

Cassandra 1.0 改进：数据压缩

Cassandra 1.0 提供了基于ColumnFamily的数据压缩，这也是一个人民群众呼声很高的功能。压缩功能能够有效地减少数据体积，同时也能减少磁盘I/O，特别是对那些读多的应用场景。

压缩有哪些好处

压缩可以有效的减小数据体积，可以在相同的内存和磁盘上存储更多的数据。除此之外，通过只解压指定部分的数据块，Cassandra对从磁盘上读取数据的性能也有提升。

与传统数据库系统不同，传统的数据库系统使用压缩通常会对写性能有负面影响，因为传统方法需要先解压原始数据，个性原始数据，再将个性后的数据进行 压缩存储。而由于Cassandra的个性操作是追加进行，并不需要对原始数据进行操作，从而避免了二次压缩的问题，写性能会比不使用压缩时提高10%左右。

总的来说，Cassandra使用压缩会得到下面一些性能提升：

- 数据大小可能只占原来的一半到四分之一
- 25-35% 的读性能提升
- 5-10% 的写性能提升

什么时候会使用数据压缩

在多行并且字段都差不多的ColumnFamilies进行数据压缩会比较合适，比如一个存储了username, email等用户信息的ColumnFamily。当各行数据中相同的值越多时，压缩效果也就越好。

而相反的，如果对于每一行字段都不太相同的数据，使用压缩的效果就不太好了。

ColumnFamily的压缩配置

当你创建或者修改一个column family时，你可以设置相关的压缩选项，压缩选项包含下面两个：

- `sstable_compression`：这个选项用于配置具体使用的压缩算法，Cassandra支持两种内置的压缩算法：SnappyCompressor 和DeflateCompressor，这两种算法各有千秋，Snappy在压缩和解压速度上更快，而Deflate在压缩比上会比较高。具体选择哪一种 压缩算法，可以根本你自己的应用场景决定。如果是一个读多的应用，建议使用Snappy算法。另外，你也可以在Cassandra提供的接口上开发自己的 压缩算法，只要实现了org.apache.cassandra.io.compress.ICompressor接口就行。
- `chunk_length_kb`：这个选项用于设置压缩块的大小，默认是 64kb，64kb这个默认值的设定是比较合适的，对于项目比较多的行，你可以不用解压整行就能获取到其中的64kb的数据，对于项目比较少的行，虽然设定为64kb可能会使用你需要解压的数据比你需要的数据还多，但是其压缩率也比较可观，可以说这个数字也是在压缩率和解压开销 间权衡后的一个比较合适的数。当然，你可以针对自己的应用场景对这个数进行个性，以使针对你比较典型的读写方式，能够取得相对高的性能。

你可以在创建一个column family的时候就设定其压缩选项，也可以对一个已经存在的column family的压缩选项进行个性。但是这个修改只会对后续数据产生影响，之前保存的SSTable是不会自动重新压缩

的。当然，如果你非要让老数据也进行 重新压缩，你可以通过使用Cassandra提供的nodetool scrub工具来手动进行操作。

下面是一个例子，使用Cassandra的命令行方式创建了一个带压缩的column family：

```
[default@demo] CREATE COLUMN FAMILY users  
WITH key_validation_class=UTF8Type  
AND column_metadata = [  
  {column_name: name, validation_class: UTF8Type}  
  {column_name: email, validation_class: UTF8Type}  
  {column_name: state, validation_class: UTF8Type}  
  {column_name: gender, validation_class: UTF8Type}  
  {column_name: birth_year, validation_class: LongType}  
]  
AND compression_options={sstable_compression:SnappyCompressor, chunk_length_kb:64};
```

总结

在Cassandra 1.0 中，使用压缩以减小数据尺寸，从而提升性能变得非常容易。你也可以在把Cassandra 升级后再按需求调整其块大小。