1 开发环境

Ubuntu16.04

Java 1.8.0_162

2 实验原理

本次实验使用在 Mesos 集群环境下的 Spark 计算工具，借用 Zeppelin 开源框架，在 Web 上实现数据可视化的 SQL 查询平台。

2.1 Apache Mesos 简介

Apache Mesos 是一个通用的集群管理器，起源于 Google 的数据中心资源管理系统 Borg。Mesos master 负责决定分配给每个框架多少资源，任务调度器负责如何使用这些资源，这取决于每个框架的调度器如何根据自身需求去实现的负载提供服务。Mesos 的设计原则是：资源分配和任务调度的分离。Mesos 的总体框架如图 2.1 所示。

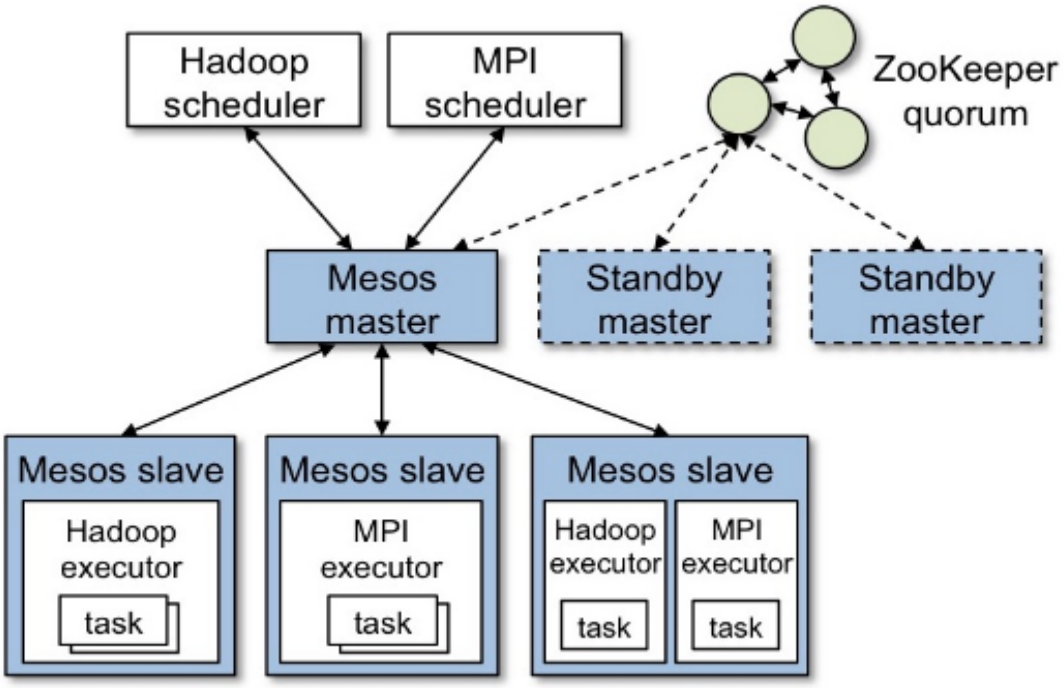


图 2.1 Mesos 总体框架

总体上看，Mesos 是一个 master/slave 结构，其中 master 是非常轻量级的，仅保存了 framework 和 mesos slave 的一些状态，而这些状态很容易通过 framework 和 slave 重新注册而重构，因而很容易使用 zookeeper 解决 mesos master 的单点故障问题。

Mesos master 实际上是一个全局资源调度器，采用某种策略将某个 slave 上的空闲资源分配给某一个 framework，各种 framework 通过自己的调度器向 Mesos master 注册，以接入到 Mesos 中。而 Mesos slave 主要功能是汇报任务的状态和启动各个 framework 的 executor。

2.2 Spark 简介

Spark 是一个围绕速度、易用性和复杂分析构建的大数据处理框架。Spark 通过管理多台计算机形成集群，以解决大规模数据计算或复杂数据计算问题，其架构如图 2.2 所示。

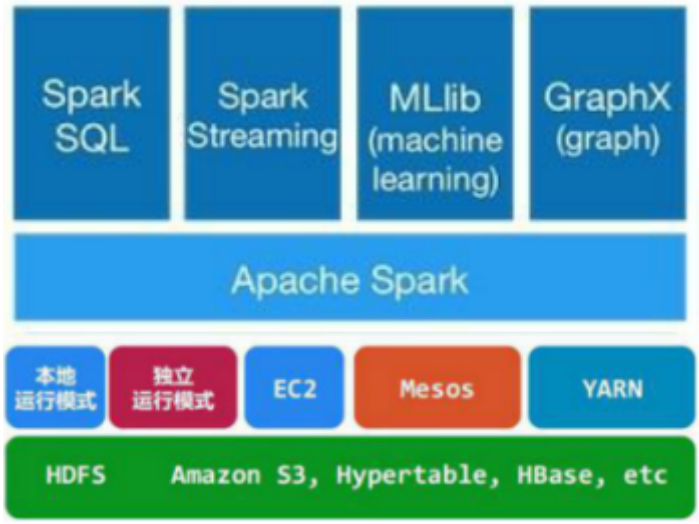


图 2.2 Spark 架构图

其中的 Spark SQL 提供通过 Apache Hive 的 SQL 变体 Hive 查询语言（HiveQL）与 Spark 进行交互的 API。每个数据库表被当做一个 RDD，Spark SQL 查询被转换为 Spark 操作。

Spark 本身并没有提供分布式文件系统，因此 Spark 的分析大多依赖于 Hadoop 的分布式文件系统 HDFS。Hadoop 的 Mapreduce 与 Spark 都可以进行数

据计算，而相比于 Mapreduce，Spark 的速度更快并且提供的功能更加丰富。关系图如图 2.3 所示。

图 2.3 Spark、Hadoop 关系图

Spark 支持使用 Scala、Java、Python 和 R 语言进行编程。由于 Spark 采用 Scala 语言进行开发，因此，建议采用 Scala 语言进行 Spark 应用程序的编写。

2.3 Zeppelin 简介

Apache Zeppelin 是一个让交互式数据分析变得可行的基于网页的开源框架，Zeppelin 提供了数据分析、数据可视化等功能。

Zeppelin 是一个提供交互数据分析且基于 Web 的笔记本。方便你做出可数据驱动的、可交互且可协作的精美文档，并且支持多种语言，包括 Scala(使用 Apache Spark)、Python(Apache Spark)、SparkSQL、 Hive、 Markdown、Shell 等等。

Zeppelin 中最核心的概念是 Interpreter，Interpreter 是一个插件，它允许用户使用一个指定的语言或数据处理器。每一个 Interpreter 都属于换一个 InterpreterGroup，同一个 InterpreterGroup 的 Interpreters 可以相互引用，例如 SparkSqlInterpreter 可以引用 SparkInterpreter 以获取 SparkContext，因为它们属于同一个 InterpreterGroup。当前已经实现的 Interpreter 有 Spark 解释器，Python 解释器，SparkSQL 解释器,JDBC，Markdown 和 shell 等。图为 Interpreter 的原理图如图 2.4 所示。

