

Текстовая расшифровка видео:

НАСТРАИВАЕМ YARN

План:

- YARN;
- Настраиваем YARN;
- Запускаем YARN.

YARN

[YARN](#) – замена встроенному и примитивному шедулеру MapReduce. У него есть промежуточные application-мастера – дополнительные менеджеры, которые запускаются на машинах и отвечают за подгруппу ресурсных контейнеров. Рапортуют не все таски, запущенные в кластере, а только application-менеджеры. Подобная архитектура позволяет YARN масштабироваться на большое количество машин.

Настраиваем YARN

Для того, чтобы правильно настроить YARN необходимо указать для MapReduce, что мы будем использовать YARN в качестве шедулера, также прокинуть переменное окружение и указать местонахождение джарников.

Рассмотрим примеры:

```
etc/hadoop/mapred-site.xml
```



```

<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.application.classpath</name>
    <value>$HADOOP_MAPRED_HOME/share/hadoop/mapreduce/*:$HADOOP_MAPRED_HOME/share/hadoop,
  </property>
</configuration>

```

etc/hadoop/yarn-site.xml

```

<configuration>

  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.nodemanager.env-whitelist</name>
    <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,CLASSPATH_PREPE
  </property>
</configuration>

```

Нам нужно подправить два файла: «mapred-site.xml» и «yarn-site.xml».

Если смотреть на свежескаченный дистрибутив, то вместо этих файлов

```

10 Unless required by applicable law or agreed to in writing, softw
11 distributed under the License is distributed on an "AS IS" BASIS
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
13 See the License for the specific language governing permissions
14 limitations under the License. See accompanying LICENSE file.
15 -->
16
17 <!-- Put site-specific property overrides in this file. -->
18
19 <configuration>
20   <property>
21     <name>dfs.replication</name>
22     <value>1</value>
23   </property>
24 </configuration>

```

могут быть файлы **template** – пустой шаблон (с пустой конфигурацией):

```
Acridity
e/h/hdfs-site.xml => e/h/hadoop-env.sh => e/h/core-site.xml => e/h/mapred-site.xml.template
  hdfs-site.xml
  httpfs-env.sh
  httpfs-log4j.properties
  httpfs-signature.secret
  httpfs-site.xml
  kms-acls.xml
  kms-env.sh
  kms-log4j.properties
  kms-site.xml
  log4j.properties
  mapred-env.cmd
  mapred-env.sh
  mapred-queues.xml.templat
  mapred-site.xml
  mapred-site.xml.template
  slaves
  ssl-client.xml.example
  ssl-server.xml.example
  yarn-env.cmd
  yarn-env.sh
  yarn-site.xml
  include/
/opt/hadoop2 NORMAL etc/hadoop/mapred-site.xml.template xml utf-8 47% 18/21= 1
"etc/hadoop/mapred-site.xml.template" 21L, 758B
[0] 0:nvim* 1:zsh 2:zsh- "pikachu" 15:17 10-июн-23
```

Необходимо создать его копию с правильным именем, например, «mapred-site.xml» и прописать в него параметры:

```
Acridity
e/h/hdfs-site.xml => e/h/hadoop-env.sh => e/h/core-site.xml => e/h/mapred-site.xml
  hdfs-site.xml
  httpfs-env.sh
  httpfs-log4j.properties
  httpfs-signature.secret
  httpfs-site.xml
  kms-acls.xml
  kms-env.sh
  kms-log4j.properties
  kms-site.xml
  log4j.properties
  mapred-env.cmd
  mapred-env.sh
  mapred-queues.xml.templat
  mapred-site.xml
  mapred-site.xml.template
  slaves
  ssl-client.xml.example
  ssl-server.xml.example
  yarn-env.cmd
  yarn-env.sh
  yarn-site.xml
  include/
/opt/hadoop2 NORMAL etc/hadoop/mapred-site.xml xml utf-8 24% 7/29= 1
"etc/hadoop/mapred-site.xml" 29L, 1064B
[0] 0:nvim* 1:zsh 2:zsh- "pikachu" 15:17 10-июн-23
```

