

华东师范大学数据科学与工程学院实验报告

课程名称：分布式模型与编程	年级：2017	上机实践成绩：
指导教师：徐辰	姓名：熊双宇	学号：10174102103
上机实践名称：Flink部署与编程		上机实践日期：2019.11.24-2019.12.04【第13-14周】
上机实践编号：实验四	组号：11	上机实践时间：18:00-19:30*

一. 实验目的

- 学习Flink的部署，理解Flink的体系架构
- 练习Flink DataStream的编程
- 深入理解Flink批流融合的工作机制

二. 实验任务

- [Storm部署](#)【第10周】：单机集中式、单机伪分布式（在个人用户下独立完成）、分布式（多位同学新建一个相同的用户，例如ecnu，协作完成）
- [Storm编程](#)【第10周】

三. 使用环境

1. Ubuntu18.04
2. flink-1.7.2
3. hadoop-2.9.2

四. 实验过程

Flink 部署

1 单机集中式部署

1.1 准备工作

- 操作系统的安装详情参考: [Prepare.md](#)
- 安装Flink: 去[官网](#)下载Flink压缩包，并解压。本此实验以flink-1.7.2-bin-hadoop28-scala_2.11.tgz为例。

```
1 >>> tar -zxvf flink-1.7.2-bin-hadoop28-scala_2.11.tgz
2 >>> mv flink-1.7.2 ~
```

1.2 运行Flink DataStream程序

- 使用shell运行DataStream程序
 - 本地模式启动Scala-Shell

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ export TERM=xterm-color
syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ ~/flink-1.7.2/bin/start-scala-shell.sh
local
Starting Flink Shell:

Starting local Flink cluster (host: localhost, port: 8081).

Connecting to Flink cluster (host: localhost, port: 8081).
```

- 在 `scala>` 后输入 scala 代码

```

1 val textstreaming=senv.fromElements("a a b b c")
2 val countsstreaming=textstreaming.flatMap { _.toLowerCase.split("\\W+") } .map { (_, 1)
  }.keyBy(0).sum(1)
3 countsstreaming.print()
4 senv.execute() // 提交作业

```

运行结果如下图所示:

```

scala> val textstreaming=senv.fromElements("a a b b c")
textstreaming: org.apache.flink.streaming.api.scala.DataStream[String] = org.apache.fli
nk.streaming.api.scala.DataStream@48dbf9d

scala> val countsstreaming=textstreaming.flatMap { _.toLowerCase.split("\\W+") } .map
{ (_, 1) }.keyBy(0).sum(1)
countsstreaming: org.apache.flink.streaming.api.scala.DataStream[(String, Int)] = org.
apache.flink.streaming.api.scala.DataStream@73a706b1

scala> countsstreaming.print()
res2: org.apache.flink.streaming.api.datastream.DataStreamSink[(String, Int)] = org.ap
ache.flink.streaming.api.datastream.DataStreamSink@5b00d92c

scala> senv.execute()
(a,1)
(a,2)
(b,1)
(b,2)
(c,1)
res3: org.apache.flink.api.common.JobExecutionResult = org.apache.flink.api.common.Job
ExecutionResult@2dbc56fb

scala>

```

- 通过提交 jar 包运行DataStream程序

flink不可以直接提交jar包运行, 首先仍需要在 终端1 本地模式启动Scala-Shell之后另起终端, 提交jar包

- 默认模式提交

终端2 中启动socket服务作为数据源

```

syx@syx-OptiPlex-7050:~$ nc -l 9000
I have a pen
An apple pen
twinkle twinkle little star
how I wonder what you are

```

终端3 中提交jar包

```

syx@syx-OptiPlex-7050:~$ ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/stre
aming/SocketWindowWordCount.jar --port 9000
Starting execution of program

```

向 终端2 中输入数据进行wordcount计算, 在 终端1 运行结果如下图所示:

```

scala> I : 1
pen : 1
a : 1
have : 1
An : 1
pen : 1
apple : 1
twinkle : 2
star : 1
little : 1
how : 1
are : 1
you : 1
what : 1
wonder : 1
I : 1

```

在运行过程中另起一个终端执行 jps 查看进程, 此时会出现 CliFrontend 进程, 计算运行结束后该进程消失

```

syx@syx-OptiPlex-7050:~$ jps
29132 Jps
28381 FlinkShell
28751 CliFrontend
4255 Main

```

DataStream程序终止的两种方法

- 在WebUI上 `cancel`
- 命令行终止 `flink cancel JobID` , `JobID` 通过 `flink list` 查询

```
syx@syx-OptiPlex-7050:~$ ~/flink-1.7.2/bin/flink list
Waiting for response...
----- Running/Restarting Jobs -----
28.11.2019 19:05:29 : 52c287c2f1f043fbae46955c0150432e : Socket Window WordCount
(RUNNING)
-----
No scheduled jobs.

syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ flink cancel 52c287c2f1f043fbae46955c0150432e
Cancelling job 52c287c2f1f043fbae46955c0150432e.
Cancelled job 52c287c2f1f043fbae46955c0150432e.
```

注：杀掉客户端进程是无法停止程序的

- detached模式提交

终端2 中启动本地服务

