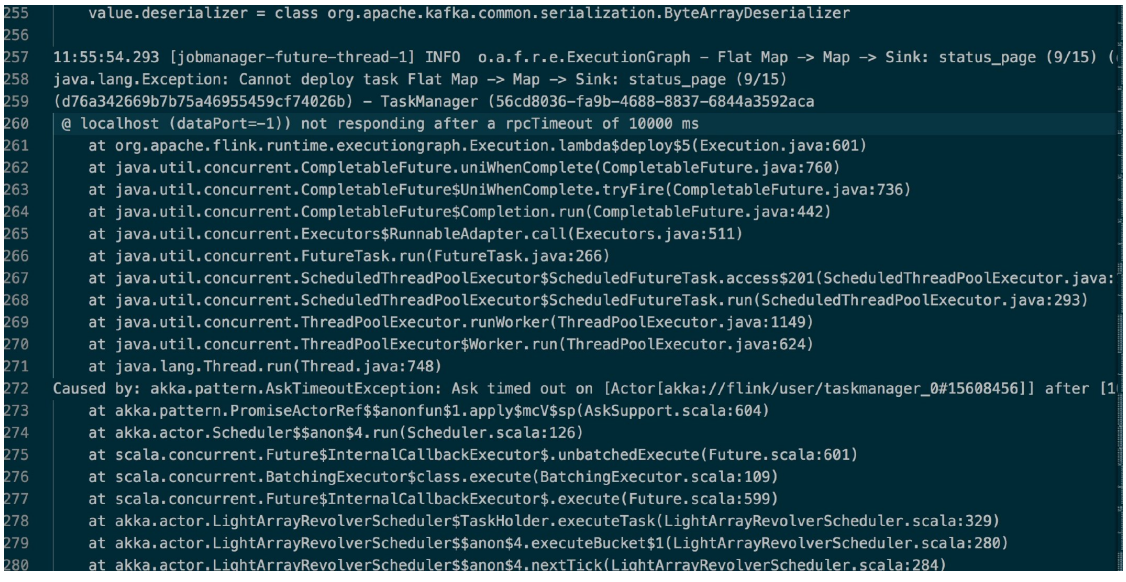
toc: true title: 《从0到1学习Flink》—— Flink parallelism 和 Slot 介绍 date: 2019-01-14 tags:

