1. **实验目的**

* 学习Flink的部署，简单使用Scala Shell
* 查看Flink的运行日志，体会与Storm运行过程中的日志区别
* 通过系统部署理解体系架构,体会流计算系统与批处理系统之间的区别。

1. **实验任务**

* **完成Flink的单机伪分布式部署以及分布式部署**
* **两种部署方法下分别以默认和托管模式运行示例程序**

1. **实验环境**

* **操作系统：Ubuntu 18.04**
* **JDK版本：1.8**
* **Flink版本：1.12.1**

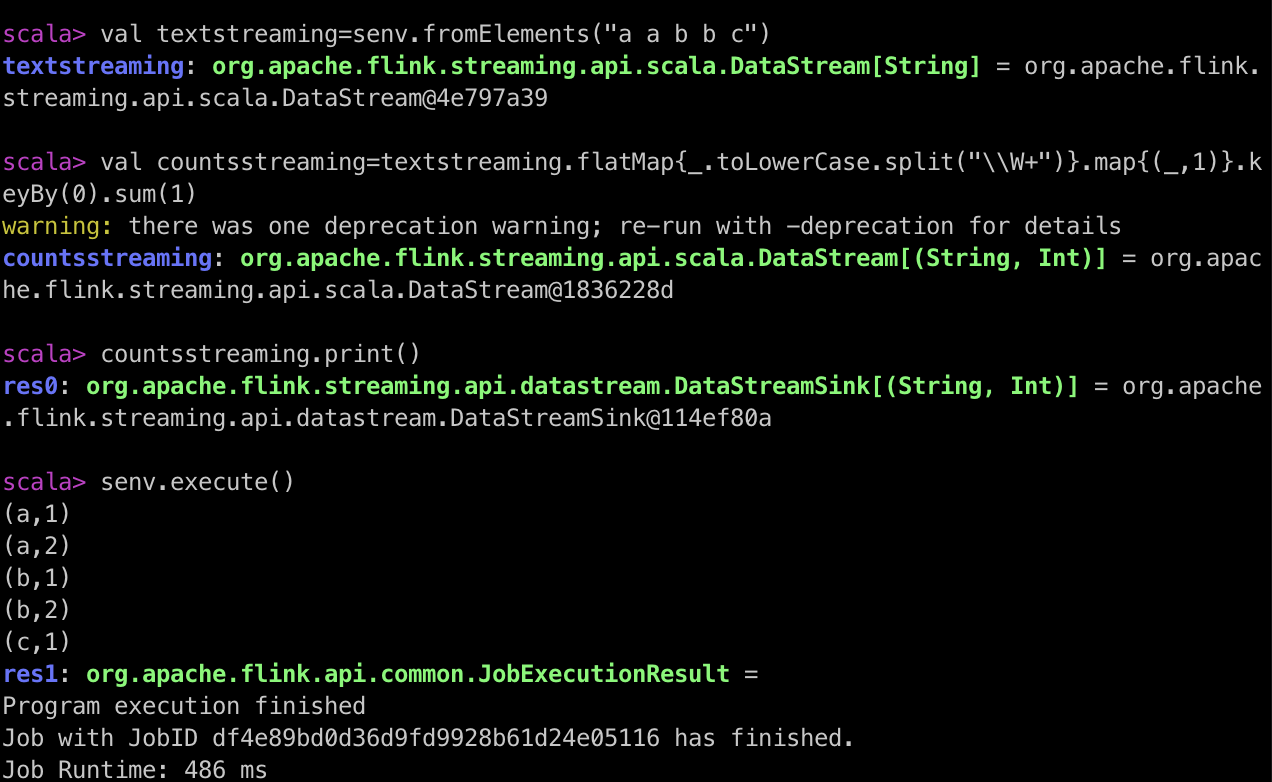
1. **实验步骤**

**13.4.1 单机集中式部署**

1. **准备工作**
2. **运行Flink DataStream程序**

**本地模式启动ScalaShell**

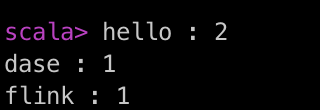