```
syx@syx-OptiPlex-7050:~$ nc -l 9000
sky like your eye
the answer is blown in the wind
life sucks!!!
life sucks !!!
I will win !!!
```

终端3 中提交程序jar包

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
Job has been submitted with JobID d0118aab5deaf6dc06c6977d718a6012
```

程序提交完后会自动退出客户端，不再打印作业进度等信息

向 终端2 中输入数据进行wordcount计算，在 终端1 运行结果如下图所示：

```
scala> sky : 1
eye : 1
your : 1
like : 1
the : 2
wind : 1
in : 1
blown : 1
is : 1
answer : 1
life : 1
sucks!!! : 1
life : 1
! : 3
sucks : 1
I : 1
! : 3
win : 1
will : 1
```

在运行过程中另起一个终端执行 `jps` 查看进程，此时不会出现 `CliFrontend` 进程

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ jps
31556 Jps
30810 FlinkShell
4255 Main
```

DataStream程序终止的两种方法

- 在WebUI上 `cancel`
- 命令行终止 `flink cancel JobID` , `JobID` 通过 `flink list` 查询

```

syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ flink list
Waiting for response...
----- Running/Restarting Jobs -----
28.11.2019 19:18:09 : ffb4f0b82af298630f7162d07575f6e4 : Socket Window WordCount
(RUNNING)
-----
No scheduled jobs.

syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ flink cancel ffb4f0b82af298630f7162d07575f6e4
Cancelling job ffb4f0b82af298630f7162d07575f6e4.
Cancelled job ffb4f0b82af298630f7162d07575f6e4.

```

注：杀掉客户端进程是无法停止程序的

输入 `:q` 退出Scala-shell

```

scala> sky : 1
eye : 1
your : 1
like : 1
the : 2
wind : 1
in : 1
blown : 1
is : 1
answer : 1
life : 1
sucks!!! : 1
life : 1
! : 3
sucks : 1
I : 1
! : 3
win : 1
will : 1
:q
good bye ..

```

2 单机伪分布式部署

2.1 准备工作

- 完成[单机集中式部署](#)

2.2 修改Flink配置

- 更改配置文件flink-conf.yaml

```
1 | >>> vim ~/flink-1.7.2/conf/flink-conf.yaml
```

修改如下：

```

1 | jobmanager.rpc.address: localhost #配置JobManager进行RPC通信的地址，使用默认即可
2 | jobmanager.rpc.port: 6123        #配置JobManager进行RPC通信的端口，使用默认即可
3 | rest.port: 8081 # 客户端访问端口与可视化端口，使用默认值即可
4 | taskmanager.numberOfTaskSlots: 2 #配置TaskManager 提供的任务 slots 数量大小，默认为1
5 | taskmanager.memory.preallocate: false #配置是否在Flink集群启动时候给TaskManager分配内存，默认不进行预分配，
   | 这样在我们不适用flink集群时候不会占用集群资源
6 | parallelism.default: 1 # 配置程序默认并行计算的个数，默认为1

```

注意：flink-conf.yaml中配置key/value时候在“:”后面需要有一个空格，否则配置不会生效。注释与配置项不要在同一行

以下还有一些非常重要的配置值（需要调节时更改，本例中不做更改）：

每个JobManager（jobmanager.heap.mb）的可用内存量

每个TaskManager（taskmanager.heap.mb）的可用内存量

每台机器的可用CPU数量（taskmanager.numberOfTaskSlots）

集群中的CPU总数（parallelism.default）

临时目录（taskmanager.tmp.dirs）#内存不够用时，写入到taskmanager.tmp.dirs指定的目录中。如果未显式指定参数，Flink会将临时数据写入操作系统的临时目录。

- 更改配置文件slaves

```
1 >>>vi ~/flink-1.7.2/conf/slaves
2 # 文件中默认内容为localhost，本例中不做修改
```

2.3 启动Flink服务

- 启动命令

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ start-cluster.sh
Starting cluster.
Starting standalone session daemon on host syx-OptiPlex-7050.
Starting task executor daemon on host syx-OptiPlex-7050.
```

- 查看进程，验证是否成功启动服务
 - 使用jps命令，因为在此单机伪分布式部署模式下，该节点既充当JobManager角色，又充当TaskManager角色，故该节点上会有两个进程：一个JobManager进程和一个TaskManager进程。若同时出现JobManager进程和TaskManager进程，则表明配置成功以及启动成功。

