

华东师范大学数据科学与工程学院实验报告

课程名称：分布式系统与编程	年级：2017 级	上机实践成绩：
指导教师：徐辰	姓名：吴双	
上机实践名称：Flink 部署与编程	学号：10164102141	上机实践日期：2019/11/20
上机实践编号：04	组号：01	上机实践时间：Week 13-14

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

- Flink 部署【第 13 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）
- Flink 编程【第 14 周】

三、使用环境

- Hadoop 2.9.2
- Ubuntu LTS 18.04
- Flink 1.7.2

四、实验过程

Flink 部署

1 单机集中式部署

1.1 准备工作并运行 Flink 批处理程序

- 使用 shell 运行批处理程序
 - 本地模式启动 Scala-Shell

遇到问题：

```
HINT: You can only print a DataStream to the shell in local mode.
2.3 启动Flink服务
[ERROR] Failed to construct terminal; falling back to unsupported
java.lang.NumberFormatException: For input string: "0x100"
    at java.lang.NumberFormatException.forInputString(NumberFormatException.java:65)
    at java.lang.Integer.parseInt(Integer.java:580)
    at java.lang.Integer.valueOf(Integer.java:766) scala 代码
3 at scala.tools.jline_embedded.internal.InfoCmp.parseInfoCmp(InfoCmp.java:59)
  at scala.tools.jline_embedded.UnixTerminal.parseInfoCmp(UnixTerminal.java:247)
  at scala.tools.jline_embedded.UnixTerminal.<init>(UnixTerminal.java:65)
  at scala.tools.jline_embedded.UnixTerminal.<init>(UnixTerminal.java:50) Case
```

查询 [stackoverflow](#) 得知，此问题出现的原因是因为在 Ubuntu LTS18.04 中，系统会尝试将 Scala 放进一个 tmux session 中运行，而 tmux 会将 TERM 参数设置为 xterm-256color。

所以解决方法为在/etc/profile 或者 shell 的 zsh 启动脚本 ~/.zshrc 中设置一个 export 的命令，将 TERM 设置成 xterm-color（这里使用了后者，即在 shell 启动脚本中设置）：

```
81 export JAVA_HOME=/usr/local/jdk1.8
82 export JRE_HOME=${JAVA_HOME}/jre
83 export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
84 export PATH=.:${JAVA_HOME}/bin:$PATH
85 export TERM=xterm-color
```

重新启动 shell，在运行 scala-shell，成功。结果如下：

```
Streaming - Use the 'senv' variable

* val dataStream = senv.fromElements(1, 2, 3, 4)
* dataStream.countWindowAll(2).sum(0).print()
* senv.execute("My streaming program")

2 The file /
package: n
be either a

HINT: You can only print a DataStream to the shell in local mode.

scala>
```

- 在 scala> 后输入 scala 代码。运行结果如下：

```
scala> val text=senv.fromElements("a a b b c")
text: org.apache.flink.api.scala.DataSet[String] = org.apache.flink.api.scala.DataSet@2ea0161f

scala> val counts = text.flatMap { _.toLowerCase.split("\\W+") }.map { (_, 1) }.groupBy(0).sum(1)
counts: org.apache.flink.api.scala.AggregateDataSet[(String, Int)] = org.apache.flink.api.scala.AggregateDataSet@1464a177

scala> counts.print()
(a,2)
(b,2)
(c,1)
```

- 通过提交 jar 包运行批处理程序

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。

另起一个终端，提交 jar 包

- 默认模式提交

```
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run ~/Downloads/flink-1.7.2/
examples/batch/WordCount.jar
Starting execution of program
Executing WordCount example with default input data set.
Use --input to specify file input.
Printing result to stdout. Use --output to specify output path.
(a,5)
(action 1)
```

在运行过程中另起一个终端执行 jps 查看进程：

```
@wushuangyoyo jps
21840 CliFrontend
20290 FlinkShell
22260 Jps
```

- 通过提交 jar 包运行批处理程序
- flink不可以直接提交jar包运行，首先仍需

- detached 模式提交

运行结果与运行时进程情况如下：

```
Thu 28 Nov - 17:12 ~ Thu 28 Nov - 17:10 ~
@wushuangyoyo jps @wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run -d ~/Downloads/flink-1.7.2/examples/batch/WordCount.jar --output /home/wushuangyoyo/flink/output
20290 FlinkShell Starting execution of program
23471 Jps Executing WordCount example with default input data set.
Thu 28 Nov - 17:12 ~ Use --input to specify file input.
@wushuangyoyo Job has been submitted with JobID ce11d839a29cf6f86e1b3da5c59ec39c
```

查看打印的结果如下：

```
@wushuangyoyo cat ~/flink/output
a 5
action 1
```

1.3 运行 Flink 流计算程序

- 使用 shell 运行流计算程序

- 本地模式启动 Scala-Shell，同上。
- 在 scala> 后输入 scala 代码

```
scala> val textstreaming=senv.fromElements("a a b b c")
textstreaming: org.apache.flink.streaming.api.scala.DataStream[String] = org.apache.flink.streaming.ap
i.scala.DataStream@13b666b4

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.str
eaming.api.scala.DataStream@46b3c1c2

scala> countsstreaming.print()
res1: org.apache.flink.streaming.api.datastream.DataStreamSink[(String, Int)] = org.apache.flink.strea
ming.api.datastream.DataStreamSink@3ecec11

scala> senv.execute()
(a,1)
(a,2)
(b,1)
(b,2)
(c,1)
res2: org.apache.flink.api.common.JobExecutionResult = org.apache.flink.api.common.JobExecutionResult@
5f065ddc
```

- 通过提交 jar 包运行流计算程序

同上，需要在终端本地模式启动 Scala-Shell。

之后提交 jar 包

- 默认模式提交

在另一个终端中启动 socket 服务作为数据源。显示如下：

```
Thu 28 Nov - 17:12 ~
@wushuangyoyo nc -lk 9000
```

在第三个终端中提交 jar 包

```
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run ~/Downloads/flink-1.7.2/e
xamples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
```

