Easysearch 请求限速漫谈

杨帆 极限科技

2025-01-17

分享嘉宾介绍

杨帆，极限科技的售前架构师。主要服务国内私有化部署的 ES 客户，银行、保险、制造集团等。 提供 ES 国产化替代，ES 服务治理的服务和解决方案，ES 容灾、请求分析与管控、限流限速等等。

INFINI Labs

Agenda

1. 为什么要限速
2. 以往经验
3. 限速介绍

为啥要限速

问题框架

想加节点？
先申请资源

申请资源？
走流程申请、
审批

集群受资源限制、操作流程限制，规模比较**固定**，需要保障集群的稳定和服务可用性（重要业务）。

弹性架构：根据负载情况自动扩缩容。

Target

欢迎发表意见
一起探讨

保障服务的稳定、可用

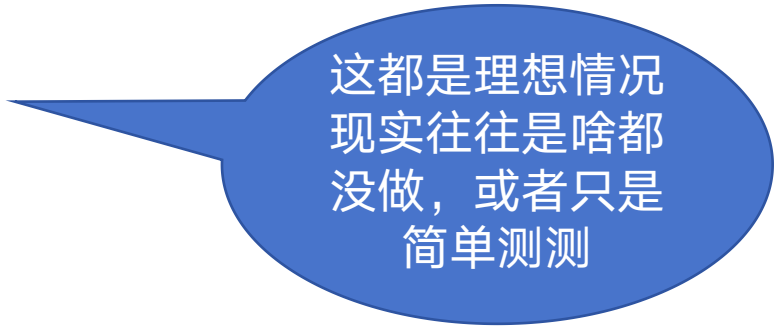
避免进程OOM

是管理措施

应该限多少

服务倒推：应用服务的 “要求”
比如： 查询 2s 内响应
写入：30s 写入实时数据

系统测试：性能测试
不同读并发及其响应时间曲线图
不同写并发及其响应时间曲线图
不同读写并发及其响应时间曲线图



经验之谈

读数据(查询)导致OOM的情况

- qps 太大 （query per second）
- 查询语句太复杂，聚合查询、嵌套聚合
- 查询语句范围太广，range？size？模糊匹配？
- 误操作/好奇，kibana 里数据分析，机器学习分析功能

