设为首页 收藏本站

首页 论坛 | 社区

导读

博客

专题

浏览

阅读推荐

群组 分类搜索

guaren2009 | 🦺 QQ帐号绑定 | 我的 | 设置 | 消息 | 提醒 | 退出

积分: 562 | 用户组: 中级会员

请输



论坛|社区 技术学习(版主发帖区) 大数据学习 Solr|Nutch|Lucene|ElasticSearch Elasticsearch性能优化-大咖实践经验总结

返回列表

[总结型] Elasticsearch性能优化-大咖实践经验总结 [复制链接]



🚺 hyj 发表于 2019-4-17 16:03:12 | 只看该作者 | 只看大图 🕨

问题导读

- 1.集群规划有哪些优化措施?
- 2.磁盘该如何选择?
- 3.内存该如何分配中?
- 4.索引优化有哪些方法?
- 5.数据模型优化包含哪些内容?

0、题记

Elasticsearch性能优化的最终目的: 用户体验爽。

Elasticsearch的爽点就是: 快、准、全!

关于Elasticsearch性能优化,阿里、腾讯、京东、携程、滴滴、58等都有过很多深入的实践总结,都是非常好的参考。本文换一个思路,基于Elasticsearch的爽点,进行性能优化相关探讨。

1、集群规划优化实践

1.1 基于目标数据量规划集群

在业务初期,经常被问到的问题,要几个节点的集群,内存、CPU要多大,要不要SSD? 最主要的考虑点是: 你的目标存储数据量是多大? 可以针对目标数据量反推节点多少。

1.2 要留出容量Buffer

注意: Elasticsearch有三个警戒水位线,磁盘使用率达到85%、90%、95%。

不同警戒水位线会有不同的应急处理策略。

这点,磁盘容量选型中要规划在内。控制在85%之下是合理的。

当然,也可以通过配置做调整。

1.3 ES集群各节点尽量不要和其他业务功能复用一台机器。

除非内存非常大。

举例:普通服务器,安装了ES+Mysql+redis,业务数据量大了之后,势必会出现内存不足等问题。

1.4 磁盘尽量选择SSD

Elasticsearch官方文档肯定推荐SSD,考虑到成本的原因。需要结合业务场景,

如果业务对写入、检索速率有较高的速率要求,建议使用SSD磁盘。

阿里的业务场景,SSD磁盘比机械硬盘的速率提升了5倍。

但要因业务场景而异。

1.5 内存配置要合理

官方建议: 堆内存的大小是官方建议是: Min(32GB, 机器内存大小/2)。

Medcl和wood大叔都有明确说过,不必要设置32/31GB那么大,建议: 热数据设置: 26GB,冷数据: 31GB。

总体内存大小没有具体要求,但肯定是内容越大,检索性能越好。

经验值供参考:每天200GB+增量数据的业务场景,服务器至少要64GB内存。

除了JVM之外的预留内存要充足,否则也会经常OOM。

1.6 CPU核数不要太小

CPU核数是和ESThread pool关联的。和写入、检索性能都有关联。

建议: 16核+。

1.7 超大量级的业务场景,可以考虑跨集群检索

除非业务量级非常大,例如:滴滴、携程的PB+的业务场景,否则基本不太需要跨集群检索。

1.8 集群节点个数无需奇数

请输

1.9 节点类型优化分配

集群节点数: <=3,建议: 所有节点的master: true, data: true。既是主节点也是路由节点。 集群节点数: >3,根据业务场景需要,建议: 逐步独立出Master节点和协调/路由节点。

1.10 建议冷热数据分离

热数据存储SSD和普通历史数据存储机械磁盘,物理上提高检索效率。

2、索引优化实践

Mysql等关系型数据库要分库、分表。Elasticserach的话也要做好充分的考虑。

2.1 设置多少个索引?

建议根据业务场景进行存储。

不同通道类型的数据要分索引存储。举例:知乎采集信息存储到知乎索引;APP采集信息存储到APP索引。

2.2 设置多少分片?

建议根据数据量衡量。

经验值:建议每个分片大小不要超过30GB。

2.3 分片数设置?

