

# Prepear

## Download

```
1. wget --no-cookies --no-check-certificate --header "Cookie: gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-x64.tar.gz"
2. wget --no-cookies --no-check-certificate --header "Cookie: gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u66-b17/jdk-8u66-linux-x64.tar.gz"
3. wget https://dl.bintray.com/sbt/native-packages/sbt/0.13.9/sbt-0.13.9.tgz
4. wget http://ftp.jaist.ac.jp/pub/apache/maven/maven-3/3.3.9/binaries/apache-maven-3.3.9-bin.tar.gz
5. wget http://downloads.typesafe.com/scala/2.11.7/scala-2.11.7.tgz
6. wget http://d3kbcqa49mib13.cloudfront.net/spark-1.5.2-bin-hadoop2.6.tgz
7. wget http://ftp.riken.jp/net/apache/hadoop/common/hadoop-2.6.2/hadoop-2.6.2.tar.gz
8. wget http://archive.cloudera.com/cdh5/cdh/5/hadoop-2.6.0-cdh5.5.1.tar.gz
9. wget https://archive.apache.org/dist/spark/spark-2.1.0/spark-2.1.0-bin-hadoop2.7.tgz
```

## unzip

```
1. tar -zxvf jdk-7u79-linux-x64.tar.gz
2. tar -zxvf jdk-8u66-linux-x64.tar.gz
3. tar -zxvf sbt-0.13.9.tgz
4. tar -zxvf apache-maven-3.3.9-bin.tar.gz
5. tar -zxvf scala-2.11.7.tgz
6. tar -zxvf spark-1.5.2-bin-hadoop2.6.tgz
7. tar -zxvf hadoop-2.6.2.tar.gz
8. tar -zxvf hadoop-2.6.0-cdh5.5.1.tar.gz
```

## Spark Standalone

modify profile

```
1. vim ~/.bash_profile
2.
3. export SOFT_BASE_PATH=/app/soft
4. # Spark Standalone
5. export SPARK_BASE_PATH=/app/spark-standalone
6. export JAVA_HOME=$SOFT_BASE_PATH/jdk1.8.0_66
7. export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
8. export SCALA_HOME=$SOFT_BASE_PATH/scala-2.11.7
9. export SPARK_HOME=$SPARK_BASE_PATH/spark-1.6.0-bin-hadoop2.6
10. export HADOOP_HOME=$SOFT_BASE_PATH/hadoop-2.7.1
11. export HADOOP_CONF_DIR=$SOFT_BASE_PATH/hadoop-2.7.1/etc/hadoop
12. export PATH=$PATH:$JAVA_HOME/bin:$SCALA_HOME/bin:$SPARK_HOME/bin:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

## load profile

```
1. source ~/.bash_profile
```

## 确认java , scala环境

```
1. java -version
2. scala -version
```

## 配置文件spark-env.sh

```
1. cp $SPARK_HOME/conf/spark-env.sh.template $SPARK_HOME/conf/spark-env.sh
2. vim $SPARK_HOME/conf/spark-env.sh
```

## 添加

```
1. export SCALA_HOME=/app/soft/scala-2.11.7
2. export SPARK_MASTER_IP=spark-centos-01
3. export SPARK_WORKER_MEMORY=2G
4. export JAVA_HOME=/app/soft/jdk1.8.0_66
5. # export HIVE_HOME=/app/soft/apache-hive-1.2.1-bin
6. # export SPARK_CLASSPATH=$HIVE_HOME/lib/mysql-connector-java-5.1.15-bin.jar:$SPARK_CLASSPATH
```

## 配置文件slaves

```
1. cp $SPARK_HOME/conf/slaves.template $SPARK_HOME/conf/slaves
2. vim $SPARK_HOME/conf/slaves
```

## 在slaves最后添加下面