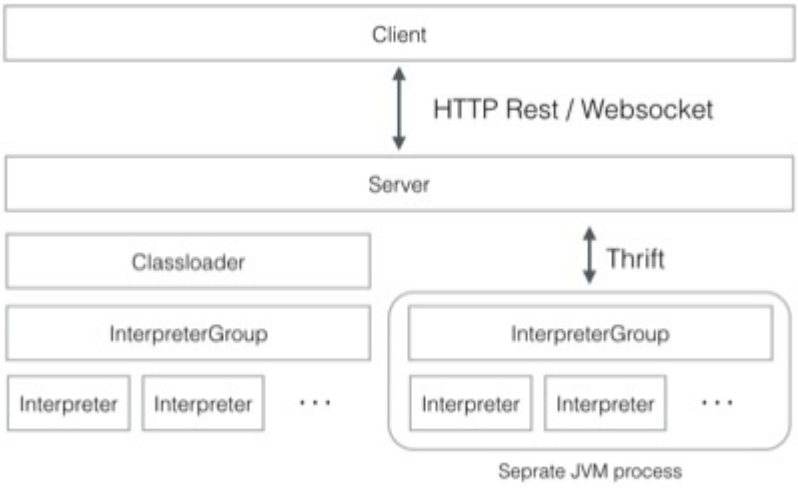


图 2.4 Zeppelin 原理图

3 配置及安装过程

3.1 Apache Mesos 安装与配置

3.1.1 下载源文件及依赖

从 Apache Mesos 下载实验环境所需的 Mesos 源文件及其依赖文件

$ wget http://www.apache.org/dist/mesos/1.4.1/mesos-1.4.1.tar.gz

$ tar -zxf mesos-1.4.1.tar.gz

如图 3.1 所示。

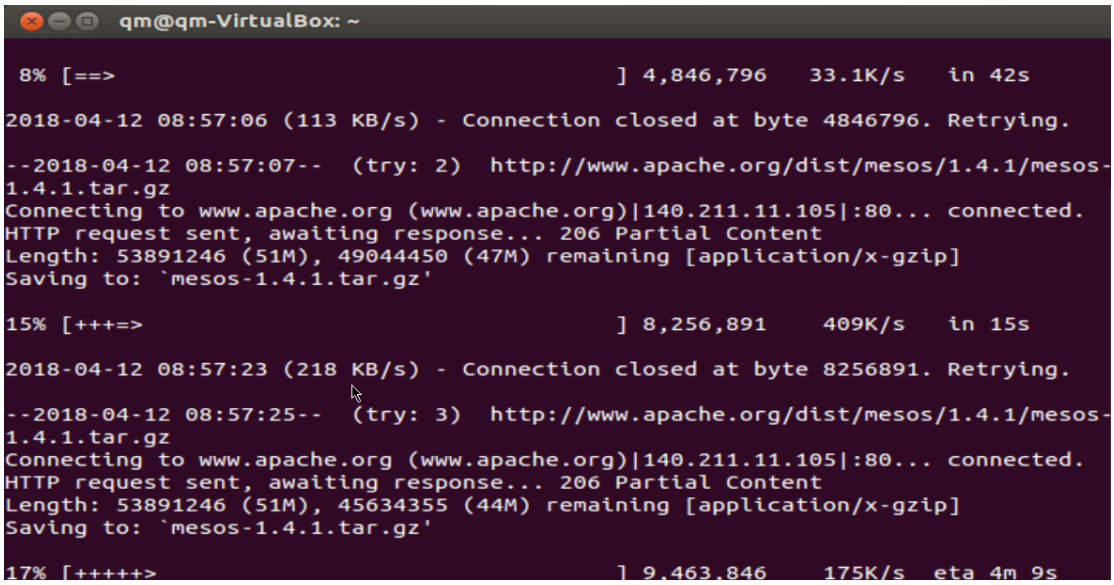


图 3.1 下载 Mesos 图

$ sudo apt-get update

$ sudo apt-get install -y tar wget git

$ sudo apt-get install -y openjdk-8-jdk

$ sudo apt-get install -y autoconf libtool

$ sudo apt-get -y install build-essential python-dev python-six python-virtualenv libcurl4-nss-dev libsasl2-dev libsasl2-modules maven libapr1-dev libsvn-dev zlib1g-dev iputils-ping

