

KvStore 作业问题说明

作业布置也有一段时间了，但是看作业完成情况并不是很好，大家的项目仍然存在某些问题，这篇文档，主要说明一下比较常见的问题，以及再详细说明一下本地调试的方法。

1)

有些同学对于processor里方法的返回值不是很清楚。

get 方法，返回的是一个map，在这里如果查询成功了，自然map中保存着查询结果，但是查询失败的时候，有些同学也向map中写点信息，这是不行的，当没有查询到key相应的数据时，请返回一个空的map或者null。

put和batchput方法，返回的是布尔值，写成功就true，失败就是false。这里得说明一下，当失败后，我们会尝试重发请求。

2)

我们提供的pom文件，并不是说让大家直接当做自己项目的pom文件，而是告诉大家，大家可以使用哪些依赖。

3)

大家需要在项目的根目录添加一个名为config的文件，文件中写入processorClass=\${your processor class name}。例如 processorClass=cn.helium.kvstore.MockProcessor。

4)

关于sdk这个jar包，请将它导入项目中并添加进mvn依赖。大家可以再将jar包放入项目lib文件夹，然后在pom文件中添加如下依赖。

```
<dependency>
  <groupId>cn.helium</groupId>
  <artifactId>kvstore</artifactId>
  <version>1.0-SNAPSHOT</version>
  <scope>system</scope>
  <systemPath>${project.basedir}/lib/original-kvstore-1.0-SNAPSHOT.jar
</systemPath>
</dependency>
```

5)

大家通过`KvStoreConfig.getHdfsUrl()`获得hdfs的url样式为：
`hdfs://ip:port`。请注意最后面是没有“/”的。这里之前的sdk中写的有些问题，将hdfs写成了hfsd。已修正，大家更新一下sdk吧。

KvStore 本地调试

貌似上一份文档有些没写清楚，大家还是不是很清楚如何在本地运行kvstore。这里再说明一下。

首先，大家需要一个hdfs环境。如何安装运行hadoop，就请大家自寻探索。

然后，大家需要将项目打包。这里不建议将所有的依赖以及自己的code打成一个大包。大家可以使用`mvn clean package dependency:copy-dependencies`命令来打包项目，这样打包出来的jar包很小，只包含你自己的code编译出来的class文件，而将项目所有的依赖全部放入target下的dependency文件夹下。需要说明的是，我们测试的时候发现，有些组的项目打包出来是一个空的jar包，里面并没有编译好的class文件。

接下来，就是运行了，大家可以使用`java -cp yours.jar:target/dependency/* cn.helium.kvstore.main.Main [--paramName paramValue]` 命令来运行自己的程序，这里说明一下，`java -cp`需要传入程序运行所需要的所有的class文件，也就是说，大家需要把自己项目打包出来的jar包以及依赖的jar包都传入。jar包之间使用：连接。传入所有的class文件后，需要指明MainClass。以及初始化一些参数。具体参数说明请看上一份文档。

当然，大家在起的时候，也可以设置一下java虚拟机的一些参数。