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ jps
2321 StandaloneSessionClusterEntrypoint
2935 Jps
2795 TaskManagerRunner
4255 Main
```

在standalone模式下，Jobmanager的进程名为StandaloneSessionClusterEntrypoint

- 在 <http://localhost:8081> (端口号为在配置文件 `flink-conf.yaml` 中设置的 `rest.port`) 确认flink是否正常运行

Overview

Version: 1.7.2

Commit: ceba8af

1

Task Managers

2

Task Slots

1

Available Task Slots

Total Jobs

Running1

Finished1

Canceled0

Failed0

Running Jobs

Start Time	End Time	Duration	Job Name	Job ID	Tasks	Status
2019-11-28, 19:46:12	2019-11-28, 19:51:49	5m 36s	Socket Window WordCount	0d65df2b40607e502f81f0287e92adb2	2000200000	RUNNING

Completed Jobs

Start Time	End Time	Duration	Job Name	Job ID	Tasks	Status
2019-11-28, 19:43:36	2019-11-28, 19:43:36	240ms	Flink Streaming Job	47f734fc3ddcb69bcf10edb32018104a	3000030000	FINISHED

因为本例在flink-conf.yaml中设置taskmanager.numberOfTaskSlots的值为2，故每个TaskManager有2个slot。

2.4 运行Flink DataStream程序

- 使用shell运行DataStream程序
 - 远程模式启动Scala-Shell

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ export TERM=xterm-color
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ ~/flink-1.7.2/bin/start-scala-shell.sh
remote localhost 8081
Starting Flink Shell:

Connecting to Flink cluster (host: localhost, port: 8081).
```



- 在 `scala>` 后输入 scala 代码

运行结果如下图所示：

```
scala> val textstreaming=senv.fromElements("a a b b c")
textstreaming: org.apache.flink.streaming.api.scala.DataStream[String] = org.apa
che.flink.streaming.api.scala.DataStream@19c1f6f4

scala> val countsstreaming=textstreaming.flatMap { _.toLowerCase.split("\\W+") }
.map { (_, 1) }.keyBy(0).sum(1)
countsstreaming: org.apache.flink.streaming.api.scala.DataStream[(String, Int)]
= org.apache.flink.streaming.api.scala.DataStream@7916568d

scala>

scala> countsstreaming.print()
res0: org.apache.flink.streaming.api.datastream.DataStreamSink[(String, Int)] =
org.apache.flink.streaming.api.datastream.DataStreamSink@4c0989f7

scala> senv.execute()
res1: org.apache.flink.api.common.JobExecutionResult = org.apache.flink.api.common
on.JobExecutionResult@3b75fdd0
```

另起终端中打开log目录下的out文件会显示flink的执行结果

```

syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ ls
flink-syx-client-syx-OptiPlex-7050.log
flink-syx-scala-shell-local-syx-OptiPlex-7050.log
flink-syx-scala-shell-remote-syx-OptiPlex-7050.log
flink-syx-standalonesession-0-syx-OptiPlex-7050.log
flink-syx-standalonesession-0-syx-OptiPlex-7050.log.1
flink-syx-standalonesession-0-syx-OptiPlex-7050.log.2
flink-syx-standalonesession-0-syx-OptiPlex-7050.out
flink-syx-standalonesession-0-syx-OptiPlex-7050.out.1
flink-syx-standalonesession-0-syx-OptiPlex-7050.out.2
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log.1
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log.2
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out.1
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out.2
syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ tail -f flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
(a,1)
(a,2)
(b,1)
(b,2)
(c,1)

```

输入 `:q` 退出Scala-Shell

- 通过提交 jar 包运行DataStream程序
 - 默认模式提交

终端1 中启动本地服务

```

syx@syx-OptiPlex-7050:~$ nc -l 9000
nc once agin
nc once again
I have nothing to see
no story

```

终端2 中提交程序jar包

```

syx@syx-OptiPlex-7050:~$ ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program

```

终端3 中打开log目录下的out文件会显示flink的执行结果

向 终端1 中输入数据进行wordcount计算, 在 终端3 运行结果如下图所示:

```

syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ tail -f flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
nc : 1
again : 1
once : 1
I : 1
story : 1
no : 1
see : 1
to : 1
nothing : 1
have : 1

```

在运行过程中另起一个终端执行 `jps` 查看进程 此时会出现 `CliFrontend` 进程, 计算运行结束后该进程消失

```

syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ jps
2321 StandaloneSessionClusterEntrypoint
2935 Jps
2795 TaskManagerRunner
4255 Main

```

DataStream程序终止的两种方法

- 在WebUI上 `cancel`

Overview

Version: 1.7.2

Commit: ceba8af

Task Managers

1

Task Slots

2

Available Task Slots

1

Total Jobs

Running	1
Finished	1
Canceled	0
Failed	0

Running Jobs

Start Time	End Time	Duration	Job Name	Job ID	Tasks	Status
2019-11-28, 19:46:12	2019-11-28, 19:51:49	5m 36s	Socket Window WordCount	0d65df2b40607e502f81f0287e92adb2	2 0 0 0 2 0 0 0 0	RUNNING

Completed Jobs

Start Time	End Time	Duration	Job Name	Job ID	Tasks	Status
2019-11-28, 19:43:36	2019-11-28, 19:43:36	240ms	Flink Streaming Job	47f734fc3ddcb69bcf10edb32018104a	3 0 0 0 0 3 0 0 0	FINISHED

