=CN -Duser.language=zh -Djava.awt.headless=true -
Xms512m -Xmx2048m -XX:MaxPermSize=512m -Dmail.mime.splitlongparameters=false -XX:+HeapDumpOnOutOfMemoryError
-XX:+UseG1GC"
```

赋予setenv.sh相关权限

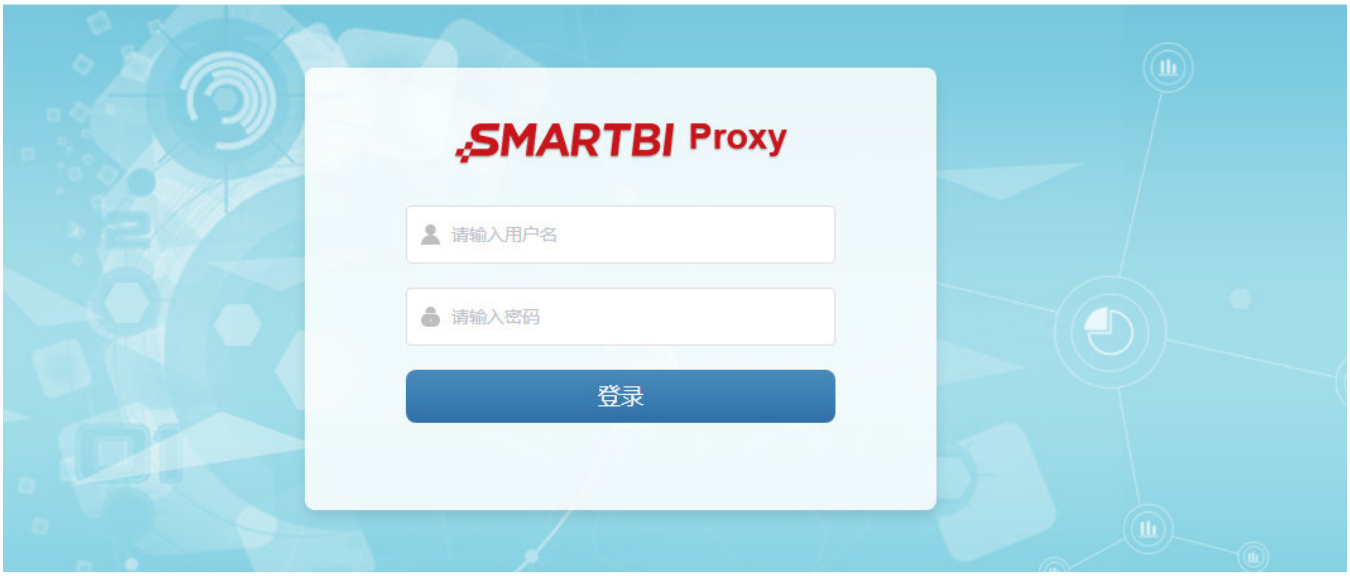
```
chmod 755 setenv.sh
```

3、上传Smartbi-Proxy war包到目录 /data/apache-tomcat-8.5.57/webapps/ 目录

4、启动Tomcat服务

```
cd /data/apache-tomcat-8.5.57/bin
./startup.sh
```

5、启动完成后，浏览器访问Smartbi-Proxy控制台，http://IP:PORT/smartbi/proxy 控制台初始账号密码都是admin。



Smartbi-Proxy部署完成。

3.2 部署服务引擎

1、解压数据挖掘安装包到指定的目录

```
tar -zxvf SmartbiMiningEngine-V10.0.64186.21183.tar.gz -C /data
```

> **注意事项**
>
> 执行引擎与服务引擎部署在相同服务器，无需重复解压安装包，使用相同安装包即可。

2、启动数据挖掘服务引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
chmod +x *.sh
./service-daemon.sh start
```

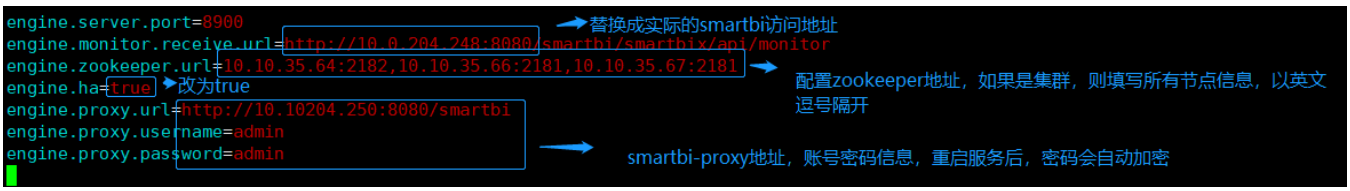
> **注意事项**
>
> 首次启动服务引擎只是为了生成相关配置文件

3、修改服务引擎配置文件

进入配置文件目录，注意：下面的配置文件是服务引擎首次启动时生成的。

```
cd /data/smartbi-mining-engine-bin/conf
vi service-application.properties
```

service-application.properties配置文件具体修改如下图：



4、停止服务引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./service-daemon.sh stop
```

5、将数据挖掘安装包分发到其他节点

假设当前的系统用户为root命令如下：

```
scp -r /data/smartbi-mining-engine-bin  root@10-10-35-65:/data/
```

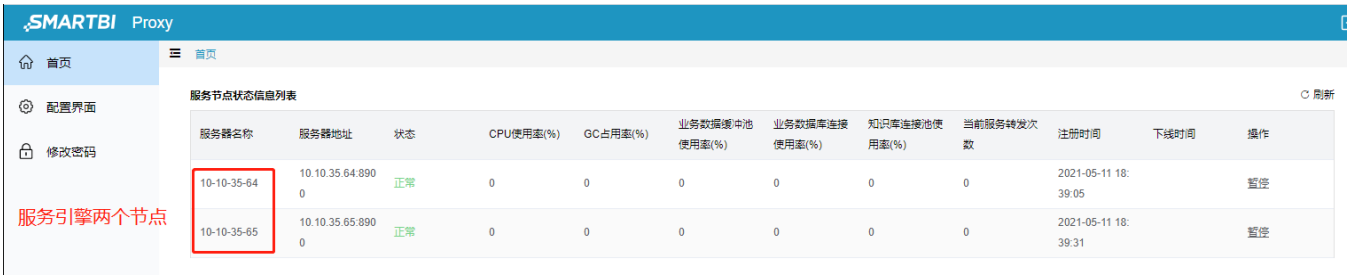
> **注意事项**
>
> 如果服务引擎集群与执行引擎部署在相同的服务器时，请直接启动服务引擎，并修改服务引擎的配置文件，无需再重复分发安装包

6、启动数据挖掘服务引擎集群

分别登陆两个节点，执行脚本启动服务引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./service-daemon.sh start
```

7、服务引擎启动后，可以登陆Smartbi-Proxy控制台查看服务引擎节点信息，如下图：



3.2 运维操作

1、启动/重启/查看服务引擎

```
cd /data/smartbi-mining-engine-bin/engine/sbin/
./experiment-daemon.sh restart   #
./experiment-daemon.sh stop      #
./experiment-daemon.sh status    #
```

2、测试服务引擎

参考 测试数据挖掘集群