```
1. spark-centos-01
2. spark-centos-02
3. spark-centos-03
```

使用scp命令，将配置修改后的spark代码发送到其他节点（spark-centos-02、spark-centos-03）

```
1. scp -r /app/spark-standalone root@spark-centos-02:/app/
2. scp -r /app/spark-standalone root@spark-centos-03:/app/
```

## 启动停止命令

```
1. # 启动全部节点
2. $SPARK_HOME/sbin/start-all.sh
3. # 启动master
4. $SPARK_HOME/sbin/start-master.sh
5. # 启动worker
6. $SPARK_HOME/sbin/start-slaves.sh
7. # 停止全部节点
8. $SPARK_HOME/sbin/stop-all.sh
```

## 启动后的截图

master节点：<http://192.168.101.141:8080/>

slave-01节点：<http://192.168.101.141:8081/>

slave-02节点：<http://192.168.101.142:8081/>

slave-03节点：<http://192.168.101.143:8081/>

Spark Master at spark://spark-centos-01:7077

URL: spark://spark-centos-01:7077  
REST URL: spark://spark-centos-01:6066 (cluster mode)  
Alive Workers: 3  
Cores in use: 3 Total, 0 Used  
Memory in use: 6.0 GB Total, 0.0 B Used  
Applications: 0 Running, 0 Completed  
Drivers: 0 Running, 0 Completed  
Status: ALIVE

Worker Id	Address	State	Cores	Memory
worker-20160221082208-192.168.101.141-42277	192.168.101.141:42277	ALIVE	1 (0 Used)	2.0 GB (0.0 B Used)
worker-20160221082208-192.168.101.142-36133	192.168.101.142:36133	ALIVE	1 (0 Used)	2.0 GB (0.0 B Used)
worker-20160221082208-192.168.101.143-56260	192.168.101.143:56260	ALIVE	1 (0 Used)	2.0 GB (0.0 B Used)

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
----------------	------	-------	-----------------	----------------	------	-------	----------

Completed Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
----------------	------	-------	-----------------	----------------	------	-------	----------

Spark Master at spark://spa...Spark Worker at 192.168.10...Spark Worker at 192.168.10...Spark Worker at 192.168.10...+

192.168.101.141:8081

Spark1.6.0

Spark Worker at 192.168.101.141:42277

ID: worker-20160221082208-192.168.101.141-42277

Master URL: spark://spark-centos-01:7077

Cores: 1 (0 Used)

Memory: 2.0 GB (0.0 B Used)

[Back to Master](#)

Running Executors (0)

ExecutorID	Cores	State	Memory	Job Details	Logs
------------	-------	-------	--------	-------------	------

Spark Master at spark://spa...Spark Worker at 192.168.10...Spark Worker at 192.168.10...Spark Worker at 192.168.10...+

192.168.101.142:8081

Spark1.6.0

Spark Worker at 192.168.101.142:36133

ID: worker-20160221082208-192.168.101.142-36133

