Zeppelin对接FusionInsight HD

适用场景

> Zeppelin-0.7.2 ↔ FusionInsight V100R002C60U20

安装Zepplin

操作场景

安装Zepplin0.7.2

前提条件

- 已完成FusionInsight HD客户端的安装。

操作步骤

- 将软件包zeppelin-0.7.2-bin-all.tgz上传至/opt目录下，解压生成zeppelin-0.7.2-bin-all目录。

```
tar -zxvf zeppelin-0.7.2-bin-all.tgz
```

- 启动和停止Zepplin

```
bin/zeppelin-daemon.sh start
bin/zeppelin-daemon.sh stop
```

```
[root@localhost zeppelin-0.7.2-bin-all]# bin/zeppelin-daemon.sh start
Log dir doesn't exist, create /opt/zeppelin-0.7.2-bin-all/logs
Pid dir doesn't exist, create /opt/zeppelin-0.7.2-bin-all/run
Zeppelin start                                             [  OK  ]
[root@localhost zeppelin-0.7.2-bin-all]# bin/zeppelin-daemon.sh stop
Zeppelin stop                                              [  OK  ]
```

- 配置Zeppelin环境变量，在profile文件中加入如下变量

```
vi /etc/profile
export ZEPPELIN_HOME=/opt/zeppelin-0.7.2-bin-all
export PATH=$ZEPPELIN_HOME/bin:$PATH
```

- 编辑zeppelin-env.sh文件，位置/opt/zeppelin-0.7.2-bin-all/conf

```
cd /opt/zeppelin-0.7.2-bin-all/conf/
cp zeppelin-env.sh.template zeppelin-env.sh
vi zeppelin-env.sh
```

加入如下内容：

```
export JAVA_HOME=/opt/jdk1.7.0_51/
```

编辑zeppelin-site.xml文件，位置/opt/zeppelin-0.7.2-bin-all/conf/

```
cp zeppelin-site.xml.template zeppelin-site.xml
```

将zeppelin-site.xml中端口8080替换成18081（可自定义，也可以不改）

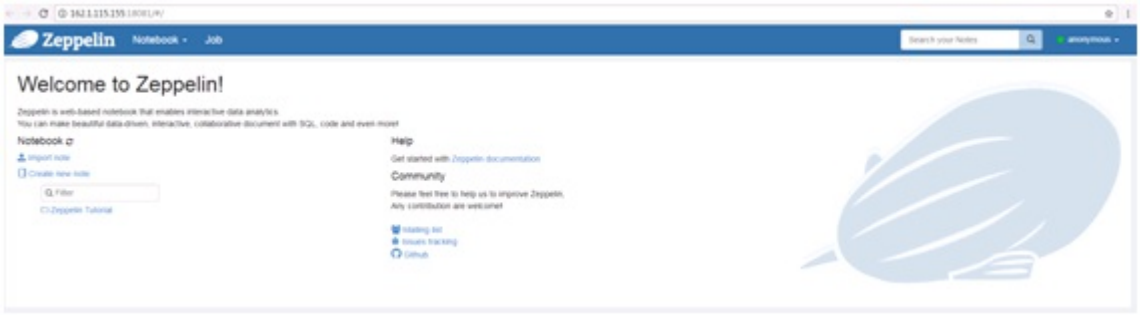
```
sed -i 's/8080/18081/' zeppelin-site.xml
```

```
<property>
  <name>zeppelin.server.port</name>
  <value>18081</value>
  <description>Server port.</description>
</property>
```

- 运行zeppelin

```
cd /opt/zeppelin-0.7.2-bin-all/
./bin/zeppelin-daemon.sh start
```

- 在浏览器中输入地址zeppelin_ip:18081登陆，zeppelin_ip为安装zeppelin的虚拟机IP。



- 根据产品文档创建用户test，并赋予足够权限，下载用户test的keytab文件user.keytab，上传至/opt/目录下。
- 编辑zeppelin-site.xml文件，将zeppelin.anonymous.allowed参数的true修改为false。

```
<property>
  <name>zeppelin.anonymous.allowed</name>
  <value>false</value>
  <description>Anonymous user allowed by default</description>
</property>
```

- 编辑shiro.ini文件，位置/opt/zeppelin-0.7.2-bin-all/conf/shiro.ini

```
cp shiro.ini.template shiro.ini
vi shiro.ini
```

[urls]authc表示对任何url访问都需要验证

```
[urls]
# This section is used for url-based security.
# You can secure interpreter, configuration and credential information by u
# anon means the access is anonymous.
# authc means Form based Auth Security
# To enfore security, comment the line below and uncomment the next one
/api/version = anon
#/api/interpreter/** = authc, roles[admin]
#/api/configurations/** = authc, roles[admin]
#/api/credential/** = authc, roles[admin]
#/** = anon
/** = authc
```

[users]下增加用户test，密码Huawei@123

```
[users]
# List of users with their password allowed to access Zeppelin.
# To use a different strategy (LDAP / Database / ...) check the shi
admin = password1, admin
user1 = password2, role1, role2
user2 = password3, role3
user3 = password4, role2
test = Huawei@123
```

- 重启zeppelin。

```
cd /opt/zeppelin-0.7.2-bin-all/
./bin/zeppelin-daemon.sh restart
```

- 使用test用户登陆Zeppelin

Zeppelin连接Hive

操作场景

Zepplin中配置JDBC解析器，对接Hive的JDBC接口。

前提条件

- 已经完成Zeppelin 0.7.2的安装；
- 已完成FusionInsight HD客户端的安装，包含Hive组件。

操作步骤

- 将`/opt/hadoopclient/Hive/Beeline/lib/`下的jar包拷贝至`/opt/zeppelin-0.7.2-bin-all/ interpreter/jdbc/`目录下。
- 将从新拷贝过来的jar包的属主和权限修改为和/opt/zeppelin-0.7.2-bin-all/ interpreter/jdbc/下原有的jar包相同

```
chown 501:wheel *.jar
chmod 644 *.jar
```