向第二个终端中输入数据进行 wordcount 计算，在第一个终端运行结果如下：

```
@wushuangyoyo nc -lk 9000
sdhnaskljhflsa
sjduiolahyerfuaie3
jdashikghia
sda
sds
da
dsa
sda
5 / 3539 Words

scala> sdhnaskljhflsa : 1
sda : 1
jdashikghia : 1
sjduiolahyerfuaie3 : 1
sds : 1
dsa : 1
da : 1
sda : 1
```

在运行过程中另起一个终端执行 `jps` 查看进程：

```
Thu 28 Nov - 17:24 ~
@wushuangyoyo jps
23936 CliFrontend
20290 FlinkShell
24552 Jps
```

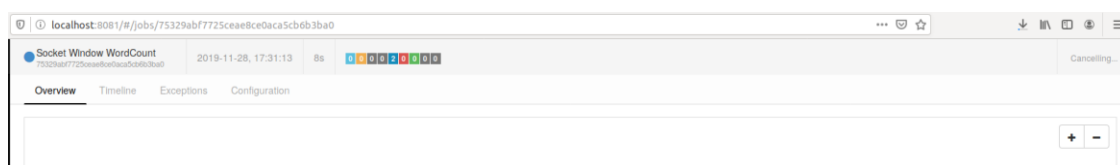
流计算任务的终止

- 在作为数据源的终端中，用键盘 `ctrl + c` 杀掉 socket 服务：

```
Thu 28 Nov - 17:15 ~
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run ~/Downloads/
examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
Program execution finished
Job with JobID d0cc357b6401999c8681380fb51e057f has finished.
Job Runtime: 475148 ms
```

由于第三个 terminal 中的进程是通过监控数据源的状态来运行的，于是 kill 掉数据源的进程就可以 kill 掉 wordcount 运算，而 scala-shell 的进程不依赖于数据源，所以不会受影响。

或者在 WebUI 上 cancel：



cancel 后的任务显示 canceled：

Aggregate task statistics by TaskManager									
Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks
2019-11-28, 17:31:13	2019-11-28, 17:31:23	9s	Source: Socket Stream -> Flat Map	0 B	0	0 B	0	1	1
2019-11-28, 17:31:13	2019-11-28, 17:31:23	9s	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction) -> Sink: Print to Std. Out	0 B	0	0 B	0	1	1

- detached 模式提交

终端 2 中启动本地服务

```
Thu 28 Nov - 19:02 ~
@wushuangyoyo nc -l 9000
单机集中式部署

Thu 28 Nov - 19:02 ~
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run ~/Downloads/
examples/streaming/SocketWindowWordCount.jar --port 9000
```

终端 3 中提交任务 jar 包

```
Thu 28 Nov - 19:02 ~
@wuchuangyoyo ~:~/Downloads/flink-1.7.2/bin/flink run -d ~/Downloads/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
Job has been submitted with JobID f8b66d2e0281f575d20b86f529a93b2e
```

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：

```
Thu 28 Nov - 19:02 ~
@wuchuangyoyo nc -l 9000
jdklfsahakl
fjdsklhauieyh
dljfie
dkfls;ljoife

scala> jdklfsahakl : 1
dljfie : 1
fjdsklhauieyh : 1
dkfls;ljoife : 1
```

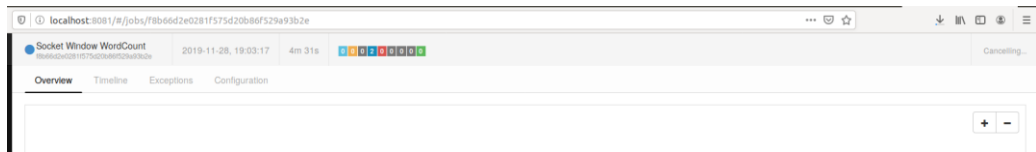
在运行过程中另起一个终端执行 jps 查看进程，此时不会出现 CliFrontend 进程

```
Thu 28 Nov - 19:07 ~
@wuchuangyoyo jps
5365 jps
2766 FlinkShell
```

流计算任务的终止

在终端 2, ctrl + c 杀掉 socket 服务，同上。

或者在 WebUI 上 cancel



cancel 后的任务显示 canceled:

Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2019-11-28, 19:03:17	2019-11-28, 19:07:50	4m 32s	Source: Socket Stream -> Flat Map	0 B	0	0 B	4	1	1	CANCELED
2019-11-28, 19:03:17	2019-11-28, 19:07:50	4m 32s	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction) -> Sink: Print to Std. Out	110 B	4	0 B	0	1	1	CANCELED

2 伪分布式部署

2.1 修改 Flink 配置

- 更改配置文件 flink-conf.yaml。由于其他都是按照默认的设置使用，只修改其中的两个参数：

- 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：

```
# We recommend to set this value to 'true' only in setups for pure batch
# processing (DataSet API). Streaming setups currently do not use the
# TaskManager's
# managed memory: The 'rocksdb' state backend uses RocksDB's own memory
# management,
# while the 'memory' and 'filesystem' backends explicitly keep data as
# objects
# to save on serialization cost.
#
taskmanager.memory.preallocate: false
```

2. 配置程序默认并行计算的个数:

```

53 # The parallelism used for programs that did not specify and other
    parallelism.
54
55 parallelism.default: 2
56

```

3. 配置 TaskManager 提供的任务 slots 数量大小:

```

# The number of task slots that each TaskManager offers. Each slot runs
one parallel pipeline.

taskmanager.numberOfTaskSlots: 2

```

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：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

```

@wushuangyoyo ./bin/start-cluster.sh
Starting cluster.
Starting standalone session daemon on host Master-yoyo.
Starting taskexecutor daemon on host Master-yoyo.