Master URL: spark://spark-centos-01:7077

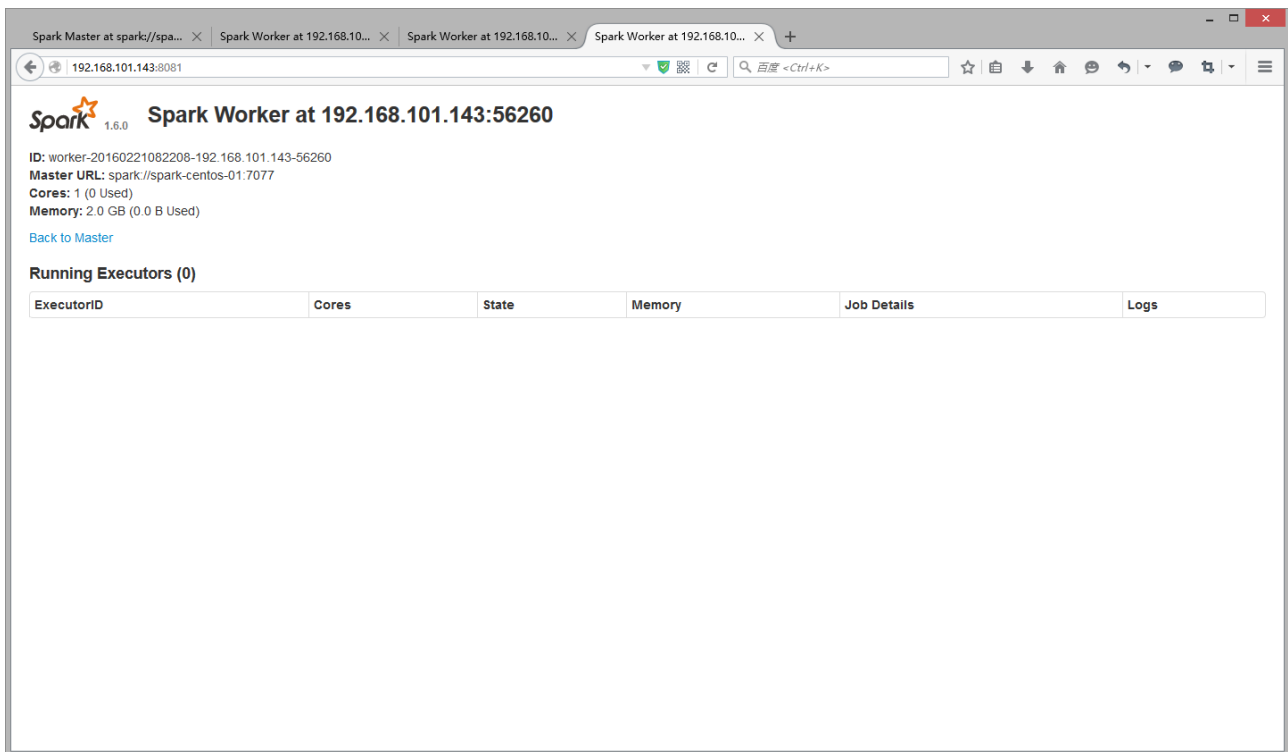
Cores: 1 (0 Used)

Memory: 2.0 GB (0.0 B Used)

[Back to Master](#)

Running Executors (0)

ExecutorID	Cores	State	Memory	Job Details	Logs
------------	-------	-------	--------	-------------	------



## spark-shell

1. `$SPARK_HOME/bin/spark-shell --master spark://spark-centos-01:7077`
2. `# -Dspark.master=spark://spark-centos-01:7077`
3. `# -Dspark.master=local`

## HelloWorld

1. `# test localhost file`
2. `scala > val textFile = sc.textFile("file:///app/spark-standalone/spark-1.6.0-bin-hadoop2.6/README.md")`
3. `scala > textFile.count()`
4. `# test hdfs file`
5. `scala > val textFile = sc.textFile("hdfs://hadoop-centos-01:9000/input/README.txt")`
6. `scala > textFile.count()`

## spark-submit

1. `$SPARK_HOME/bin/spark-submit --master spark://spark-centos-01:7077 --class org.apache.spark.examples.SparkPi --executor-memory 2g --total-executor-cores 2 lib/spark-examples-1.6.0-hadoop2.6.0.jar 1000`

## 启动后的截图

master节点：<http://192.168.101.141:8080/>

slave-01节点：<http://192.168.101.141:8081/>

slave-02节点：<http://192.168.101.142:8081/>

slave-03节点：<http://192.168.101.143:8081/>

启动多个spark shell后，监控界面端口4040,4041自动依次递增（第二个spark shell启动的时

候会出现端口绑定错误 )

spark shell job : <http://192.168.101.141:4040/jobs/>

Spark Master at spark...Application: Spark shellSpark shell - Spark JobsSpark shell - Details f...Spark Worker at 192...Spark Worker at 192...Spark Worker at 192...

192.168.101.141:8080

Spark 1.6.0

Spark Master at spark://spark-centos-01:7077

URL: spark://spark-centos-01:7077  
REST URL: spark://spark-centos-01:6066 (cluster mode)  
Alive Workers: 3  
Cores in use: 3 Total, 3 Used  
Memory in use: 6.0 GB Total, 3.0 GB Used  
Applications: 1 Running, 0 Completed  
Drivers: 0 Running, 0 Completed  
Status: ALIVE

Workers

Worker Id	Address	State	Cores	Memory
<a href="#">worker-20160221082208-192.168.101.141-42277</a>	192.168.101.141:42277	ALIVE	1 (1 Used)	2.0 GB (1024.0 MB Used)
<a href="#">worker-20160221082208-192.168.101.142-36133</a>	192.168.101.142:36133	ALIVE	1 (1 Used)	2.0 GB (1024.0 MB Used)
<a href="#">worker-20160221082208-192.168.101.143-56260</a>	192.168.101.143:56260	ALIVE	1 (1 Used)	2.0 GB (1024.0 MB Used)

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