如图 3.2 所示。

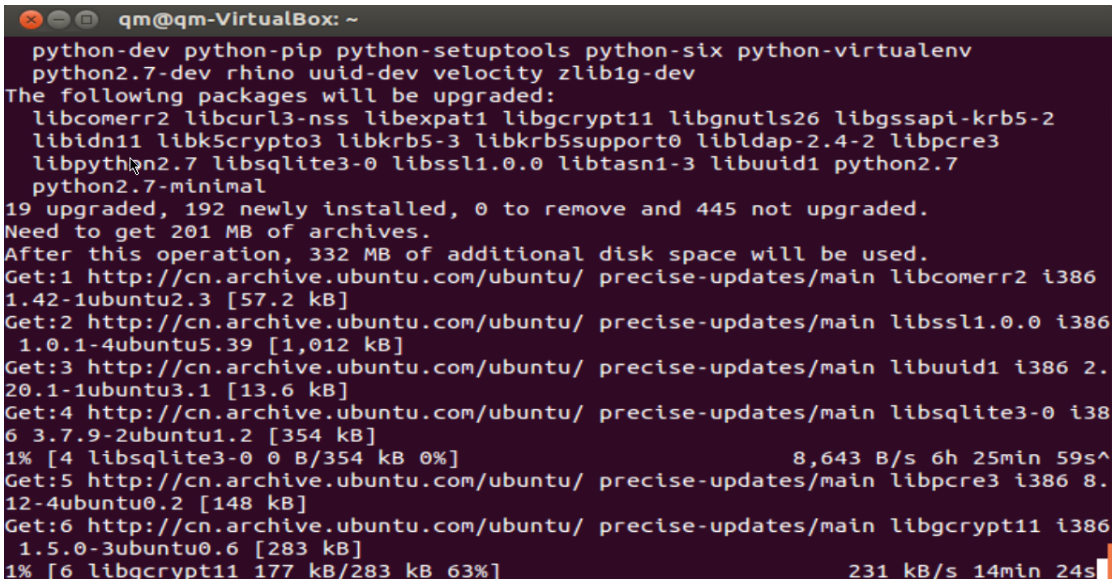


图 3.2 下载 Mesos 依赖文件

3.1.2 编译 Mesos 环境

创建 Mesos 所需要的目录，并进行配置检查和编译

$ cd mesos

$ mkdir build

$ cd build

$ ../configure

$ make

$ make install

如图 3.3 所示。

```
Setting up libjsch-java (0.1.53-1ubuntu1) ...
Setting up libjsoup-java (1.8.3-1) ...
Setting up libjsr305-java (0.1~+svn49-9) ...
Setting up libguava-java (19.0-1) ...
Setting up libplexus-classworlds-java (1.5.0-5) ...
Setting up libxbean-java (4.5-1) ...
Setting up libplexus-containers-java (1.0~beta3.0.7-8) ...
Setting up libplexus-interactivity-api-java (1.0-alpha-6-7) ...
Setting up libwagon-java (1.0.0-7) ...
Setting up libeclipse-aether-java (1.0.2-1) ...
Setting up libaopalliance-java (20070526-5) ...
Setting up libguice-java (4.0-2) ...
Setting up libjdom1-java (1.1.3-1) ...
Setting up libservlet2.5-java (6.0.45+dfsg-1) ...
Setting up libjetty-java (6.1.26-5) ...
Setting up liblog4j1.2-java (1.2.17-7ubuntu1) ...
Setting up libmaven-parent-java (21-2) ...
Setting up libplexus-container-default-java (1.0-alpha-9-stable-1-7) ...
Setting up libplexus-ant-factory-java (1.0~alpha2.1-3) ...
Setting up libplexus-io-java (1.0~alpha5-2) ...
Setting up libplexus-archiver-java (2.2-1) ...
Setting up libplexus-bsh-factory-java (1.0~alpha7-3.1) ...
Setting up libplexus-cipher-java (1.7-1) ...
Setting up libplexus-sec-dispatcher-java (1.3.1-6) ...
Setting up libmaven2-core-java (2.2.1-24) ...
Setting up libplexus-classworlds2-java (2.5.2-1) ...
Setting up libplexus-utils2-java (3.0.22-1) ...
Setting up libplexus-container-default1.5-java (1.6-2) ...
Setting up libplexus-component-annotations-java (1.6-2) ...
Setting up libplexus-cli-java (1.2-5) ...
Setting up libqdox2-java (2.0~M3-2) ...
Setting up libplexus-component-metadata-java (1.6-2) ...
Setting up libplexus-containers1.5-java (1.6-2) ...
Setting up libsisu-inject-java (0.3.2-1) ...
Setting up libsisu-plexus-java (0.3.2-1) ...
Setting up libwagon2-java (2.10-3) ...
Setting up libmaven3-core-java (3.3.9-3) ...
Setting up libpython2.7-dev:amd64 (2.7.12-1ubuntu0~16.04.3) ...
Setting up libpython-dev:amd64 (2.7.12-1~16.04) ...
Setting up libsasl2-dev (2.1.26.dfsg1-14build1) ...
Setting up libsvn1:amd64 (1.9.3-2ubuntu1.1) ...
Setting up libxom-java (1.2.10-1) ...
Setting up maven (3.3.9-3) ...
update-alternatives: using /usr/share/maven/bin/mvn to provide /usr/bin/mvn (mvn) in auto mode
Setting up python2.7-dev (2.7.12-1ubuntu0~16.04.3) ...
Setting up python-dev (2.7.12-1~16.04) ...
Setting up python-pip-whl (8.1.1-2ubuntu0.4) ...
Setting up python-pkg-resources (20.7.0-1) ...
Setting up python-six (1.10.0-3) ...
Setting up python-virtualenv (15.0.1+ds-3ubuntu1) ...
Setting up python3-virtualenv (15.0.1+ds-3ubuntu1) ...
Setting up virtualenv (15.0.1+ds-3ubuntu1) ...
Setting up zlib1g-dev:amd64 (1:1.2.8.dfsg-2ubuntu4.1) ...
Setting up libsvn-dev (1.9.3-2ubuntu1.1) ...
Processing triggers for libc-bin (2.23-0ubuntu3) ...
qm@qm-VirtualBox:~$
```

图 3.3 下载 Mesos 图

3.2 Hadoop 安装与配置

3.2.1 安装 SSH 并配置 SSH 无线密码

集群、单节点模式都需要用到 SSH 登陆（类似于远程登陆，你可以登录某台 Linux 主机，并且在上面运行命令）

执行如下命令测试一下 SSH 是否可用:

ssh -X localhost

配置无密码登录：

exit

cd ~/.ssh/

ssh-keygen -t rsa

cat id_rsa.pub >> authorized_keys

chmod 600 ./authorized_keys

3.2.2 安装 Hadoop

sudo tar -zxf ~/Downloads/hadoop-2.6.0.tar.gz -C /usr/local

cd /usr/local/

sudo mv ./hadoop-2.6.0/ ./hadoop

sudo chown -R hadoop:hadoop ./hadoop

cd /usr/local/hadoop

./bin/hadoop version

3.2.3 配置环境变量

vim ~/.bashrc

export HADOOP_HOME=/usr/local/hadoop

export HADOOP_INSTALL=$HADOOP_HOME

export HADOOP_MAPRED_HOME=$HADOOP_HOME

export HADOOP_COMMON_HOME=$HADOOP_HOME

export HADOOP_HDFS_HOME=$HADOOP_HOME

export YARN_HOME=$HADOOP_HOME

export

HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native

export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin

source ~/.bashrc

3.3 Spark 安装与配置

3.3.1 下载及安装

sudo tar Downloads/Spark-2.2.0-bin-without-hadoop.tgz -C /usr/local/

cd /usr/local

sudo mv ./Spark-2.2.0-bin-without-hadoop/ ./Spark

sudo chown -R hadoop:hadoop ./Spark

如图 3.4 所示。

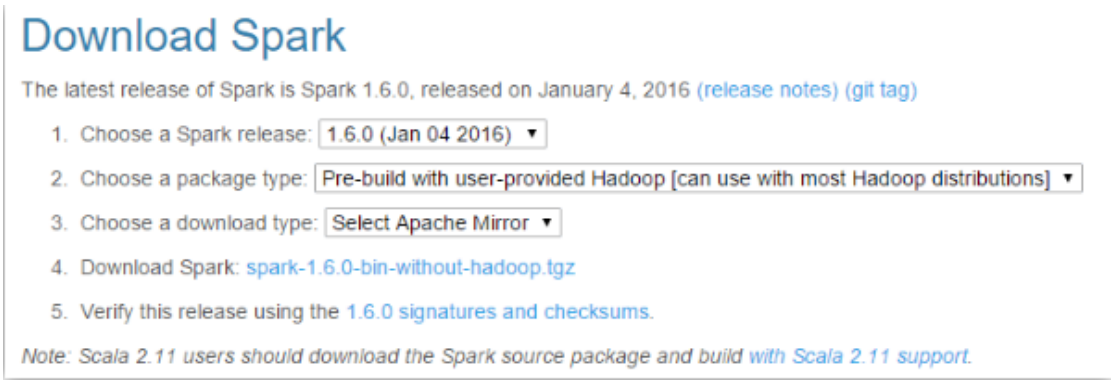


图 3.4 下载 Spark

3.3.2 文件配置并启动 Spark shell

cd /usr/local/Spark

cp ./conf/Spark-env.sh.template ./conf/Spark-env.sh

vim ./conf/Spark-env.sh

export SPARK_DIST_CLASSPATH=$(/usr/local/hadoop/bin/hadoop classpath)

./bin/Spark-shell

3.3.3 Spark 与 Mesos 连接所需配置

cd /usr/local/Spark/conf

cat Spark-defaults.conf.template > Spark-defaults.conf

vim Spark-defaults.conf

修改 Spark-defaults.conf 内容，添加

Spark.io.compression.codec lzf

Cat Spark-env.sh.template > Spark-env.sh

vim Spark-env.sh

修改 Spark-env.sh 内容，添加

export MESOS_NATIVE_JAVA_LIBRARY=/usr/local/mesos/mesos/lib/libmesos.so

export SPARK_EXECUTOR_URI=/usr/local/Spark/Spark.tar.gz

之后就可以启动 mesos 集群了

cd /usr/local/Spark/sbin

./start-mesos-dispatcher.sh –master mesos://127.0.0.1:5050

这个时候可以在 Spark-shell 里面尝试简单的任务，观察 mesos 管理网页的反应，来确定 Spark 与 mesos 的连接情况。

cd /usr/local/Spark/Spark-2.0.1-bin-hadoop2.7/bin

./Spark-shell --master mesos://127.0.0.1:5050

scala> val a = sc.parallelize(2 to 1000)

