字节跳动的ES智能运维和数据管理中台演进

高英举
12/2019, 广告数据平台, 字节跳动

Agenda

- ES运维和用户服务的痛点
- Datapalace 数据管理中台介绍
- ES智能诊断系统介绍
- 数据管理中台和智能诊断系统的设计与实现
- 未来计划

ES运维和用户服务的痛点

特定(集群、数据)规模有特定问题

elastic

ES运维和用户服务的痛点

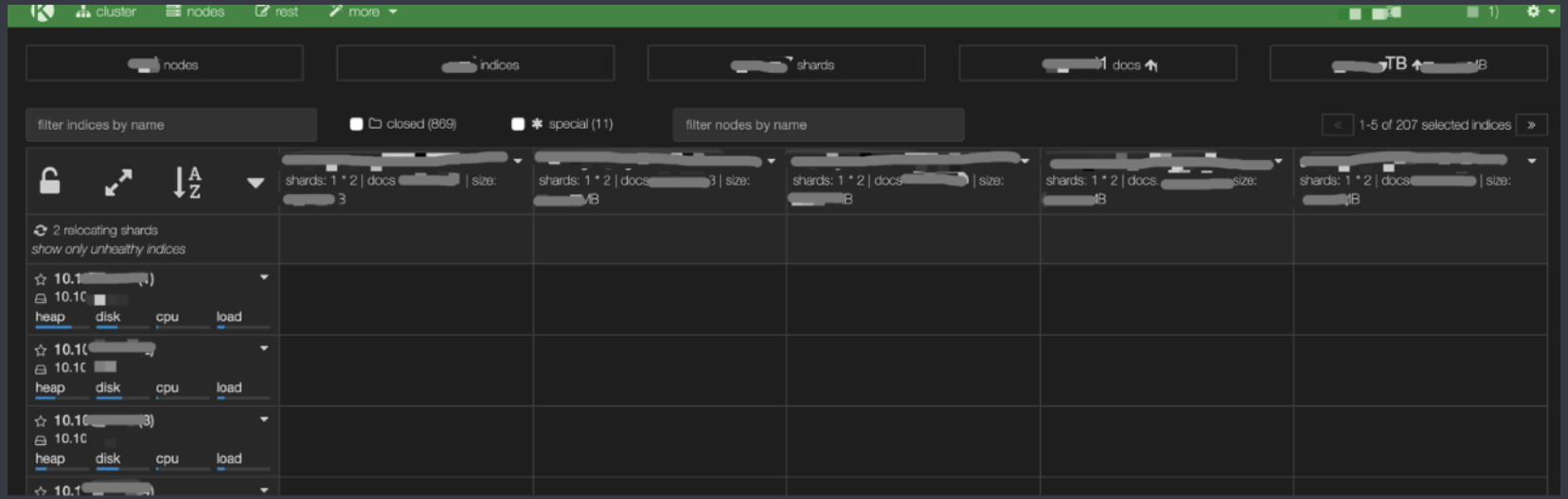
痛点列举

- 开源集群、索引管理工具
  - 缺少对公司业务的感知和数据治理能力
  - 缺乏与公司基础设施的联动（如部署、监控、报警）
- ES集群、索引的运维
  - 操作的安全性待优化（加个密码访问就够吗？）
  - 缺乏标准化手段，流程不健全，运维工程师不自觉
- 线上问题排查
  - 较多依赖手动操作和人工经验判断
  - 解决问题的最佳实践没有固化

