**分片策略文档**

V1.0.0

吴志彬 @ Enlink 数据中心

**目录**

[1. 概述 3](#_Toc13641)

[1.1. 说明 3](#_Toc15902)

[2. Cluster级别分片策略 4](#_Toc25585)

[2.1. 分片设置 4](#_Toc30449)

[2.2. 分片重新均衡配置 4](#_Toc8247)

[2.3. 指定何时允许分片重新平衡 5](#_Toc28287)

[2.4. 基于磁盘的分片分配 6](#_Toc4746)

[3. 分片过滤 8](#_Toc29232)

# 概述

本文档主要是对分片策略进行描述与使用。

## 说明

一个分片代表一个搜索引擎，分片是ES的基本组成，虽然不能改变主分片的数量，但是我们可以控制分片的位置。更高的在数据存储倾斜后很好的控制磁盘，防止意外。

或者在需要更换节点后，将需要更换的节点上的分片都移出去，待新节点加进来后再进行分片平衡

# Cluster级别分片策略

分片分配是将分片分配给节点的过程，这可能发生在初始恢复，副本分配，重新平衡或添加或删除节点期间

## 分片设置

启用或禁用分片分配

允许为各种分片分配分片（默认）,索引的主分片和副本分片会均分到各个节点

|  |
| --- |
| cluster.routing.allocation.enable:all |

仅允许分配主分片，主分片会均分到各个节点，副本分片处于unassigined状态

|  |
| --- |
| cluster.routing.allocation.enable:primaries |

仅允许为新索引分配主分片，新增的索引的主分片会均分到各节点，副本分片处于unassigined状态，之前的索引则不会影响

|  |
| --- |
| cluster.routing.allocation.enable:new\_primaries |

不允许任何索引的分片，新增的索引不论主分片还是副本分片都处于unassigined状态

|  |
| --- |
| cluster.routing.allocation.enable:none |

示例：

临时设置分片设置

表 2-1临时设置分片设置

|  |
| --- |
| PUT \_cluster/settings  {  "transient": {  "cluster.routing.allocation.enable": "primaries"  }  } |

## 分片重新均衡配置

启用或禁用分片重新均衡配置

允许各种分片的分片平衡（默认）

|  |
| --- |
| cluster.routing.rebalance.enable:all |

仅允许主分片的分片平衡

|  |
| --- |
| cluster.routing.rebalance.enable:primaries |

仅允许副本分片进行分片平衡

|  |
| --- |
| cluster.routing.rebalance.enable:replicas |

任何分片都不能进行分片平衡

|  |
| --- |
| cluster.routing.rebalance.enable:none |

解释：

该参数主要含义是：当我们在cluster中新增node时，新增的node是否均衡索引的分片。比如10个分片，5主5副，当前2个节点已经均分了，当新增第三个节点时，如cluster.routing.rebalance.enable设置为none，则新增的节点不会被分配到分片

表 2-2 分片重新均衡分配

|  |
| --- |
| PUT \_cluster/settings  {  "transient": {  "cluster.routing.rebalance.enable":"replicas"  }  } |

## 指定何时允许分片重新平衡

始终允许重新平衡

|  |
| --- |
| cluster.routing.allocation.allow\_rebalance:always |

仅在集群中的所有主分片分配完成后

|  |
| --- |
| cluster.routing.allocation.allow\_rebalance:indices\_primaries\_active |

仅当分配了集群中所有分片（主分片和副本分片）（默认）

|  |
| --- |
| cluster.routing.allocation.allow\_rebalance:indices\_all\_active |

默认的意味着等到所有分片均处于可用状态时，才会进行重新平衡分配操作，可以减少集群初始启动时机器之间的交互

表 2-3 指定何时允许重新分片

|  |
| --- |
| PUT \_cluster/settings  {  "transient": {  "cluster.routing.allocation.allow\_rebalance": "always"  }  } |

## 基于磁盘的分片分配

在确定是将新分片分配给该节点还是主动从该节点重新将分片分配出去之前，ElasticSearch会检查该节点的可用磁盘空间

不能在这些设置里面混合使用百分比值和字节值，要么全部设置为百分比，要么全部设置为字节值，这样我们就可以验证设置是否内部一致（即低磁盘阀值不超过高磁盘阀值，高阀值不超过最高阀值

禁用磁盘分配决定器(默认为true)

|  |
| --- |
| cluster.routing.allocation.disk.threshold\_enabled:false |

设置磁盘最低阀值,默认85%。此设置不会影响新创建索引的分片，或者之前未分配过的分片

|  |
| --- |
| cluster.routing.allocation.disk.watermark.low:85% |

设置磁盘高阀值，默认90%。当容量达到90%时，ES会考虑重新定位分片。会影响所有分片的分配，无论之前是否分配过

|  |
| --- |
| cluster.routing.allocation.disk.watermark.high:90% |

设置磁盘最高阀值，默认95%。当节点次磁盘容量达到95%时，索引会被置为只读模式，不允许继续写数据。可以在空间释放后，解除只读模式

|  |
| --- |
| cluster.routing.allocation.disk.watermark.flood\_stage:95% |

禁用磁盘分配决定器

表 2-4 禁用磁盘分配决定器

|  |
| --- |
| PUT \_cluster/settings  {  "transient": {  "cluster.routing.allocation.disk.threshold\_enabled": false  }  } |

解除只读模式

表 2-5 解除只读模式

|  |
| --- |
| PUT \_settings  {  "index.blocks.read\_only\_allow\_delete": null  } |

修改ES多久检测一次磁盘使用情况，默认30秒。该项设置也可以用来避免节点频繁重启而引发的分片移动，造成磁盘IO 过高，以至于节点丢失的问题

表 2-6　修改磁盘检测时间

|  |
| --- |
| PUT \_cluster/settings/  {  "transient": {  "cluster.info.update.interval": "5m"  }  } |

关于分片移动时，ES会在计算节点的磁盘使用情况时考虑当前正在重定位到目标节点的分片，但是，将重定位分片的大小考虑在内可能意味着节点的磁盘使用量偏差偏高，因为重定位可能超过90%，最近检索的磁盘使用量包括将重定位分片的大小以及正在运行的使用的空间。

所以，先关闭检测，待分片移动结束后在开启检测

表 2-7 关闭磁盘用量检测

|  |
| --- |
| PUT \_cluster/settings/  {  "transient": {  "cluster.routing.allocation.disk.include\_relocations": false/true (关闭/开启)  }  } |

# 分片过滤

虽然索引分片分配提供了每个索引设置来控制分片到节点的分配，但是集群级别分片分配过滤允许或禁止分片从任何索引到节点的分配

会影响到现有索引和新建的索引

使用动态设置：

1. cluster.routing.allocation.include.{attr}:将分片分配给{attr}至少具有一个逗号分隔的节点

2. cluster.routing.allocation.required.{attr}:将分片分配给{attr}具有所有逗号分隔的节点

3. cluster.routing.allocation.exclude.{attr}:不将分片分配给{attr}具有所有逗号的节点

支持的属性：

1. \_name:节点名称

2. \_ip：IP地址

3. \_host：主机名称

排除某个节点，IP支持使用\*通配符。如192.168.225.\*

|  |
| --- |
| PUT \_cluster/settings  {  "transient" : {  "cluster.routing.allocation.exclude.\_ip" : "192.168.225.135,192.168.225.136"  }  } |

允许某个节点

|  |
| --- |
| PUT \_cluster/settings  {  "transient" : {  "cluster.routing.allocation.include.\_ip" : "192.168.225.137"  }  } |

测试下来，exculde的优先级更高