##### Spark-2.3.2-bin-hadoop2.6镜像构建

###### 实现目的

基于容器技术，搭建Spark-Hadoop开发环境，以供开发人员快速掌握并应用到实际开发中，有关Spark的优点等知识这里不再赘述。暂且把优点总结如下：

1. 技术积累
2. 省时省力省钱

###### 环境准备

1. sequenceiq/docker-spark源码

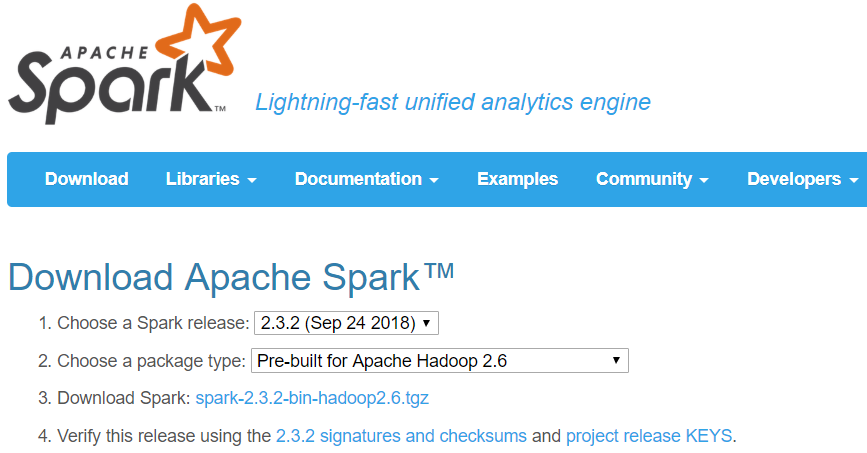
下载sequenceiq/ docker-spark镜像构建源码，下载命令如下：

|  |
| --- |
| *git clone https://github.com/sequenceiq/docker-spark* |

1. sequenceiq/hadoop-docker:2.6.0 官网镜像

|  |
| --- |
| docker pull sequenceiq/hadoop-docker:2.6.0 |

1. spark-2.3.2-bin-hadoop2.6.tgz



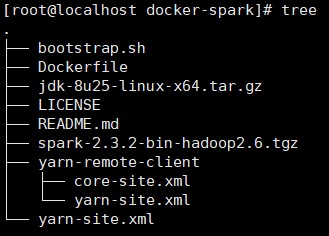
1. jdk-8u25-linux-x64.tar.gz

请自行下载。

###### 镜像构建

1. 目录结构

基础目录结构如下图所示：



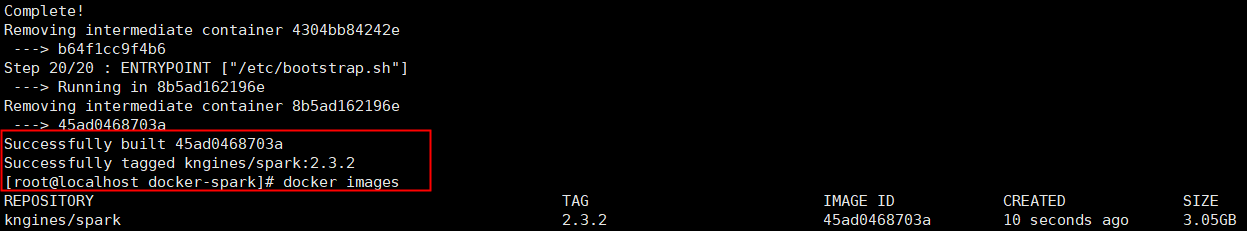
其中，docker-spark为sequenceiq/docker-spark源码目录，Dockerfile及yarn-site.xml如下附件。



1. 构建命令

|  |
| --- |
| *docker build --rm -t kngines/spark:2.3.2 .* |

1. 结果输出



1. 创建容器

|  |
| --- |
| *docker run -it -p 8088:8088 -p 8042:8042 -p 4040:4040 -h sandbox kngines/spark:2.3.2 bash* |