- Flink
- 大数据
- 流式计算

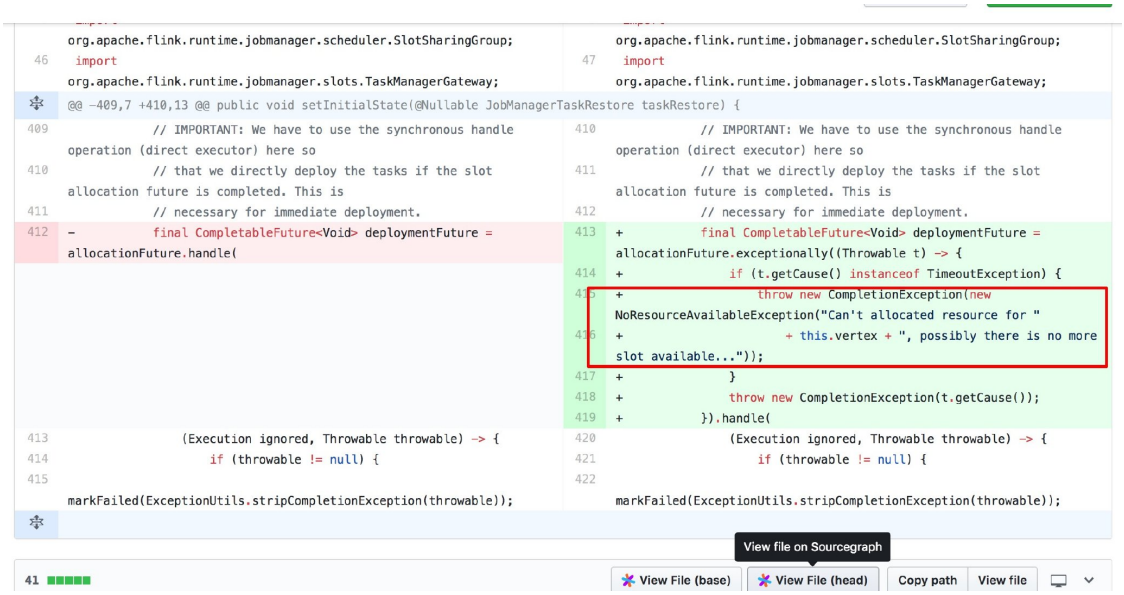
前言

之所以写这个是因为前段时间自己的项目出现过这样的一个问题：

```
Caused by: akka.pattern.AskTimeoutException:
Ask timed out on [Actor[akka://flink/user/taskmanager_0#15608456]]
after [10000 ms].
Sender[null] sent message of type
"org.apache.flink.runtime.rpc.messages.LocalRpcInvocation".
```



跟着这问题在 Flink 的 Issue 列表里看到了一个类似的问题：https://issues.apache.org/jira/browse/FLINK-9056 ，看下面的评论差不多就是 TaskManager 的 slot 数量不足的原因，导致 job 提交失败。在 Flink 1.63 中已经修复了变成抛出异常了。



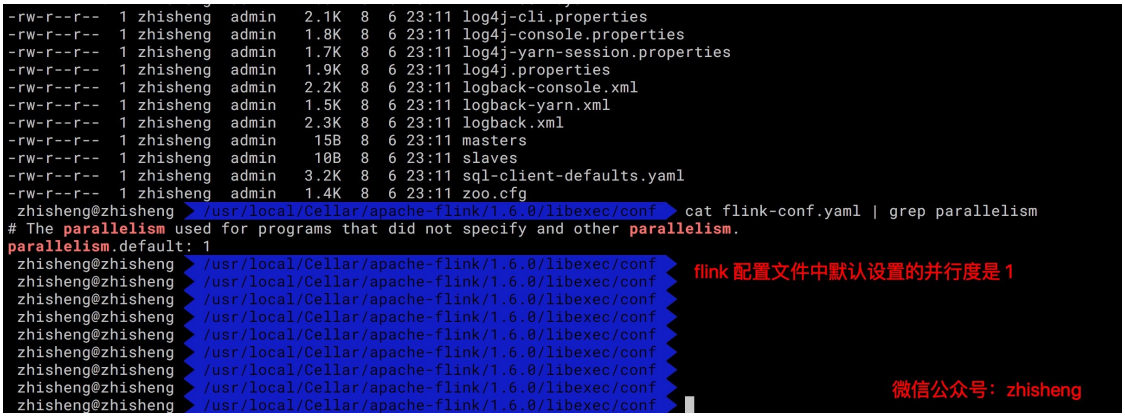
竟然知道了是因为 slot 不足的原因了，那么我们就要先了解下 slot 是什么东东呢？不过文章这里先介绍下 parallelism。

什么是 parallelism?

如翻译这样，parallelism 是并行的意思，在 Flink 里面代表每个任务的并行度，适当的提高并行度可以大大提高 job 的执行效率，比如你的 job 消费 kafka 数据过慢，适当调大可能就消费正常了。

那么在 Flink 中怎么设置并行度呢?

如何设置 parallelism?



如上图，在 flink 配置文件中可以查看到默认并行度是 1，

```
cat flink-conf.yaml | grep parallelism

# The parallelism used for programs that did not specify and other
parallelism.
parallelism.default: 1
```

所以你如何在你的 flink job 里面不设置任何的 parallelism 的话，那么他也会有一个默认的 parallelism = 1。那也意味着你可以修改这个配置文件的默认并行度。

如果你是用命令行启动你的 Flink job，那么你也可以这样设置并行度(使用 -p 并行度)：

```
./bin/flink run -p 10 ../word-count.jar
```

你也可以通过这样来设置你整个程序的并行度：

```
StreamExecutionEnvironment env =
StreamExecutionEnvironment.getExecutionEnvironment();
env.setParallelism(10);
```

注意：这样设置的并行度是你整个程序的并行度，那么后面如果你的每个算子不单独设置并行度覆盖的话，那么后面每个算子的并行度就都是这里设置的并行度的值了。

如何给每个算子单独设置并行度呢?

```
data.keyBy(new xxxKey())
    .flatMap(new XxxFlatMapFunction()).setParallelism(5)
    .map(new XxxMapFunction).setParallelism(5)
    .addSink(new XxxSink()).setParallelism(1)
```