总结：特定(集群、数据)规模有特定问题，不一定适用所有业务场景

ES运维和用户服务的痛点

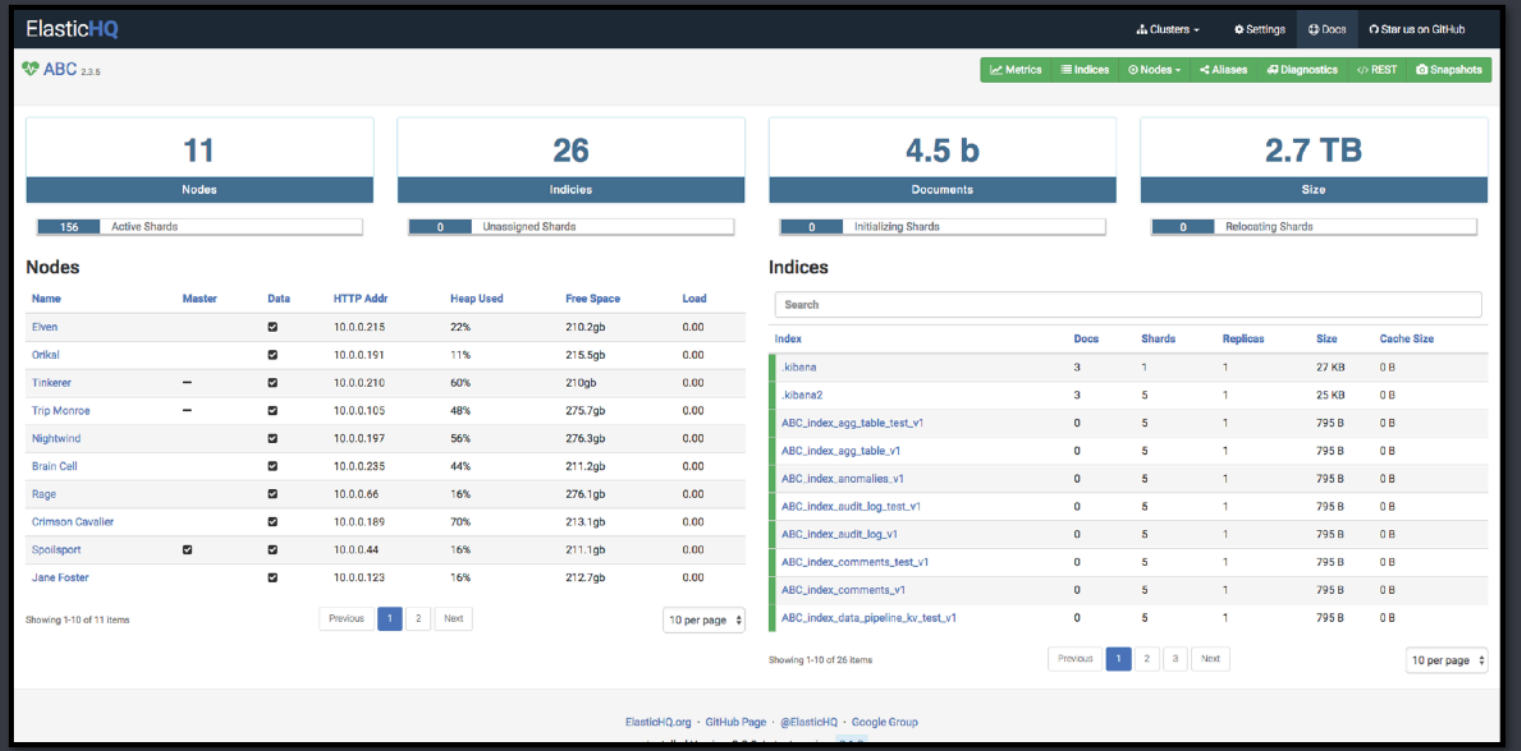
Kopf(Cerebro): https://github.com/lmenezes/cerebro