查询是应用发送的，不可控，只能找应用限流。

a. 业务来不及操作或应用没有限流功能，oom风险大增

提出需求：能不能对查询做管控？

经验之谈

写数据也会导致OOM

- 有时候是在导数据
- 大量数据写入场景
  - 如 日志场景，日增数据量大，为了尽快进数

提出需求：能不能对写入数据限速

经验之谈

限速难点

- 架构：分布式，应用连接多个节点
- 并发设计：并发控制方式--乐观锁

解决方案

- 外层网关：所有流量走网关，便于控制
- 引擎层实现：原来的用法不变，底层通过代码实现限速功能

读请求限速

- qps 太大 （query per second）

应用限流

INFINI Gateway 开源产品 Repo：https://github.com/infinilabs/gateway

- request_path_limiter 定义请求的限速规则，可以实现索引级别的限速。

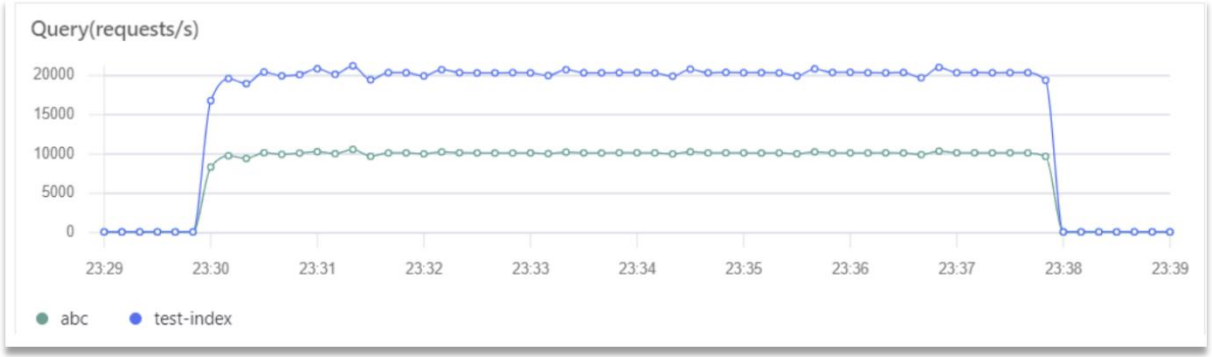
参数说明

名称	类型	说明
message	string	设置达到限速条件的请求的返回消息

INFINI Labs

读请求限速

```
- name: default_flow
  filter:
    - request_path_limiter:
        message: "Hey, You just reached our request limit!"
        rules:
          - pattern: "/(?P<index_name>abc)/_search"
            max_qps: 10000
            group: index_name
          - pattern: "/(?P<index_name>test-index)/_search"
            max_qps: 20000
            group: index_name
    - elasticsearch:
        elasticsearch: prod
        max_connection_per_node: 1000
```



INFINI Labs

读请求过滤

- 查询语句太复杂，嵌套聚合
- 查询语句范围太广，range？size？模糊匹配？
- 误操作/好奇，kibana 里数据分析，机器学习分析功能

INFINI Gateway 开源产品

- context_filter 过滤器用来按请求上下文来过滤流量。
- request_body_json_del 用来删除 JSON 格式的请求体里面的部分字段。
- request_body_json_set 修改 JSON 格式的请求体。
- request_body_regex_replace 使用正则表达式来替换请求体正文的字符串内容。

读请求过滤

内置请求上下文

HTTP 请求内置的 `_ctx` 上下文对象主要包括如下：

名称	类型	说明
id	uint64	请求的唯一 ID
tls	bool	表示请求是否 TLS
remote_ip	string	客户端来源 IP
remote_addr	string	客户端来源地址，包含端口
local_ip	string	网关本地 IP
local_addr	string	网关本地地址，包含端口
elapsed	int64	请求已执行时间（毫秒）
request.*	object	描述请求信息
response.*	object	描述响应信息

```
- name: main_flow
  filter:
    - context_filter:
        context: _ctx.request.to_string
        action: redirect_flow
        status: 403
        flow: log4j_matched_flow
        must_not: # any match will be filtered
          regex:
            - \$\{.*?\}
            - "%24%7B.*?%7D" #urlencode
          contain:
            - "jndi:"
            - "jndi:ldap:"
            - "jndi:rmi:"
            - "jndi%3A" #urlencode
            - "jndi%3Aldap%3A" #urlencode
            - "jndi%3Armi%3A" #urlencode
    - elasticsearch:
        elasticsearch: es-server
- name: log4j_matched_flow
  filter:
    - echo:
        message: 'Apache Log4j 2, Boom!'
```

INFINI Labs

读请求过滤

INFINI Ga

- contex

```
[infini@kylin ~]$ curl  -u admin:1ef0c661d8562aaa06be -XGET http://localhost:8000/yf-test-1shard/_search
{
  "size": 10,
  "query": {
    "wildcard": {
   "path": {
        "value": "a*"
      }
}
  }
}
'
Request blocked. Reason: Forbidden. Please contact the administrator.010-111111[infini@kylin ~]$
```

```
flow:
  - name: forbidden
    filter:
      - dump:
          request: true
          response: true
      - echo:
          message: 'Request blocked. Reason: Forbidden. Please contact the administrator.010-111111'
  - name: default_flow
    filter:
      - context_filter:
          context: _ctx.request.to_string
          action: redirect_flow
          status: 403
          flow: forbidden
          must_not:
            contain:
              - "wildcard"
```

读请求改写

INFINI Gateway 开源产品

- request_body_json_del 用来删除 JSON 格式的请求体里面的部分字段。

```
flow:
  - name: test
    filter:
      - request_body_json_del:
          path:
          - query.bool.should.[0]
          - query.bool.must
```

INFINI Labs

读请求改写

INFINI Gateway 开源产品

- request_body_json_set 修改 JSON 格式的请求体。

```
flow:
  - name: test
    filter:
      - request_body_json_set:
          path:
          - aggs.total_num.terms.field -> "name"
          - aggs.total_num.terms.size -> 3
          - size -> 0
```

