构造IndexWriter对象（二）

构造一个IndexWriter对象的流程总体分为下面三个部分：

- 设置索引目录Directory
- 设置IndexWriter的配置信息IndexWriterConfig
- 调用IndexWriter的构造函数

在文章构造IndexWriter对象（一）中我们讲到了设置IndexWriter的配置信息IndexWriterConfig中不可配置的内容，接着我们继续介绍可配置的内容。

设置IndexWriter的配置信息IndexWriterConfig

可变配置

可变配置包含的内容有：MergePolicy、MaxBufferedDocs、RAMBufferSizeMB、MergedSegmentWarmer、UseCompoundFile、CommitOnClose、CheckPendingFlushUpdate。

可变配置指的是在构造完IndexWriter对象后，在运行过程也可以随时调整的配置。

MergePolicy

MergePolicy是段的合并策略，它用来描述如何从索引目录中找到满足合并要求的段集合（segment set），在前面的文章了已经介绍了LogMergePolicy、TieredMergePolicy两种合并策略，这里不赘述。

MergePolicy可以通过IndexWriterConfig.setMergePolicy(MergePolicy mergePolicy)方法设置，在版本Lucene7.5.0中默认值使用TieredMergePolicy，如果修改了MergePolicy，那么下一次的段的合并会使用新的合并策略。

MaxBufferedDocs、RAMBufferSizeMB

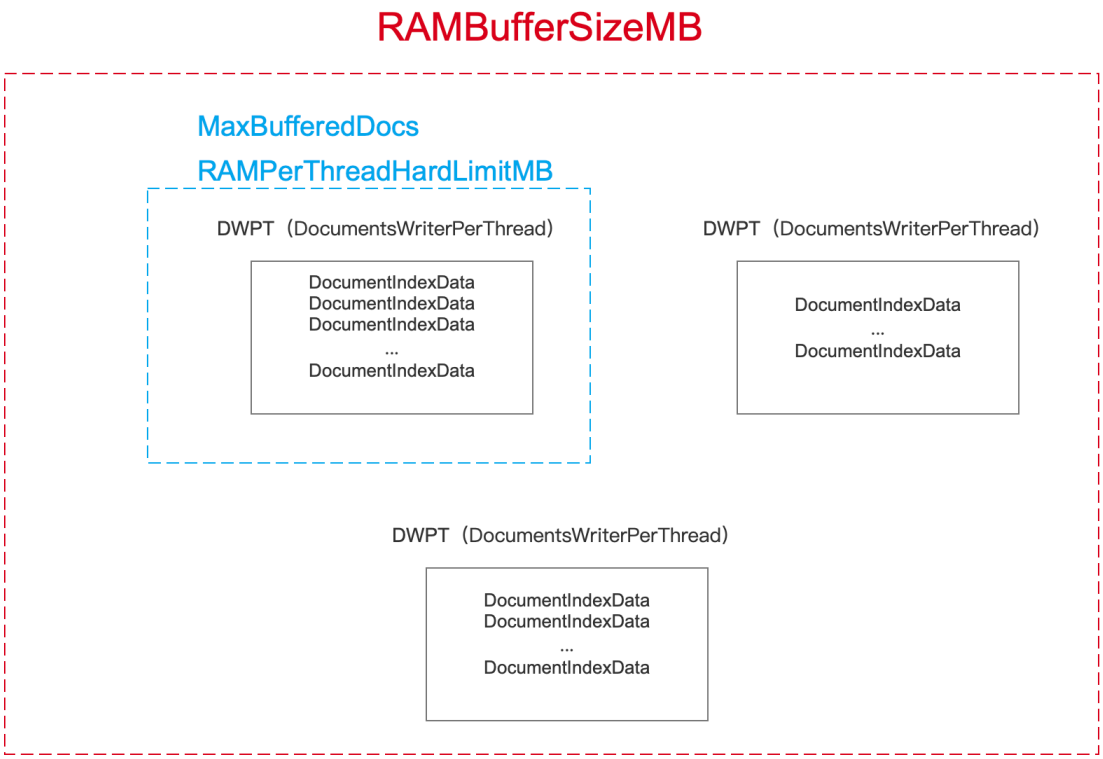
RAMBufferSizeMB描述了索引信息被写入到磁盘前暂时缓存在内存中允许的最大使用内存值，而MaxBufferedDocs则是描述了索引信息被写入到磁盘前暂时缓存在内存中允许的文档最大数量，**这里注意的是，MaxBufferedDocs指的是一个DWPT允许添加的最大文档数量，在多线程下，可以同时存在多个DWPT（DWPT的概念见文档的增删改（中）），而MaxBufferedDocs并不是所有线程的DWPT中添加的文档数量和值。**

