

Chapter 4. Setting Up a Development Environment for Apache Flink (启动一个flink开发环境)

前言
本章中，学习如何 设置环境开发、运行、调试Flink应用程序。讨论程序所需的环境，并获取本书的代码示例。使用这些示例来展示如何在IDE中执行和调试Flink应用程序。最后，会展示了如何引导Flink Maven项目

Required Software 需要的软件

开发和运行的环境
*unix环境有丰富的软件支持，也可以使用Linux (WSL) , Cygwin或Linux虚拟机的windows子系统
*可以使用java或者scala开发，java需要1.8或更高
*maven
*IntelliJ IDEA

Run and Debug Flink Applications in an IDE 在ide中调试和运行flink程序

Import the Book's Examples in an IDE (导入本书的例子到IDE)
*git clone https://github.com/streaming-with-flink/examples-scala
*也可以使用命令下载zip压缩包
> wget https://github.com/streaming-with-flink/examples-scala/archive/master.zip
> unzip master.zip
*项目结构如图
*使用idea打开项目，这里就不展开了

Run Flink Applications in an IDE (在ide中运行flink程序)
a 程序运行说明
找到 AverageSensorReadings 这个类，程序逻辑 是读取多个传感器的温度，并从华氏温度转换为摄氏温度，并每秒计算每个传感器的平均温度。程序的source, sink, operators都在main方法中，启动动程序后将在控制台输出结果。

b 程序最后执行逻辑
在调用StreamExecutionEnvironment.execute() 方法后 main() 方法将组合的dataflow提交到远程JobManager(master), Jobmanager将分发task给不同的TaskManager(worker), Jobmanager和TaskManager是运行在不同机器上的jvm进程
使用IDE调试的模式，在调用execute()后，在相同的jvm中启动jobmanager和taskmanager(默认slot和cpu线程数一样多)，这时整个flink程序是多线程的并在同一jvm中，这种模式适合在ide中启动flink

Debug Flink Applications in an IDE
*在ide调试模式下，可以像普通程序一样打断点调试
*默认启动的线程数和cpu线程数相同，是多线程的程序
*与通过将Flink程序发送到远程JobManager来运行Flink程序相反，此模式为单个JVM中执行，类的加载无法正常调试
*虽然在单个jvm中运行，records是要进行序列化进行跨线程通信

Bootstrap a Flink Maven Project (引导flink maven项目)

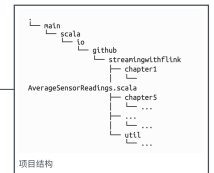
主要写了如何使用maven命令创建一个flink项目

使用如下maven命令创建一个名为quickstart的项目
mvn archetype:generate
-DarchetypeGroupId=org.apache.flink
-DarchetypeArtifactId=flink-quickstart-scala
-DarchetypeVersion=1.7.1
-DgroupId=org.apache.flink.quickstart
-DartifactId=flink-scala-project
-Dversion=0.1
-Dpackage=org.apache.flink.quickstart
-DinteractiveMode=false
项目的结构层次如图

最后使用如下命令在target目录生成jar包：mvn clean package -Pbuild-jar

Summary

本章讲了创建一个flink项目，5、6、7、8主要讲解api和operator、stateful functions、source and sink connectors



项目结构

```
2> SensorReading(sensor_31,1515014051000,23.92465618384873
3>
4> SensorReading(sensor_32,1515014051000,4.118569040962492
5>
6> SensorReading(sensor_38,1515014051000,14.78183542024247
7>
8> SensorReading(sensor_34,1515014051000,23.8714332525808
9>
```

控制台的输出结果



项目结构层次