INFINI Labs

读请求改写

INFINI Gateway 开源产品

- request_body_regex_replace 使用正则表达式来替换请求体正文的字符串内容。

```
flow:
  - name: test
    filter:
      - request_body_regex_replace:
          pattern: '"size": 10000'
          to: '"size": 100'
      - elasticsearch:
          elasticsearch: prod
      - dump:
          request: true
```

写请求限速

- 应用控制写入速度
- Easysearch 引擎限速
- INFINI Gateway 开源产品
  - bulk_request_throttle 限制每秒写入到文档数

优点：
使用简单
能对不同索引限制不同的速度

缺点：
数据在内存中，虽然有重试机制，还是有丢数据的可能
应用要对请求返回值做判断，必要时应用进行重试、回避

写请求限速

```
flow:
  - name: bulk_request_mutate
    filter:
      - bulk_request_throttle:
          indices:
            test:
              max_requests: 5
              action: drop
              message: "test writing too fast。"
              log_warn_message: true
            filebeat-*:
              max_bytes: 512
              action: drop
              message: "filebeat indices writing too fast。"
              log_warn_message: true
```

indices	map	用于限速的索引，可以分别设置限速规则
indices.[NAME].interval	string	评估限速的单位时间间隔，默认为 `1s`
indices.[NAME].max_requests	int	单位间隔内最大的请求次数限额
indices.[NAME].burst_requests	int	单位间隔内极限允许的请求次数
indices.[NAME].max_bytes	int	单位间隔内最大的请求流量限额
indices.[NAME].burst_bytes	int	单位间隔内极限允许的流量限额
indices.[NAME].action	string	触发限速之后的处理动作，分为 `retry` 和 `drop` 两种，默认为 `retry`
indices.[NAME].status	string	设置达到限速条件的返回状态码，默认 `429`
indices.[NAME].message	string	设置达到限速条件的请求的拒绝返回消息
indices.[NAME].retry_delay_in_ms	int	限速重试的时间间隔，单位毫秒，默认 `10`，即 10 毫秒
indices.[NAME].max_retry_times	int	限速重试的最大重试次数，默认 `1000`
indices.[NAME].failed_retry_message	string	设置达到最大重试次数的请求的拒绝返回消息
indices.[NAME].log_warn_message	bool	是否输出警告消息到日志

写请求限速

```
env: #use $[[env.LOGGING_ES_ENDPOINT]] in config instead
  LOGGING_ES_ENDPOINT: http://localhost:9200/
  LOGGING_ES_USER: admin
  LOGGING_ES_PASS: 1ef0c661d8562aaa06be
  PROD_ES_ENDPOINT: http://localhost:9200/
  PROD_ES_USER: admin
  PROD_ES_PASS: 1ef0c661d8562aaa06be
  GW_BINDING: "0.0.0.0:8000"
  API_BINDING: "0.0.0.0:2900"
  THROTTLE_BULK_INDEXING_MAX_BYTES: 40485760 #40MB/s
  THROTTLE_BULK_INDEXING_MAX_REQUESTS: 2000 #10k docs/s
  THROTTLE_BULK_INDEXING_ACTION: retry #retry,drop
  THROTTLE_BULK_INDEXING_MAX_RETRY_TIMES: 10 #1000
  THROTTLE_BULK_INDEXING_RETRY_DELAY_IN_MS: 100 #10
```

```
- name: bulking_indexing_limit
  filter:
    - bulk_request_throttle:
        indices:
          "yf-test":
            max_bytes: $[[env.THROTTLE_BULK_INDEXING_MAX_BYTES]]
            max_requests: $[[env.THROTTLE_BULK_INDEXING_MAX_REQUESTS]]
            action: $[[env.THROTTLE_BULK_INDEXING_ACTION]]
            retry_delay_in_ms: $[[env.THROTTLE_BULK_INDEXING_RETRY_DELAY_IN_MS]]
            max_retry_times: $[[env.THROTTLE_BULK_INDEXING_MAX_RETRY_TIMES]]
            message: "bulk writing too fast" #触发限流告警message自定义
            log_warn_message: true
```

写请求限速

Easysearch 1.8.0 新增了限速功能，支持节点、索引、分片 三种不同的限速方法

- 节点限速：限制整个数据节点的全量写入速度，不论哪个索引、分片，全部管控。
- 索引限速：限制特定索引的写入速度。
- 分片限速：限制任何单个分片的写入速度，不论哪个索引。