Те же самые действия применимы для «yarn-site.xml»:

```
Acridity
e/h/hdfs-site.xml => e/h/hadoop-env.sh => e/h/core-site.xml => e/h/mapred-site.xml => e/h/yarn-site.xml
  hdfs-site.xml
  httpfs-env.sh
  httpfs-log4j.properties
  httpfs-signature.secret
  httpfs-site.xml
  kms-acls.xml
  kms-env.sh
  kms-log4j.properties
  kms-site.xml
  log4j.properties
  mapred-env.cmd
  mapred-env.sh
  mapred-queues.xml.templa
  mapred-site.xml
  mapred-site.xml.template
  slaves
  ssl-client.xml.example
  ssl-server.xml.example
  yarn-env.cmd
  yarn-env.sh
  yarn-site.xml
  include/
/opt/hadoop2 NORMAL etc/hadoop/yarn-site.xml xml utf-8 28% 7/25= 1
"etc/hadoop/yarn-site.xml" 25L, 1016B
[0] 0:nvim* 1:zsh 2:zsh- "pikachu" 15:18 10-июн-23
```

Запускаем YARN

Запускается YARN тоже одной командой: `~$ sbin/start-yarn.sh`

Рассмотрим на примере:

```
meow-nofer@pikachu /opt/hadoop2 $ ./sbin/start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /opt/hadoop2/logs/yarn-meow-nofer-resourcemanager-pikachu.out
localhost: starting nodemanager, logging to /opt/hadoop2/logs/yarn-meow-nofer-nodemanager-pikachu.out
meow-nofer@pikachu /opt/hadoop2 $
```

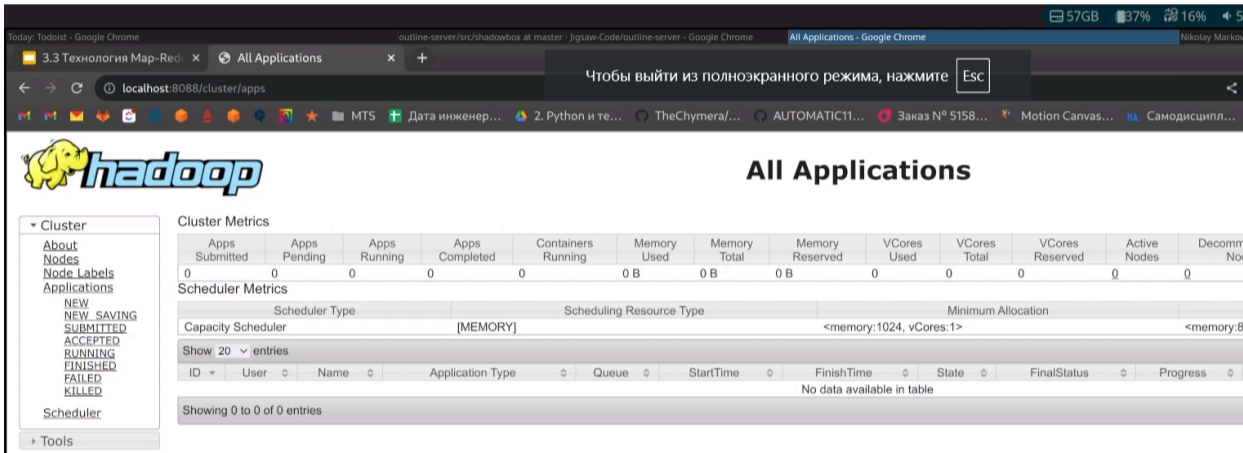
У нас запустятся ожидаемые компоненты. Ранее на схеме, посвященной описанию YARN, мы видели наличие Produce manager, а также автоматический запуск дополнительного Node Manager (отслеживает нагрузку конкретной ноды). Эти две позиции должны быть в коде, их мы и видим.

Проверим jps:

```
meow-nofer@pikachu /opt/hadoop2 $ jps
1359284 ResourceManager
1354705 NameNode
1359980 Jps
1359597 NodeManager
1355034 SecondaryNameNode
1354843 DataNode
meow-nofer@pikachu /opt/hadoop2 $
```

Все работает!

Как и с HDFS есть универсальный интерфейс. Это универсальный интерфейс вся YARN. Когда запускаются приложения и job'ы, мы можем воспользоваться им (называется «Applications»).



Как вам урок?



Изучил, далее >

Слёрм ©

+7 (495) 248-05-80

[Лицензия №ДЛ-1368 от 22.08.2019](#)

[Политика конфиденциальности](#)

[Публичная оферта](#)