- 编辑zeppelin-env.sh文件，位置/opt/zeppelin-0.7.2-bin-all/conf，加入以下三个配置内容

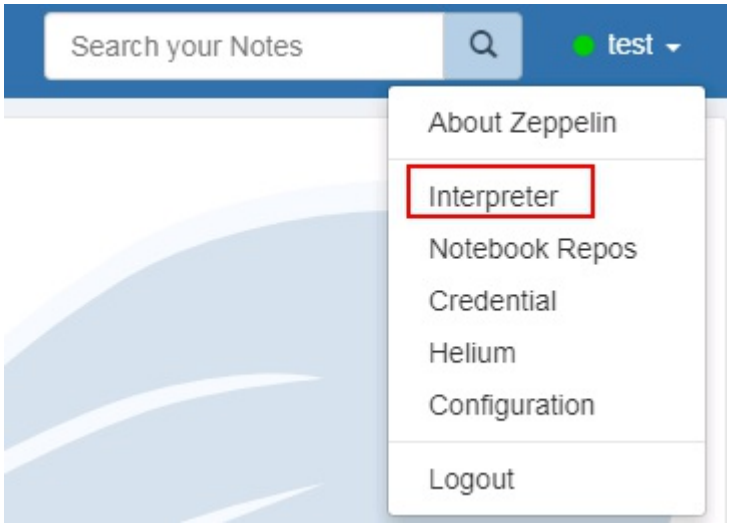
```
export JAVA_HOME=/opt/hadoopclient/JDK/jdk
export ZEPPELIN_INTP_JAVA_OPTS="-Djava.security.krb5.conf=/etc/krb5.conf -Djava.security.auth.login.config=/opt/zeppelin-0.7.2-bin-all/conf/jaas.conf -Dzookeeper.server.principal=zookeeper/hadoop.hadoop.com -Dzookeeper.request.timeout=120000"
export HADOOP_CONF_DIR=/opt/hadoopclient/HDFS/hadoop/etc/hadoop
```

- 从FusionInsight客户端下载用户test的user.keytab和krb5.conf文件，将krb5.conf文件放在/etc/下
- 使用`vi /opt/zeppelin-0.7.2-bin-all/conf/`新建hbase的认证文件jaas.conf，内容如下:

```
Client {
com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
keyTab="/opt/user.keytab"
principal="test"
useTicketCache=false
storeKey=true
debug=true;
};
```

> 其中用户为在FusionInsight Manager中创建的test用户，将test的keytab文件user.key放在/opt/目录下

- 登陆Zepplin，选择右上角菜单中的 Interpreter



- 选择JDBC，点击 **edit** 编辑，修改default.driver和default.url参数，点击 **save** 保存

```
default.driver：org.apache.hive.jdbc.HiveDriver
```

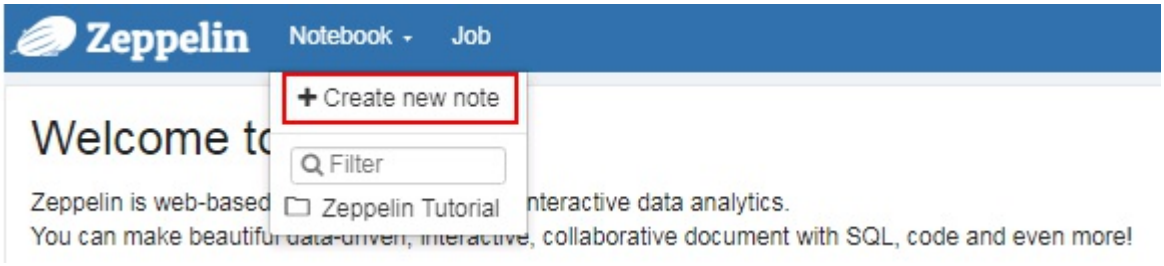
```
default.url:
jdbc:hive2://162.1.93.103:24002,162.1.93.102:24002,162.1.93.101:24002/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2;sasl.qop=auth-conf;auth=KERBEROS;principal=hive/hadoop.hadoop.com@HADOOP.COM
```

default.driver	org.apache.hive.jdbc.HiveDriver
default.password	
default.url	jdbc:hive2://162.1.93.103:24002,162.1.93.102:24002,162.1.93.101:24002/;serviceDiscoveryMode=zooKeeper;zooKeeperNamespace=hiveserver2;sasl.qop=auth-conf;auth=KERBEROS;principal=hive/hadoop.hadoop.com@HADOOP.COM

- 重启zeppelin。

```
source /opt/hadoopclient/bigdata_env
kinit –kt /opt/user.keytab test
cd /opt/zeppelin-0.7.2-bin-all/bin
./zeppelin-daemon.sh restart
```