每次执行文档的增删改后，会调用FlushPolicy（flush策略）判断是否需要执行自动flush（见文档提交之flush（一）），在Lucene7.5.0版本中，仅提供一个flush策略，即FlushByRamOrCountsPolicy，该策略正是依据MaxBufferedDocs、RAMBufferSizeMB来判断是否需要执行自动flush。

在文档的增删改系列文章中，详细介绍了自动flush，以及FlushByRamOrCountsPolicy的概念，这里不赘述。

另外在文章中构造IndexWriter对象（一）中我们说到一个不可配置值，即RAMPerThreadHardLimitMB，该值被允许设置的值域为0~2048M，它用来描述每一个DWPT允许缓存的最大的索引量。

图1：



如果你没有看过文档的增删改系列文章，那么可以简单的将DWPT理解为一个容器，存放每一篇文档对应转化后的索引信息，在多线程下执行文档的添加操作时，每个线程都会持有一个DWPT，然后将一篇文档的信息转化为索引信息（DocumentIndexData），并添加到DWPT中。

如果每一个DWPT中的DocumentIndexData的**个数**超过MaxBufferedDocs时，那么就会触发自动flush，将DWPT中的索引信息生成为一个段，如图1所示，MaxBufferedDocs影响的是一个DWPT。

如果每一个DWPT中的所有DocumentIndexData的**索引内存占用量**超过RAMPerThreadHardLimitMB，那么就会触发自动flush，将DWPT中的索引信息生成为一个段，如图1所示，RAMPerThreadHardLimitMB影响的是一个DWPT。

如果所有DWPT（例如图1中的三个DWPT）中的DocumentIndexData的**索引内存占用量**超过RAMBufferSizeMB，那么就会触发自动flush，将DWPT中的索引信息生成为一个段，如图1所示，RAMPerThreadHardLimitMB影响的是所有的DWPT。

为什么要提供不可配置RAMPerThreadHardLimitMB：

- 为避免翻译歧义，直接给出源码中的英文注释

```
Sets the maximum memory consumption per thread triggering a forced flush if
exceeded. A DocumentsWriterPerThread(DWPT) is forcefully flushed once it
exceeds this limit even if the RAMBufferSizeMB has not been exceeded. This is a
safety limit to prevent a DocumentsWriterPerThread from address space
exhaustion due to its internal 32 bit signed integer based memory addressing.
The given value must be less that 2GB (2048MB)
```

- 上文中的forcefully flushed即自动flush

MaxBufferedDocs、RAMBufferSizeMB分别可以通过IndexWriterConfig.setMaxBufferedDocs(int maxBufferedDocs)、IndexWriterConfig.setRAMBufferSizeMB(double ramBufferSizeMB)方法设置，其中MaxBufferedDocs默认值为-1，表示在flush策略中不依据该值，RAMBufferSizeMB默认值为16M。

MergedSegmentWarmer

MergedSegmentWarmer即预热合并后的新段，它描述的是在执行段的合并期间，提前获得合并后生成的新段的信息，由于段的合并和文档的增删改是并发操作，所以使用该配置可以提高性能，至于为什么能提高性能，以及提高了什么性能可以看文章执行段的合并（四）关于`生成IndexReaderWarmer`的介绍。

MergedSegmentWarmer可以通过IndexWriterConfig.setMergedSegmentWarmer(IndexReaderWarmer mergeSegmentWarmer)方法设置，MergedSegmentWarmer默认为null。

UseCompoundFile

UseCompoundFile是布尔值，当该值为true，那么通过flush、commit的操作生成索引使用的数据结构都是复合索引文件，即索引文件.cfs、.cfe。

UseCompoundFile可以通过IndexWriterConfig.setUseCompoundFile(boolean useCompoundFile)方法设置，UseCompoundFile默认为true。