Socket Window WordCount

0d65df2b40607e502f81f0287e92adb2

2019-11-28, 19:46:12

6m 38s

0 2 0 0 0 0 0 0 0

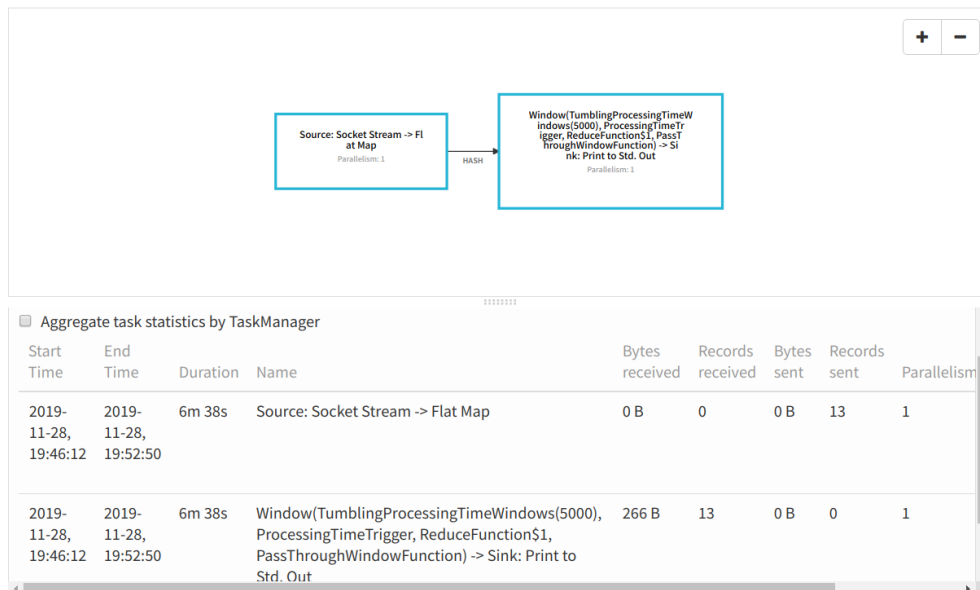
Cancel

Overview

Timeline

Exceptions

Configuration



- 命令终止 `flink cancel JobID` , `JobID` 通过 `flink list` 查询

注：杀掉客户端进程是无法停止程序的

- detached模式提交

终端1 中启动本地服务

```
syx@syx-OptiPlex-7050:~$ nc -l 9000
sky like your eye
the answer is blown in the wind
life sucks!!!
life sucks !!!
I will win !!!
```

终端2 中提交程序jar包

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
Job has been submitted with JobID d0118aab5deaf6dc06c6977d718a6012
```

程序提交完后会自动退出客户端，不再打印作业进度等信息

终端3 中打开log目录下的out文件会显示flink的执行结果


```

syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ ls
flink-syx-client-syx-OptiPlex-7050.log
flink-syx-scala-shell-local-syx-OptiPlex-7050.log
flink-syx-scala-shell-remote-syx-OptiPlex-7050.log
flink-syx-standalonesession-0-syx-OptiPlex-7050.log
flink-syx-standalonesession-0-syx-OptiPlex-7050.log.1
flink-syx-standalonesession-0-syx-OptiPlex-7050.log.2
flink-syx-standalonesession-0-syx-OptiPlex-7050.out
flink-syx-standalonesession-0-syx-OptiPlex-7050.out.1
flink-syx-standalonesession-0-syx-OptiPlex-7050.out.2
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log.1
flink-syx-taskexecutor-0-syx-OptiPlex-7050.log.2
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out.1
flink-syx-taskexecutor-0-syx-OptiPlex-7050.out.2
syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ tail -f flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
(a,1)
(a,2)
(b,1)
(b,2)
(c,1)

```

向 `终端1` 中输入数据进行wordcount计算，在 `终端3` 运行结果如下图所示：

```

scala> sky : 1
eye : 1
your : 1
like : 1
the : 2
wind : 1
in : 1
blown : 1
is : 1
answer : 1
life : 1
sucks!!! : 1
life : 1
! : 3
sucks : 1
I : 1
! : 3
win : 1
will : 1

```

在运行过程中另起一个终端执行 `jps` 查看进程 此时不会出现 `CliFrontend` 进程

```

syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ jps
2321 StandaloneSessionClusterEntrypoint
2935 Jps
2795 TaskManagerRunner
4255 Main

```

DataStream程序终止的两种方法

- 在WebUI上 `cancel`
- 命令行终止 `flink cancel JobID`，`JobID` 通过 `flink list` 查询

注：杀掉客户端进程是无法停止程序的

2.5 停止Flink服务

- 停止命令

```
1 >>> ~/flink-1.7.2/bin/stop-cluster.sh
```

```
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ stop-cluster.sh
Stopping taskexecutor daemon (pid: 7592) on host syx-OptiPlex-7050.
Stopping standalone-session daemon (pid: 7125) on host syx-OptiPlex-7050.
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ jps
9908 Jps
4255 Main
```

- 查看进程，验证是否成功停止服务
 - 若成功停止，JobManager进程和TaskManager进程应消失。

3 分布式部署

3.1 准备工作

请确认是否已完成以下内容：

- 有至少两台的服务端器每台机器上的已完成 [Prepare.md](#)
- 服务器之间实现[免密登录](#)
- 在其中一台机器上完成[单机集中式部署](#)
- 已完成 [HDFS v2 分布式部署](#)并启动（由于程序的输入输出需要）

3.2 修改配置文件

- 更改配置文件flink-conf.yaml