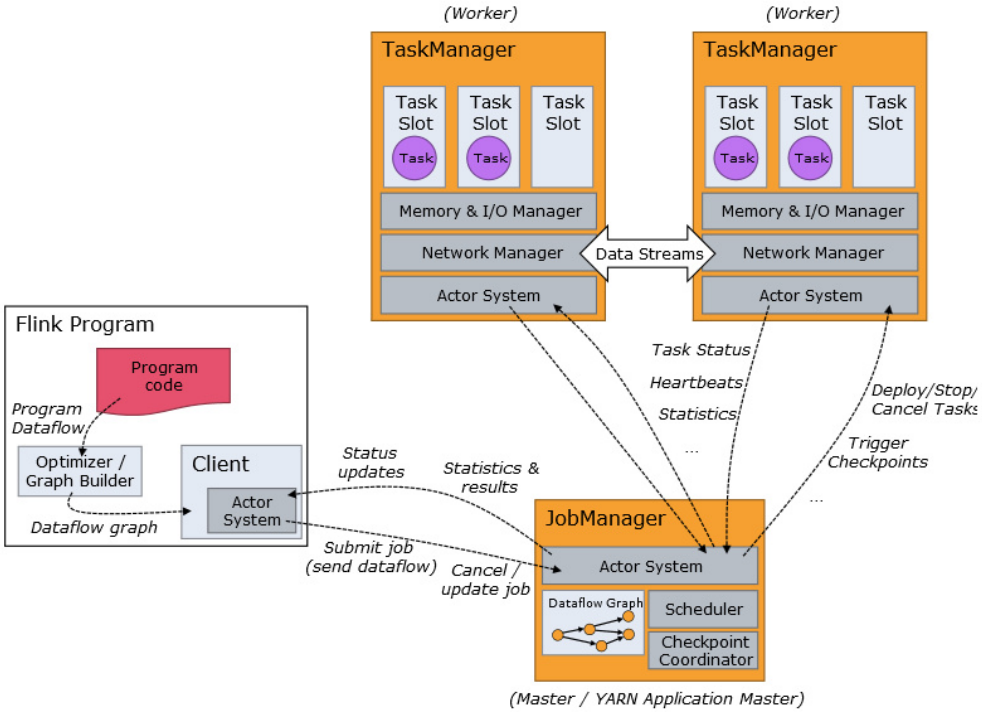
如上，就是在每个算子后面单独的设置并行度，这样的话，就算你前面设置了 env.setParallelism(10) 也是会被覆盖的。

这也说明优先级是：算子设置并行度 > env 设置并行度 > 配置文件默认并行度

并行度讲到这里应该都懂了，下面 zhisheng 就继续跟你讲讲 什么是 slot?

什么是 slot?

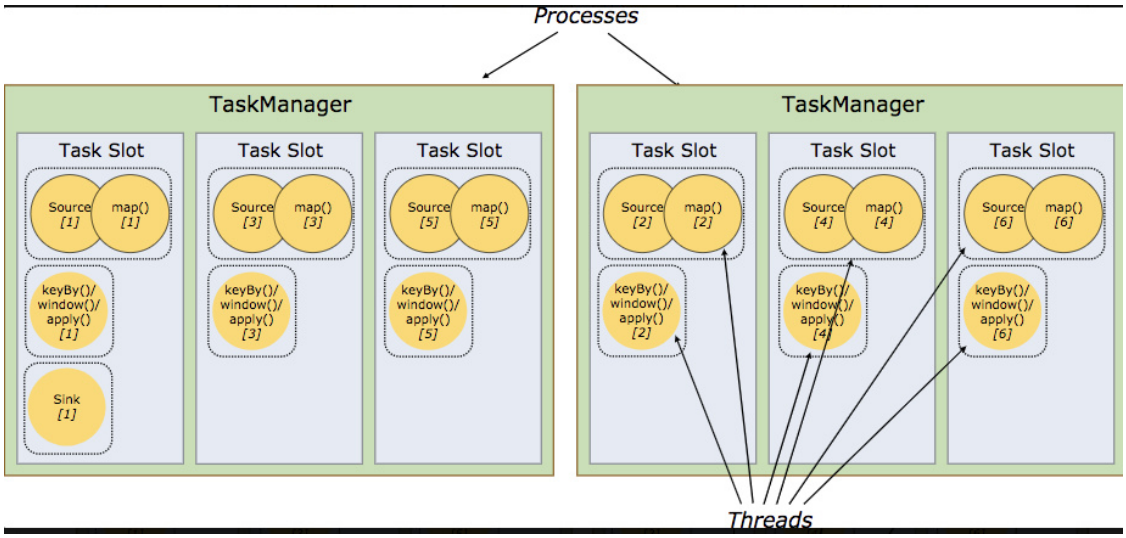
其实什么是 slot 这个问题之前在第一篇文章 《从0到1学习Flink》—— Apache Flink 介绍 中就介绍过了，这里再讲细一点。