```




- 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。其中，在 standalone 模式下，Jobmanager 的进程名为 StandaloneSessionClusterEntrypoint。结果如下：

```

@wushuangyoyo jps
7156 TaskManagerRunner
6709 StandaloneSessionClusterEntrypoint
7374 Jps

```

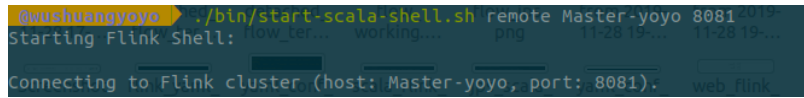
- 在 Web 中查看 flink 状态:

Overview	Version: 1.7.2	Commit: ceba8af
	1	Total Jobs
	2	Running
	2	Finished
		Canceled
		Failed

2.4 运行 Flink 批处理程序

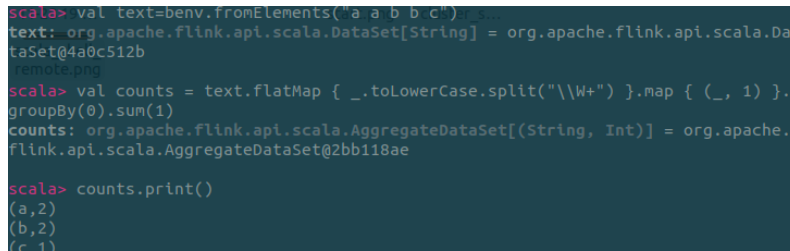
- 使用 shell 运行批处理程序

- 远程模式启动 Scala-Shell



```
@wushuangyoyo ~$ ./bin/start-scala-shell.sh remote Master-yoyo 8081
Starting Flink Shell: now ter... working... png 11-28 19:00 11-28 19:00
Connecting to Flink cluster (host: Master-yoyo, port: 8081): web.flink
```

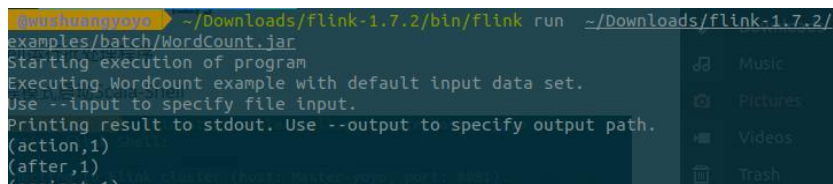
- 在 scala> 后运行 wordcount 程序，结果如下图所示：



```
scala> val text=benv.fromElements("a a b b c")
text: org.apache.flink.api.scala.DataSet[String] = org.apache.flink.api.scala.Data
DataSet@4a0c512b
remote.png
scala> val counts = text.flatMap { _.toLowerCase.split("\\W+") }.map { (_, 1) }.
groupBy(0).sum(1)
counts: org.apache.flink.api.scala.AggregateDataSet[(String, Int)] = org.apache.
flink.api.scala.AggregateDataSet@2bb118ae
scala> counts.print()
(a,2)
(b,2)
(c,1)
```

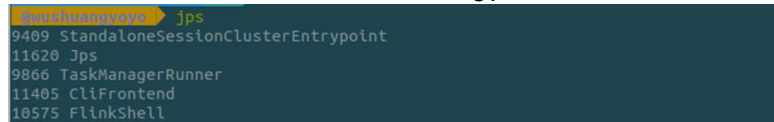
- 通过提交 jar 包运行批处理程序

- 默认模式提交。运行结果如下：



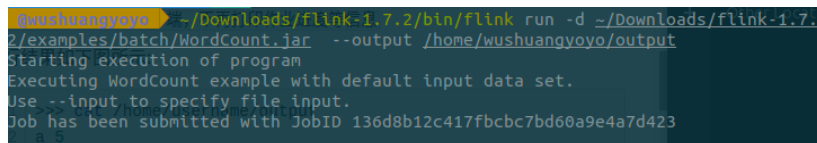
```
@wushuangyoyo ~$ ./Downloads/Flink-1.7.2/bin/flink run ~/Downloads/flink-1.7.2/
examples/batch/WordCount.jar
Starting execution of program
Executing WordCount example with default input data set.
Use --input to specify file input.
Printing result to stdout. Use --output to specify output path.
(action,1)
(after,1)
```

在运行过程中另起一个终端执行 jps 查看进程：



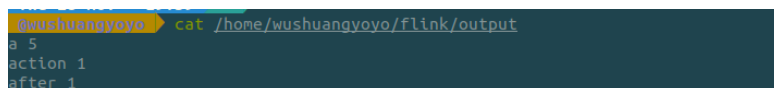
```
@wushuangyoyo ~$ jps
9409 StandaloneSessionClusterEntrypoint
11620 Jps
9866 TaskManagerRunner
11405 CliFrontend
10575 FlinkShell
```

- detached 模式提交



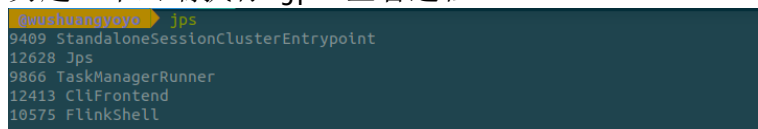
```
@wushuangyoyo ~$ ./Downloads/flink-1.7.2/bin/flink run -d ~/Downloads/flink-1.7.
2/examples/batch/WordCount.jar --output /home/wushuangyoyo/output
Starting execution of program
Executing WordCount example with default input data set.
Use --input to specify file input.
Job has been submitted with JobID 136d8b12c417fbc7bd60a9e4a7d423
```

运行结果如下图所示：



```
@wushuangyoyo ~$ cat /home/wushuangyoyo/flink/output
a 5
action 1
after 1
```

另起一个终端执行 jps 查看进程



```
@wushuangyoyo ~$ jps
9409 StandaloneSessionClusterEntrypoint
12628 Jps
9866 TaskManagerRunner
12413 CliFrontend
10575 FlinkShell
```


2.5 运行 Flink 流计算程序

- 使用 shell 运行流计算程序

- 远程模式启动 Scala-Shell，同上。
- 在 `scala>` 后输入 scala 代码

```
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@26ab3c1b

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@29701f1f

scala> countsstreaming.print()
res1: org.apache.flink.streaming.api.datastream.DataStreamSink[(String, Int)] =
org.apache.flink.streaming.api.datastream.DataStreamSink@7a065f73

scala> senv.execute()
res2: org.apache.flink.api.common.JobExecutionResult = org.apache.flink.api.comm
on.JobExecutionResult@699e81db
```