```
1 | >>> vi ~/flink-1.7.2/conf/flink-conf.yaml
```

主要的修改内容有：

```
1 | jobmanager.rpc.address: 219.228.135.207 #配置JobManager进行RPC通信的地址
```

说明：此处的Master应用实际配置过程中Master的ip地址代替；其余部分配置信息在伪分布式部署已经列出，按需配置。

- 更改配置文件slaves

`vi conf/slaves` 文件中默认内容为localhost，本例中修改为：

```
1 | 219.228.135.207
2 | 219.228.135.42
```

说明：此处的Slave应用实际配置过程中Worker的ip地址代替

- 将配置好的Flink同步到其他节点

```
1 | scp -r flink-1.7.2 219.228.135.42:/home/ecnu/
```

3.3 启动flink服务

- 启动命令

```
1 | ~/flink-1.7.2/bin/start-cluster.sh
```

- 查看进程，验证是否成功启动服务
 - 因为在此分布式部署模式下，Master节点充当Master角色，各Slaves节点充当Worker角色，故在Master节点上会存在一个JobManager进程，各Slaves节点上会存在一个TaskManager进程。分别在Master和Slaves上使用jps命令，若在Master上出现StandaloneSessionClusterEntrypoint进程，且在Slaves上出现TaskManagerRunner进程，则表明配置成功且启动成功。如下图所示：

```
ecnu@may-lab:/home/ecnu/hadoop-2.9.2$ jps
29345 Jps
21249 TaskManagerRunner
29204 SecondaryNameNode
28955 DataNode
20734 StandaloneSessionClusterEntrypoint
28751 NameNode
```

```
ecnu@syx-OptiPlex-7050:~$ jps
9172 DataNode
10181 Jps
10022 TaskManagerRunner
9323 NodeManager
```

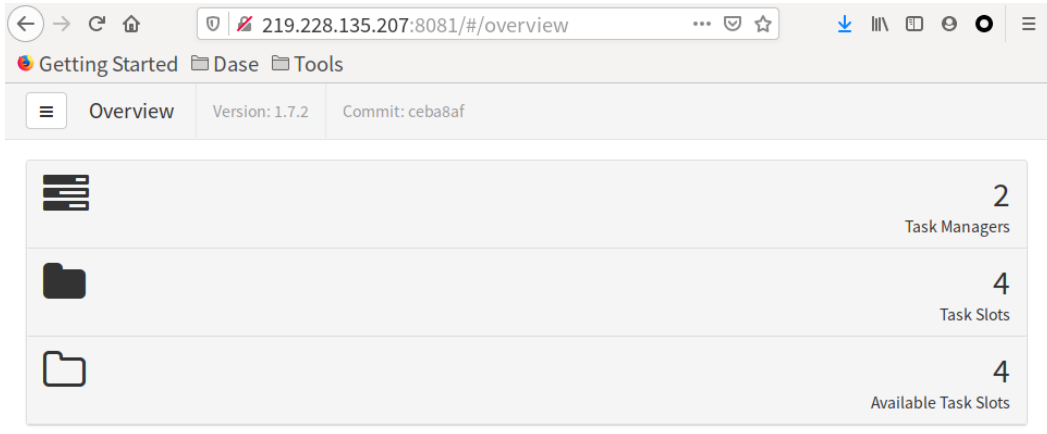
在standalone模式下，Jobmanager进程名为StandaloneSessionClusterEntrypoint

- 查看flink服务信息

- 查看flink服务日志

日志信息在 /flink-1.7.2/log 目录

- 访问flink web界面



当前有2个TaskManager（即Slave1、Slave2），因为更改配置文件taskmanager.numberOfTaskSlots项其为2，故Task Slots的总数为4。

3.4 运行Flink DataStream程序

- 通过提交 jar 包运行DataStream程序

- 默认模式提交，可以在客户端看到应用程序运行过程中的信息

