Hadoopz主要有4个项目组成：

\*Common

\*HDFS

\*yarn

\*Mapreduce

主要配置一下

第一:

Hadoop-env.sh yarn-env.sh mapred-eng.sh 里面加入JAVA\_HOME

在配置hadoop环境的时候可以围绕上面的4个项目分别进行配置

用户的配置文件主要包括

core-site.xml

hdfs-site.xml

yar-site.xml

mapred-site.xml

系统默认的配置在如下配置文件中，配置文件所在的位置在对应的包下。

Core-default.xml

Hdfs-default.xml

Yarn-default.xml

Mapred-default.xml

Hadoop有3中模式：本地，伪分布，分布式

本地模式下：不要配置什么，直接可以使用，使用了系统默认的配置

伪分布和分布式基本相同

Common的配置

Core-site.xml

Fs.defaultFs配置文件系统的入口

其值例如：hdfs://master:9000

在这个地方的master，其实也是namenode的地址，在这个地方指定了namenode

还需要指定一下hadoop.tmp.dir

这个的作用就是防止每次linux重新启动的时候数据丢失



Common最小的基本配置完成，什么叫配置，就是覆盖了系统默认的，所以还可以配置的更多。

Hdfs的配置

Hdfs-env.sh配置JAVA\_HOME

Hdfs-site.xml

Dfs.replication这个是指定文件的副本数

Dfs.name.secondary.http-address这个是指定secondary的http地址

Dfs.namenode.name.dir

Dfs.namenode.data.dir



配置datenode

配置datanode就是配置slaves文件。在里面加入从节点的主机名或者ip就可以了，这个也同时使nodemanager的配置，所以slaves里面的指定的主机就是datanode，同时也是nodemanager所在的节点位置

然后配置yarn

Yarn-env.sh配置JAVA\_HOME

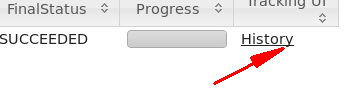
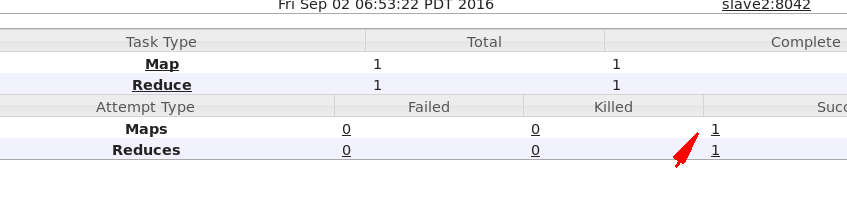
Yarn-site.xml

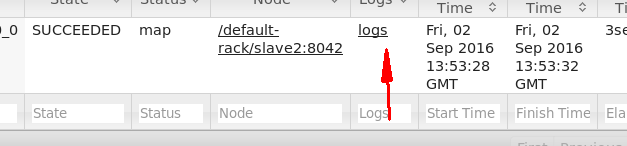
Yarn-resroucemanager.hostname这个是指定resoucemanager的地址，其nodemanager在slaves文件里面指定