- 通过提交 jar 包运行流计算程序

- 默认模式提交

终端 1 中启动本地服务，同上。

终端 2 中提交任务 jar 包

```
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run ~/Downloads/flink-1.7.2/e
xamples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
```

终端 3 中打开 log 目录下的 out 文件会统计 flink 的执行结果。结果如下：

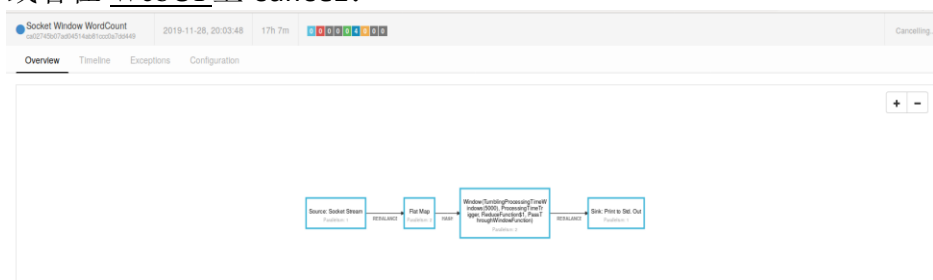
```
@wushuangyoyo ~ nc -l 9000 @wushuangyoyo ~ tell -f Downloads/flink-1.7.2/log/flink-wushuangyoyo-taskexecut
jhlakdhlaik or-0-Master-yoyo.out
fsadkljfgalluge jhlakdhlaik : 1
fdhakigheu fdhakigheu : 1
sdhuja fsadkljfgalluge : 1
daf sdhuja : 1
sda link run ~/flink- fdas : 1
fdas/SocketWindowWordCount fdas : 1
fdas sda : 1
hdfjklasf daf : 1
fjasoihf hdfjklasf : 1
fjasoihf fjasoihf : 1
```

在运行过程中另起一个终端执行 `jps` 查看进程。结果如下：

```
Fri 29 Nov - 12:57 ~
@wushuangyoyo ~ jps
9409 StandaloneSessionClusterEntrypointe
13542 CliFrontend
13575 Jps
9866 TaskManagerRunner
10575 FlinkShell
```

流计算任务的终止

- 在终端 2, `ctrl + c` 杀掉 socket 服务，同上一部分。
- 或者在 WebUI 上 cancel:



canceled 后的任务状态显示为 canceled:

Subtasks										
Aggregate task statistics by TaskManager										
Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2019-11-28, 20:03:48	2019-11-29, 13:11:03	17h 7m	Source: Socket Stream	0 B	0	0 B	10	1	1	CANCELED
2019-11-28, 20:03:48	2019-11-29, 13:11:03	17h 7m	Flat Map	134 B	10	0 B	10	2	2	CANCELED
2019-11-28, 20:03:48	2019-11-29, 13:11:03	17h 7m	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction)	244 B	10	0 B	10	2	2	CANCELED
2019-11-28, 20:03:48	2019-11-29, 13:11:03	17h 7m	Sink: Print to Std. Out	324 B	10	0 B	0	1	1	CANCELED

- detached 模式提交

终端 1 中启动本地服务，同上一部分。

终端 2 中提交任务 jar 包

```
@wushuangyoyo ~/Downloads/flink-1.7.2/bin/flink run -d ~/Downloads/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --port 9000
Starting execution of program
Job has been submitted with JobID 4c02e9581981f546f28ff0cb9338ca38
```

终端 3 中打开 log 目录下的 out 文件会统计 flink 的执行结果

```
@wushuangyoyo nc -l 9000
djuahgiluwyrhrgilqauge\
dalhdal
daf
fa
fdasdfa
fawfa
fassa
sfda
das
dasfd
fa
khdla
统计flink的执行结果
Fri 29 Nov - 14:15 ~/Downloads/flink-1.7.2/log origin 60master 60*30
14-7+
@wushuangyoyo tail -f flink-wushuangyoyo-taskexecutor-0-Master-yoyo.out
start.png
a_flink scala_flink scala_flink
hell_ batch_ pdos_
ch_1... wordcod... batch_jp...
batch_ problem_ problem_
k_log dos_1 dos_1
flink log dos_1 dos_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

```
>>>jps
19138 StandaloneSessionClusterEntrypoint
19604 TaskManagerRunner
9098 Jps
```

流计算任务的终止，同上一部分。

2.6 停止 Flink 服务

- 停止命令 并通过 jps 来判断是否关闭成功。

```
Sat 30 Nov - 20:04 ~
@wushuangyoyo Downloads/flink-1.7.2/bin/stop-cluster.sh
Stopping taskexecutor daemon (pid: 9866) on host Master-yoyo.
Stopping standalone session daemon (pid: 9409) on host Master-yoyo.
Sat 30 Nov - 20:13 ~
@wushuangyoyo jps
6975 Jps
10575 FlinkShell
```

3 分布式部署

3.1 修改配置文件

- 更改配置文件 `flink-conf.yaml`:

```
33 jobmanager.rpc.address: 219.228.135.148
34
35 # The RPC port where the JobManager is reachable.
36
37 jobmanager.rpc.port: 6124
```

- 更改配置文件 `slaves`

```
slaves
219.228.135.64
219.228.135.103
```

- 将配置好的 Flink 同步到其他节点

```
ecnu@Master-yoyo:~$ scp flink-1.7.2/conf/flink-conf.yaml 219.228.135.64:/home/ecnu/flink-1.7.2/conf/
flink-conf.yaml
ecnu@Master-yoyo:~$ scp flink-1.7.2/conf/flink-conf.yaml 219.228.135.103:/home/ecnu/flink-1.7.2/conf/
flink-conf.yaml
ecnu@Master-yoyo:~$ flink-1.7.2/bin/start-cluster.sh
```

3.3 启动 flink 服务

- 启动命令