scala> a.collect

如图 3.5 所示。

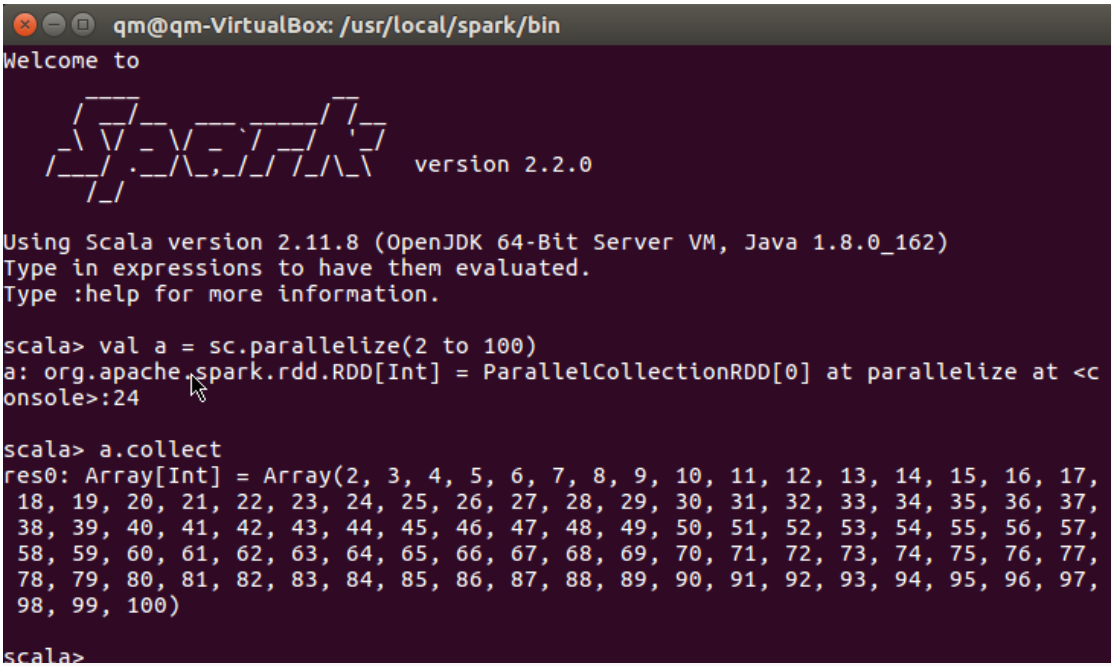


图 3.5 测试 Spark

Mesos 管理界面，如图 3.6 所示。

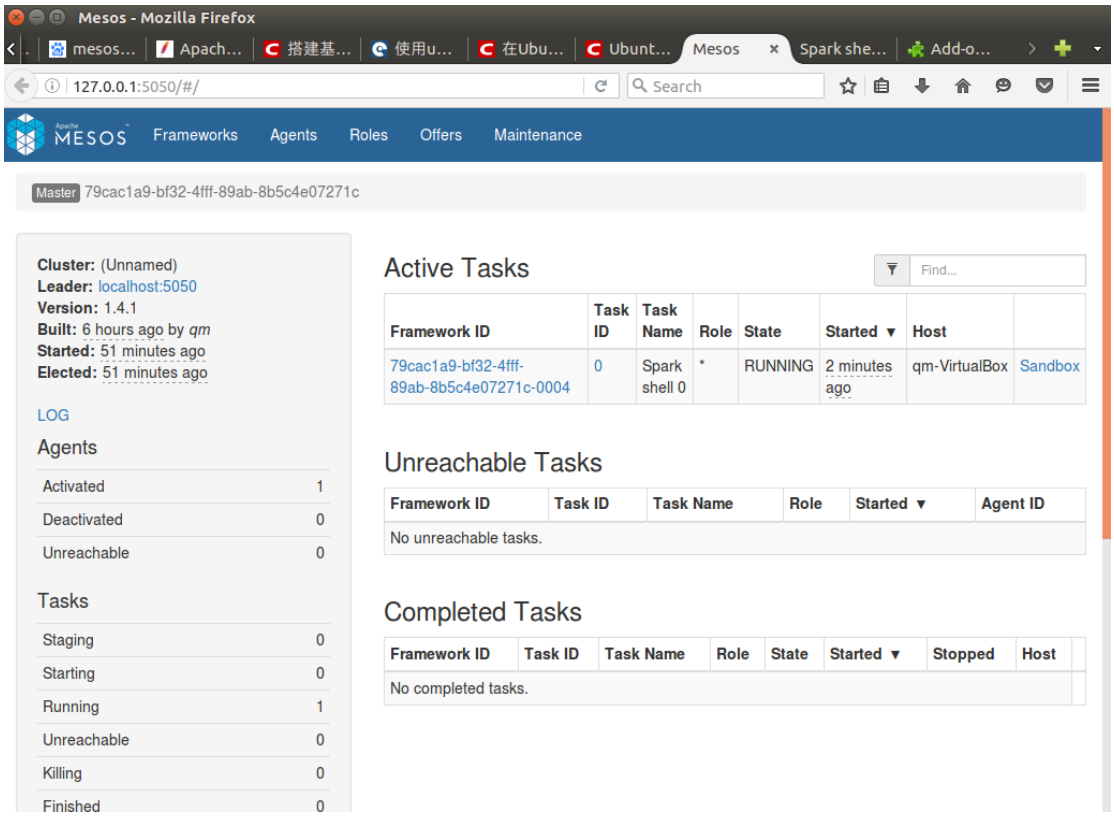


图 3.6 Mesos 管理界面

Spark 管理界面，如图 3.7 所示。

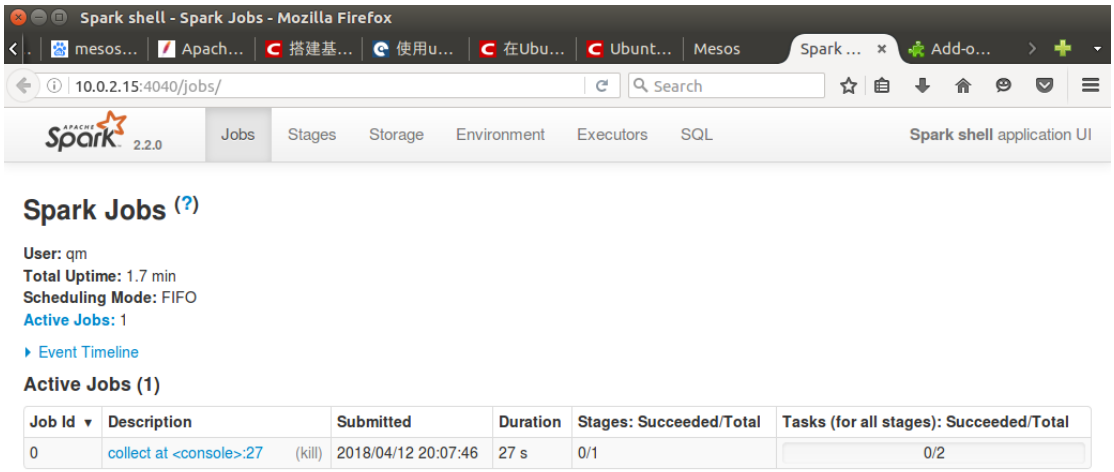


图 3.7 Spark 管理界面

3.4 Zeppelin 安装与配置

3.4.1 下载安装

在官网上下载并解压到对应文件夹。

wget http://mirrors.shu.edu.cn/apache/zeppelin/zeppelin-0.7.3/zeppelin-0.7.3-bin-netinst.tgz