- 页面选择Notebook -> Create new note



- 自定义note名称，例如hive

- 编辑note，点击右侧“执行”按钮。

```
%jdbc
Show tables;
Select * from workers_info;
```

- 查看结果

Zeppelin连接HBase

操作场景

Zeppelin中配置Hbase解析器，对接Hbase

前提条件

- 已经完成Zeppelin 0.7.2的安装；
- 已完成FusionInsight HD客户端的安装，包含HBase组件。

操作步骤

- 将 `/opt/hadoopclient/HBase/hbase/lib/` 以下的jar包拷贝至 `/opt/zeppelin-0.7.2-bin-all/interpreter/hbase/` 目录下，overwrite选择n

```
[root@localhost hbase]# cp /opt/hadoopclient/HBase/hbase/lib/*.jar .
cp: overwrite `./activation-1.1.jar'? n
cp: overwrite `./aopalliance-1.0.jar'? n
cp: overwrite `./apacheds-i18n-2.0.0-M15.jar'? n
cp: overwrite `./apacheds-kerberos-codec-2.0.0-M15.jar'? n
cp: overwrite `./api-asn1-api-1.0.0-M20.jar'? n
cp: overwrite `./api-util-1.0.0-M20.jar'? n
cp: overwrite `./asm-3.1.jar'? n
cp: overwrite `./avro-1.7.4.jar'? n
```

- 在/opt/zeppelin-0.7.2-bin-all/interpreter/hbase/下新建目录zeppelin_hbase_jar

```
mkdir /opt/zeppelin-0.7.2-bin-all/interpreter/hbase/zeppelin_hbase_jar
```

- 将/opt/zeppelin-0.7.2-bin-all/interpreter/hbase/下与FusionInsight冲突的38个jar包移动到zeppelin_hbase_jar目录中
  - commons-codec-1.5.jar
  - commons-collections-3.2.1.jar
  - commons-configuration-1.9.jar
  - commons-lang-2.5.jar
  - commons-logging-1.1.1.jar
  - guava-15.0.jar
  - hadoop-annotations-2.6.0.jar
  - hadoop-auth-2.5.1.jar
  - hadoop-client-2.5.1.jar
  - hadoop-common-2.5.1.jar
  - hadoop-hdfs-2.5.1.jar
  - hadoop-mapreduce-client-app-2.5.1.jar
  - hadoop-mapreduce-client-common-2.5.1.jar
  - hadoop-mapreduce-client-core-2.5.1.jar
  - hadoop-mapreduce-client-jobclient-2.5.1.jar
  - hadoop-mapreduce-client-shuffle-2.5.1.jar
  - hadoop-yarn-api-2.6.0.jar
  - hadoop-yarn-client-2.5.1.jar
  - hadoop-yarn-common-2.6.0.jar
  - hadoop-yarn-server-common-2.5.1.jar
  - hbase-annotations-1.0.0.jar
  - hbase-client-1.0.0.jar
  - hbase-common-1.0.0.jar
  - hbase-common-1.0.0-tests.jar
  - hbase-hadoop2-compat-1.0.0.jar
  - hbase-hadoop-compat-1.0.0.jar
  - hbase-prefix-tree-1.0.0.jar
  - hbase-protocol-1.0.0.jar
  - hbase-server-1.0.0.jar
  - httpclient-4.5.1.jar
  - httpcore-4.4.1.jar
  - jettison-1.1.jar
  - netty-3.6.2.Final.jar
  - slf4j-api-1.7.10.jar
  - slf4j-log4j12-1.7.10.jar
  - xmlenc-0.52.jar
  - zookeeper-3.4.6.jar

- 最终/opt/zeppelin-0.7.2-bin-all/interpreter/hbase/有152个jar包

```
[root@localhost hbase]# cd zeppelin_hbase_jar/
[root@localhost zeppelin_hbase_jar]# ll  |wc -l
38
[root@localhost zeppelin_hbase_jar]# cd ../
[root@localhost hbase]# ll | wc -l
152
[root@localhost hbase]#
```

- 编辑zeppelin-env.sh文件，位置/opt/zeppelin-0.7.2-bin-all/conf，加入以下三个配置内容

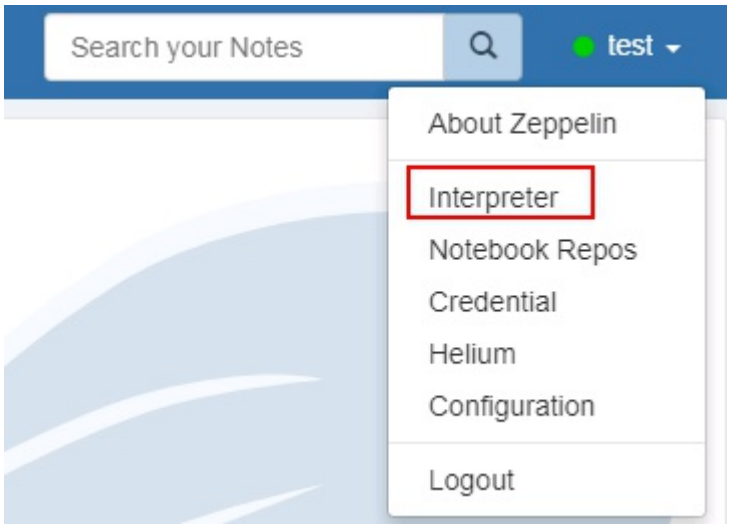
```
export JAVA_HOME=/opt/hadoopclient/JDK/jdk
export ZEPPELIN_INTP_JAVA_OPTS="-Djava.security.krb5.conf=/etc/krb5.conf -Djava.security.auth.login.config=/opt/zeppelin-0.7.2-bin-all/conf/jaas.conf -Dzookeeper.server.principal=zookeeper/hadoop.hadoop.com -Dzookeeper.request.timeout=120000"
export HBASE_HOME=/opt/hadoopclient/HBase/hbase
```

- 从FusionInsight客户端下载用户test的user.keytab和krb5.conf文件，将krb5.conf文件放在/etc/下
- 使用 `vi /opt/zeppelin-0.7.2-bin-all/conf/` 新建hbase的认证文件jaas.conf，内容如下:

```
Client {
com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
keyTab="/opt/user.keytab"
principal="test"
useTicketCache=false
storeKey=true
debug=true;
};
```