注意的是执行段的合并后生成的新段对应的索引文件，即使通过上述方法另UseCompoundFile为true，但还是有可能生成非复合索引文件，其原因可以看文章执行段的合并（三）中`生成复合索引文件`的流程介绍。

CommitOnClose

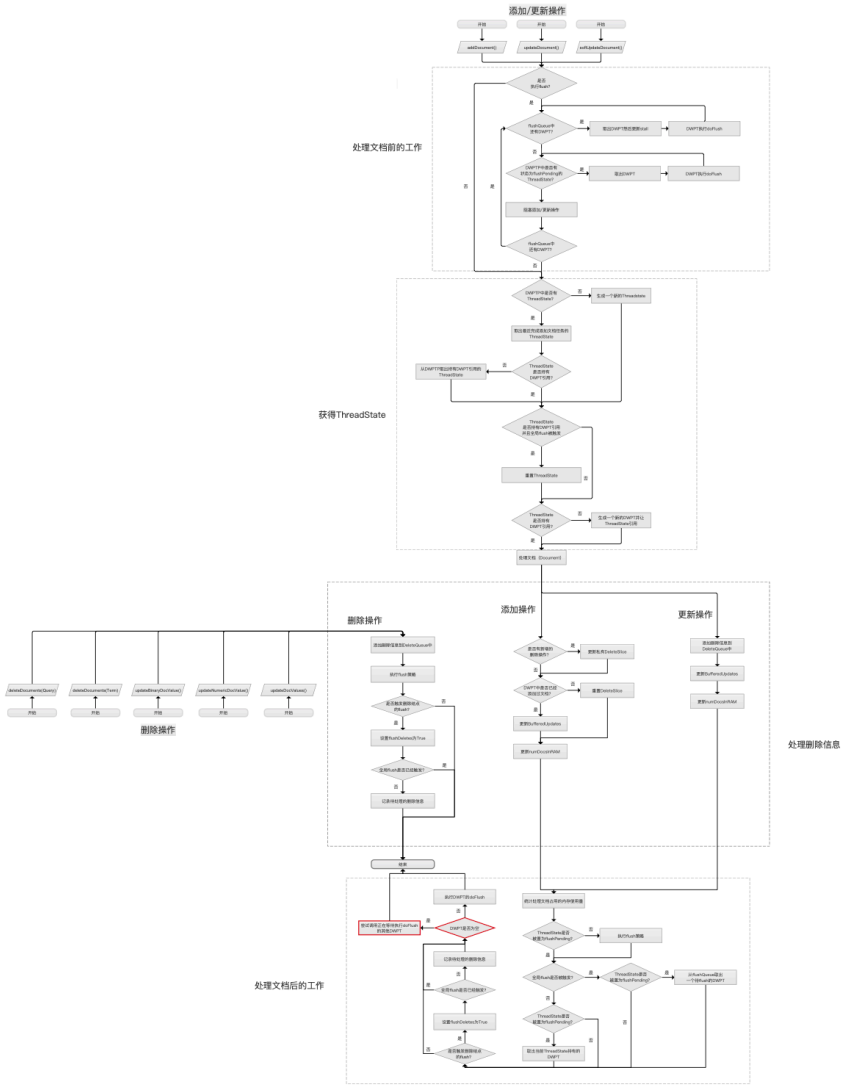
该值为布尔值，它会影响IndexWriter.close()的执行逻辑，如果设置为true，那么会先应用（apply）所有的更改，即执行commit操作，否则上一次commit操作后的所有更改都不会保存，直接退出。

CommitOnClose可以通过IndexWriterConfig.setCommitOnClose(boolean commitOnClose)方法设置，CommitOnClose默认为true。

CheckPendingFlushUpdate

该值为布尔值，如果设置为true，那么当一个执行添加或更新文档操作的线程完成处理文档的工作后，会尝试去帮助待flush的DWPT，其执行的时机点见下图中红框标注的两个流程点，图2为文档的增删改的完整流程图：

图2：



点击查看大图

结语

在下一篇文章中，我们继续介绍构造一个IndexWriter对象的流程的剩余部分，即调用IndexWriter的构造函数。

点击下载附件