如图 3.8 所示。

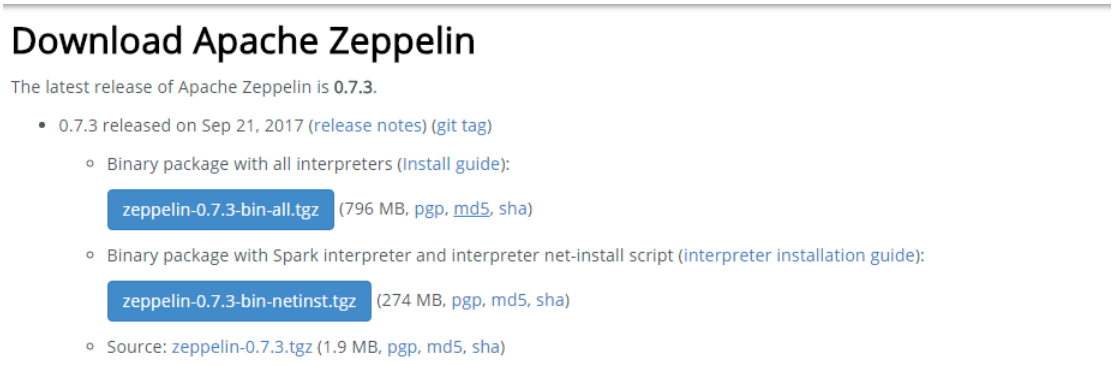


图 3. 8 下载 Zeppelin

sudo tar -zxf ~/Downloads/zeppelin-0.7.3-bin-netinst.tgz -C /usr/local

cd /usr/local/

sudo mv ./zeppelin-0.7.3-bin-netinst/ ./zeppelin

3.4.2 配置

配置环境变量

cd /usr/local/zeppelin/conf

vim zeppelin-env.sh

如图 3.9、图 3.10 所示。

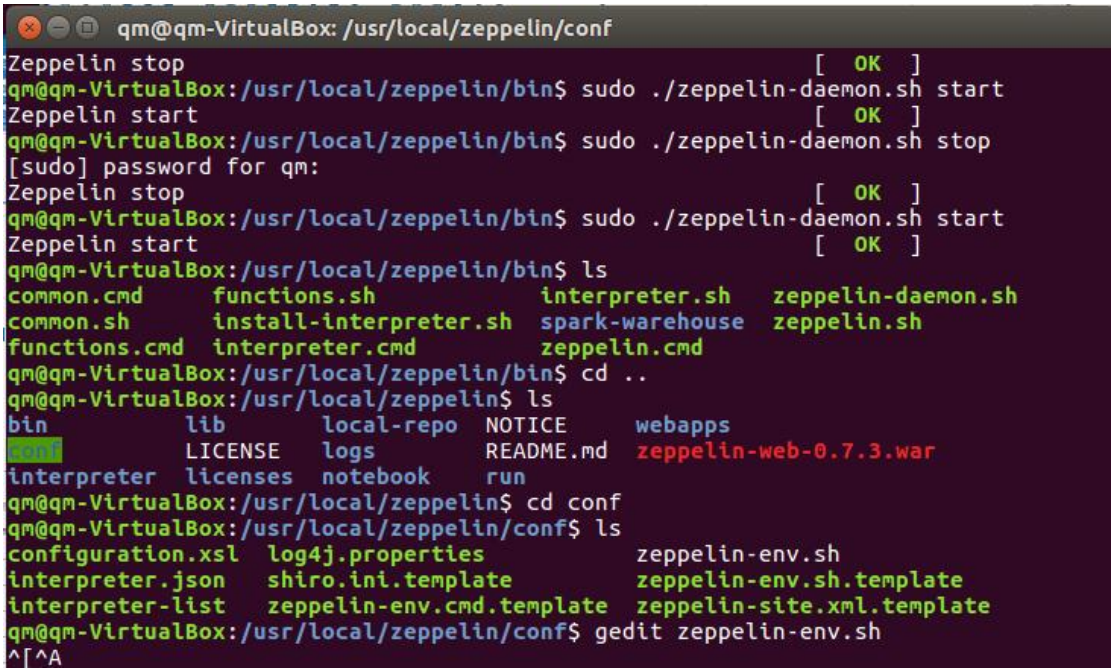


图 3. 9 Zeppelin 配置图

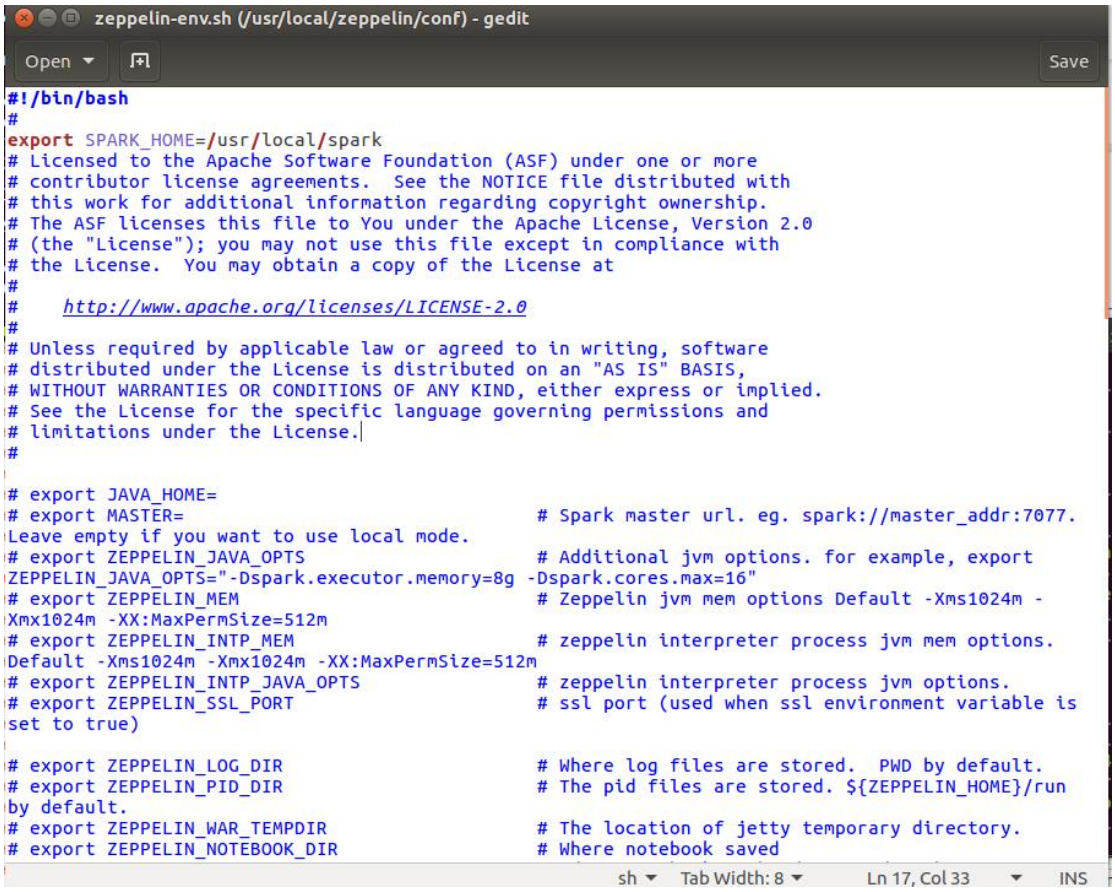


图 3. 10 Zeppelin 配置图

3.4.3 与 Spark-Mesos 集群连接

对 Zeppelin 进行配置

cd /usr/local/zeppelin/bin

./zeppelin-daemon.sh start

在浏览器中输入：localhost:8080/#/ ，显示如下图所示，配置成功。

如图 3.11 所示。

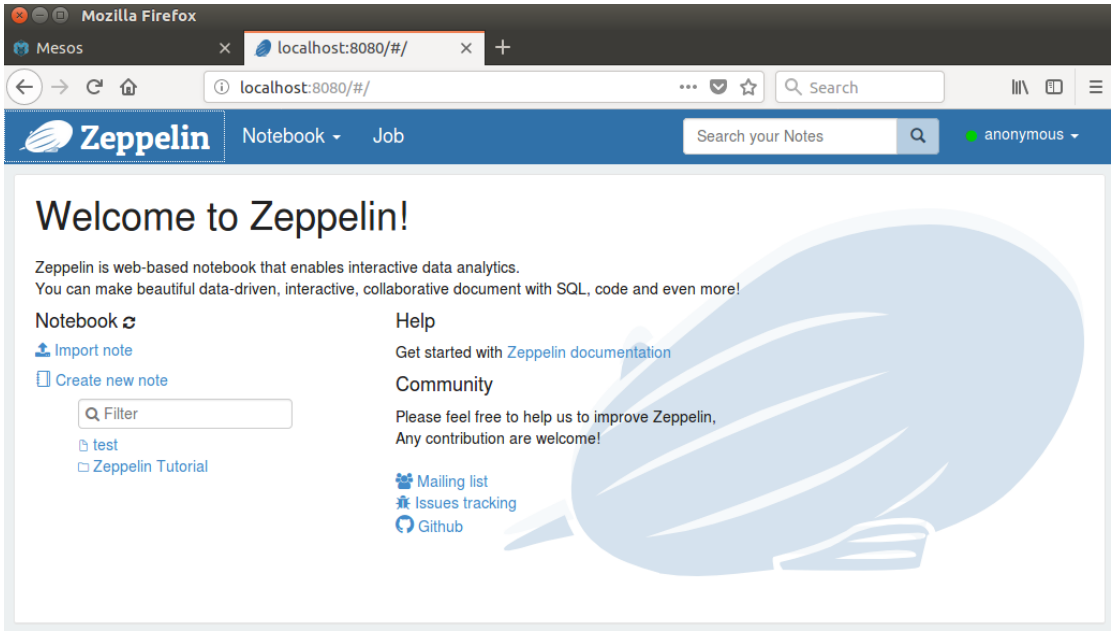


图 3. 11 测试 Zeppelin

首先配置使用的 spart 解释器的若干属性，包括 master ip，资源数量等，如图 3.12.