> 其中用户为在FusionInsight Manager中创建的test用户，将test的keytab文件user.key放在/opt/目录下

- 登陆Zepplin，选择右上角菜单中的 Interpreter



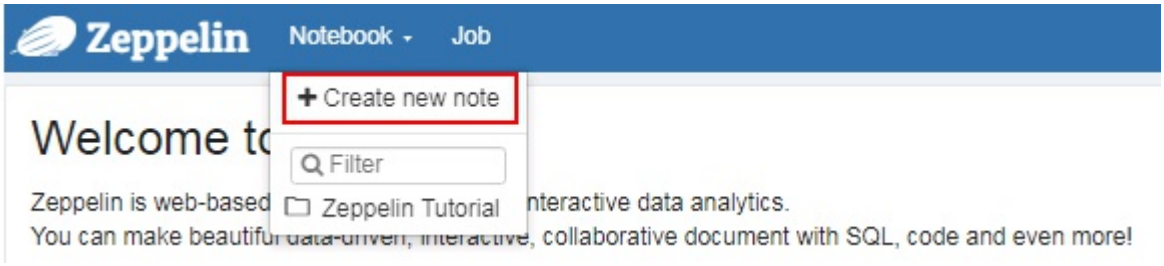
- 选择hbase，点击 **edit** 编辑，修改hbase.home参数，点击 **save** 保存

```
hbase.home：/opt/hadoopclient/HBase/hbase
```

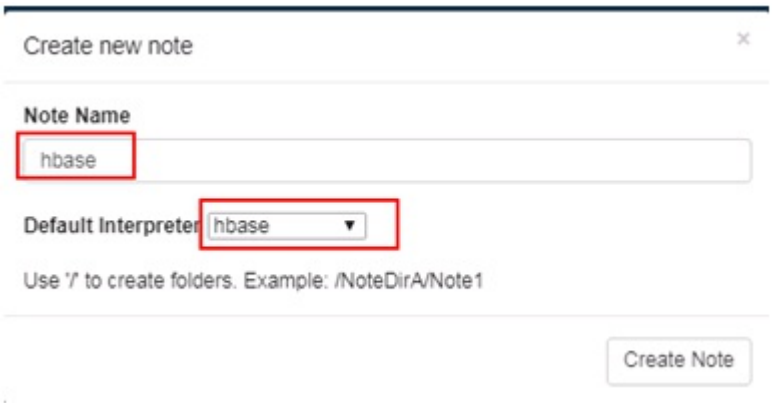
- 重启zeppelin

```
source /opt/hadoopclient/bigdata_env
kinit -kt /opt/user.keytab test
cd /opt/zeppelin-0.7.2-bin-all/bin
./zeppelin-daemon.sh restart
```

- 页面选择Notebook -> Create new note

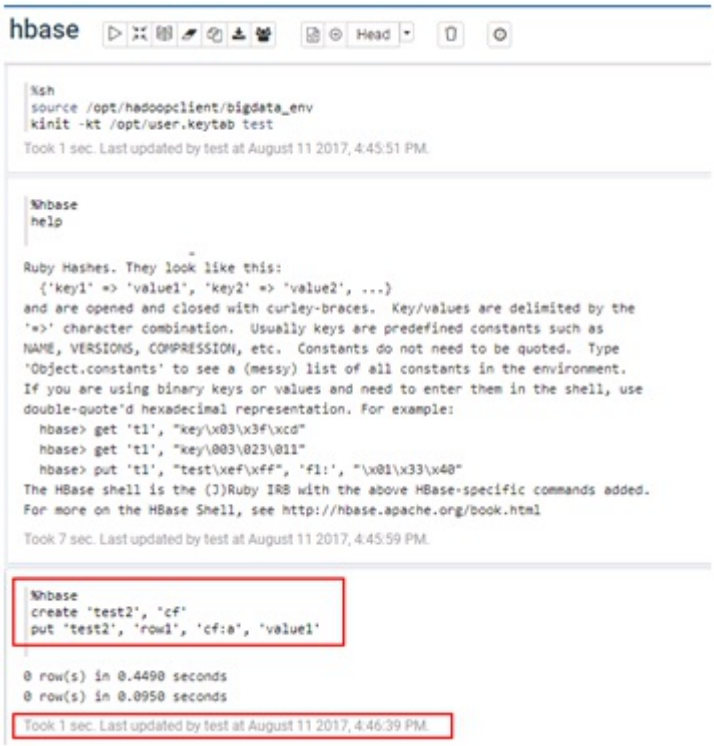


- 自定义note名称，例如hbase



- 编辑note，点击右侧 **执行** 按钮

```
%hbase
create 'test2', 'cf'
put 'test2', 'row1', 'cf:a', 'value1'
```



- 在FusionInsight的客户端下可以看到创建的hbase表test2和数据

```
hbase(main):003:0> scan "test2"
ROW                           COLUMN+CELL
 row1                         column=cf:a, timestamp=1502441199545, value=value1
1 row(s) in 0.0300 seconds
```

Zeppelin连接Spark

操作场景

Zepplin中配置Spark解析器

前提条件

- 完成Zeppelin0.7.2的安装；
- 已完成FusionInsight HD V100R002C60U20和客户端的安装，包含Spark组件。
- 参考http://zeppelin.apache.org/docs/latest/interpreter/spark.html