****

****

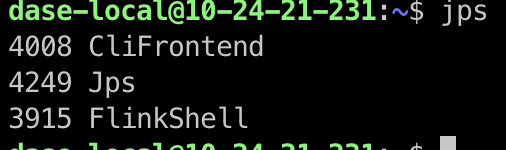
**通过提交Jar包运行DataStream程序**

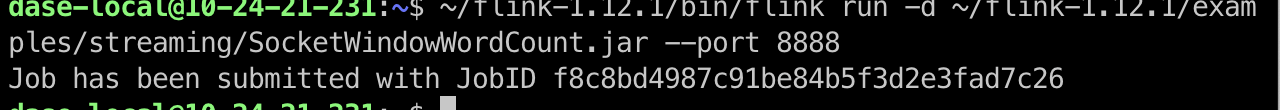
* **默认方式提交：**

**默认方式提交程序运行结果：**

****

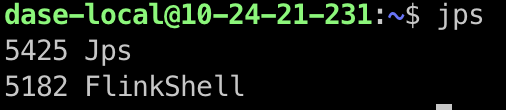
**默认方式提交程序运行JPS：**





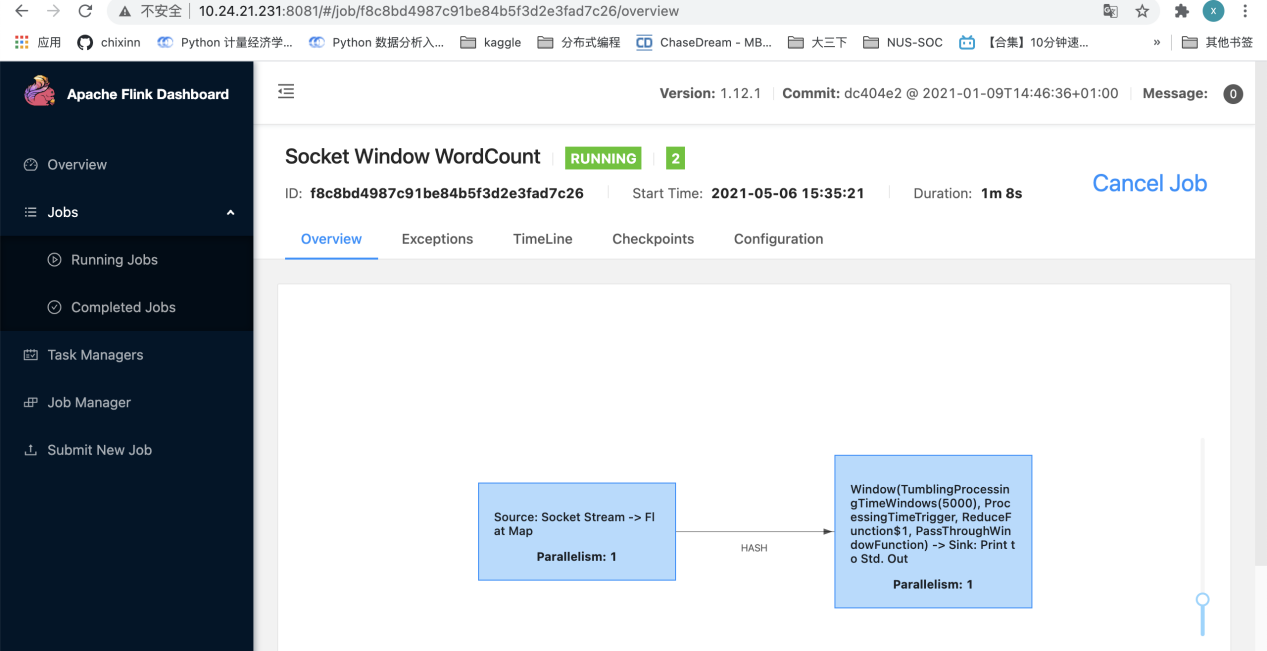
* Detached模式提交：

Detached模式提交的进程



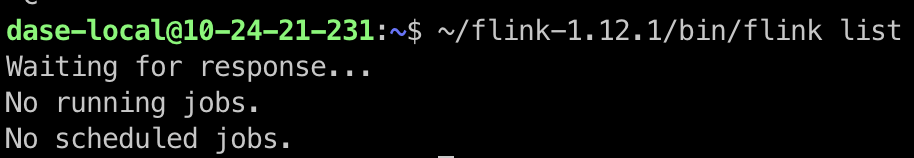
此时只能看到一个FlinkShell进程，而不会出现CliFronted进程。

FlinkWebUI:



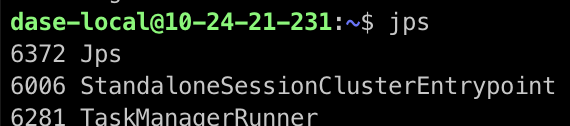
* 停止DataStream程序

停止以后，使用list命令列出正在运行的Job



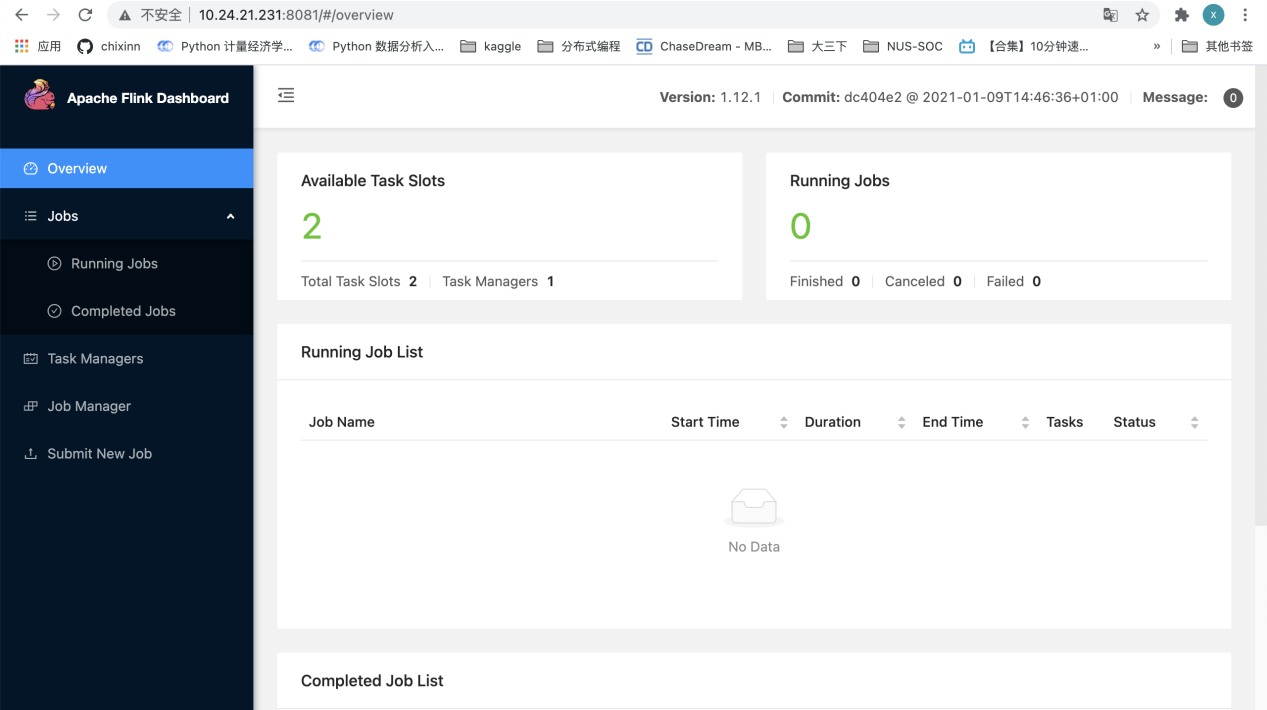
13.4.2 单机伪分布式部署

1. 修改配置
2. 启动服务
3. 查看Flink服务信息



FlinkUI界面

FlinkUI界面显示，有两个Avaliable TaskSlots 和配置文件更改的内容一致。

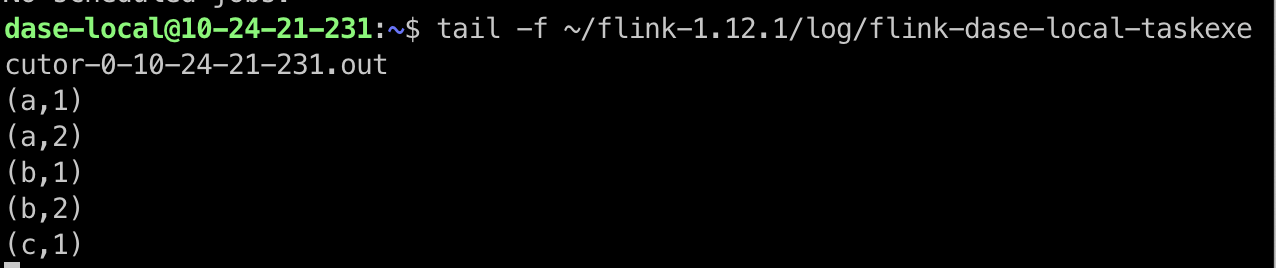