图中 Task Manager 是从 Job Manager 处接收需要部署的 Task，任务的并行性由每个 Task Manager 上可用的 slot 决定。每个任务代表分配给任务槽的一组资源，slot 在 Flink 里面可以认为是资源组，Flink 将每个任务分成子任务并且将这些子任务分配到 slot 来并行执行程序。

例如，如果 Task Manager 有四个 slot，那么它将为每个 slot 分配 25% 的内存。可以在一个 slot 中运行一个或多个线程。 同一 slot 中的线程共享相同的 JVM。 同一 JVM 中的任务共享 TCP 连接和心跳消息。Task Manager 的一个 Slot 代表一个可用线程，该线程具有固定的内存，注意 Slot 只对内存隔离，没有对 CPU 隔离。默认情况下，Flink 允许子任务共享 Slot，即使它们是不同 task 的 subtask，只要它们来自相同的 job。这种共享可以有更好的资源利用率。

文字说的比较干，zhisheng 这里我就拿下面的图片来讲解：



上面图片中有两个 Task Manager，每个 Task Manager 有三个 slot，这样我们的算子最大并行度那么就可以达到 6 个，在同一个 slot 里面可以执行 1 至多个子任务。

那么再看上面的图片，source/map/keyby/window/apply 最大可以有 6 个并行度，sink 只用了 1 个并行。

每个 Flink TaskManager 在集群中提供 slot。 slot 的数量通常与每个 TaskManager 的可用 CPU 内核数成比例。一般情况下你的 slot 数是你每个 TaskManager 的 cpu 的核数。

但是 flink 配置文件中设置的 task manager 默认的 slot 是 1。

```
zhisheng@zhisheng /usr/local/Cellar/apache-flink/1.6.0/libexec/conf
zhisheng@zhisheng /usr/local/Cellar/apache-flink/1.6.0/libexec/conf
zhisheng@zhisheng /usr/local/Cellar/apache-flink/1.6.0/libexec/conf cat flink-conf.yaml | grep taskmanager
taskmanager.heap.size: 1024m
taskmanager.numberOfTaskSlots: 1
# taskmanager.memory.preallocate: false
# taskmanager.network.memory.fraction: 0.1
# taskmanager.network.memory.min: 67108864
# taskmanager.network.memory.max: 1073741824
zhisheng@zhisheng /usr/local/Cellar/apache-flink/1.6.0/libexec/conf
```