操作步骤

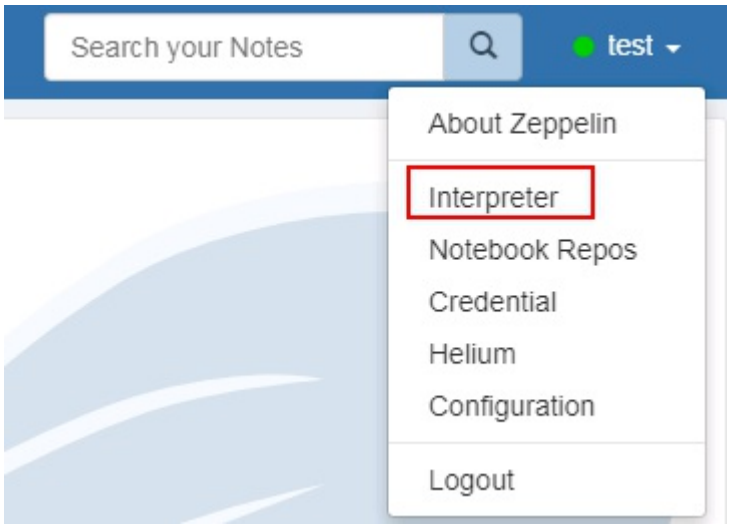
- 将`/opt/zeppelin-0.7.2-bin-all/lib/`目录下的原有的相关的jar包删除
  - hadoop-auth-2.6.0.jar
  - hadoop-common-2.6.0.jar
  - scala-compiler-2.11.7.jar
  - scala-library-2.11.7.jar
  - scala-parser-combinators_2.11-1.0.4.jar
  - scala-reflect-2.11.7.jar
  - scala-xml_2.11-1.0.2.jar
- 将`/opt/hadoopclient/Spark/adapter/dev_lib/`下的以下jar包拷贝到`/opt/zeppelin-0.7.2-bin-all/lib/`目录下
  - hadoop-auth-2.7.2.jar
  - hadoop-common-2.7.2.jar
  - scala-compiler-2.10.4.jar
  - scala-library-2.10.4.jar
  - scala-reflect-2.10.4.jar
- 将`/opt/zeppelin-0.7.2-bin-all/lib/`下的jackson的相关jar包删除
  - jackson-annotations-2.5.0.jar
  - jackson-core-2.5.3.jar
  - jackson-core-asl-1.9.13.jar
  - jackson-databind-2.5.3.jar
  - jackson-mapper-asl-1.9.13.jar
- 将`/opt/hadoopclient/Spark/adapter/dev_lib/`下的jackson相关的jar包拷贝到`/opt/zeppelin-0.7.2-bin-all/lib/`下
  - jackson-annotations-2.4.0.jar
  - jackson-core-2.4.4.jar
  - jackson-core-asl-1.9.13.jar
  - jackson-databind-2.4.4.jar
  - jackson-jaxrs-1.9.13.jar
  - jackson-mapper-asl-1.9.13.jar
  - jackson-module-scala_2.10-2.4.4.jar
  - jackson-xc-1.9.13.jar
- 将步骤1和步骤2所有从spark客户端拷贝过来的jar包的属主和权限修改为和`/opt/zeppelin-0.7.2-bin-all/lib/`下原有的jar包相同

```
chown 501:wheel *.jar
chmod 644 *.jar
```

- 编辑zeppelin-env.sh文件，位置`/opt/zeppelin-0.7.2-bin-all/conf`，加入以下内容

```
export MASTER=yarn-client
export SPARK_HOME=/opt/hadoopclient/Spark/spark
export HADOOP_CONF_DIR=/opt/hadoopclient/HDFS/hadoop/etc/hadoop
```

- 登陆Zepplin，选择右上角菜单中的 Interpreter



- 选择Spark，点击 **edit** 编辑，将 Master 参数改为 yarn-client，点击 **save** 保存

Properties

name	value
args	
master	yarn-client
spark.app.name	Zeppelin

- 重启zeppelin

```
source /opt/hadoopclient/bigdata_env
kinit -kt /opt/user.keytab test
cd /opt/zeppelin-0.7.2-bin-all/bin
./zeppelin-daemon.sh restart
```

- 执行zeppelin的spark样例代码zeppelin Tutorial -> Basic Features(Spark)

> 样例代码需要访问Internet上的资源，所以保证zeppelin所在的节点可以联网，检测是否能打开以下链接

Load data into table

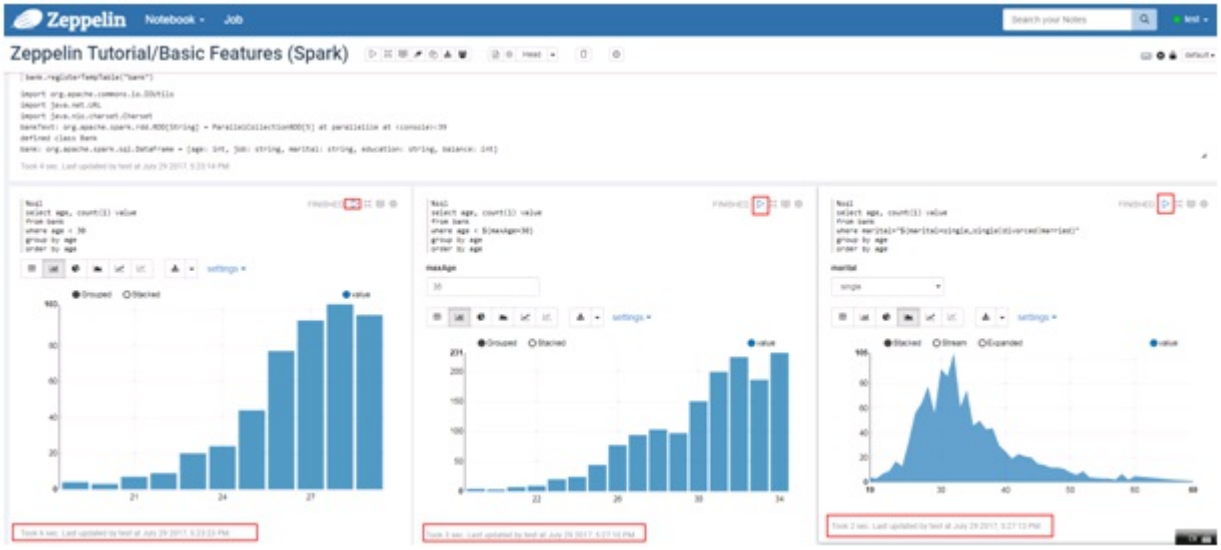
```
import org.apache.commons.io.IOUtils
import java.net.URL
import java.nio.charset.Charset

// 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.parallelize(
    IOUtils.toString(
        new URL("https://s3.amazonaws.com/apache-zeppelin/tutorial/bank/bank.csv"),
        Charset.forName("utf8")).split("\n"))

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")
```