1. 运行Flink DataStream程序

* 使用Shell运行DataStream程序

另起一个终端，输入tail命令可以看到如下运行结果

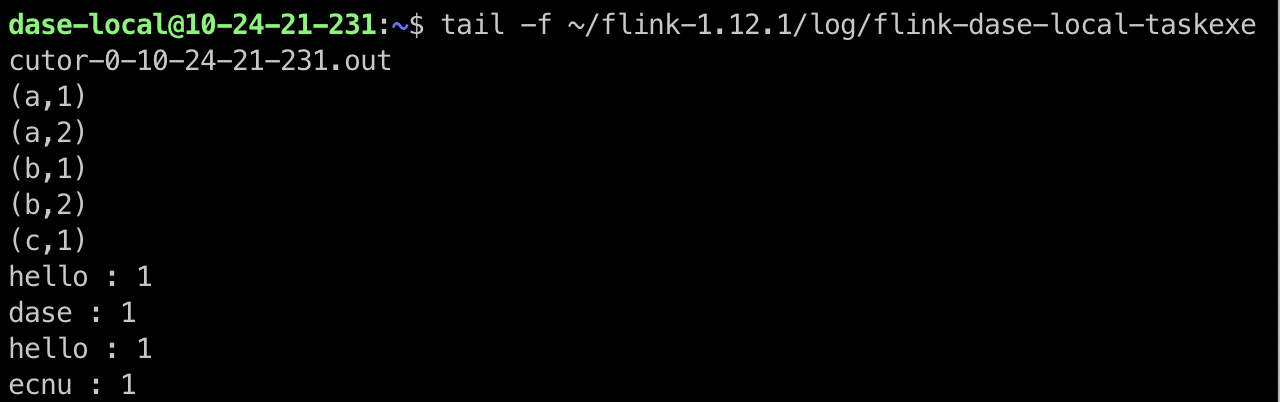
运行结果：



* 通过提交jar包运行DataStream程序

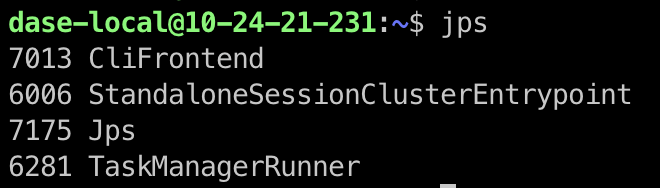
1. 启动Netcat服务
2. 提交jar包

默认方式方式提交

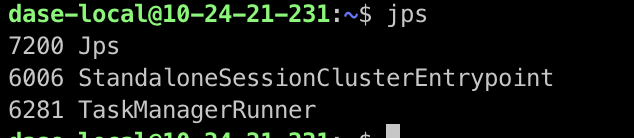