写请求限速

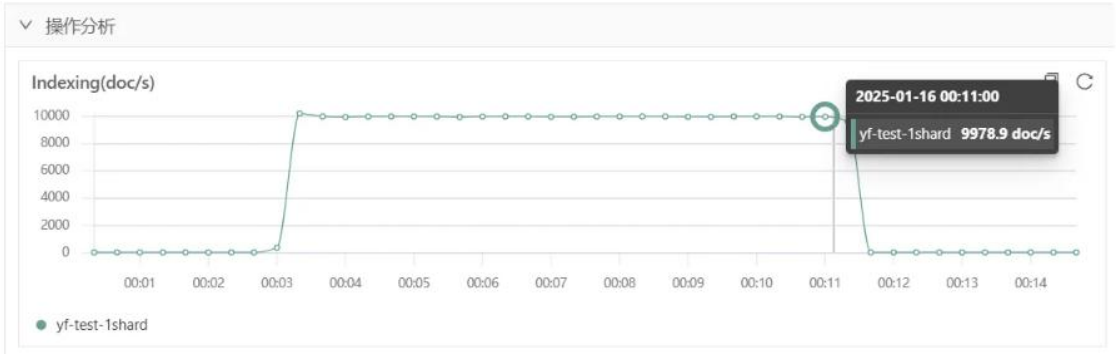
节点限速：限制整个数据节点的全量写入速度，不论哪个索引、分片，全部管控。

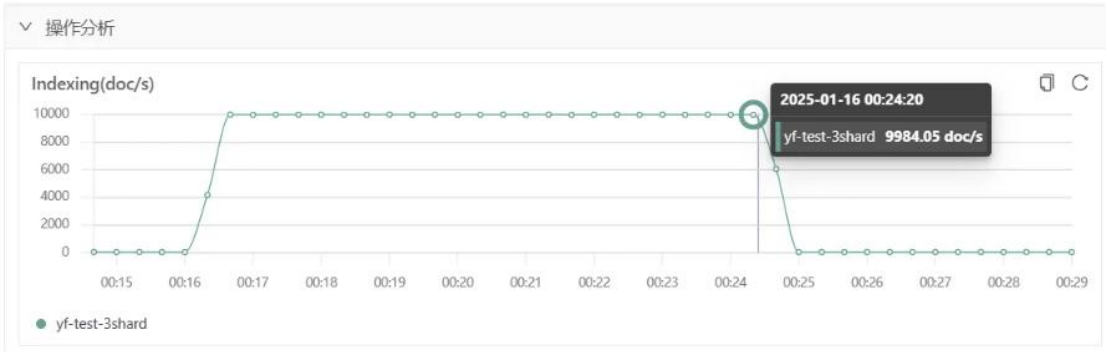
- 支持动态调整
- action 支持 retry 和 drop

```
PUT _cluster/settings
{
  "transient": {
    "cluster.throttle.node.write": true
    "cluster.throttle.node.write.max_requests": 10000,
    "cluster.throttle.node.write.action": "retry"
  }
}
```

写请求限速

节点限速：限制整个数据节点的全量写入速度，不论哪个索引、分片，全部管控。





场景：

压测过单节点的极限写入性能，为了保障集群的稳定，设置一个比极限值低一些的阈值进行保护。

写请求限速

索引限速：限制特定索引的写入速度。

在索引的 settings 里进行设置，动态的

```
PUT yf-test-3shard/_settings
{
    "index.throttle.write.max_requests": 2000,
    "index.throttle.write.action": "retry",
    "index.throttle.write.enable": true
}
```