elastic

ES运维和用户服务的痛点

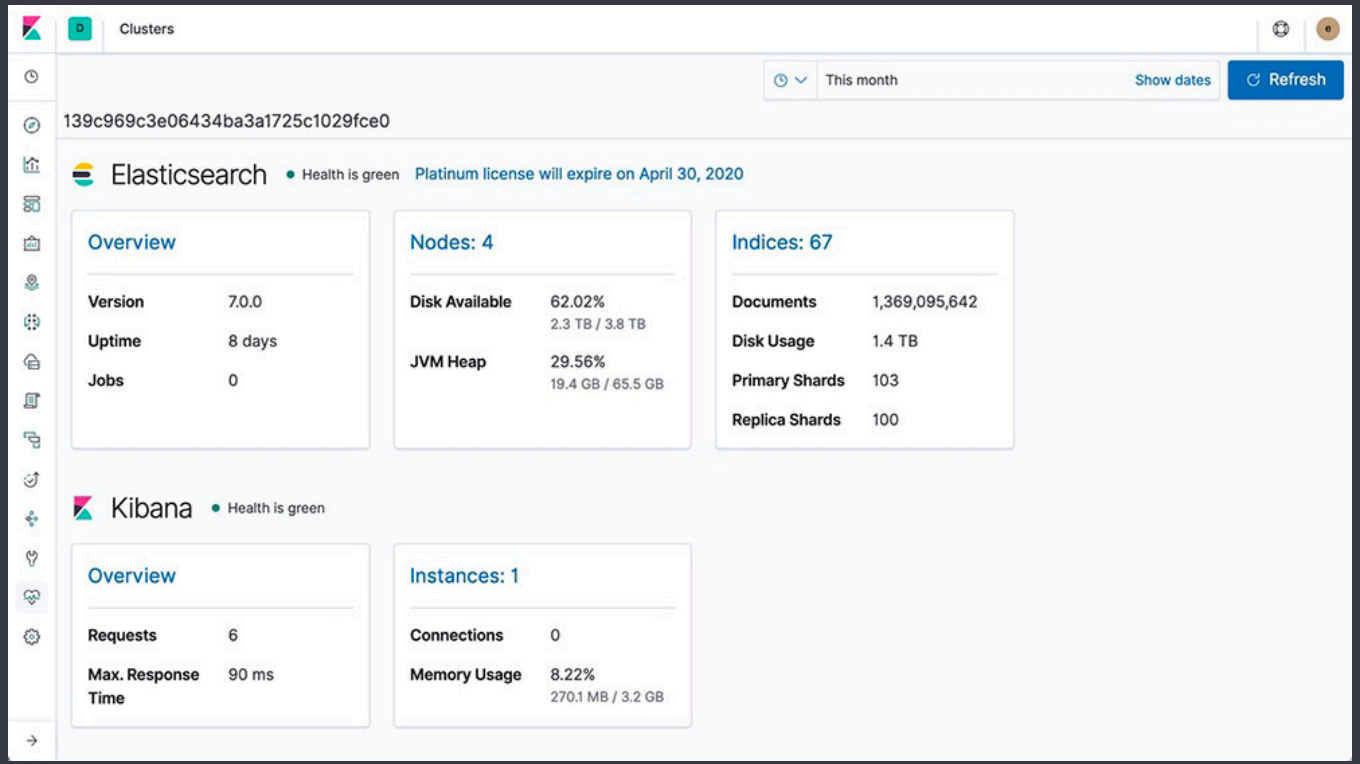
ElasticHQ: https://github.com/ElasticHQ/elasticsearch-HQ



图片来源：非字节内部图片

ES运维和用户服务的痛点

Kibana Full Stack Monitoring



图片来源：非字节内部图片

elastic

Datapalace 数据管理中台

公司级一站式ES集群、索引托管和数据治理平台

elastic

Datapalace 数据管理中台

功能介绍

1. ES集群功能：

- 创建集群
- 更新集群元数据
- 申请集群运维变更
- 查看集群数据链路
- 查看集群存储容量统计
- 查看节点加入退出记录
- 查看Master切换记录
- 查看集群的诊断信息
- 下线、删除集群
- 自动绑定计费服务树