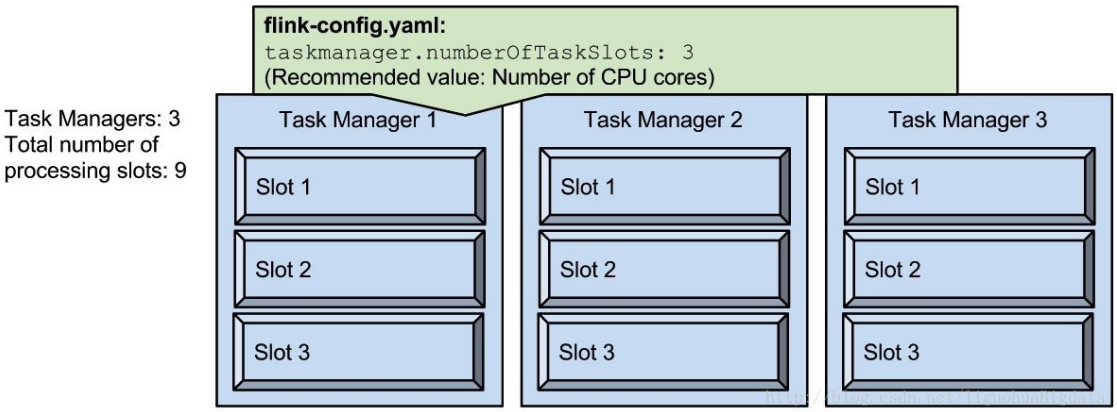
Task Manager 中默认就只有一个 slot

微信公众号：zhisheng

slot 和 parallelism

下面给出官方的图片来更加深刻的理解下 slot：

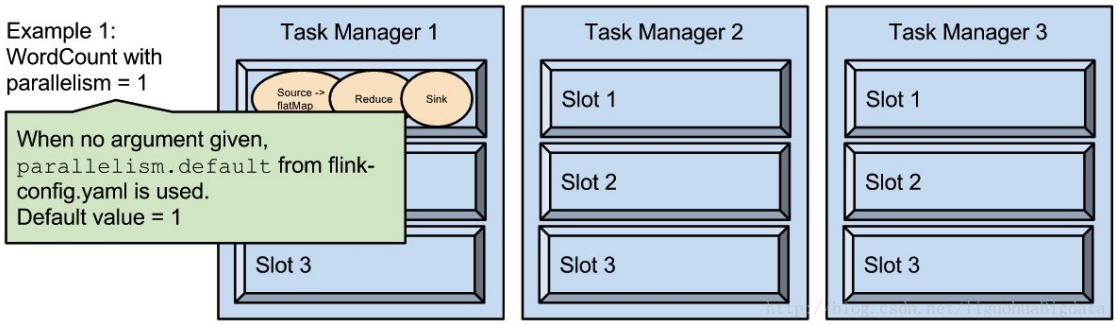
1、slot 是指 taskmanager 的并发执行能力



taskmanager.numberOfTaskSlots:3

每一个 taskmanager 中的分配 3 个 TaskSlot, 3 个 taskmanager 一共有 9 个 TaskSlot。

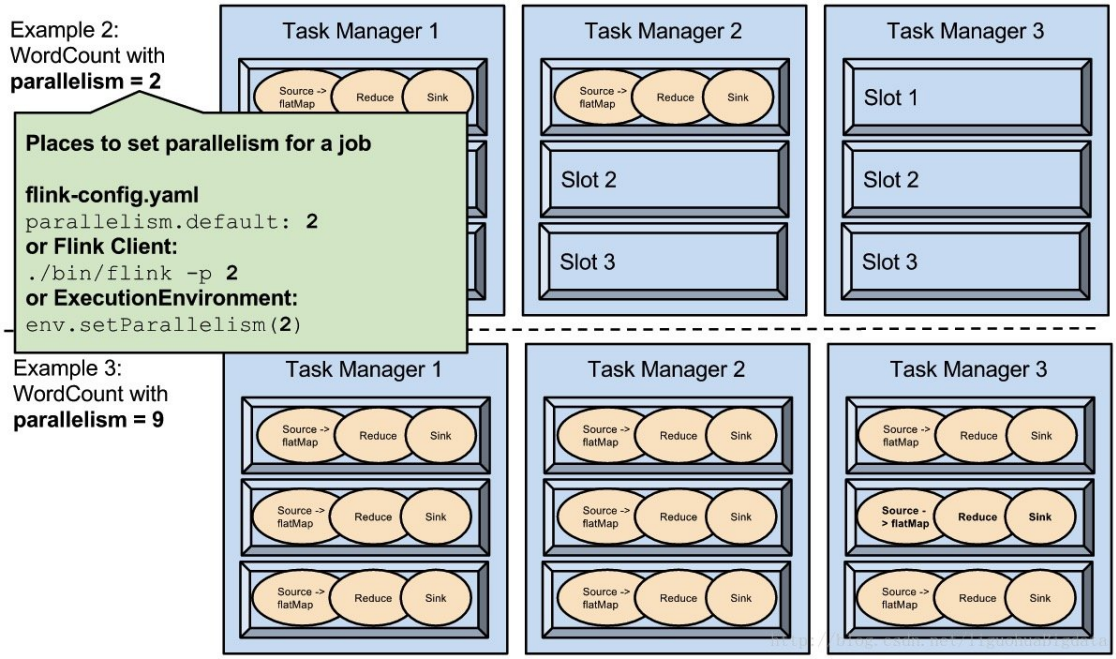
2、parallelism 是指 taskmanager 实际使用的并发能力



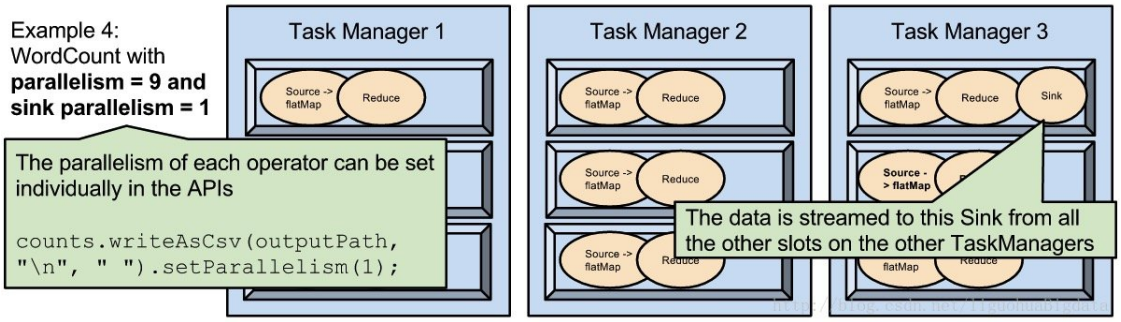
parallelism.default:1

运行程序默认的并行度为 1，9 个 TaskSlot 只用了 1 个，有 8 个空闲。设置合适的并行度才能提高效率。

3、parallelism 是可配置、可指定的



上图中 example2 每个算子设置的并行度是 2， example3 每个算子设置的并行度是 9。



example4 除了 sink 是设置的并行度为 1，其他算子设置的并行度都是 9。

好了，既然并行度和 slot zhisheng 都带大家过了一遍了，那么再来看文章开头的问题：slot 资源不够。

问题原因

现在这个问题的答案其实就已经很明显了，就是我们设置的并行度 parallelism 超过了 Task Manager 能提供的最大 slot 数量，所以才会报这个错误。

再来拿我的代码来看吧，当时我就是只设置了整个项目的并行度：

```
env.setParallelism(15);
