8.5 容错处理

8.5.1 从检查点恢复故障

8.5.2 故障恢复中的限制

8.5.1 从检查点恢复故障



- Spark程序一般要长时间运行，由于有时会因为系统故障或者JVM故障导致程序退出、系统重启等，因此，Spark程序需要具备恢复能力。
- 正确配置容错环境包括选择容错的输入源，记录输入源位置偏移量，保存检查点和预写日志中间状态，以及使用容错的接收器。其中，记录输入源位置偏移量、检查点、预写日志由Spark引擎完成。使用者只需要提供检查点路径，Spark引擎就会保存恢复的必要的数据。
- 比如，在8.4.2节中的实例中，在文件接收器内设置选项checkpointLocation，如果发生故障，由于检查点和预写日志保存了位置偏移量等信息，因此Spark程序可以通过这些信息恢复到之前的查询的进度和状态。

8.5.2 故障恢复中的限制



- 虽然Spark设计了分离的输入源和接收器，然而在程序停止运行或者故障发生后，有时候为了解决故障问题，我们可能需要修改部分程序代码并重启查询。
- 但是，这里需要说明的是，如果这时仍使用旧的检查点数据来恢复程序运行，那么，修改部分代码的做法是不被允许的；或者即使允许，也会导致运行的结果可能有无法预期的错误。

8.5.2 故障恢复中的限制



以下是关于参数修改的一些规则。

•输入源的类型和数量的更改：不被允许。

•输入源的参数的更改：部分不会影响到检查点状态的参数可以修改，如修改Kafka的maxOffsetsPerTrigger等限速参数；而修改Kafka的主题或者文件路径则不被允许。

•接收器类型的更改：File接收器改为Kafka接收器是允许的，但是Kafka只能接收到新的数据；Kafka接收器改为File接收器，则不被允许；Kafka接收器和Foreach接收器可以互相替换。

•接收器参数的更改：更改File接收器的路径不被允许；Kafka接收器的输出主题允许被更改；Foreach接收器自定义的函数可以被更改。

•projection、filter、map等类似操作的更改：部分允许，如增加或者删除过滤条件等。

•有状态的操作的更改：有些流计算查询操作需要保存状态数据到检查点，以便在数据持续到来时更新查询结果，这种操作的更改不被允许。