2. ES集群组功能：

- 关联集群组
- 集群组扩容，缩容
- 集群组上索引的操作

3. ES索引功能：

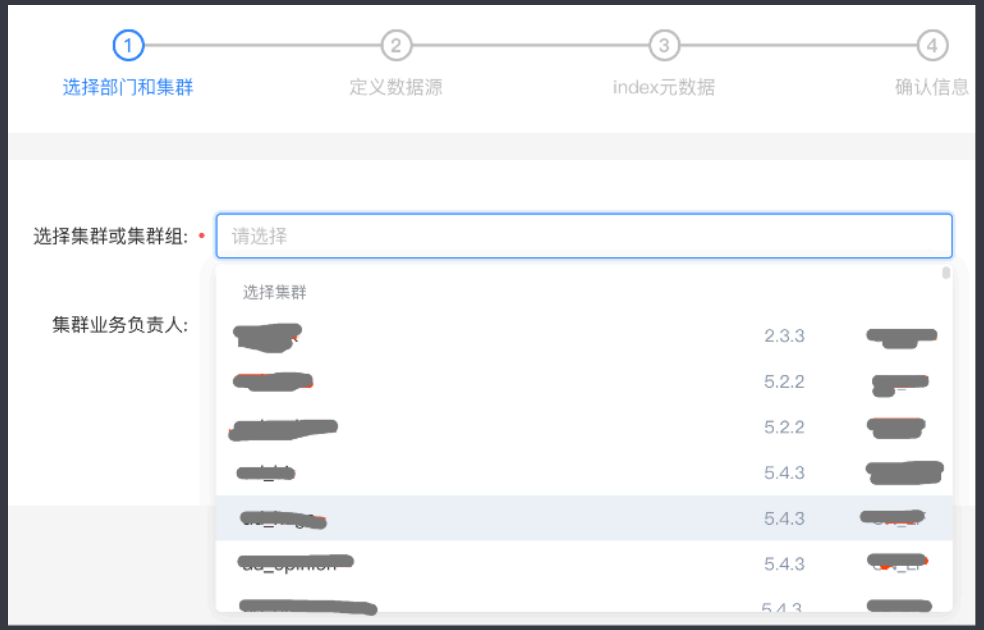
- 接入索引[集群、集群组]
- 索引数据读写的交互式文档
- 更新索引元数据
- 更新索引settings（支持历史索引的更新）
- 更新索引mappings（支持历史索引的更新）
- 历史索引的查看、打开、关闭、删除
- 使用别名管理索引，别名切换
- 查看索引存储容量统计
- 查看索引数据链路
- 查看索引的诊断信息
- 下线、删除索引（报警联动）

4. 数据导入导出

- 自助式数据导入支持：Spark，Flink，Hive2ES，Logstash
- 自助式数据导出支持：Spark，Flink
- 字节云，物理机自动部署 Logstash
- 自动绑定计费服务树

Datapalace 数据管理中台

索引接入



索引概念梳理：

- 单集群索引(组)
- 集群组索引(组)
- 索引(组)的物理索引

elastic

Datapalace 数据管理中台

索引接入

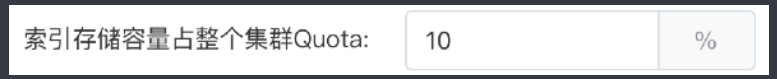
定义数据源的目的:

- 自动建立数据链路
- 与数据导入导出工具联动
- 自动推测索引Mappings

elastic

Datapalace 数据管理中台

索引接入



elastic

Datapalace 数据管理中台

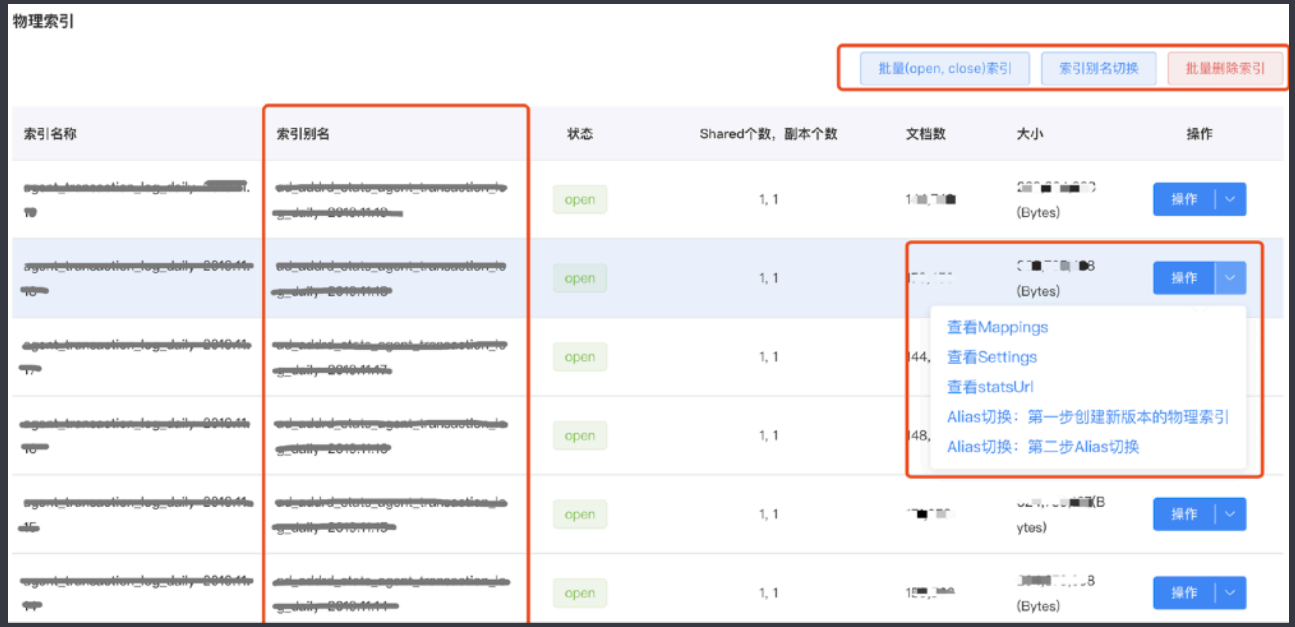
索引接入

业务负责人、运维负责人参与索引接入的审批，确保业务和运维上的合理性

elastic

Datapalace 数据管理中台

物理索引操作

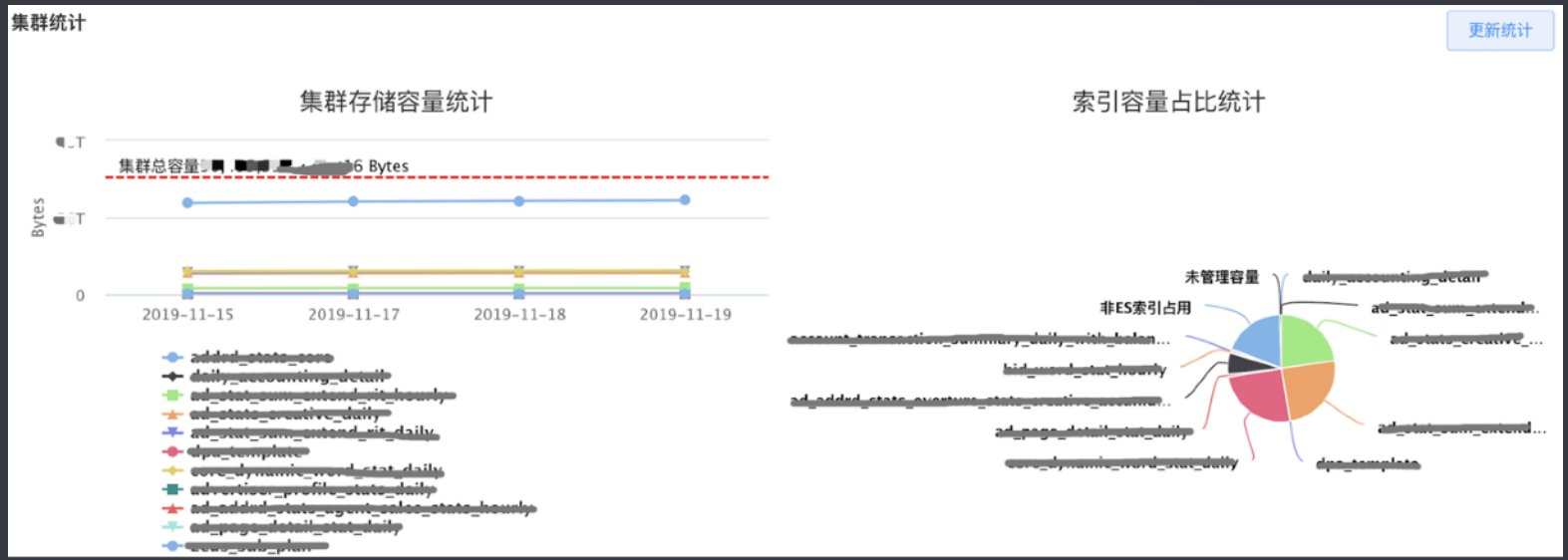


- 批量open，close索引
- 批量删除索引
- 批量切换索引的别名
- 查看索引的别名
- 查看索引的mappings，settings，stats

elastic