```

为什么要设置 15 呢，因为我项目消费的 Kafka topic 有 15 个 parttion，就想着让一个并行去消费一个 parttion，没曾想到 Flink 资源的不够，稍微降低下 并行度为 10 后就没出现这个错误了。

总结

本文由自己项目生产环境的一个问题来讲解了自己对 Flink parallelism 和 slot 的理解，并告诉大家如何去设置这两个参数，最后也指出了问题的原因所在。

Github 代码仓库

https://github.com/zhisheng17/flink-learning/

以后这个项目的所有代码都将放在这个仓库里，包含了自己学习 flink 的一些 demo 和博客

相关文章

1、《从0到1学习Flink》—— Apache Flink 介绍

2、《从0到1学习Flink》—— Mac 上搭建 Flink 1.6.0 环境并构建运行简单程序入门

3、《从0到1学习Flink》—— Flink 配置文件详解

4、《从0到1学习Flink》—— Data Source 介绍

5、《从0到1学习Flink》—— 如何自定义 Data Source ？

6、《从0到1学习Flink》—— Data Sink 介绍

7、《从0到1学习Flink》—— 如何自定义 Data Sink ？

8、《从0到1学习Flink》—— Flink Data transformation(转换)

9、《从0到1学习Flink》—— 介绍Flink中的Stream Windows

10、《从0到1学习Flink》—— Flink 中的几种 Time 详解

11、《从0到1学习Flink》—— Flink 写入数据到 ElasticSearch

12、《从0到1学习Flink》—— Flink 项目如何运行?

13、《从0到1学习Flink》—— Flink 写入数据到 Kafka