建议根据集群节点的个数规模,分片个数建议>=集群节点的个数。

5节点的集群,5个分片就比较合理。

注意:除非reindex操作,分片数是不可以修改的。

2.4副本数设置?

除非你对系统的健壮性有异常高的要求,比如:银行系统。可以考虑2个副本以上。

否则,1个副本足够。

注意: 副本数是可以通过配置随时修改的。

2.5不要再在一个索引下创建多个type

即便你是5.X版本,考虑到未来版本升级等后续的可扩展性。

建议:一个索引对应一个type。6.x默认对应_doc, 5.x你就直接对应type统一为doc。

2.6 按照日期规划索引

随着业务量的增加,单一索引和数据量激增给的矛盾凸显。

按照日期规划索引是必然选择。

好处1: 可以实现历史数据秒删。很对历史索引delete即可。注意: 一个索引的话需要借助delete_by_query+force_merge操作,慢且删除不彻底。

好处2: 便于冷热数据分开管理,检索最近几天的数据,直接物理上指定对应日期的索引,速度快的一逼!

操作参考:模板使用+rollover API使用。

2.7 务必使用别名

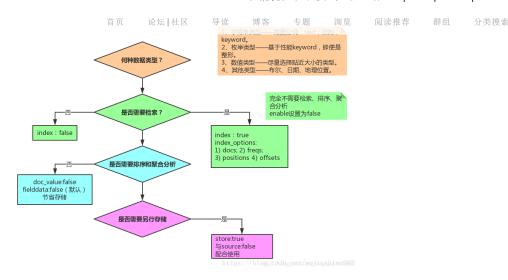
ES不像mysql方面的更改索引名称。使用别名就是一个相对灵活的选择。

3、数据模型优化实践

3.1 不要使用默认的Mapping

默认Mapping的字段类型是系统自动识别的。其中: string类型默认分成: text和keyword两种类型。如果你的业务中不需要分词、检索,仅需要精确匹配,仅设置为keyword即可。 根据业务需要选择合适的类型,有利于节省空间和提升精度,如: 浮点型的选择。

3.2 Mapping各字段的选型流程



3.3 选择合理的分词器

常见的开源中文分词器包括:ik分词器、ansj分词器、hanlp分词器、结巴分词器、海量分词器、"ElasticSearch最全分词器比较及使用方法"搜索可查看对比效果。如果选择ik,建议使用ik_max_word。因为:粗粒度的分词结果基本包含细粒度ik_smart的结果。

3.4 date、long、还是keyword

根据业务需要,如果需要基于时间轴做分析,必须date类型; 如果仅需要秒级返回,建议使用keyword。

4、数据写入优化实践

4.1 要不要秒级响应?

Elasticsearch近实时的本质是:最快1s写入的数据可以被查询到。

如果refresh interval设置为1s,势必会产生大量的segment,检索性能会受到影响。

所以,非实时的场景可以调大,设置为30s,甚至-1。

4.2 减少副本,提升写入性能。

写入前,副本数设置为0,

写入后,副本数设置为原来值。

4.3 能批量就不单条写入

批量接口为bulk,批量的大小要结合队列的大小,而队列大小和线程池大小、机器的cpu核数。

4.4 禁用swap

在Linux系统上,通过运行以下命令临时禁用交换:

[Bash shell] 纯文本查看 复制代码

1 sudo swapoff -a

5、检索聚合优化实战

5.1 禁用 wildcard模糊匹配

数据量级达到TB+甚至更高之后,wildcard在多字段组合的情况下很容易出现卡死,甚至导致集群节点崩溃宕机的情况。

后果不堪设想。

替代方案:

