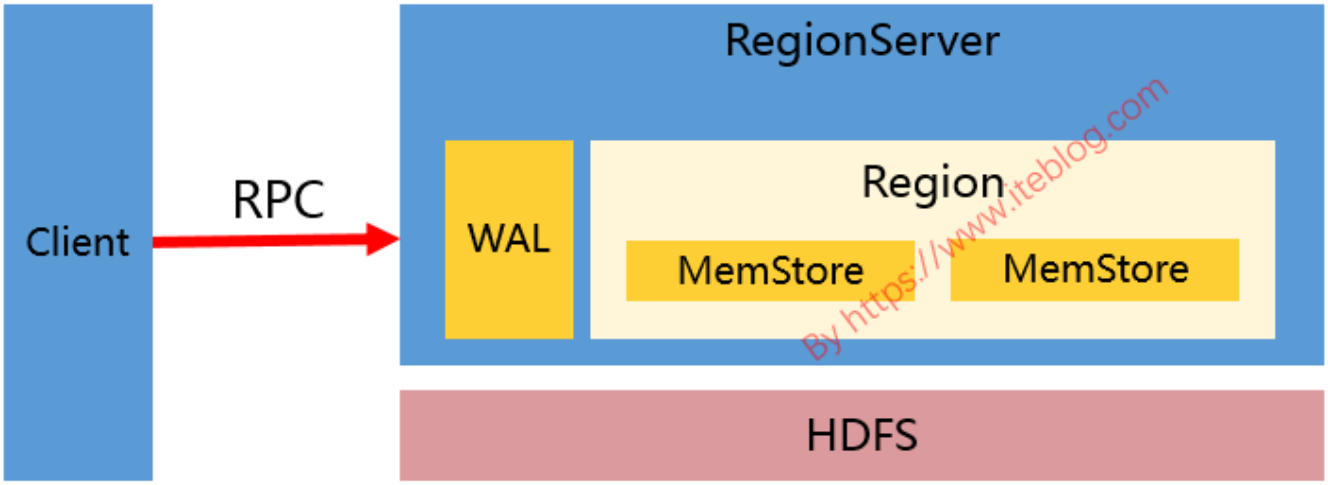
为什么不建议在 HBase 中使用过多的列族

我们知道，一张 HBase 表包含一个或多个列族。HBase 的官方文档中关于 HBase 表的列族的个数有两处描述：A typical schema has between 1 and 3 column families per table. HBase tables should not be designed to mimic RDBMS tables. 以及 HBase currently does not do well with anything above two or three column families so keep the number of column families in your schema low.

上面两句话其实都是说一件事，HBase 中每张表的列族个数建议设在1~3之间。其实，HBase 支持的列族个数并没有限制，但为什么文档建议在1~3之间呢？我将从几个方面来阐述这么做的原因。