- 执行zeppelin的spark样例代码Zeppelin Tutorial -> Matplotlib (Python • PySpark)

  安装python-matplotlib

  ```
  yum install python-matplotlib
  ```

  安装Anaconda2-4.4

  ```
  wget https://repo.continuum.io/archive/Anaconda2-4.4.0-Linux-x86_64.sh
  sh Anaconda2-4.4.0-Linux-x86_64.sh
  ```

  配置环境变量PATH，将python换成安装Anaconda安装目录中的python

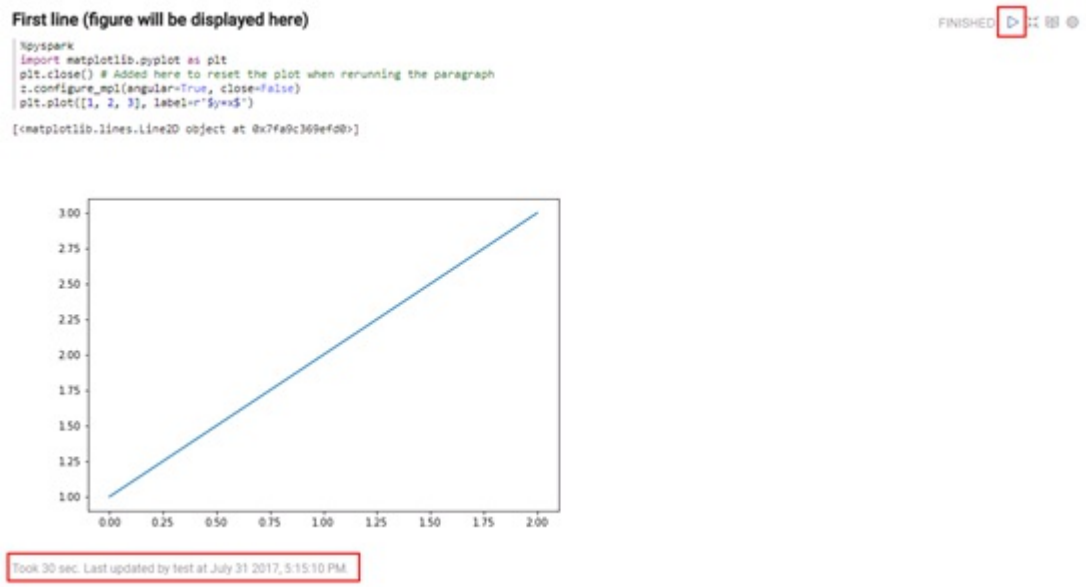
  ```
  export PATH=/root/anaconda2/bin/:$PATH
  ```

  在zeppelin的界面中，选择右上角的 Interpreter

  选择Spark，点击 **edit** 编辑，将 zeppelin.pyspark.python 参数改为Anaconda安装目录中的python，点击 **save** 保存

zeppelin.pyspark.python	/root/anaconda2/bin/python

  执行zeppelin的pyspark样例代码Zeppelin Tutorial -> Matplotlib

Zeppelin连接SparkR

操作场景

Zepplin中配置Spark解析器，连接SparkR

前提条件

- 完成Zeppelin0.7.2的安装；
- 已完成FusionInsight HD V100R002C60U20和客户端的安装，包含Spark组件。
- 参考http://zeppelin.apache.org/docs/latest/interpreter/spark.html

操作步骤

- 安装R，通过yum源安装所需rpm包

```
yum install gcc-c++ gcc-gfortran zlib zlib-devel bzip2 bzip2-devel perl openssl-devel
yum install xz-devel.x86_64 pcre pcre-devel libcurl libcurl-devel readline-devel libXt-devel
```

- 安装curl-7.47.1（安装R需要curl版本高于7.22.0）

```
  wget --no-check-certificate https://curl.haxx.se/download/curl-7.47.1.tar.gz
  tar -zxvf curl-7.47.1.tar.gz
  cd curl-7.47.1
  ./configure
  make
  make install
```

- 下载并解压R-3.4.1

```
cd /root
wget http://cran.stat.nus.edu.sg/src/base/R-3/R-3.4.1.tar.gz
tar -zxvf R-3.4.1.tar.gz
```

- 编译安装R-3.4.1

```
mkdir /opt/R-3.4.1
./configure --prefix /opt/R-3.4.1 --enable-R-shlib --with-x --with-cairo --with-libpng
make
make install
```