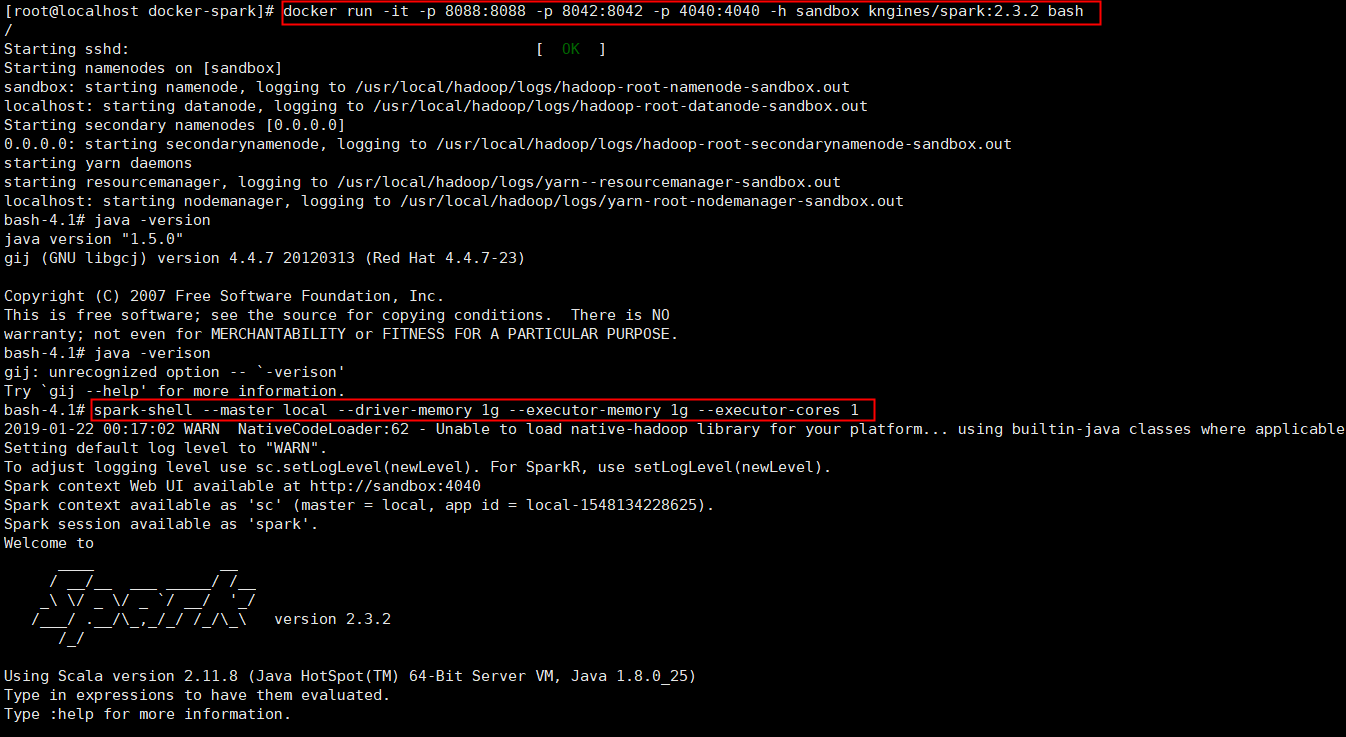
1. 运行模式
2. Local模式命令

*spark-shell --master local --driver-memory 1g --executor-memory 1g --executor-cores 1*

1. Yarn模式命令

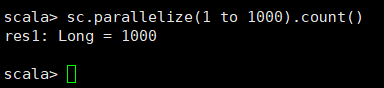
*spark-shell --master yarn --deploy-mode client --driver-memory 512m --executor-memory 512m --executor-cores 1*

1. 运行效果

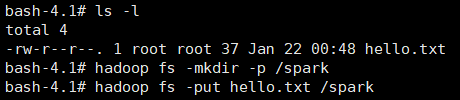


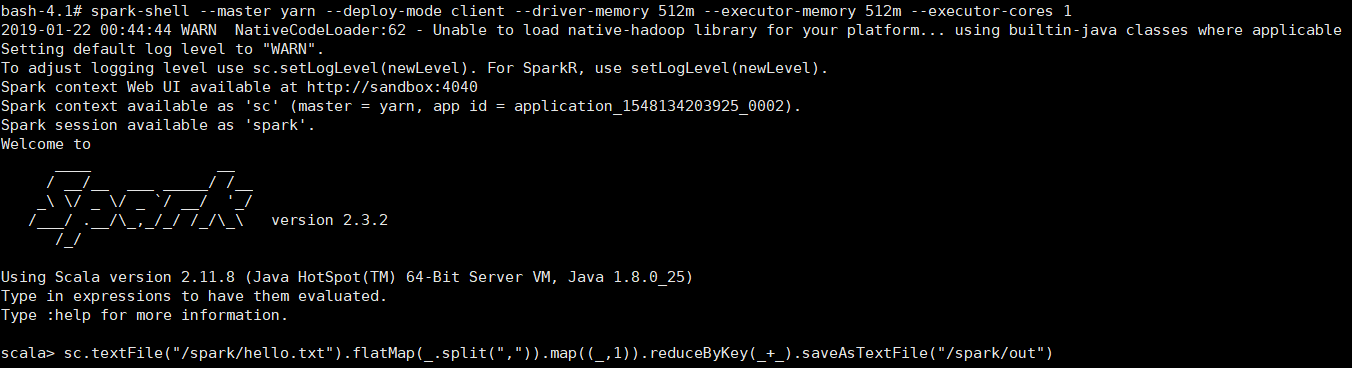
###### 测试运行

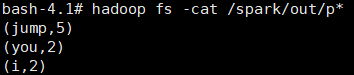
1. 简单测试



1. 复杂测试（Hadoop-Spark）







说明：

sc是SparkContext对象，提交spark程序的入口

textFile("/spark/hello.txt")是从hdfs中读取数据

flatMap(\_.split(" "))先map再压平

map((\_,1))将单词和1构成元组

reduceByKey(\_+\_)按照key进行reduce，并将value累加

saveAsTextFile("/spark/out")将结果写入到hdfs中

相关命令：

hadoop fs -mkdir -p /spark

hadoop fs -put hello.txt /spark

hadoop fs -cat /spark/out/p\*

###### Web UI

1. Spark Web UI

