

## Flink : Scala Shell使用指南

Flink内置支持交互式的Scala Shell，我们既可以在本地安装模式下或者集群模式下运行它。我们可以通过下面的命令在单机模式下启动Shell:

```
bin/start-scala-shell.sh local
```

同样，我们可以通过启动Shell时指定remote参数，并提供JobManager的hostname和port等信息，如下：

```
bin/start-scala-shell.sh remote <hostname> <portnumber>
```

### 用法

Flink Scala Shell支持Batch和Streaming模式。在启动Shell的时候，它会自动初始化好相应的ExecutionEnvironments，在Batch模式下，可以使用benv；而在Streaming模式下可以使用senv。（注意，在Flink 1.0.0版本的时候，Scala Shell只支持Batch模式，对应的ExecutionEnvironment为env。）。下面的程序将展示如何在Scala shell中运行WordCount程序：

```
Scala-Flink> val text = env.fromElements(
  | "To be, or not to be,--that is the question:--",
  | "Whether 'tis nobler in the mind to suffer",
  | "The slings and arrows of outrageous fortune",
  | "Or to take arms against a sea of troubles,")
text: org.apache.flink.api.scala.DataSet[String] =
org.apache.flink.api.scala.DataSet@94aa195
Scala-Flink> 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@24a3a224
Scala-Flink> counts.print()
(a,1)
(against,1)
(and,1)
(arms,1)
(arrows,1)
(be,2)
```

```
(fortune,1)
(in,1)
(is,1)
(mind,1)
(nobler,1)
(not,1)
(of,2)
(or,2)
(outrageous,1)
(question,1)
(sea,1)
(slings,1)
(suffer,1)
(take,1)
(that,1)
(the,3)
(tis,1)
(to,4)
(troubles,1)
(whether,1)
```

上面程序的print()命令将自动地向JobManager发送task以便得到运行，然后会在终端显示计算的结果。

当然，你完全可以将计算的结果存入到文件中。在这种情况下，你需要显示地调用execute来运行你的程序：

```
env.execute("MyProgram")
```

上面的Batch模式的程序也可以在Streaming模式下运行：

```
Scala-Flink> val textStreaming = env.fromElements(
  "To be, or not to be,--that is the question:--",
  "Whether 'tis nobler in the mind to suffer",
  "The slings and arrows of outrageous fortune",
  "Or to take arms against a sea of troubles,")
Scala-Flink> val countsStreaming = textStreaming.flatMap {
  >_.toLowerCase.split("WWW+").map { (_, 1)}.keyBy(0).sum(1)
Scala-Flink> countsStreaming.print()
Scala-Flink> env.execute("Streaming Wordcount")
```

需要注意的是，在Streaming模式下，print不会自动地触发运行。

## 在YARN模式下运行Scala Shell

Scala shell可以连接到Flink cluster on YARN，我们可以通过下面命令实现：

```
bin/start-scala-shell.sh yarn
```

这个shell会从.yarn-properties文件中读取到Flink集群的部署信息，其实是通过配置文件里面的yarn.properties-file.location参数配置的目录或者临时目录获取的。如果没有Flink集群部署在YARN上，shell将会打印出相应的错误信息。

YARN容器的数量可以通过-n 参数指定。

Flink Shell支持显示历史命令和自动补全的功能。

## 添加外部依赖

给Scala shell添加外部依赖再正常不过了。它将你的shell程序和添加的依赖包自动地发送到obmanager。使用参数-a 或者 --addclasspath 添加额外的类，具体使用如下：

```
bin/start-scala-shell.sh [local | remote <host> <port>] --addclasspath <path/to/jar.jar>
```

本文翻译自：[https://ci.apache.org/projects/flink/flink-docs-master/apis/scala\\_shell.html](https://ci.apache.org/projects/flink/flink-docs-master/apis/scala_shell.html)

本博客文章除特别声明，全部都是原创！  
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。  
本文链接：[【】（）](#)