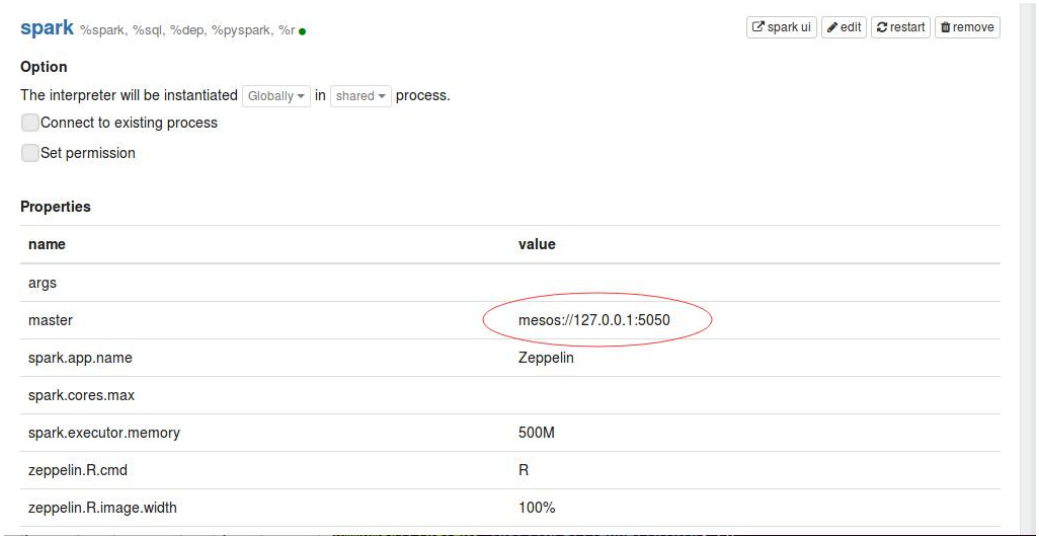


图 3. 12 配置 Spark 解释器若干属性

新建 web 笔记，进行与 Spark 的连接，如图 3.13 所示。

图 3. 13 新建 web 笔记

可以看到正确输出了 Spark 的版本，说明已经成功连接了 Spark。

在 Mesos 管理界面中找到 Framework，如图 3.14 所示。

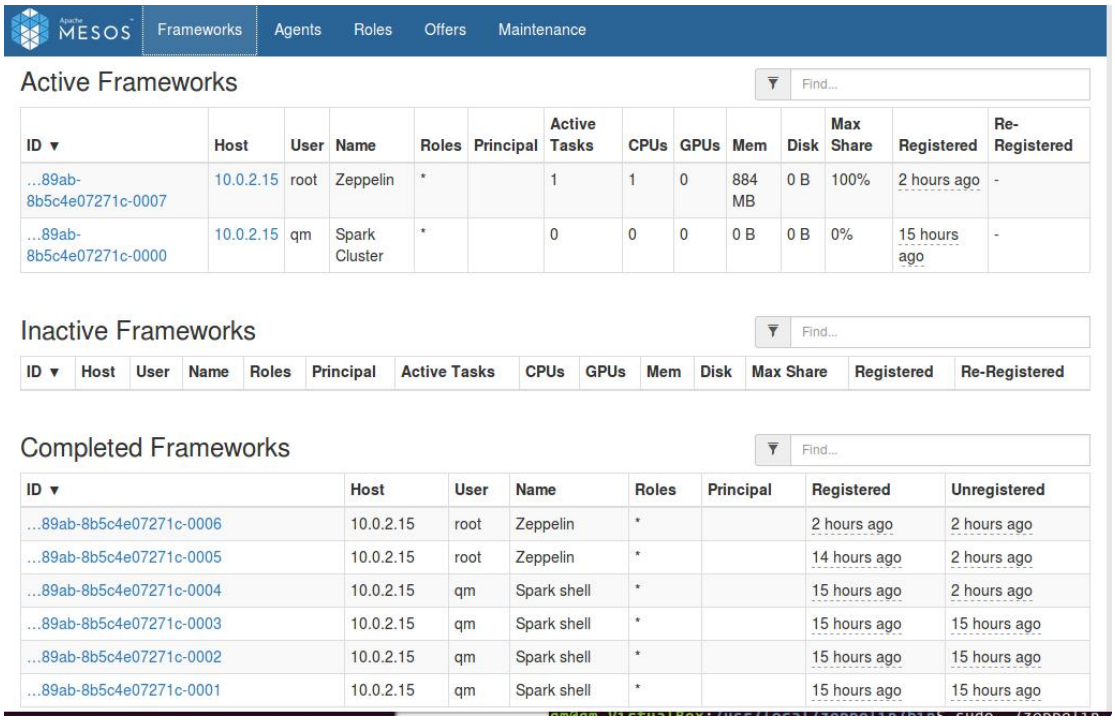


图 3. 14 Mesos 管理界面

4 可视化 SQL 查询实例

首先使用本地文件，加载 bank 数据表，本地数据表的内容如图 4.1 所示。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	y
2	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct	79	1	-1	0	unknown	no
3	33	services	married	secondary	no	4789	yes	yes	cellular	11	may	220	1	339	4	failure	no
4	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr	185	1	330	1	failure	no
5	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun	199	4	-1	0	unknown	no
6	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may	226	1	-1	0	unknown	no
7	35	management	single	tertiary	no	747	no	no	cellular	23	feb	141	2	176	3	failure	no
8	36	self-employed	married	tertiary	no	307	yes	no	cellular	14	may	341	1	330	2	other	no
9	39	technician	married	secondary	no	147	yes	no	cellular	6	may	151	2	-1	0	unknown	no
10	41	entrepreneur	married	tertiary	no	221	yes	no	unknown	14	may	57	2	-1	0	unknown	no
11	43	services	married	primary	no	-88	yes	yes	cellular	17	apr	313	1	147	2	failure	no
12	39	services	married	secondary	no	9374	yes	no	unknown	20	may	273	1	-1	0	unknown	no
13	43	admin.	married	secondary	no	264	yes	no	cellular	17	apr	113	2	-1	0	unknown	no
14	36	technician	married	tertiary	no	1109	no	no	cellular	13	aug	328	2	-1	0	unknown	no
15	20	student	single	secondary	no	502	no	no	cellular	30	apr	261	1	-1	0	unknown	yes
16	31	blue-collar	married	secondary	no	360	yes	yes	cellular	29	jan	89	1	241	1	failure	no
17	40	management	married	tertiary	no	194	no	yes	cellular	29	aug	189	2	-1	0	unknown	no
18	56	technician	married	secondary	no	4073	no	no	cellular	27	aug	239	5	-1	0	unknown	no
19	37	admin.	single	tertiary	no	2317	yes	no	cellular	20	apr	114	1	152	2	failure	no
20	25	blue-collar	single	primary	no	-221	yes	no	unknown	23	may	250	1	-1	0	unknown	no
21	31	services	married	secondary	no	132	no	no	cellular	7	jul	148	1	152	1	other	no
22	38	management	divorced	unknown	no	0	yes	no	cellular	18	nov	96	2	-1	0	unknown	no
23	42	management	divorced	tertiary	no	16	no	no	cellular	19	nov	140	3	-1	0	unknown	no
24	44	services	single	secondary	no	106	no	no	unknown	12	jun	109	2	-1	0	unknown	no
25	44	entrepreneur	married	secondary	no	93	no	no	cellular	7	jul	125	2	-1	0	unknown	no
26	26	housemaid	married	tertiary	no	543	no	no	cellular	30	jan	169	3	-1	0	unknown	no
27	41	management	married	tertiary	no	5883	no	no	cellular	20	nov	182	2	-1	0	unknown	no
28	55	blue-collar	married	primary	no	627	yes	no	unknown	5	may	247	1	-1	0	unknown	no
29	67	retired	married	unknown	no	696	no	no	telephone	17	aug	119	1	105	2	failure	no
30	56	self-employed	married	secondary	no	784	no	yes	cellular	30	jul	149	2	-1	0	unknown	no
31	53	admin.	married	secondary	no	105	no	yes	cellular	21	aug	74	2	-1	0	unknown	no