> configure的过程中，可能会有各种不同的错误，针对报错信息网上搜索解决方法，另外make的过程耗时较多

- 配置R的环境变量: `vi /etc/profile` 在文件最后一行增加 `export PATH=/opt/R-3.4.1/bin:$PATH`
- 检查R是否可用

```
source /etc/profile
./R
```

- 正常启动如下图所示

```
[root@fusetest bin]# ./R

R version 3.4.1 (2017-06-30) -- "Single Candle"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

- FusionInsight客户端下测试是否可以使用sparkR

```
source /opt/hadoopclient/bigdata_env
kinit test
sparkR
```

- 正常启动如下图所示

```
[root@fusetest ~]# source .bash_profile
[root@fusetest ~]# source /opt/hadoopclient/bigdata_env
[root@fusetest ~]# kinit test
Password for test@HADOOP.COM:
[root@fusetest ~]# cd /opt/hadoopclient/Spark/spark/bin/
[root@fusetest bin]# sparkR

R version 3.4.1 (2017-06-30) -- "Single Candle"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

Launching java with spark-submit command /opt/hadoopclient/Spark/spark/bin/spark-submit   "spar
Warning: Ignoring non-spark config property: hadoop_server_path=/opt/huawei/Bigdata/hadoop/
17/08/02 16:38:41 WARN SparkConf: In Spark 1.0 and later spark.local.dir will be overridden by
DIRS in mesos/standalone and LOCAL_DIRS in YARN).
17/08/02 16:38:42 WARN MetricsSystem: Using default name DAGScheduler for source because spark.

 Welcome to
    ____              __
   / __/__  ___ _____/ /__
  _\ \/ _ \/ _ `/ __/  '_/
 /___/ .__/\_,_/_/ /_/\_\   version  1.5.1
    /_/

 Spark context is available as sc, SQL context is available as sqlContext
>
```

- 参考http://zeppelin.apache.org/docs/0.7.2/interpreter/r.html#using-the-r-interpreter 在R的命令行中安装sparkR样例需要的R的libraries

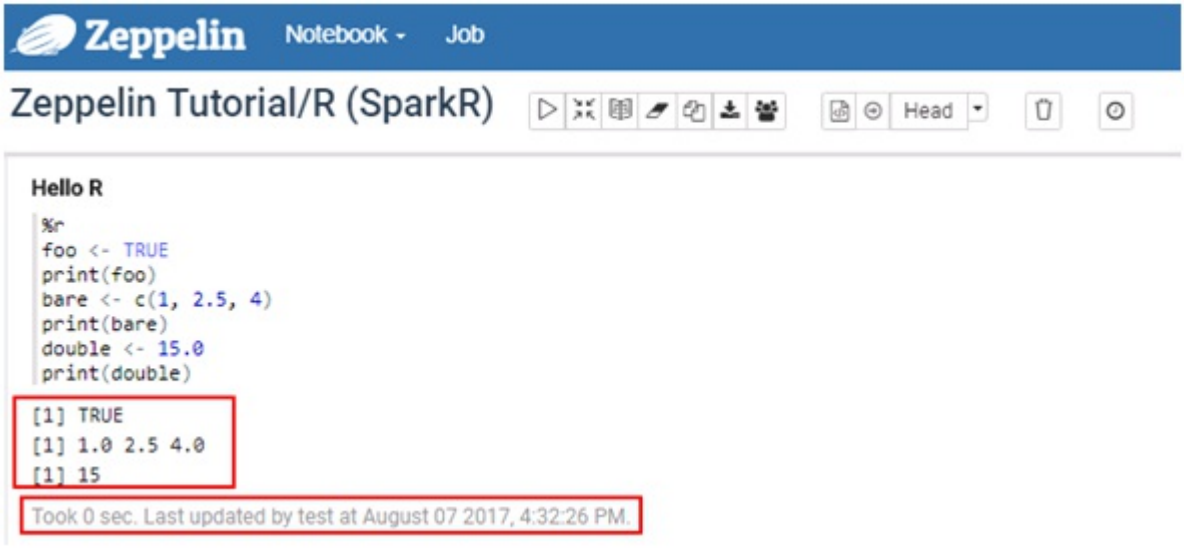
```
install.packages('devtools')
install.packages('knitr')
install.packages('ggplot2')
install.packages(c('devtools','mplot','googleVis'))
install.packages('data.table')
install.packages('sqldf')
install.packages('glmnet')
install.packages('pROC')
install.packages('caret')
install.packages('sqldf')
install.packages('wordcloud')
```

- 在zeppelin的界面中，选择右上角的 Interpreter
- 选择Spark，点击 **edit** 编辑，将 zeppelin.R.cmd 参数改为R的启动文件，点击 **save** 保存

- 重启zeppelin

```
cd /opt/zeppelin-0.7.2-bin-all/bin/
./zeppelin-daemon.sh restart
```

- 在Zeppelin中执行Zeppelin Tutorial -> R (SparkR)样例



FAQ

- 连接hbase出现AuthFialed for /hwbackup/hbase