Datapalace 数据管理中台

集群、索引容量统计：做好容量规划

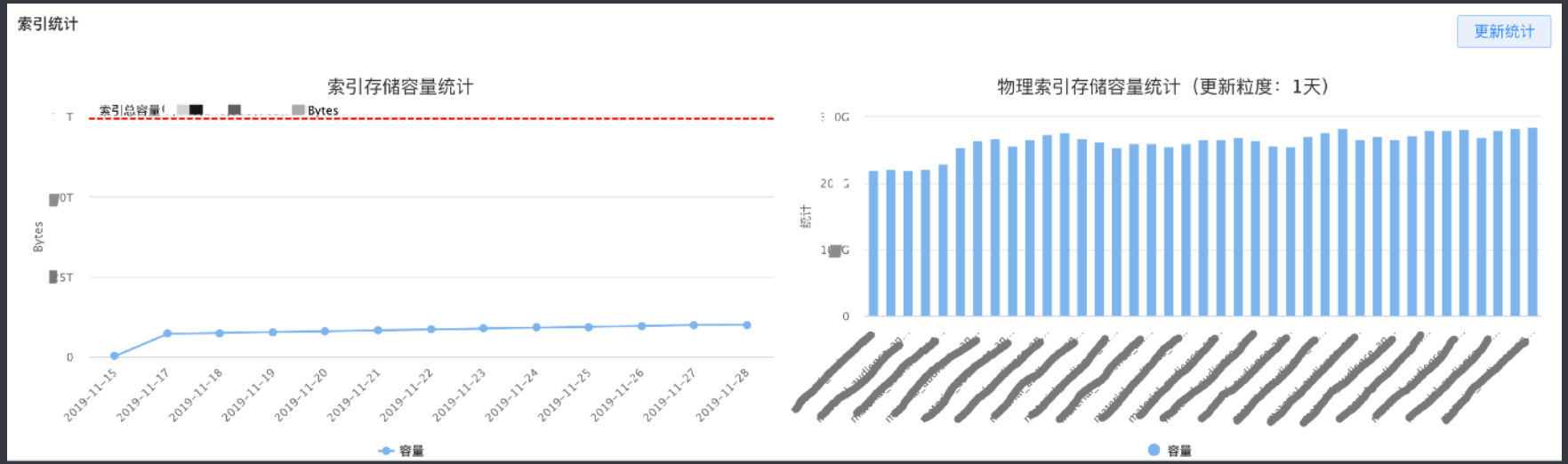


索引(组)为单位的容量统计

elastic

Datapalace 数据管理中台

集群、索引容量统计：做好容量规划

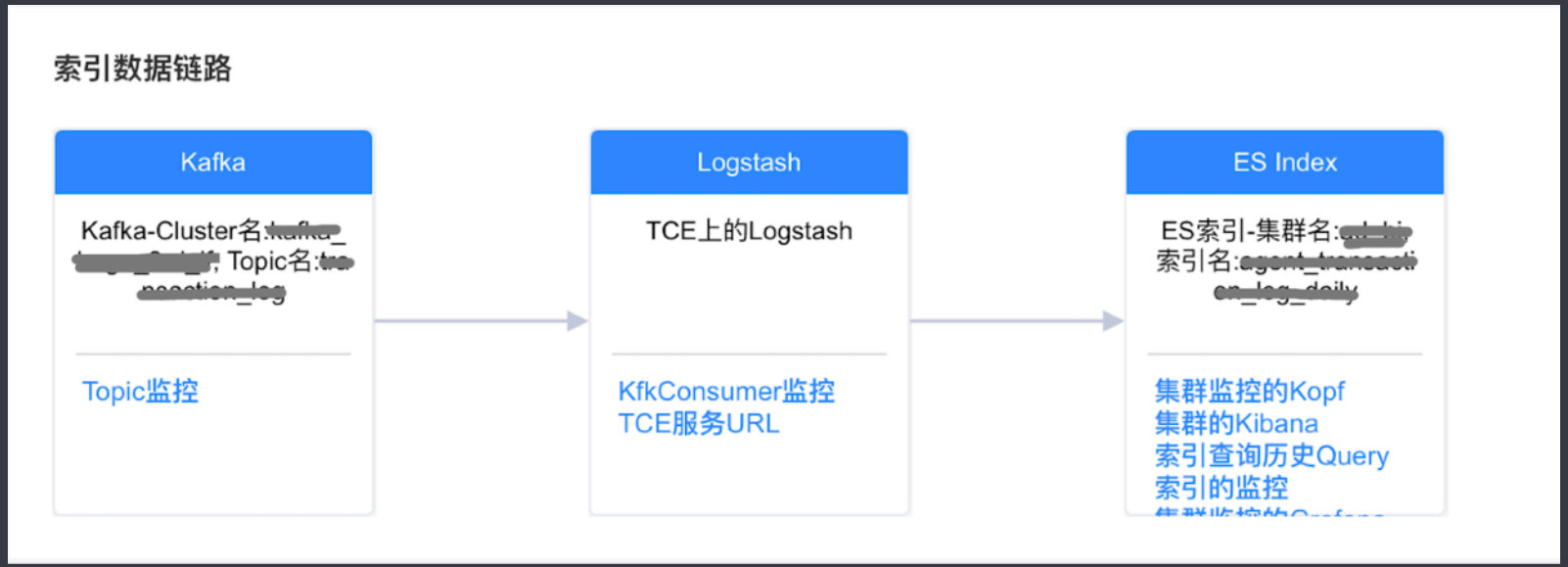


物理索引为单位的统计

elastic

Datapalace 数据管理中台

索引的数据链路：大大简化了业务和运维负责人管理和学习数据的成本



数据链路上的任意节点提供：监控页面入口，报警管理入口

elastic

Datapalace 数据管理中台

索引的数据链路：大大简化了业务和运维负责人管理和学习数据的成本

多机房容灾场景下的数据链路

elastic

Datapalace 数据管理中台

ES集群组：多机房容灾的支持

集群组列表

集群组中的索引

elastic

Datapalace 数据管理中台

ES集群组：多机房容灾的支持

更新集群组索引的Mappings，多个集群中的索引被同步更新

elastic

Datapalace 数据管理中台

集群运维变更工单

把”线上集群运维变更“这件事程序化、标准化，以达到运维效率、人力成本、线上集群稳定性的平衡

elastic

Datapalace 数据管理中台

数据导入导出，多源数据关联分析

数据源：
- Kafka
- Hive
- HDFS
- MySQL
- ES

导入导出工具：
- Dorado[公司平台]
- Waterdrop[开源项目]