32	68	retired	divorced	secondary	no	4189	no	no	telephone	14	jul	897	2	-1	0	unknown	yes
33	31	technician	married	secondary	no	171	no	no	cellular	27	aug	81	3	-1	0	unknown	no
34	59	management	married	secondary	no	42	no	no	cellular	21	nov	40	1	-1	0	unknown	no
35	32	management	single	tertiary	no	2536	yes	no	cellular	26	aug	958	6	-1	0	unknown	yes
36	49	technician	married	tertiary	no	1235	no	no	cellular	13	aug	354	3	-1	0	unknown	yes
37	42	admin.	divorced	secondary	no	1811	yes	no	unknown	14	may	150	1	-1	0	unknown	no
38	78	retired	divorced	primary	no	229	no	no	telephone	22	oct	97	1	-1	0	unknown	yes
39	32	blue-collar	married	secondary	no	2089	yes	no	cellular	14	nov	132	1	-1	0	unknown	yes
40	33	management	married	secondary	no	3935	yes	no	cellular	6	may	765	1	342	2	failure	yes
41	23	services	single	tertiary	no	363	yes	no	unknown	30	may	16	18	-1	0	unknown	no
42	38	management	single	tertiary	no	11971	yes	no	unknown	17	nov	609	2	101	3	failure	no
43	36	management	single	tertiary	no	553	no	no	cellular	11	aug	106	2	-1	0	unknown	no
44	52	blue-collar	married	secondary	no	1117	yes	no	cellular	13	may	365	1	-1	0	unknown	no
45	32	technician	married	tertiary	no	396	yes	no	cellular	13	may	205	3	-1	0	unknown	no

图 4.1 bank.csv 内容

代码：

val bankText =sc.textFile("file:/home/qm/Downloads/bank.csv")

case class Bank(age:Integer, job:String, marital:String, education:String, balance : Integer)

val bank = bankText.map(s=>s.split(";")).filter(s=>s(0)!="\"age\"").map(s=>Bank(s(0).toInt,

s(1).replaceAll("\"", ""),s(2).replaceAll("\"", ""),

s(3).replaceAll("\"", ""),s(5).replaceAll("\"", "").toInt)).toDF()

bank.registerTempTable("bank")

如图 4.2 所示。

```
import org.apache.commons.io.IOUtils
import java.net.URL
import java.nio.charset.Charset

// Zeppelin creates and injects sc (SparkContext) and sqlContext (HiveContext or SqlContext)
// So you don't need create them manually

// load bank data
val bankText = sc.textFile("file:/home/qm/Downloads/bank.csv")

case class Bank(age: Integer, job: String, marital: String, education: String, balance: Integer)

val bank = bankText.map(s => s.split(";")).filter(s => s(0) != "\"age\"").map(
    s => Bank(s(0).toInt,
            s(1).replaceAll("\"", ""),
            s(2).replaceAll("\"", ""),
            s(3).replaceAll("\"", ""),
            s(5).replaceAll("\"", "").toInt
        )
).toDF()
bank.registerTempTable("bank")
```

FINISHED

```
import org.apache.commons.io.IOUtils
import java.net.URL
import java.nio.charset.Charset
bankText: org.apache.spark.rdd.RDD[String] = file:/home/qm/Downloads/bank.csv MapPartitionsRDD[138] at textFile at <console>:38
defined class Bank
bank: org.apache.spark.sql.DataFrame = [age: int, job: string ... 3 more fields]
warning: there was one deprecation warning; re-run with -deprecation for details
```

Took 30 sec. Last updated by anonymous at April 13 2018, 3:24:34 PM.