```
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.207 --port 9001
Starting execution of program
```

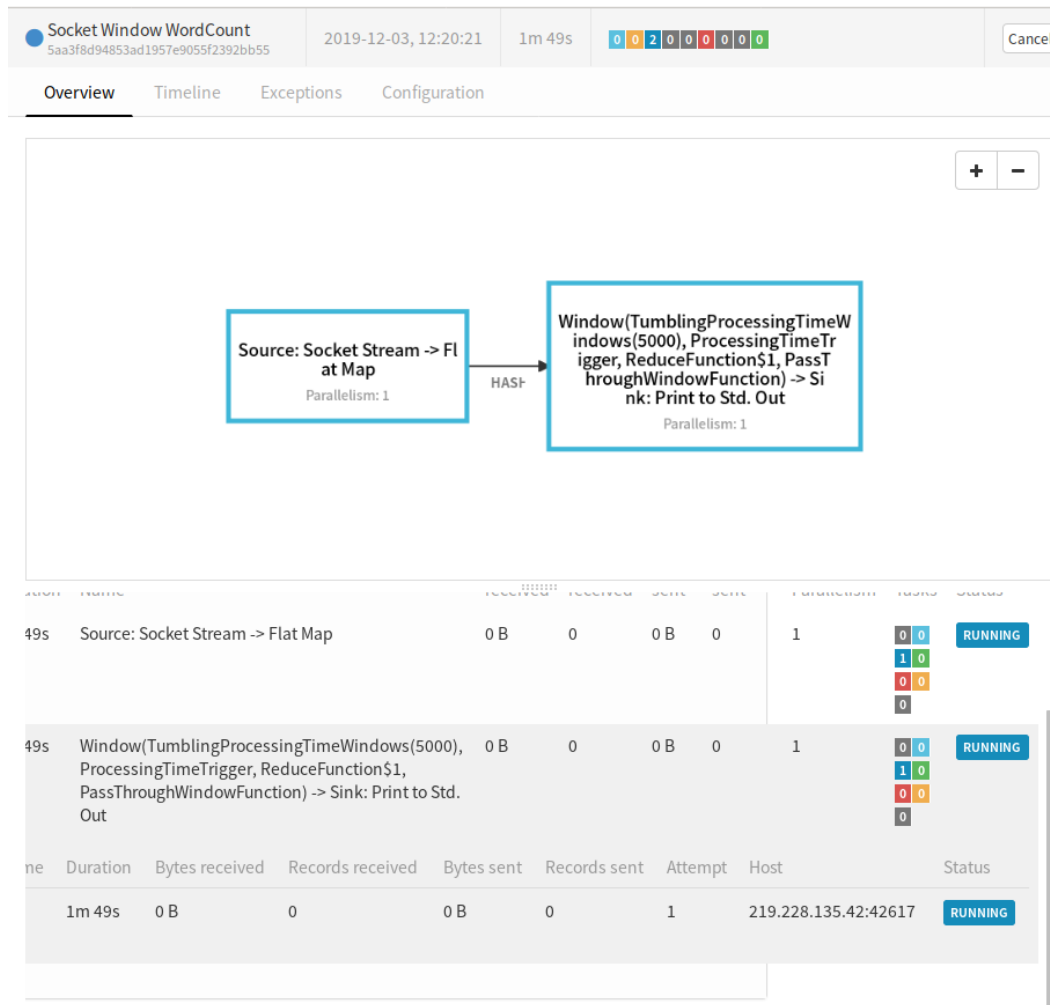
使用jps命令，默认模式提交出现CliFrontend进程，用于提交作业并接受返回信息，应用程序运行结束后该进程消失。

```
may-lab:/home/ecnu/flink-1.7.2/log$ jps
TaskManagerRunner
SecondaryNameNode
CliFrontend
Jps
DataNode
NameNode
StandaloneSessionClusterEntrypoint
```

在程序运行位置的flink log目录下输入命令：tail -f flink-xxx-taskexecutor-x-xxx.out，查看运行结果：

```
ecnu@may-lab:/home/ecnu/flink-1.7.2/log$ nc -l 9001
I have a pen
ecnu@may-lab:/home/ecnu/flink-1.7.2/log$ tail -f flink-ecnu-taskexecutor-0-may-lab.out.3
I : 1
pen : 1
a : 1
have : 1
```

访问flink web界面查看程序运行位置



DataStream程序终止的两种方法

- 在WebUI上 cancel
- 命令行终止 `flink cancel JobID`, JobID 通过 `flink list` 查询

注: 杀掉客户端进程是无法停止程序的

- detached模式提交, 在客户端看不到应用程序运行过程中的信息
- 在程序运行位置的flink log目录下输入命令: `tail -f flink-xxx-taskexecutor-x-xxx.out`, 查看运行结果:

```
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.207 --port 9001
Starting execution of program
Job has been submitted with JobID 980e92b42472322d47c8b93a2571a960

ecnu@may-lab:/home/ecnu/flink-1.7.2/log$ nc -l 9001
and you have a pencil

y-lab.out.1
and : 1
pencil : 1
a : 1
have : 1
you : 1
```

程序提交完毕后退出客户端

detached模式下无CliFrontend进程

```

ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ ~/flink-1.7.2/bin/flink run -d ~/f
flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --hostname 219.228.
135.207 --port 9001
Starting execution of program
Job has been submitted with JobID 21ba9893f0f354a79ca6d24e87f220db
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ jps
12641 Jps
10632 StandaloneSessionClusterEntrypoint
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ecnu@syx-OptiPlex-7050:~$ jps
19684 TaskManagerRunner
19932 Jps

```

访问flink web界面查看程序运行位置

Getting Started Dase Tools

Socket Window WordCount
21ba9893f0f354a79ca6d24e87f220db 2m 27s Cancel

Overview Timeline Exceptions Configuration

Source: Socket Stream -> Flat Map
Parallelism: 1

HASh

Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction) -> Sink: Print to Std. Out
Parallelism: 1

Metrics by TaskManager

TaskManager	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2-03, 6	2m 27s	Source: Socket Stream -> Flat Map	0 B	0	0 B	0	1	0/0	RUNNING
2-03, 6	2m 27s	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction) -> Sink: Print to Std. Out	0 B	0	0 B	0	1	0/0	RUNNING

- 停止flink正在运行中的程序
 - 在flink web界面中，进入running jobs，选取正在运行的程序，点击右上角cancel按钮
 - 使用命令行停止。先使用命令./bin/flink list获取正在运行的程序及其ID，再使用命令./bin/flink cancel ID根据程序ID停止程序

