

MapReduce 调优

目录

1、MapReduce 参数调优	1
1.1、资源相关参数.....	1
1.2、容错相关参数.....	2
1.3、本地运行 mapreduce 作业	2
1.4、效率和稳定性相关参数.....	3

1、MapReduce 参数调优

1.1、资源相关参数

以下参数是在用户自己的 **mr** 应用程序中配置就可以生效：

1、`mapreduce.map.memory.mb`：一个 Map Task 可使用的资源上限(单位:MB)，默认为 1024。
如果 Map Task 实际使用的资源量超过该值，则会被强制杀死。

2、`mapreduce.reduce.memory.mb`：一个 Reduce Task 可使用的资源上限（单位:MB），默认为 1024。如果 Reduce Task 实际使用的资源量超过该值，则会被强制杀死。

3、`mapreduce.map.java.opts`：Map Task 的 JVM 参数，你可以在此配置默认的 java heap size 等参数, e.g.

“`-Xmx1024m -verbose:gc -Xloggc:/tmp/@taskid@.gc`”（@taskid@会被 Hadoop 框架自动换为相应的 taskid），默认值：“”

4、`mapreduce.reduce.java.opts`：Reduce Task 的 JVM 参数，你可以在此配置默认的 java heap size 等参数, e.g.

“`-Xmx1024m -verbose:gc -Xloggc:/tmp/@taskid@.gc`”，默认值：“”

5、`mapreduce.map.cpu.vcores`：每个 Map task 可使用的最多 cpu core 数目，默认值: 1

6、`mapreduce.reduce.cpu.vcores`：每个 Reduce task 可使用的最多 cpu core 数目，默认值: 1

以下参数应该在 **yarn** 启动之前就配置在服务器的配置文件中才能生效：

7、`yarn.scheduler.minimum-allocation-mb=1024` 给应用程序 container 分配的最小内存

8、`yarn.scheduler.maximum-allocation-mb=8192` 给应用程序 container 分配的最大内存

9、`yarn.scheduler.minimum-allocation-vcores=1`

10、`yarn.scheduler.maximum-allocation-vcores=32`

11、`yarn.nodemanager.resource.memory-mb=8192`

shuffle 性能优化的关键参数，应在 yarn 启动之前就配置好：

12、`mapreduce.task.io.sort.mb=100` //shuffle 的环形缓冲区大小，默认 100m

13、`mapreduce.map.sort.spill.percent=0.8` //环形缓冲区溢出的阈值，默认 80%

1.2、容错相关参数

1、`mapreduce.map.maxattempts=4`: 每个 Map Task 最大重试次数，一旦重试参数超过该值，则认为 Map Task 运行失败

2、`mapreduce.reduce.maxattempts=4`: 每个 Reduce Task 最大重试次数，一旦重试参数超过该值，则认为 Map Task 运行失败

3、`mapreduce.map.failures.maxpercent=0`: 当失败的 Map Task 失败比例超过该值为，整个作业则失败。如果你的应用程序允许丢弃部分输入数据，则该值设为一个大于 0 的值，比如 5，表示如果有低于 5% 的 Map Task 失败（如果一个 Map Task 重试次数超过 `mapreduce.map.maxattempts`，则认为这个 Map Task 失败，其对应的输入数据将不会产生任何结果），整个作业仍然认为成功。

4、`mapreduce.reduce.failures.maxpercent=0`: 当失败的 Reduce Task 失败比例超过该值为，整个作业则失败

5、`mapreduce.task.timeout`: Task 超时时间，经常需要设置的一个参数，该参数表达的意思为：如果一个 task 在一定时间内没有任何进入，即不会读取新的数据，也没有输出数据，则认为该 task 处于 block 状态，可能是卡住了，也许永远会卡主，为了防止因为用户程序永远 block 住不退出，则强制设置了一个该超时时间（单位毫秒），默认是 300000。如果你的程序对每条输入数据的处理时间过长（比如会访问数据库，通过网络拉取数据等），建议将该参数调大，该参数过小常出现的错误提示是“ AttemptID:attempt_14267829456721_123456_m_000224_0 Timed out after 300 secsContainer killed by the ApplicationMaster.”。

1.3、本地运行 mapreduce 作业

设置以下几个参数:

`mapreduce.framework.name=local`

`mapreduce.jobtracker.address=local`

1.4、效率和稳定性相关参数

- 1、`mapreduce.map.speculative`: 是否为 Map Task 打开推测执行机制，默认为 `false`
- 2、`mapreduce.reduce.speculative`: 是否为 Reduce Task 打开推测执行机制，默认为 `false`
- 3、`mapreduce.job.user.classpath.first` & `mapreduce.task.classpath.user.precedence`: 当同一个 class 同时出现在用户 jar 包和 hadoop jar 中时, 优先使用哪个 jar 包中的 class, 默认为 `false`, 表示优先使用 hadoop jar 中的 class。
- 4、`mapreduce.input.fileinputformat.split.minsize`: `FileInputFormat` 做切片时的最小切片大小
- 5、`mapreduce.input.fileinputformat.split.maxsize`: `FileInputFormat` 做切片时的最大切片大小 (切片的默认大小就等于 `blocksize`, 即 134217728)