```
>>> ~/flink-1.7.2/flink-1.7.2/bin/start-cluster.sh
```

遇到问题:

```
ecnu@Master-yoyo:~$ flink-1.7.2/bin/start-cluster.sh
Starting cluster.
Starting standalone session daemon on host Master-yoyo.
Starting taskexecutor daemon on host cyy-OptiPlex-7050.
Starting taskexecutor daemon on host lcc-OptiPlex-7050.
ecnu@Master-yoyo:~$ jps
20890 Jps
```

运行了分布式的启动程序，但是 `jps` 却没有显示进程正在运行，查看 `log` 文件，错误信息为:

```
org.apache.flink.runtime.entrypoint.ClusterEntryPointException: Failed to initialize the cluster endpoint StandaloneSessionClusterEndpoint.
    at org.apache.flink.runtime.entrypoint.ClusterEntryPoint.startCluster(ClusterEntryPoint.java:181)
    at org.apache.flink.runtime.entrypoint.ClusterEntryPoint.runClusterEntryPoint(ClusterEntryPoint.java:517)
    at org.apache.flink.runtime.entrypoint.StandaloneSessionClusterEntryPoint.main(StandaloneSessionClusterEntryPoint.java:65)
Caused by: java.net.BindException: Could not start actor system on any port in port range 6123
    at org.apache.flink.runtime.clusterframework.BootstrapTools.startActorSystem(BootstrapTools.java:181)
    at org.apache.flink.runtime.clusterframework.BootstrapTools.startActorSystem(BootstrapTools.java:121)
    at org.apache.flink.runtime.clusterframework.BootstrapTools.startActorSystem(BootstrapTools.java:96)
    at org.apache.flink.runtime.rpc.akka.AkkaRpcServiceUtils.createRpcService(AkkaRpcServiceUtils.java:78)
```

不可以使用 6123 这个端口，查看网络信息，发现校园网并不可以使用 6123 这样的 ipv6 的端口。于是将 `taskmanager.rpc.port` 设置为 6124，再次运行，依然遇到问题:

```
Caused by: java.net.BindException: Address already in use
    at sun.nio.ch.Net.bind0(Native Method)
    at sun.nio.ch.Net.bind(Net.java:433)
    at sun.nio.ch.Net.bind(Net.java:425)
    at sun.nio.ch.ServerSocketChannelImpl.bind(ServerSocketChannelImpl.java:223)
    at org.apache.flink.shaded.netty4.io.netty.channel.socket.nio.NioServerSocketChannel.doBind(NioServerSocketChannel.java:131)
    at org.apache.flink.shaded.netty4.io.netty.channel.AbstractChannel$AbstractUnsafe.bind(AbstractChannel.java:558)
```

显示问题为地址已被使用，一开始以为是 6124 端口被占用，于是尝试了 `kill -9`:

```
ecnu@Master-yoyo:~$ ps -ef | grep 6123
ecnu 28851 14842 0 22:00 pts/4 00:00:00 grep --color=auto 6123
ecnu@Master-yoyo:~$ ps -ef | grep 6124
ecnu 28853 14842 0 22:00 pts/4 00:00:00 grep --color=auto 6124
ecnu@Master-yoyo:~$ kill -9 6124
bash: kill: (6124) - No such process
```

结果发现没有在使用。发现其实是 `rest.port` 已被使用，将其修改为 8082。再次运行，成功。并运行 `jps` 检查进程情况，结果如下：

```
ecnu@Master-yoyo:~$ flink-1.7.2/bin/start-cluster.sh
Starting cluster.
Starting standalone session daemon on host Master-yoyo.
Starting task executor daemon on host cyy-OptiPlex-7050.
Starting task executor daemon on host lcc-OptiPlex-7050.
ecnu@Master-yoyo:~$ jps
31274 Jps
31210 StandaloneSessionClusterEntrypoint
ecnu@Master-yoyo:~$ netstat -anp | grep 6124
```

从节点进程情况如下：

```
ecnu@cyy-OptiPlex-7050:~$ jps
4519 Jps
23950 TaskManagerRunner
```

- 查看 flink 服务信息
 - 查看 flink 服务日志(`./log/flink-ecnu-standalonesession-1-Master-yoyo.log` / `./log/flink-ecnu-taskexecutor-1-cyy-OptiPlex-7050.log`)，其中 1 表示运行成功。




- 主节点：

```
ecnu@Master-yoyo:~$ cat flink-1.7.2/log/flink-ecnu-standalonesession-1-Master-yoyo.log
2019-11-28 22:12:37,649 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - Starting StandaloneSessionClusterEntrypoint (Version: 1.7.2, Revision: ceba8af, Date: 11.02.2019 @ 14:17:09 UTC)
2019-11-28 22:12:37,650 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - OS current user: ecnu
2019-11-28 22:12:37,691 WARN org.apache.hadoop.util.NativeCodeLoader - Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2019-11-28 22:12:37,935 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - Current Hadoop/Kerberos user: ecnu
2019-11-28 22:12:37,935 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - JVM: Java HotSpot(TM) 64-Bit Server VM - Oracle Corporation - 1.8.0_221-b11
2019-11-28 22:12:37,935 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - Maximum heap size: 981 MBytes
2019-11-28 22:12:37,935 INFO org.apache.flink.runtime.entrypoint.ClusterEntrypoint - Java home: /usr/local/jdk1.8.0_221
```

- 从节点：

