# **ElasticSearch优化系列七：优化建议**

[](http://www.jianshu.com/u/92a1227beb27)

[jacksu在简书](http://www.jianshu.com/u/92a1227beb27) 关注

2016.09.21 23:23\* 字数 821 阅读 7097评论 0喜欢 10

尽量运行在Sun/Oracle JDK1.7以上环境中，低版本的jdk容易出现莫名的bug，ES性能体现在在分布式计算中，一个节点是不足以测试出其性能，一个生产系统至少在三个节点以上。

ES集群节点规划良好，master、node、client分离开来，data节点关闭http功能。

合理利用内存。

a) JVM内存设置不要超过机器的一半内存，并且不超过32G。（./bin/elasticsearch -Xmx10g -Xms10g或者修改./bin/elasticsearch.in.sh文件：

\*\* 一般分配主机1/4-1/2的内存\*\*

if [ "x$ES\_MIN\_MEM" =

"x" ]; then

ES\_MIN\_MEM=12g

fi

if [ "x$ES\_MAX\_MEM" =

"x" ]; then

ES\_MAX\_MEM=12g

fi

JAVA\_OPTS="$JAVA\_OPTS

-Xms${ES\_MIN\_MEM}"

JAVA\_OPTS="$JAVA\_OPTS

-Xmx${ES\_MAX\_MEM}"

设置每个线程的堆栈大小, ES单线程承载的数据量比较大  
JAVA\_OPTS="$JAVA\_OPTS -Xss128m"

b) 修改swapping参数，内存不够用时才进行swapping（vm.swappiness= 1）  
c) 暂时不要修改GC方法  
d）锁定内存，不让JVM写入swapping，避免降低ES的****性能****  
bootstrap.mlockall: true  
e）缓存类型设置为Soft Reference，只有当内存不够时才会进行回收  
index.cache.field.max\_size: 50000 index.cache.field.expire: 10m index.cache.field.type: soft

4.权衡建索引的性能和检索的时效性，修改以下参数。

“index.translog.flush\_threshold\_ops”:”10000”  
“index.refresh\_interval”:1  
“number\_of\_replicas”: 0

5.倒排词典的索引需要常驻内存，无法GC，需要监控data node上segment  
memory增长趋势。

定期对不再更新的索引做optimize (ES2.0以后更改为force merge api)。这Optimze的实质是对segment file强制做合并，可以节省大量的segment memory

6.根据机器数，磁盘数，索引大小等硬件环境，根据测试结果，设置最优的分片数和备份数，单个分片最好不超过10GB，定期删除不用的索引，做好冷数据的迁移。

7.保守配置内存限制参数，尽量使用doc value存储以减少内存消耗，查询时限制size、from参数。

8.如果不使用\_all字段最好关闭这个属性，否则在创建索引和增大索引大小的时候会使用额外更多的CPU，如果你不受限CPU计算能力可以选择压缩文档的\_source。这实际上就是整行日志，所以开启压缩可以减小索引大小。

9.避免返回大量结果集的搜索与聚合。缺失需要大量拉取数据可以采用scan & scroll api来实现。

10.熟悉各类缓存作用，如field cache, filter cache, indexing cache, bulk queue等等，要设置合理的大小，并且要应该根据最坏的情况来看heap是否够用。

11.必须结合实际应用场景，并对集群使用情况做持续的监控。