实验手册

1. 安装Apache Flink
2. 下载Flink安装包(选择hadoop2.6以及scala 2.11版本)

|  |
| --- |
| wget [**http://mirrors.hust.edu.cn/apache/flink/flink-1.7.0/flink-1.7.0-bin-hadoop27-scala\_2.11.tgz**](http://mirrors.hust.edu.cn/apache/flink/flink-1.7.0/flink-1.7.0-bin-hadoop27-scala_2.11.tgz)  tar -zxvf flink-1.7.0-bin-hadoop27-scala\_2.11.tgz |

1. 启动本地 standalone集群

|  |
| --- |
| cd flink-1.7.0  bin/start-cluster.sh |

打开浏览器数据输入http://localhost:8081

检查启动日志(可选)

|  |
| --- |
| tail log/flink-\*-standalonesession-\*.log |

1. 启动第一个Flink程序

启动本地socket 端口

|  |
| --- |
| nc -lk 9999 |

启动flink streaming word count程序

|  |
| --- |
| bin/flink run -m localhost:8081 examples/streaming/SocketWindowWordCount.jar --port 9999 |

在nc终端中输入

|  |
| --- |
| hello world  hello flink  hello streaming |

检查输出

|  |
| --- |
| tail log/flink-\*-taskexecutor-\*.out |

观察UI上各个组件

关闭flink集群

|  |
| --- |
| bin/stop-cluster.sh |

1. 启动yarn集群

启动Hadoop HDFS 以及 Yarn

|  |
| --- |
| cd hadoop-2.7.3  sbin/start-all.sh |

确保配置了HADOOP\_HOME环境变量

|  |
| --- |
| echo $HADOOP\_HOME |

启动yarn集群

|  |
| --- |
| bin/yarn-session.sh -n 1 -jm 1024m -tm 1024m |

在日志中会输出

|  |
| --- |
| Flink JobManager is now running on localhost:55975 with leader id 00000000-0000-0000-0000-000000000000.  JobManager Web Interface: http://localhost:55975 |

打开浏览器输入地址http://localhost:55975

启动wordcount程序

|  |
| --- |
| bin/flink run -m localhost:55975 -yn 1 -yjm 1024m -ytm 1024m ./examples/batch/WordCount.jar |

观察UI

1. 直接在yarn上运行程序（无需预先启动yarn session）

|  |
| --- |
| bin/flink run -m yarn-cluster -yn 1 -yjm 1024m -ytm 1024m examples/batch/WordCount.jar |

1. 编写第一个Flink Batch程序

创建maven project

|  |
| --- |
| mvn archetype:generate \  -DarchetypeGroupId=org.apache.flink \  -DarchetypeArtifactId=flink-quickstart-java \  -DarchetypeVersion=1.7.0 |

生成如下目录结构

|  |
| --- |
| flink-project/  |-- pom.xml  `-- src  `-- main  |-- java  | `-- com  | `-- aura  | `-- cases  | |-- BatchJob.java  | `-- StreamingJob.java  `-- resources  `-- log4j.properties |

生成的代码中包含BatchJob和StreamingJob两个Flink程序框架

将项目导入到IDE中

修改BatchJob，在main函数中输入下列代码

|  |
| --- |
| final long numSamples = args.length > 0 ? Long.parseLong(args[0]) : 1000000;  final ExecutionEnvironment env = ExecutionEnvironment.getExecutionEnvironment();  // count how many of the samples would randomly fall into // the unit circle DataSet<Long> count =  env.generateSequence(1, numSamples)  .map(new MapFunction<Long, Long>() {  @Override  public Long map(Long aLong) throws Exception {  double x = Math.random();  double y = Math.random();  return (x \* x + y \* y) < 1 ? 1L : 0L;  }  })  .reduce((x,y) -> x + y);  long theCount = count.collect().get(0);  System.out.println("We estimate Pi to be: " + (theCount \* 4.0 / numSamples)); |

在IDE中运行BatchJob程序，输出下列结果

|  |
| --- |
| We estimate Pi to be: 3.142076 |

运行Maven打包命令

|  |
| --- |
| mvn package |

在target目录下得到flink-project-1.0-SNAPSHOT.jar和original-flink-project-1.0-SNAPSHOT.jar

使用flink运行jar(standalone模式)

|  |
| --- |
| bin/start-cluster.sh  bin/flink run -m localhost:8081 -c com.aura.cases.BatchJob ~/Desktop/flink-project-1.0-SNAPSHOT.jar |

1. 编写Flink Streaming Job（可选练习，高级）

统计10s内Socket中输入的整型数之和，并打印出来