```
ecnu@cyy-OptiPlex-7050:~$ cat flink-1.7.2/log/flink-ecnu-taskexecutor-1-cyy-OptiPlex-7050.log
2019-11-28 09:12:38,436 INFO org.apache.flink.runtime.taskexecutor.TaskManagerRunner - Starting TaskManager (Version: 1.7.2, Revision: ceba8af, Date: 11.02.2019 @ 14:17:09 UTC)
2019-11-28 09:12:38,437 INFO org.apache.flink.runtime.taskexecutor.TaskManagerRunner - OS current user: ecnu
2019-11-28 09:12:38,635 WARN org.apache.hadoop.util.NativeCodeLoader - Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2019-11-28 09:12:38,635 INFO org.apache.flink.runtime.taskexecutor.TaskManagerRunner - Current Hadoop/Kerberos user: ecnu
```

- 访问 flink web 界面

Overview			Version: 1.7.2		Commit: ceba8af	
<div><div></div><div>Task Managers</div><div>2</div></div>						
<div><div></div><div>Task Slots</div><div>4</div></div>						
<div><div></div><div>Available Task Slots</div><div>4</div></div>						
			Total Jobs			
			Running		<div>0</div>	
			Finished		<div>0</div>	
			Canceled		<div>0</div>	
			Failed		<div>0</div>	

3.4 运行 flink 批处理程序

- 运行 jar 文件并查看结果

- 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

```
>>> ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/batch/WordCount.jar
```

运行并使用 `jps` 命令。结果如下：

```
ecnu@Master-yoyo:/home/wushuangyoyo$ jps
664 CliFrontend
31210 StandaloneSessionClusterEntrypoint
879 Jps
ecnu@Master-yoyo:/home/wushuangyoyo$ 
ecnu@Master-yoyo:~$ ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/batch/WordCount.jar
Starting execution of program
Executing WordCount example with default input data set.
Use --input to specify file input.
Printing result to stdout. Use --output to specify output path.
(action,1)
(after,1)
```

- detached 模式提交，在客户端看不到应用程序运行过程中的信息

由于有一个从节点的 ip 地址发生了改变，于是修改 `hadoop-2.9.2/etc/hadoop/slaves` 中的 ip 地址至与现实一致。

同样的问题依然存在：

```
ecnu@Master-yoyo:~$ jps
2179 Jps
2020 SecondaryNameNode
1533 NameNode
ecnu@cyy-OptiPlex-7050:~$ jps
17140 Jps
ecnu@cyy-OptiPlex-7050:~$
```

查看从节点的 datanode 的 logs 日志。发现问题出现的原因是 `All specified directories are failed to load`. 即 datanode 和 namenode 相关设置不一致，只需要修改 namenode 下的 `./hadoop-2.9.2/tmp/dfs/current/name/VERSION` 和 datanode 下的 `./hadoop-2.9.2/tmp/dfs/current/data/VERSION` 至一致便可以解决问题。

于是删除 tmp 文件并重新格式化 namenode，重新启动 hadoop，成功。结果如下：

```
ecnu@lcc-OptiPlex-7050:~$ jps
30881 DataNode
31052 Jps
ecnu@lcc-OptiPlex-7050:~$ 
ecnu@Master-yoyo:~$ jps
3107 NameNode
3548 Jps
3422 SecondaryNameNode
ecnu@cyy-OptiPlex-7050:~$ jps
22384 Jps
22291 DataNode
ecnu@cyy-OptiPlex-7050:~$
```

`>>> ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/batch/WordCount.jar --output hdfs://20s209:9001/flink-data/output`

运行效果如下图：

```
ecnu@Master-yoyo:~$ ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/batch/WordCount.jar --output hdfs://219.228.135.148:9000/flink-data/output
Starting execution of program
Executing WordCount example with default input data set.
Use --input to specify file input.
Job has been submitted with JobID 4bce0ffc2a3ad1e6f2bec8042012443c
```

同时运行 `jps` 来显示此时正在运行的进程：

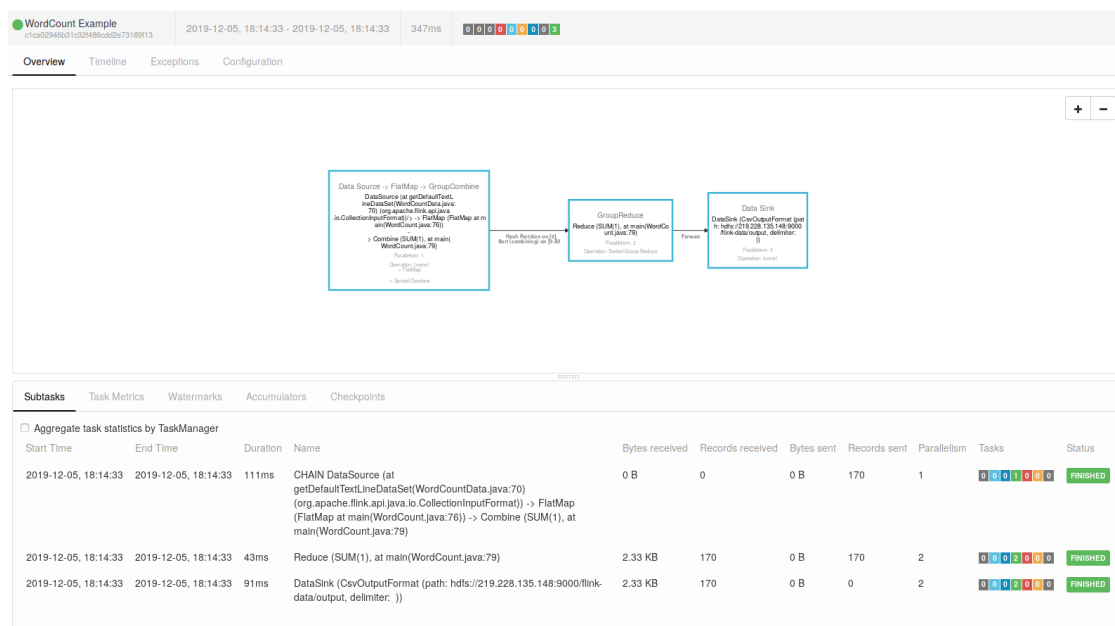
```
ecnu@Master-yoyo:/home/wushuangyoyo$ jps
5056 StandaloneSessionClusterEntrypoint
3107 NameNode
7956 Jps
7741 CliFrontend
3422 SecondaryNameNode
```

detached 模式提交并不是如文档所说的没有 `CliFrontend` 进程。感谢陈启航学长细心指导得知其中原因（事实上，除了个别看了源码解决的问题，基本上所有的网上解决不了的问题，都是陈启航学长帮忙解决的）：`CliFrontend` 进程在 detached 模式下的准备阶段运行，当任务提交以后，便自动退出。所以在 submitted 的情况下是没有 `CliFrontend` 进程的。

查看 hdfs 中的运行结果：