```
%hbase
help
```

```
org.apache.zookeeper.KeeperException$AuthFailedException: KeeperErrorCode = AuthFailed for /hwbackup/hbase
        at org.apache.zookeeper.KeeperException.create(KeeperException.java:123)
        at org.apache.zookeeper.KeeperException.create(KeeperException.java:51)
        at org.apache.zookeeper.ZooKeeper.create(ZooKeeper.java:783)
        at org.apache.hadoop.hbase.zookeeper.RecoverableZooKeeper.createNonSequential(RecoverableZooKeeper.java:512)
        at org.apache.hadoop.hbase.zookeeper.RecoverableZooKeeper.create(RecoverableZooKeeper.java:491)
        at org.apache.hadoop.hbase.zookeeper.ZKUtil.createWithParents(ZKUtil.java:1252)
        at org.apache.hadoop.hbase.zookeeper.ZKUtil.createWithParents(ZKUtil.java:1230)
        at com.huawei.hadoop.hbase.backup.zookeeper.BackupZooKeeperWatcher.createBaseZNodes(BackupZooKeeperWatcher.java:137)
        at com.huawei.hadoop.hbase.backup.zookeeper.BackupZooKeeperWatcher.<init>(BackupZooKeeperWatcher.java:78)
        at com.huawei.hadoop.hbase.backup.client.BackupAdmin.<init>(BackupAdmin.java:98)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
        at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
        at org.jruby.javasupport.JavaConstructor.newInstanceDirect(JavaConstructor.java:275)
```

  原因：zeppelin的原理hbase的jar包与从FusionInsight客户端下拷贝过来的jar冲突。

  解决：将zeppelin中原有的重名jar包移走或删除，全部用FusionInsight客户端下的相关jar包。

- Zeppelin连接spark是报如下NoSuchMethodError

```
java.lang.NoSuchMethodError: scala.reflect.api.JavaUniverse.runtimeMirror(Ljava/lang/ClassLoader;)Lscala/reflect/api/JavaMirrors$JavaMirror;
        at org.apache.spark.repl.SparkILoop.<init>(SparkILoop.scala:936)
        at org.apache.spark.repl.SparkILoop.<init>(SparkILoop.scala:70)
        at org.apache.zeppelin.spark.SparkInterpreter.open(SparkInterpreter.java:775)
        at org.apache.zeppelin.interpreter.LazyOpenInterpreter.open(LazyOpenInterpreter.java:70)
        at org.apache.zeppelin.spark.PySparkInterpreter.getSparkInterpreter(PySparkInterpreter.java:564)
        at org.apache.zeppelin.spark.PySparkInterpreter.createGatewayServerAndStartScript(PySparkInterpreter.java:208)
        at org.apache.zeppelin.spark.PySparkInterpreter.open(PySparkInterpreter.java:162)
        at org.apache.zeppelin.interpreter.LazyOpenInterpreter.open(LazyOpenInterpreter.java:70)
        at org.apache.zeppelin.interpreter.remote.RemoteInterpreterServer$InterpretJob.jobRun(RemoteInterpreterServer.java:491)
        at org.apache.zeppelin.scheduler.Job.run(Job.java:175)
        at org.apache.zeppelin.scheduler.FIFOScheduler$1.run(FIFOScheduler.java:139)
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:471)
        at java.util.concurrent.FutureTask.run(FutureTask.java:262)
        at java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.access$201(ScheduledThreadPoolExecutor.java:178)
        at java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(ScheduledThreadPoolExecutor.java:292)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1145)
```

  原因：jar包冲突

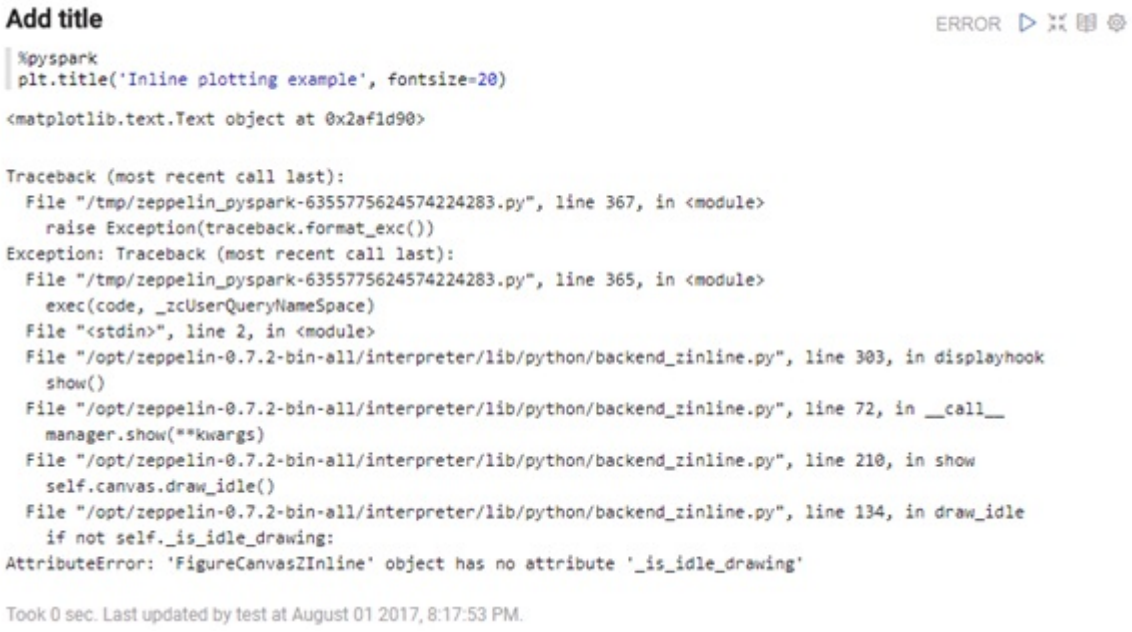
  解决：删除 `/opt/zeppelin-0.7.2-bin-all/lib/` 下原有jar包scala-reflect-2.11.7.jar，替换为FusionInsight客户端下的jar包，重启zeppelin

- Zeppelin执行Spark样例代码时报GC overhead limit exceeded

原因：内存不够

解决：安装Zeppelin的节点的内存需要16G以上

- 执行zeppelin的样例代码Zeppelin Tutorial/Matplotlib (Python PySpark)报如下错误



原因：python版本问题

解决：安装Anaconda2-4.4