Big Data

Hadoop

Prof. Leandro Batista de Almeida
leandro@utfpr.edu.br
2016

Média de recursos para Hadoop

- Cluster convencional
  - Centenas a milhares de nós
  - Menos de 100 nós é considerado um cluster experimental
- Nó convencional
  - 64 Gb RAM +
  - 12 cores +
  - 4 disks +

Instalando Hadoop 2.x

- Não é uma tarefa complexa
  - Similar a 1.x
- Modos de instalação
  - Local
  - Stand Alone
    - Pseudo-distributed
  - Cluster

Requisitos

- Roda em Unix e Windows
  - Linux é o único ambiente de produção suportado
    - Mas roda em outros Unix, como MacOS
    - Windows precisa de Cygwin e openssh
      - HortonWorks distribution
- Java
  - Versão 7 ou superior
- SSH

Configuração

- Arquivos XML
  - Diretório etc/hadoop
    - No diretório de instalação do Hadoop
- core-site.xml
  - Configuração principal
  - HDFS
- mapred-site.xml
  - MapReduce framework (YARN)
- hdfs-site.xml
  - Configuração de replicação
- yarn-site.xml
  - Configuração YARN

Stand Alone

- Em Linux
- Instalar ssh
  - `sudo apt-get install ssh`
- Criar ssh key
  - `ssh-keygen -t rsa -P ""`
  - `cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys`
- Instalar java (oracle)
  - `sudo add-apt-repository ppa:webupd8team/java`
  - `sudo apt-get update`
  - `sudo apt-get install oracle-java8-installer`

Stand Alone

- Baixar hadoop e descompactar
- Copiar arquivos do hadoop
  - Para /usr/local/hadoop, por exemplo
  - `sudo tar xzf hadoop-2.7.tar.gz`
  - `sudo mv hadoop-2.7 /usr/local/hadoop`
  - `sudo chown -R user:group /usr/local/hadoop`

Stand Alone

- Configurar .bashrc do usuário hadoop
- nano .bashrc

```
#Variaveis Hadoop
export JAVA_HOME=/usr/lib/jvm/java8oracle
export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
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=$HADOOP_HOME/lib/native
export HADOOP_OPTS="Djava.library.path=$HADOOP_HOME/lib"
unset JAVA_TOOL_OPTIONS
```

Stand Alone

- Configurar arquivos hadoop, em $HADOOP_HOME/etc/hadoop

- hadoop-env.sh
  - `export JAVA_HOME="/usr/lib/jvm/java-8-oracle"`
  - `export HADOOP_OPTS=-Djava.net.preferIPv4Stack=true`

Stand Alone

- core-site.xml (na tag configuration)

```
<property>
  <name>fs.defaultFS</name>
  <value>hdfs://localhost:9000</value>
</property>
```

Stand Alone

- mapred-site.xml (na tag configuration)
- cp mapred-site.xml.template mapred-site.xml

```
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>
```

Stand Alone

- hdfs-site.xml (na tag configuration)

```
<property>
  <name>dfs.replication</name>
  <value>1</value>
</property>
<property>
  <name>dfs.namenode.name.dir</name>
  <value>file:/usr/local/hadoop/hadoop_data/hdfs/namenode</value>
 </property>
 <property>
  <name>dfs.datanode.data.dir</name>
  <value>file:/usr/local/hadoop/hadoop_data/hdfs/datanode</value>
 </property>
```

Stand Alone

- yarn-site.xml (na tag configuration)

```
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
<property>
  <name>
    yarn.nodemanager.auxservices.mapreduce.shuffle.class
  </name>
  <value> org.apache.hadoop.mapred.ShuffleHandler</value>
 </property>
```

Stand Alone

- Criar diretórios para namenode e datanode

```
mkdir p /usr/local/hadoop/hadoop_data/hdfs/namenode
$mkdir p /usr/local/hadoop/hadoop_data/hdfs/datanode
sudo chown sawmya:sawmya R /usr/local/hadoop
```

- Formatar sistema de arquivos HDFS
  - `hdfs namenode -format`
- Iniciar deamons do hadoop
  - `start-dfs.sh`
  - `start-yarn.sh`
  - `mr-jobhistory-daemon start historyserver`
- Verificar se os deamons estão executando com jps
  - NameNode, DataNode, SecondaryNameNode
  - ResourceManager, NodeManager
  - JobHistoryServer

Stand Alone

- Ferramentas de gerenciamento
  - Linha de comando
    - hadoop, hdfs, yarn, etc
    - Hadoop docs
  - Web
    - HDFS admin
      - http://localhost:50070
    - YARN admin
      - http://localhost:8088
    - History Server
      - http://localhost:19888

Testando hadoop

- Diversos exemplos no sistema hadoop
  - Diretório $HADOOP_HOME/share/hadoop/mapreduce
  - WordCount, TeraSort, TestDFSIO, etc
- TestDFSIO
  - Grava e lê dados aleatórios em HDFS
    - Dados são gerados em jobs mapreduce
    - Usados para testar clusters e stressar o sistema

```
hadoop jar
hadoop-mapreduce-client-jobclient-2.7.2-tests.jar
TestDFSIO -write -nrFiles 5 -fileSize 10
```

Questões

Prof. Leandro Batista de Almeida
leandro@utfpr.edu.br