```
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ flink list
Waiting for response...
----- Running/Restarting Jobs -----
03.12.2019 13:32:43 : 37b1ead67f3cdc6ffdc46d08ab754855 : Socket Window WordC
ount (RUNNING)
-----
No scheduled jobs.
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ flink cancel 37b1ead67f3cdc6ffdc46d
08ab754855
Cancelling job 37b1ead67f3cdc6ffdc46d08ab754855.
Cancelled job 37b1ead67f3cdc6ffdc46d08ab754855.
```

3.5 停止Flink服务

- 停止命令

```
1 | >>> ~/flink-1.7.2/bin/stop-cluster.sh
```

- 查看进程，验证是否成功停止服务

- 若成功停止，JobManager进程和TaskManager进程应消失，如下图所示：

```
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ flink cancel 37b1ead67f3cdc6ffdc46d
08ab754855
Cancelling job 37b1ead67f3cdc6ffdc46d08ab754855.
Cancelled job 37b1ead67f3cdc6ffdc46d08ab754855.
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ stop-cluster.sh
Stopping taskexecutor daemon (pid: 31503) on host may-lab.
Stopping taskexecutor daemon (pid: 26927) on host syx-OptiPlex-7050.
Stopping standalone session daemon (pid: 30969) on host may-lab.
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$ jps
1984 Jps
ecnu@may-lab:/home/ecnu/flink-1.7.2/bin$

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ecnu@syx-OptiPlex-7050:~$ jps
27731 Jps
```

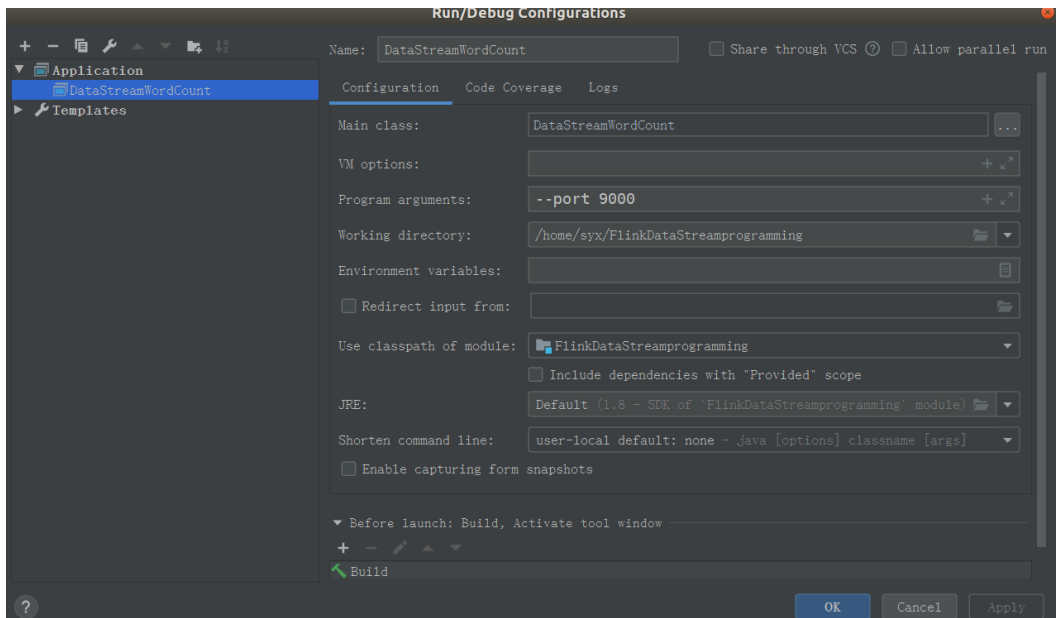
Flink编程

1. 编写Flink程序

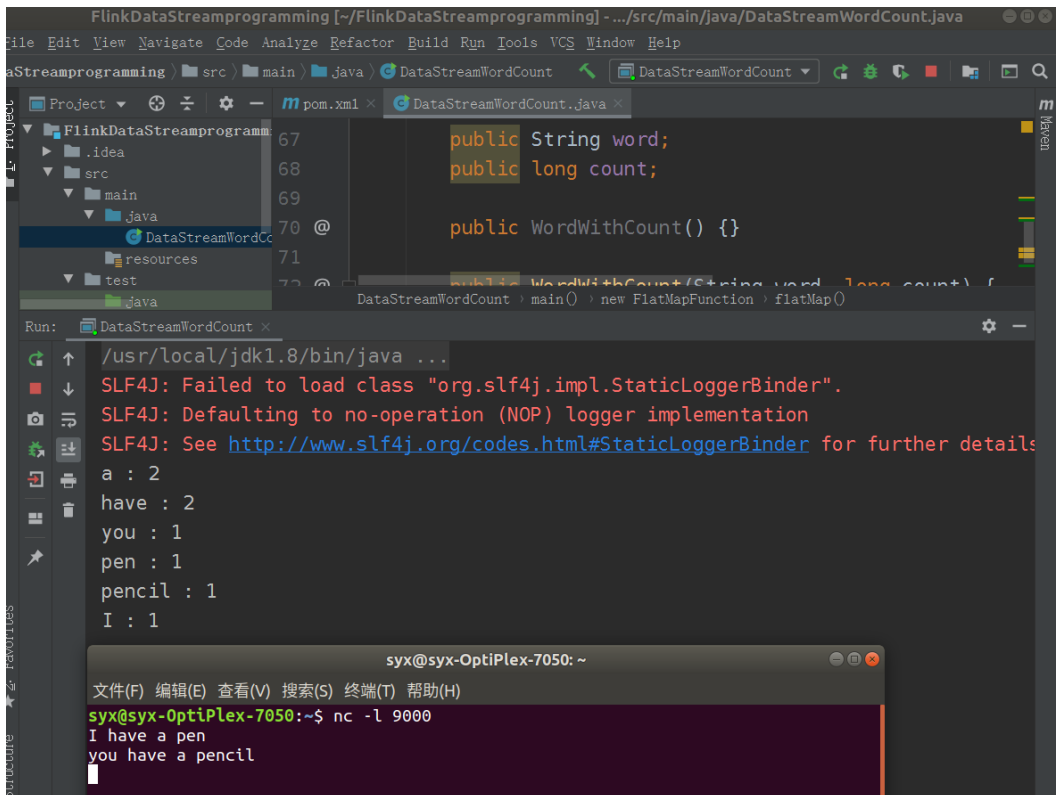
- 新建Maven项目并添加pom依赖
- IDE环境编写代码
 - 编写java代码，在src->main->java目录下新建名为DataStreamWordCount的Java类，并编写代码

2. 调试Flink程序

- IDE中直接运行
 - 配置运行环境，并进行本地调试。在Intellij菜单栏中选择Run->Edit Configuration，在弹出对话框中新建Application配置，配置Main Class为DataStreamWordCount，Program arguments为hostname port，分别为主机名和端口号，默认主机名为localhost。如下图所示：



- 配置完成后，右键->Run'DataStreamWordCount'
- 输出：



- 调试经验
 - IDE中设置断点

3. 运行Flink程序

- 利用IDE打包jar文件
- 伪分布模式下提交Flink程序
 - 在终端输入命令，向jobmanager提交作业
 - 另起终端输入如下命令查看运行结果, 如下所示：