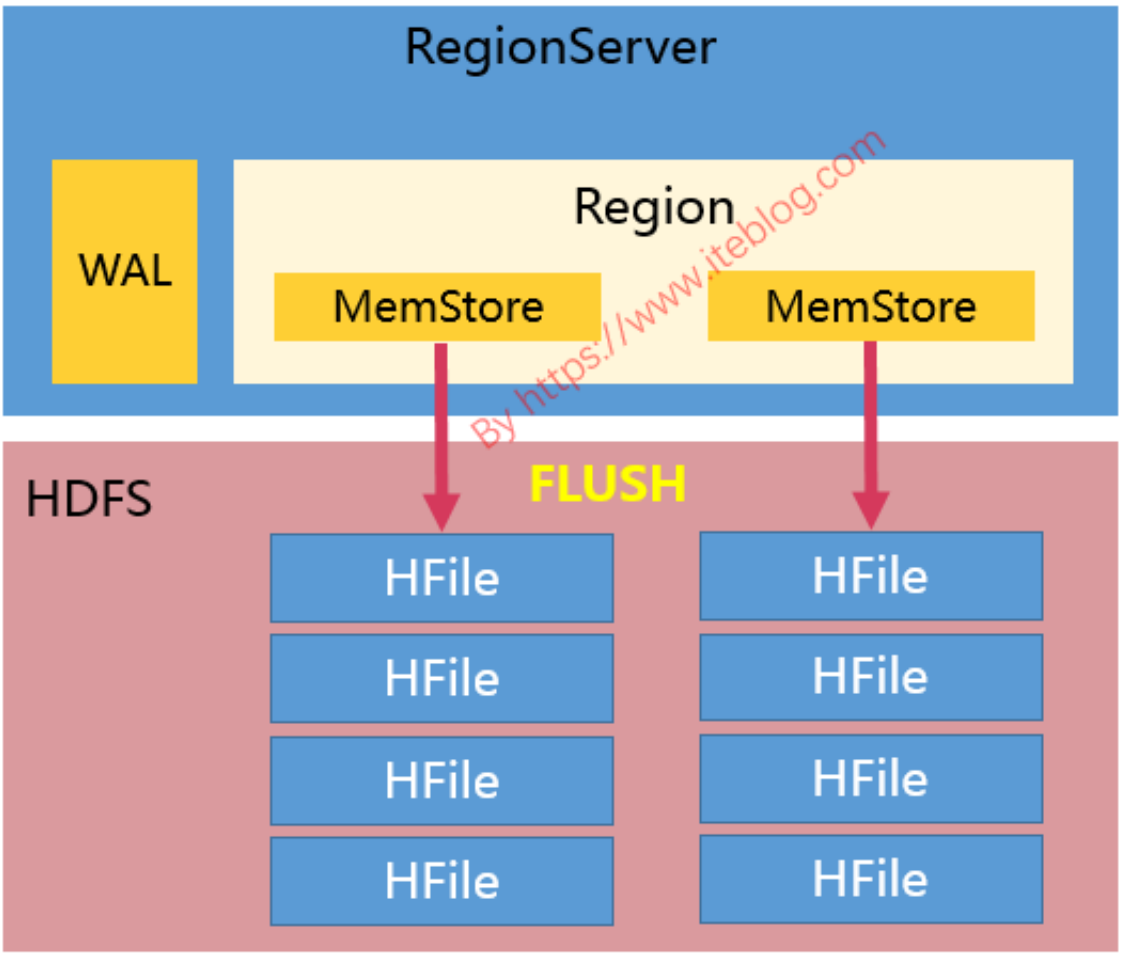
列族数对 Flush 的影响

在 HBase 中，调用 API 往对应的表插入数据是会写到 MemStore 的，而 MemStore 是一种内存结构，每个列族对应一个 MemStore（和零个或多个 HFile）。如果我们的表有两个列族，那么相应的 Region 中存在两个 MemStore，如下图：



如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

从上图可以看出，越多的列族，将会导致内存中存在越多的 MemStore；而储存在 MemStore 中的数据在满足一定条件的时候将会进行 Flush 操作；每次 Flush 的时候，每个 MemStore 将在磁盘生产一个 HFile 文件，如下：



如果想及时了
解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

这样会导致越多的列族最终持久化到磁盘的 HFile 越多。更要命的是，当前 Flush 操作是 Region 级别的（当然，从HBase 1.1，HBase 2.0 开始 Flush
已经可以设置成列族级别的了），也就是说， Region 中某个 MemStore 被 Flush，同一个 Region 的其他 MemStore 也会进行 Flush 操作。当表有很多列族，而且列族之间数据不均匀，比如一个列族有100W行，一个列族只有10行，这样会导致持久化到磁盘的文件数很多，同时有很多小文件，而且每次 Flush 操作也涉及到一定的 IO 操作。

为了解决每次 Flush 都对整个 Region 中 MemStore
进行的，HBASE-10201/HBASE-3149引入了对 Flush 策略进行选择的功能（hbase.regionserver.flush.policy），可以仅对超过阈值（hbase.hregion.percolumnfamilyflush.size.lower.bound.min）的 MemStore 进行 Flush 操作。但是如果没有 MemStore 大于这个阈值，还是会对所有的 MemStore 进行 Flush 操作。

此外，如果我们的列族数过多，这可能会导致触发 RegionServer 级别的 Flush
操作；这将会导致落在该
RegionServer上的更新操作被阻塞，而且阻塞时间可能会达到分钟级别。

列族数对 Split 的影响

我们知道，当 HBase 表中某个 Region 过大（比如大于 hbase.hregion.max.filesize 配置的大小。当然，Region 分裂并不是说整个 Region 大小加起来大于 hbase.hregion.max.filesize 就拆分，而是说 Region 中某个最大的 Store/HFile/storeFile 大于 hbase.hregion.max.filesize 才会触发 Region 拆分的 ），会被拆分成两个。如果我们有很多个列族，而这些列族之间的数据量相差悬殊，比如有些列族有 100W 行，而有些列族只有10行，这样在 Region Split 的时候会导致原本数据量很小的 HFile 文件进一步被拆分，从而产生更多的小文件。注意，Region Split 是针对所有的列族进行的，这样做的目的是同一行的数据即使在 Split 后也是存在同一个 Region 的。

列族数对 Compaction 的影响

与 Flush 操作一样，目前 HBase 的 Compaction 操作也是 Region 级别的，过多的列族也会产生不必要的 IO。

列族数对 HDFS 的影响

我们知道，HDFS 其实对一个目录下的文件数有限制的（dfs.namenode.fs-limits.max-directory-items）。如果我们有 N 个列族，M 个 Region，那么我们持久化到 HDFS 至少会产生 N*M 个文件；而每个列族对应底层的 HFile 文件往往不止一个，我们假设为 K 个，那么最终表在 HDFS 目录下的文件数将是 N*M*K，这可能会操作 HDFS 的限制。

列族数对 RegionServer 内存的影响

前面说了，一个列族在 RegionServer 中对应于一个 MemStore。而 HBase 从 0.90.1 版本开始引入了 MSLAB（Memstore-Local Allocation Buffers，参考HBASE-3455 ），这个功能默认是开启的（通过hbase.hregion.memstore.mslab.enabled），这使得每个 MemStore 在内存占用了 2MB （通过hbase.hregion.memstore.mslab.chunksize 配置）的 buffer。如果我们有很多的列族，而且一般一个 RegionServer 上会存在很多个 Region，这么算起来光 MemStore 的缓存就会占用很多的内存。要注意的是，如果没有往 MemStore 里面写数据，那么 MemStore 的 MSLAB 是不占用空间的。

关于列族数设置的建议

在设置列族之前，我们最好想想，有没有必要将不同的列放到不同的列族里面。如果没有必要最好放一个列族。如果真要设置多个列族，但是其中一些列族相对于其他列族数据量相差非常悬殊，比如1000W相比100行，是不是考虑用另外一张表来存储相对小的列族。

本文链接: 【】（ ）