<a href="#">app-20160221083026-0000</a>	(kill) <a href="#">Spark shell</a>	3	1024.0 MB	2016/02/21 08:30:26	root	RUNNING	41 min

Completed Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
----------------	------	-------	-----------------	----------------	------	-------	----------

Spark Master at spark...Application: Spark shellSpark shell - Spark JobsSpark shell - Details f...Spark Worker at 192...Spark Worker at 192...Spark Worker at 192...

192.168.101.141:8080/app/?appId=app-20160221083026-0000

Spark 1.6.0

Application: Spark shell

ID: app-20160221083026-0000  
Name: Spark shell  
User: root  
Cores: Unlimited (3 granted)  
Executor Memory: 1024.0 MB  
Submit Date: Sun Feb 21 08:30:26 JST 2016  
State: RUNNING  
[Application Detail UI](#)

Executor Summary

ExecutorID	Worker	Cores	Memory	State	Logs
2	<a href="#">worker-20160221082208-192.168.101.142-36133</a>	1	1024	RUNNING	<a href="#">stdout stderr</a>
1	<a href="#">worker-20160221082208-192.168.101.143-56260</a>	1	1024	RUNNING	<a href="#">stdout stderr</a>
0	<a href="#">worker-20160221082208-192.168.101.141-42277</a>	1	1024	RUNNING	<a href="#">stdout stderr</a>

Spark Master at spark...

Application: Spark shell

Spark shell - Spark Jobs

Spark shell - Details f...

Spark Worker at 192...

Spark Worker at 192...

Spark Worker at 192...

192.168.101.141:4040/jobs/

百度 <Ctrl+K>

Spark 1.6.0

Jobs

Stages

Storage

Environment

Executors

SQL

Spark shell application UI

## Spark Jobs (?)

Total Uptime: 42 min  
Scheduling Mode: FIFO  
Completed Jobs: 1

▶ [Event Timeline](#)

### Completed Jobs (1)

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
0	count at <console>-30	2016/02/21 09:11:21	15 s	1/1	2/2

The screenshot displays the Spark shell application UI in a web browser. The browser's address bar shows the URL `192.168.101.141:4040/jobs/job/?id=0`. The application's navigation bar includes tabs for `Jobs`, `Stages`, `Storage`, `Environment`, `Executors`, and `SQL`. The `Jobs` tab is active, showing the details for Job 0.

**Details for Job 0**

**Status:** SUCCEEDED  
**Completed Stages:** 1

- Event Timeline
- DAG Visualization

**Completed Stages (1)**

Stage Id	Description	Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
0	count at <console>:30 <a href="#">+details</a>	2016/02/21 09:11:25	11 s	2/2	4.9 KB			