方案一:针对精确度要求高的方案:两套分词器结合,standard和ik结合,使用match_phrase检索。

方案二:针对精确度要求不高的替代方案:建议ik分词,通过match_phrase和slop结合查询。

5.2极小的概率使用match匹配

中文match_match_phrase"。 match_phrase结合合理的分词词典、词库,会使得搜索结果精确度更高,避免噪音数据。

5.3 结合业务场景,大量使用filter过滤器

对于不需要使用计算相关度评分的场景,无疑filter缓存机制会使得检索更快。

举例:过滤某邮编号码。

请输

首页 论坛|社区 导读 博客 专题 浏览 阅读推荐 群组 分类搜索

5.3控制返回字段和结果

和mysql查询一样,业务开发中,select * 操作几乎是不必须的。

同理, ES中, _source 返回全部字段也是非必须的。

要通过_source 控制字段的返回,只返回业务相关的字段。

网页正文content,网页快照html_content类似字段的批量返回,可能就是业务上的设计缺陷。

显然,摘要字段应该提前写入,而不是查询content后再截取处理。

5.4 分页深度查询和遍历

分页查询使用: from+size;

遍历使用: scroll;

并行遍历使用: scroll+slice。

斟酌集合业务选型使用。

5.5 聚合Size的合理设置

聚合结果是不精确的。除非你设置size为2的32次幂-1,否则聚合的结果是取每个分片的Top size元素后综合排序后的值。

实际业务场景要求精确反馈结果的要注意。

尽量不要获取全量聚合结果——从业务层面取TopN聚合结果值是非常合理的。因为的确排序靠后的结果值意义不大。

5.6 聚合分页合理实现

聚合结果展示的时,势必面临聚合后分页的问题,而ES官方基于性能原因不支持聚合后分页。

如果需要聚合后分页,需要自开发实现。包含但不限于:

方案一:每次取聚合结果,拿到内存中分页返回。

方案二: scroll结合scroll after集合redis实现。

6、业务优化

让Elasticsearch做它擅长的事情,很显然,它更擅长基于倒排索引进行搜索。

业务层面,用户想最快速度看到自己想要的结果,中间的"字段处理、格式化、标准化"等一堆操作,用户是不关注的。

为了让Elasticsearch更高效的检索,建议:

1) 要做足"前戏"

字段抽取、倾向性分析、分类/聚类、相关性判定放在写入ES之前的ETL阶段进行;

2)"睡服"产品经理

产品经理基于各种奇葩业务场景可能会提各种无理需求。

作为技术人员,要"通知以情晓之以理",给产品经理讲解明白搜索引擎的原理、Elasticsearch的原理,哪些能做,哪些真的"臣妾做不到"。

7、小结

实际业务开发中,公司一般要求又想马儿不吃草,又想马儿飞快跑。

对于Elasticsearch开发也是,硬件资源不足(cpu、内存、磁盘都爆满)几乎没有办法提升性能的。

除了检索聚合,让Elasticsearch做N多相关、不相干的工作,然后得出结论"Elastic也就那样慢,没有想像的快"。

你脑海中是否也有类似的场景浮现呢?

提供相对NB的硬件资源、做好前期的各种准备工作、让Elasticsearch轻装上阵,相信你的Elasticsearch也会飞起来!

最新经典文章, 欢迎关注公众号



作者: 铭毅天下(公众号同名)

原文: https://blog.csdn.net/laoyang360/article/details/85109769

分享到: 🚇 QQ好友和群













请输

首页 论坛 | 社区 导读 博客 专题 浏览 阅读推荐 群组 分类搜索

请输

欢迎加入**about**云群**425860289、432264021**,云计算爱好者群,关注**about**云腾讯认证空间

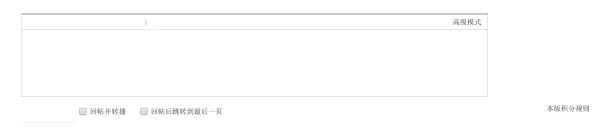


回复 使用道具 举报

还有一些帖子被系统自动隐藏,点此展开

返回列表





Powered by **Discuz!** X3.4 Licensed © 2018 Comsenz Inc.Designed by u179

<u>命 在线容询</u> | 小黑屋 | **about**云开发-学问论坛 | 社区(京ICP备12023829号) GMT+8, 2020-1-7 17:44 , Processed in 1.109375 second(s), 30 queries , Gzip On.