Yarn.nodemanager.aux-services这个是指定在yarn上运行的是什么，其值例如如下：

mapredece\_shuffle指定运行在其上面的是map reduce shuffle

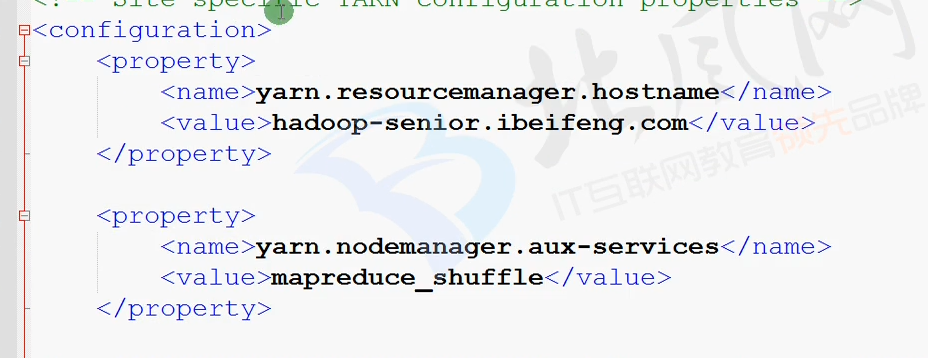
从web上面查看一个作业的运行日志

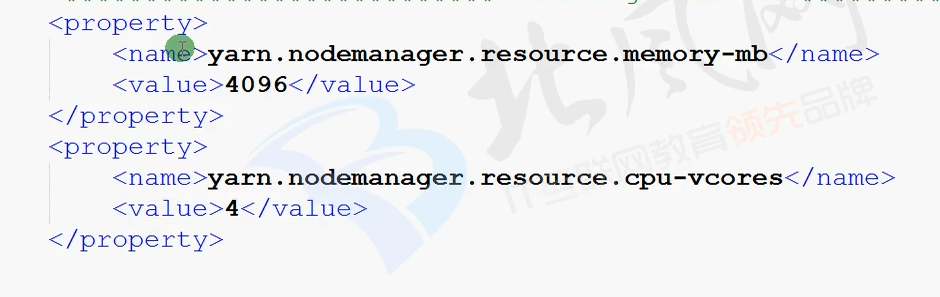
 

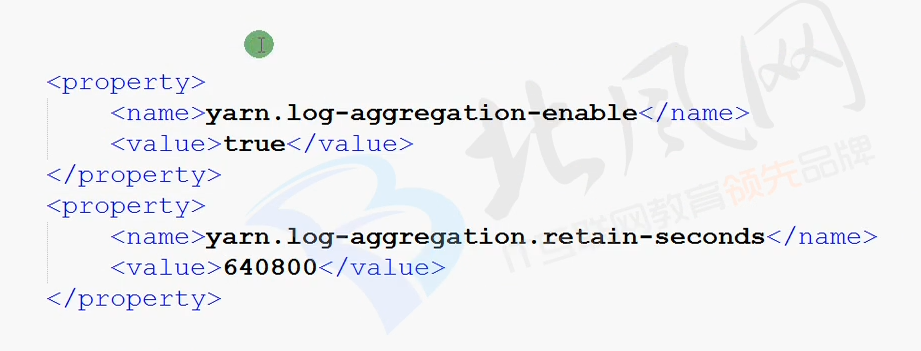
通过web想直接查看其他节点的日志，那么就需要从节点在运行完作业的时候需要将日志信息上传到hdfs。他的配置如下：

Yarn.log-aggregation-enable:true

Yarn.log-aggregation.retain-seconds指定日志保留的时间，单位为妙







配置完yarn了以后

配置mapreduce

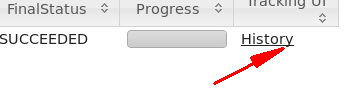
Mapred-eng.sh配置JAVA\_HOME

Mapred-site.xml

配置mapreduce在yarn上运行

Mapreduce.frameword.name:yarn

如果想通过web查看作业的历史



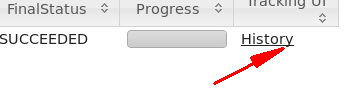
需要配置如下：

Mapreduce.jobhistory.address:master:10020

Mapreduce.jobhistory.webapp.address:master:19888

其实这两个不用配置也是可以的，在mpred-default.xml里面是默认的。

经常出现在web页面上点击



会出现错误，原因就是没有启动historyserver服务，启动的命令如下

Sbin/mr-jobhistrory-daemon.sh start historyserver



Hadoop的启动命令

主节点

Hadoop-daemon.sh start namenode启动namenode

Yarn-daemon.sh start resoucemanager 启动resoucemanager

Mr-jobhistory-daemon.sh start historyserver

启动secondarynamenode

Hadoop-daemon.sh start secondarynamenode

Secondarynamenode 的作用：

Namenode启动的时候会把元数据从磁盘全部加载到内存中，这个时候相当于namenode里面有两份元数据信息，一份在磁盘，一份在内存，在集群运行的时候，如果元数据信息发生变化，元数据直接在内存中进行改变，并且把日志记录下来，磁盘上还是以前的元数据信息，secondarynamenode会定时通过namenode的更新日志去更新磁盘上元数据的信息，进行同步

Namenode

\*fsimage镜像文件

\*edites 编辑日志

从节点

Hadoop –daemon.sh start datanode 启动 datanode

Yarn-daemon.sh start nodemanager 启动nodemanager