场景：

针对特定索引进行写入限速，避免响其他索引的读写。

写请求限速

分片限速：限制任何单个分片的写入速度，不论哪个索引。

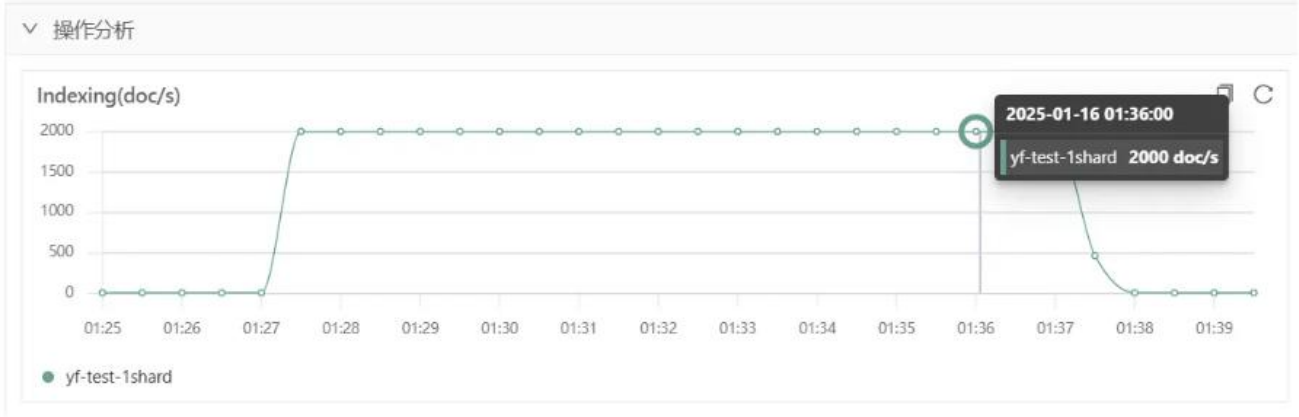
- 支持动态调整

```
PUT _cluster/settings
{
  "transient": {
    "cluster.throttle.shard.write": true,
    "cluster.throttle.shard.write.max_requests": 2000,
    "cluster.throttle.shard.write.action": "retry"
  }
}
```

写请求限速

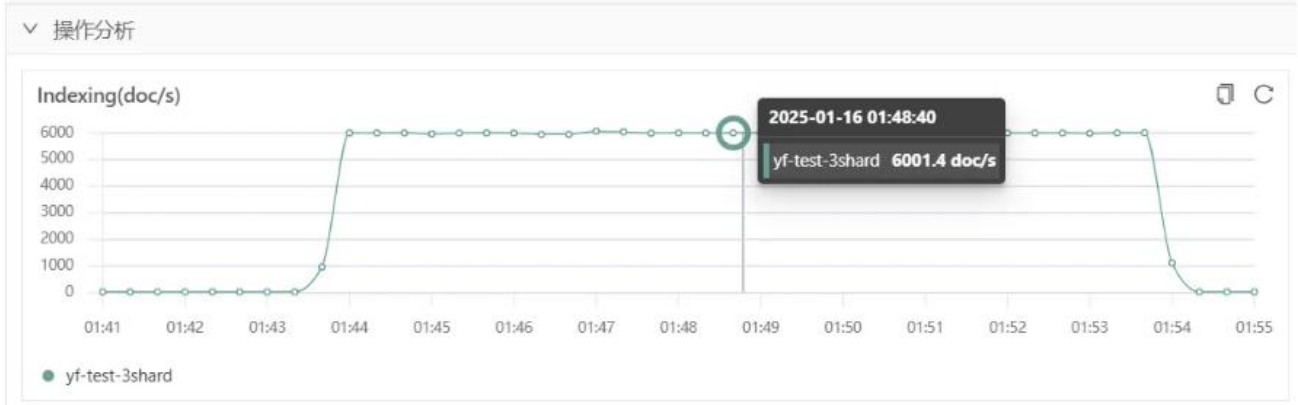
压测单个索引，1 主，0 副

1 个分片，写入速度 2000 个文档每秒。



压测单个索引，3 主，0 副

3 个分片，写入速度 6000 个文档每秒。



场景：

一个集群中有高低配置混搭主机的场景。

- 高配机器性能强悍，磁盘空间也大，分布的分片也多；
- 低配主机性能和磁盘容量都有限，分布的分片数较少；

INFINI Labs

欢迎加V交流

杨 帆

极限科技 | 解决方案架构师

yangf@infinilabs.com | www.infinilabs.com |

INFINI Labs

INFINI Labs

极限科技·让搜索更简单

THANK YOU FOR WATCHING

感谢聆听

INFINI Labs

搜索客 SearchKit

搜索人自己的社区

专业、垂直、纯粹的开源搜索技术交流社区

https://searchkit.cn