可以看出，结果直接追加在.out的日志文件的末尾。

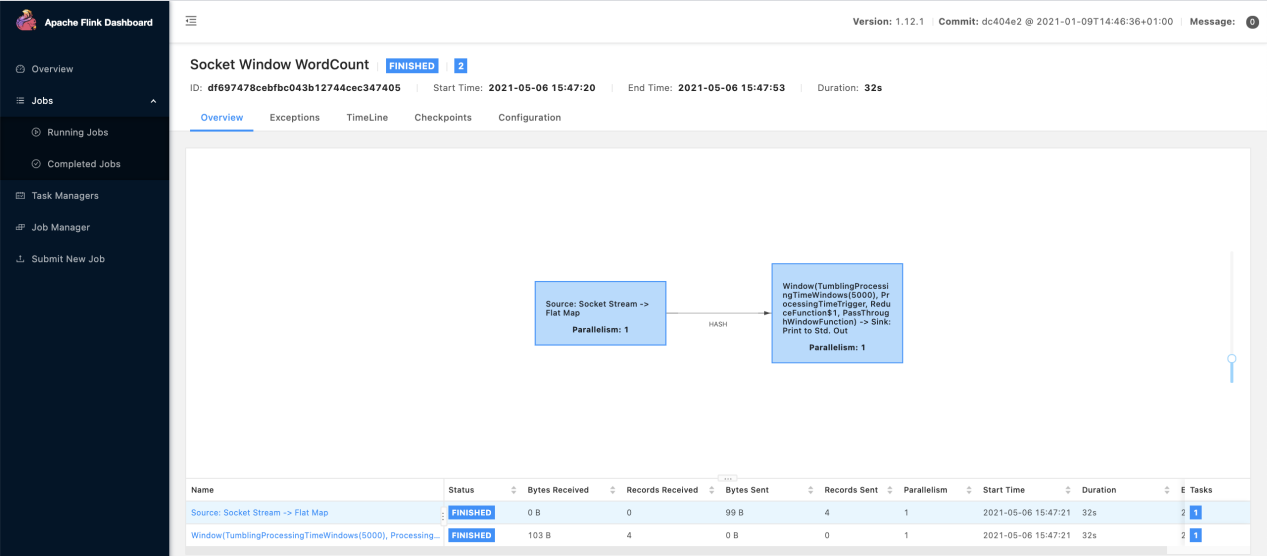
程序运行中存在的进程：

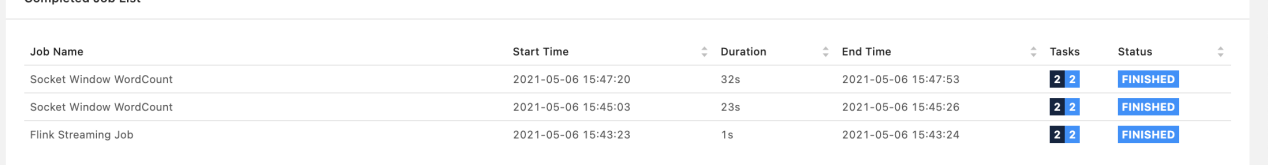


程序停止后的进程：



flinkWebUI





1. **思考题**
2. 在部署Flink的时候，如何指定每个TaskManager中的slots数量？
3. 一个Flink流计算应用程序何时停止，如何停止？请通过与Spark批处理应用程序进行对比来阐述。
4. **总结**