```
ecnu@Master-yoyo:~$ ~/hadoop-2.9.2/bin/hdfs dfs -cat /flink-data/output/2
a 5
all 2
bear 3
but 1
by 2
conscience 1
consummation 1
```

- 访问 flink web 界面



3.5 运行 flink 流计算程序

- 运行 jar 文件并查看结果

— 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

```
>>> ~/flink-1.7.2/bin/flink run ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.148 --port 9001
```

程序提交完毕后退出客户端，不再打印作业进度等信息。

遇到问题：

```
at org.apache.flink.runtime.security.HadoopSecurityContext.runSecured(HadoopSecurityContext.java:112)
at org.apache.flink.client.cli.CliFrontend.main(CliFrontend.java:1126)
Caused by: org.apache.flink.runtime.client.JobExecutionException: Job execution failed.
at org.apache.flink.runtime.jobmaster.JobResult.toJobExecutionResult(JobResult.java:112)
at org.apache.flink.client.program.rest.RestClusterClient.submitJob(RestClusterClient.java:112)
... 20 more
Caused by: java.net.ConnectException: Connection refused (Connection refused)
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
at java.net.AbstractPlainSocketImpl.connect(AbstractPlainSocketImpl.java:188)
```

一开始以为是端口被占用造成的原因，所以更换多次端口，未果。以为是 ipv6 端口不支持，再次选取 ipv4 端口在运行，未果。经徐老师提醒，应该先打开 netcat 数据源端口，再打开 dataflow 的程序。

更换运行顺序以后再次启动，成功。结果如下：

```
ecnu@Master-yoyo:~$ nc -l 9001
jfdalhfla
fjalhfla
fdak;
ecnu@Master-yoyo:/home/wushuangyoyo$ ~/flink-1.7.2/bin/flink run -r ~/flink-1.7.2/e
xamples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.148 --port 90
01
Starting execution of program
2333 Words
```

使用 jps 命令查看进程状况：

```
ecnu@Master-yoyo:~$ jps
3107 NameNode
29125 CliFrontend
24936 StandaloneSessionClusterEntrypoint
29517 Jps
3422 SecondaryNameNode
```

访问 flink web 界面查看任务运行位置

Subtasks	Task Metrics	Watermarks	Accumulators	Checkpoints	Back Pressure					
Aggregate task statistics by TaskManager										
Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2019-12-05, 18:40:43	2019-12-05, 18:51:52	11m 8s	Source: Socket Stream	0 B	0	0 B	3	1	<div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING
2019-12-05, 18:40:43	2019-12-05, 18:51:52	11m 8s	Flat Map	40 B	3	0 B	3	2	<div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING
2019-12-05, 18:40:43	2019-12-05, 18:51:52	11m 8s	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction)	73 B	3	0 B	3	2	<div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING
2019-12-05, 18:40:43	2019-12-05, 18:51:52	11m 8s	Sink: Print to Std. Out	97 B	3	0 B	0	1	<div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING

在任务运行位置的 flink log 目录下输入命令：tail -f ~/flink-1.7.2/log/flink-ecnu-taskexecutor-0-Master-yoyo.out，遇到问题：

```
ecnu@Master-yoyo:~$ tail -f flink-1.7.2/log/flink-ecnu-taskexecutor-0-Master-yoyo.out
^C
ecnu@Master-yoyo:~$ cat flink-1.7.2/log/flink-ecnu-taskexecutor-0-Master-yoyo.out
```

out 文件没有输出。经老师提醒，发现输出节点不在本台机器：

2019-12-05, 18:40:43	2019-12-05, 19:02:40	21m 57s	Window(TumblingProcessingTimeWindows(5000), ProcessingTimeTrigger, ReduceFunction\$1, PassThroughWindowFunction)	304 B	12	0 B	12	2	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING
2019-12-05, 18:40:43	2019-12-05, 19:02:40	21m 57s	Sink: Print to Std. Out	400 B	12	0 B	0	1	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div>	RUNNING
Start Time	End Time	Duration	Bytes received	Records received	Bytes sent	Records sent	Attempt	Host	Status			
2019-12-05, 18:40:43		21m 57s	400 B	12	0 B	0	1	cyy-OptiPlex-7050-41875	RUNNING			

查看相应的机器中的存储数据，结果如下：

