

损坏或不完整的检查点会导致 Channel 重启得非常慢，因为 Channel 需要从所有的数据文件读取和重新进行。

为了避免这个问题，推荐将 `useDualCheckpoints` 参数值设置为 `true`，`backupCheckpointDir` 参数也要设置，建议这个目录在不同的磁盘上而不是在最初检查点存储的磁盘上。当这些参数设置了，在检查点完成之后 Flume 将尽快备份它到 `backupCheckpointDir`。这将确保一旦 Channel 在一个短暂的时间内操作（第一个检查点写出和备份所需的足够时间），它将能够从检查点重新启动，即使最新的一个是损坏或不完整的，这大大减少了启动时间。连续检查点之间的时间间隔是由 `checkpointInterval` 参数控制的。

剩下的参数是用来调整 File Channel 的性能和磁盘使用的。Flume 在数据文件的结尾追加每个事件，并且只要文件包含的事件没有被读取和提交完毕，或事件一直被写入，Flume 就一直保留这个文件。在 Flume 关闭文件并将它作为只读文件之前，`maxFileSize` 参数控制了文件增加的最大值。该参数默认为大约 1.6GB，也是该参数的最大值。如果设置的值比默认值更大，文件仍会在它达到默认大小时保存关闭。需要注意的是，当且仅当所有写入 Channel 的事件被读取提交，文件才被删除（实际上，当检查点追踪的最后一个事件被读取和提交时文件才被删除）。如果文件需要更快地被删除，该参数则应设置为更小的值，这样所有的事件会更快地从单独的文件被读取（因为文件更小，那么它比大文件包含更少的事件）。但是该值设置得过小将导致磁盘创建过多的文件，所以最好不要从默认值减小该值。关于文件的删除 Channel 也是保守的。即使文件没有任何将要被读取和提交的事件，Channel 也总是在每个数据目录保留两个文件。Channel 也不会删除正在写入的文件。

为了确保 Channel 不写入低空间的磁盘，则可以设置 `minimumRequiredSpace` 参数。一旦指

◀ 89

定磁盘的磁盘空间低于该参数的值（或 1MB，这是更高的），Channel 就停止操作。为了保存系统资源且不影响性能，Channel 并没有在每次写入的时候检查磁盘空间，而是周期性检查，它维护内部计数器来计算磁盘的可用空间。假设没有其他 Channel 或程序写入这个磁盘，这将使可用的磁盘空间减少得更快。

前面已经讨论过，File Channel 定期写检查点到磁盘上。因为检查点反映了 Channel 写入时候的状态，所以写入检查点时，操作不能正常进行。`keep-alive` 参数与 Memory Channel 中的工作相似，指明了等待容量可用的时间，或检查点完成一个写入事件的等待时间，或从 Channel 取走的事件等待其可用的时间。

通常当 Channel 有更多可用磁盘时，会有更好的性能。如果足够的磁盘空间可用，Channel 应该配置一个大的容量来适应下游故障或通信高峰。基于每个 Channel 预想的