图 4.2 代码

进行可视化的 SQL 查询，可以使用多种类型的图表进行数据的查看，使用饼状图查看数据，如图 3.3 所示。

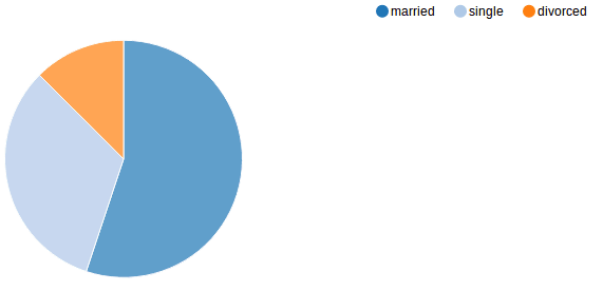
%sql

select martial, balance from bank where balance <1000

```
%sql
select marital, balance from bank where balance<1000
```

FINISHED

settings



Results are limited by 1000.

Took 2 sec. Last updated by anonymous at April 13 2018, 3:15:15 PM. (outdated)

图 4.3 饼状图查看数据

使用条状图查看数据，如图 4.4 所示。

%sql

select job, balance from bank where balance < 1000

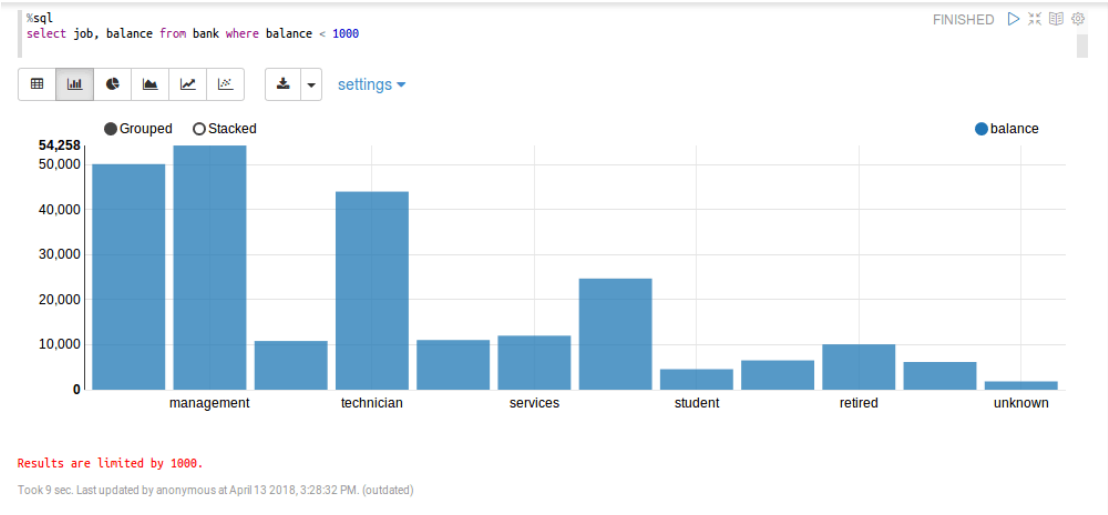


图 4.4 条状图查看数据

使用折线图查看数据，如图 4.5 所示。

%sql

select education,balance from bank sort by balance

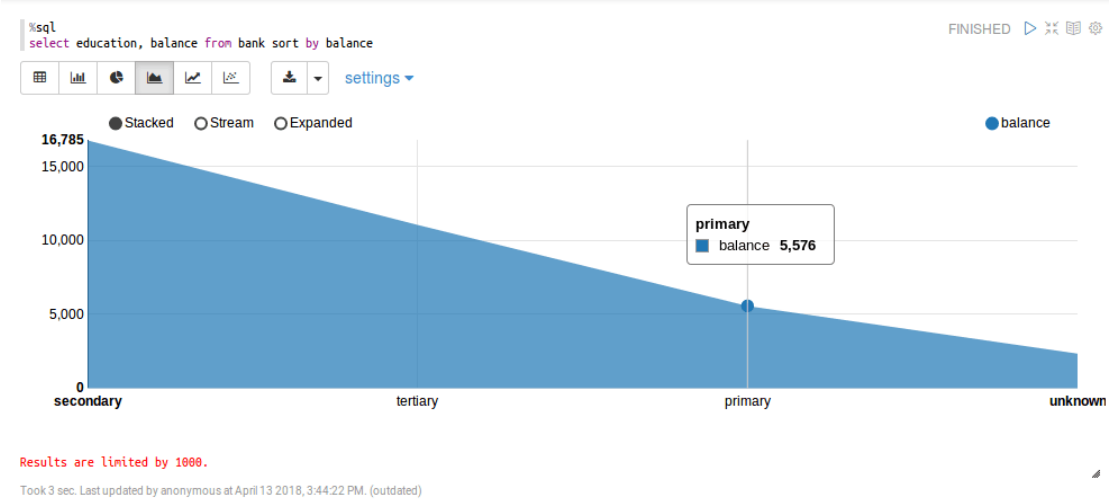


图 4.5 折线图查看数据

使用散点图查看数据，如图 4.6 所示。

%sql

select age, balance from bank sort by balance

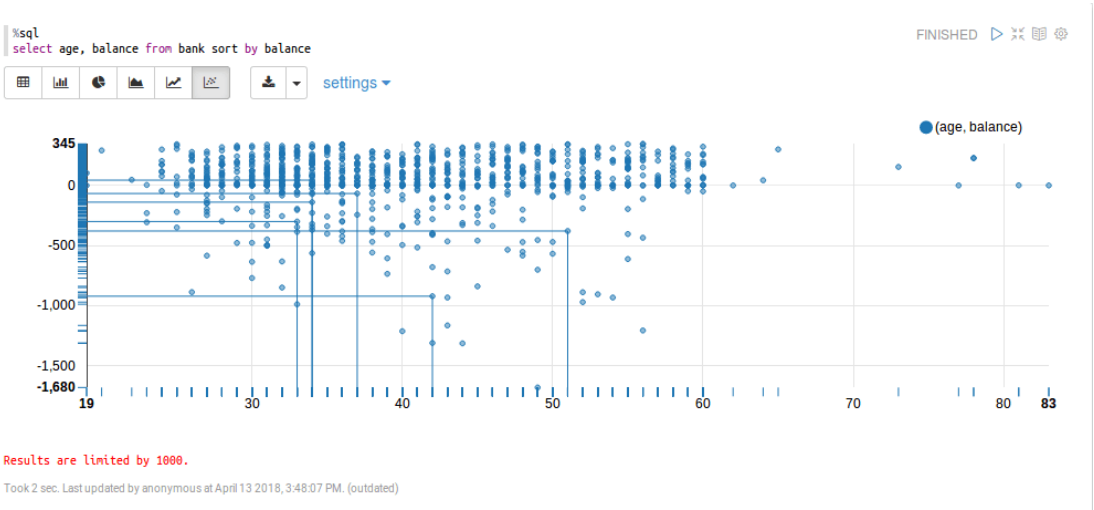


图 4.6 散点图图查看数据

总结

本次试验使用 Zeppelin 工具，对配置在 Mesos 上的 Spark 集群进行 SQL 可视化查询，通过 web 笔记的形式实现了通过多种图表的数据可视化呈现和分析。实验结果基本符合要求。接下来可以进一步的对于 Spark 集群的资源配置进行更好的优化，以达到更好的效果。