```
ecnu@Master-yoyo:~$ nc -l 9001
jfdalhfla
fjalhfla
fdak;
lloholl;fas
faslfhbla
fajkfgba
fajakgb的信息
fajlfhaekh
faklfhflaw
akfsh
ecnu@cyy-OptiPlex-7050:~$ tail -f flink-1.7.2/log/flink-ecnu-taskexecutor-0-cyy-OptiPlex-7050.out
afalhfla : 1
f;aoawleyh; : 1
lkchz;odf : 1
flakishyftio;a : 1
dhloa; : 1
fhapteh : 1
flalwh;t : 1
falfh : 1
f;lskhl : 1
cynla.knkl : 1
```

流计算任务的终止

- 在终端 2, ctrl + c 杀掉 socket 服务，同上面的步骤。
 - 或者在 WebUI 上 cancel，同上面的步骤。
- detached 模式提交，在客户端看不到应用程序运行过程中的信息

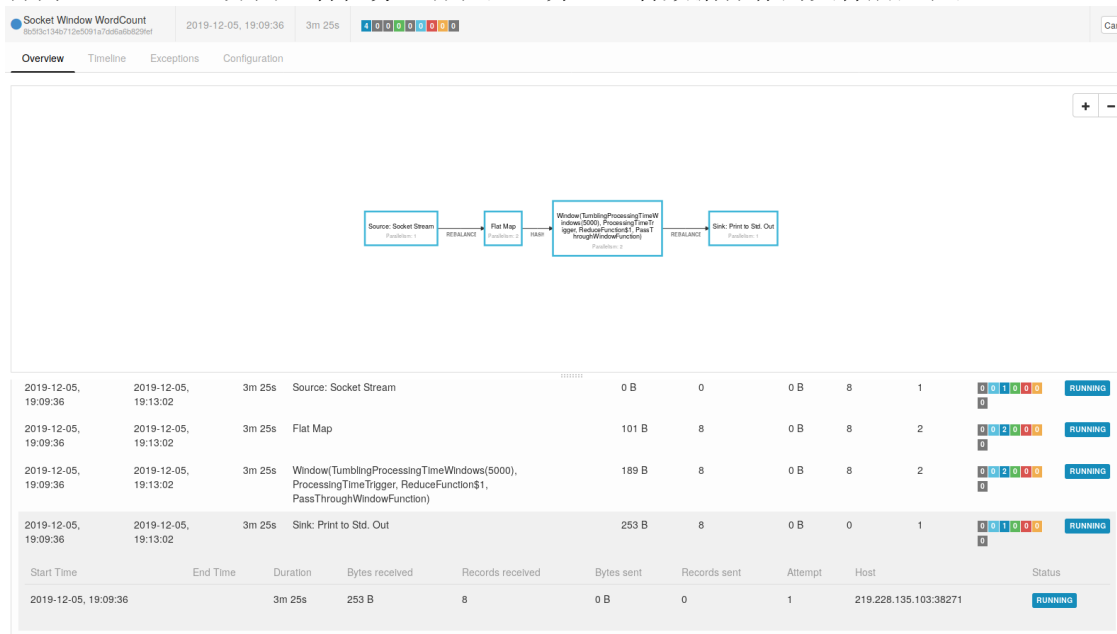
>>> ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/examples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.148 --port 9001

```
ecnu@Master-yoyo:~$ nc -l 9001
dalkf
fhaiol
fahlfujighaiol
ecnu@Master-yoyo:/home/wushuangyoyo$ ~/flink-1.7.2/bin/flink run -d ~/flink-1.7.2/
examples/streaming/SocketWindowWordCount.jar --hostname 219.228.135.148 --port
9001
Starting execution of program
Job has been submitted with JobID 8b5f3c134b712e5091a7dd6a6b829fef
```

使用 `jps` 观察进程运行状况：

```
ecnu@Master-yoyo:/home/wushuangyoyo$ jps
30481 Jps
3107 NameNode
24936 StandaloneSessionClusterEntrypoint
3422 SecondaryNameNode
```

访问 `flink web` 界面查看任务运行位置，并且查看数据保存的文件所处位置：



在任务运行位置的 `flink log` 目录下输入命令：`tail -f ~/flink-1.7.2/log/flink-ecnu-taskexecutor-0-lcc-OptiPlex-7050.out`，查看运行结果：

```
ecnu@Master-yoyo:~$ nc -l 9001
dalkf
fhaiol
fahlfujighatol
fagkljh
afhla
fahlhda
falbh
hfla
ecnu@lcc-OptiPlex-7050:~$ tail -f flink-1.7.2/log/flink-ecnu-taskexecutor-0-lcc-OptiPlex-7050.out
dalkf : 1
fhaiol : 1
fagkljh : 1
afhla : 1
fahlhda : 1
falbh : 1
fahlhda : 1
fahlfujighatol : 1
```

- 停止 `flink` 正在运行中的任务
 - 在 `web UI` 上 `cancel`，同上面的步骤。

3.6 停止 Flink 服务

- 停止命令

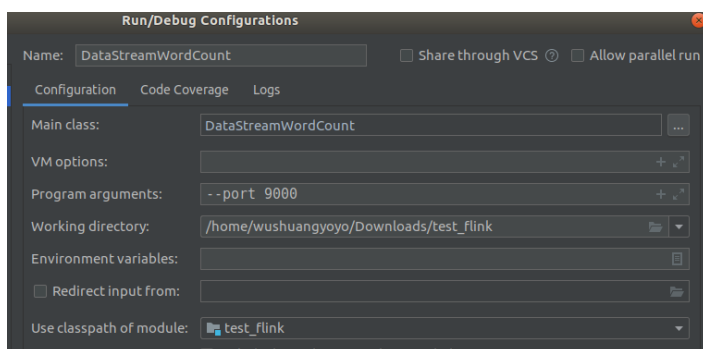

```
>>> ~/flink-1.7.2/bin/stop-cluster.sh
```
- 查看进程，验证是否成功停止服务
 - 若成功停止，`JobManager` 进程和 `TaskManager` 进程应消失，如下图所示：

```
ecnu@Master-yoyo:~$ ./flink-1.7.2/bin/stop-cluster.sh
Stopping taskexecutor daemon (pid: 22153) on host cyy-OptiPlex-7050.
Stopping taskexecutor daemon (pid: 20736) on host lcc-OptiPlex-7050.
Stopping standalone session daemon (pid: 32304) on host Master-yoyo.
ecnu@Master-yoyo:~$ jps
3107 NameNode
3422 SecondaryNameNode
543 Jps
```

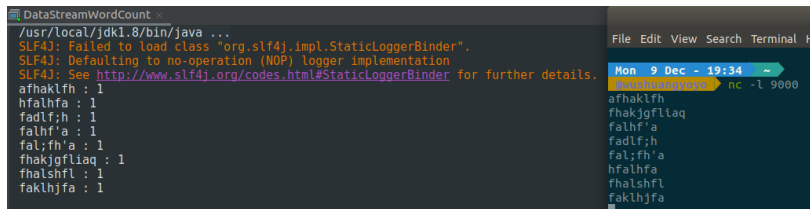

Flink 编程

1. 编写并调试 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'
- 运行结果如下：



2. 运行 Flink 程序

- 利用 IDE 打包 jar 文件并在伪分布模式下提交
 - 在终端输入命令，向 jobmanager 提交作业。并另起终端输入如下命令查看运行结果。

如下所示：