```
syx@syx-OptiPlex-7050: ~/flink-1.7.2/bin
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
syx@syx-OptiPlex-7050:~/flink-1.7.2/bin$ ./flink run -c DataStreamWordCount /home/syx/FlinkDataStreamprogramming/out/artifacts/DataStreamWordCount/DataStreamWordCount.jar --hostname localhost --port 9000

syx@syx-OptiPlex-7050: ~/flink-1.7.2/log
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
syx@syx-OptiPlex-7050:~/flink-1.7.2/log$ tail -f flink-syx-taskexecutor-0-syx-OptiPlex-7050.out
star : 1
little : 1
: 1
so : 1
what : 1
do : 1
message : 1
the : 1
recieve : 1
you : 1
: 1
and : 1
this? : 1
: 1
another : 1
: 1
? : 1
message : 1
one : 1
Facinating! : 1
: 1
you : 1
storm! : 1
boy! : 1
good : 1
are : 1

syx@syx-OptiPlex-7050: ~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
syx@syx-OptiPlex-7050:~$ nc -l 9000
I have a pen
you have a pencil
an apple pen
pine^C
syx@syx-OptiPlex-7050:~$ nc -l 9000
twinkle twinkle little star
how I wonder what you are

so what
do you recieve the message

and this?

another one message ?

Facinating!

you are good boy! storm!
```

• 分布式模式下提交flink程序

- 首先是上传jar包，执行scp命令将jar包上传至集群

```
1 | scp flinkprogram.jar ecnu@219.228.135.41:/home/ecnu
```

- 在client中输入命令,向集群中的jobmanager提交作业

另起终端，在flink log目录下输入如下命令，查看结果.out, 如下所示:

```
ecnu@may-lab: /home/ecnu/flink-1.7.2
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ecnu@may-lab: /home/ecnu/flink-1.7.2$ ./bin/flink run -m 219.228.135.207:8081 -c DataStreamWordCount /home/ecnu/DataStreamWordCount.jar --hostname 219.228.135.207 --port 9000
Starting execution of program

0 个安全更新。

Your Hardware Enablement Stack (HWE) is supported until April 2023.
*** 需要重启系统 ***
Last login: Thu Dec 5 18:43:56 2019 from 219.228.135.42
ecnu@may-lab:~$ nc -l 9000
hello world!
I am doing my homework
there is a river in the blue sky
ecnu@may-lab:~$ nc -l 9000
hello world!
I am doing my homework
there is a river in the blue sky

hello world!
I am doing my homework
there is a river in the blue sky

the bird is singing!

look at me !
how many !: !!!!
! ! !

ecnu@may-lab: /home/ecnu/flink-1.7.2/log
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ecnu@may-lab: /home/ecnu/flink-1.7.2/log$ tail -f flink-ecnu-taskexecutor-0-may-lab.out
river : 1
a : 1
is : 1
there : 1
homework : 1
my : 1
doing : 1
am : 1
I : 1
world! : 1
the : 1
singing! : 1
is : 1
bird : 1
: 2
look : 1
! : 1
me : 1
at : 1
how : 1
!!!! : 1
! : 1
many : 1
! : 3
```