数据源：
- Kafka
- Hive
- HDFS
- MySQL
- ES

功能：
- 自助式数据导入支持：Spark，Flink，Logstash
- 自助式数据导出支持：Spark，Flink

数据治理：
- 索引|docs count校验
- 索引|mappings校验
- 数值型字段min, max, avg, count校验
- 业务自定义SQL校验

Datapalace 数据管理中台

数据导入导出，多源数据关联分析

	Logstash	Spark	Flink
使用和部署成本	低	高	高
开发成本	低	高	高
海量、高吞吐的数据计算	不支持	支持	支持
多源数据关联分析	不支持	支持	支持
使用SQL表达计算逻辑	不支持	支持	支持
有状态的流式计算	不支持	做得不好	非常擅长
表达复杂的计算逻辑	不好做	支持	支持
擅长的领域	小数据量传输，无状态简单计算	海量数据，无状态复杂计算	海量数据，有状态复杂计算

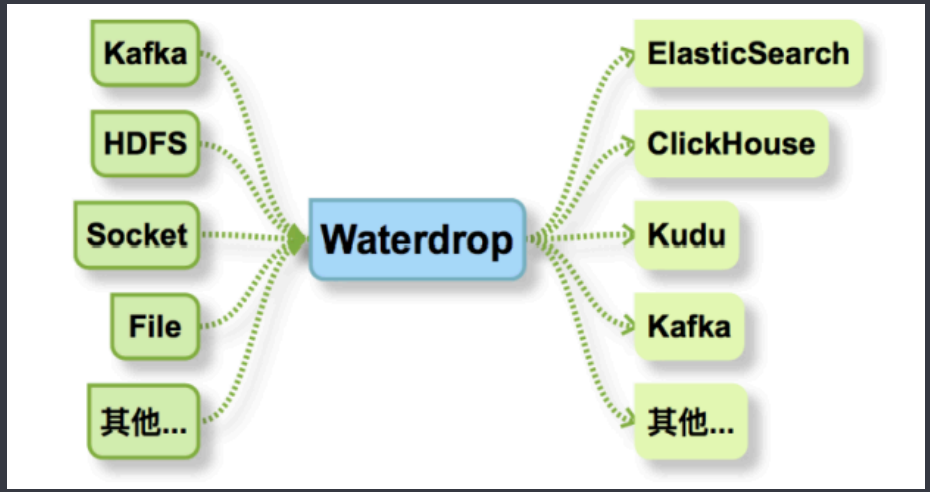
Datapalace 数据管理中台

数据导入导出，多源数据关联分析工具：Waterdrop

Waterdrop 是一个非常易用，高性能、支持实时流式和离线批处理的海量数据处理产品，架构于Apache Spark之上。(支持Flink的版本近期发布)

Waterdrop 的特性

- 简单易用，灵活配置，无需开发
- 实时流式处理
- 离线多源数据分析
- 高性能
- 海量数据处理能力
- 模块化和插件化，易于扩展
- 支持利用SQL做数据处理和聚合
- 支持Spark Structured Streaming
- 支持Spark 2.x



https://github.com/InterestingLab/waterdrop

elastic

ES智能诊断系统

把问题排查手段、实践经验的复用标准化

elastic

ES 智能诊断系统

概要介绍

- 支持14种巡检诊断项目
- 支持自动优化
- 诊断结果包含详细的数据披露和明确、可量化的优化建议
- 完全插件化的巡检、诊断、优化的开发
- 支持对接公司MS报警平台实现报警自愈
- 支持Query Profile 采样存储分析
- 使用数学、统计模型来简化问题排查难度

ES 智能诊断系统

已支持的诊断项列表 (可自动优化比例 = 8/14)

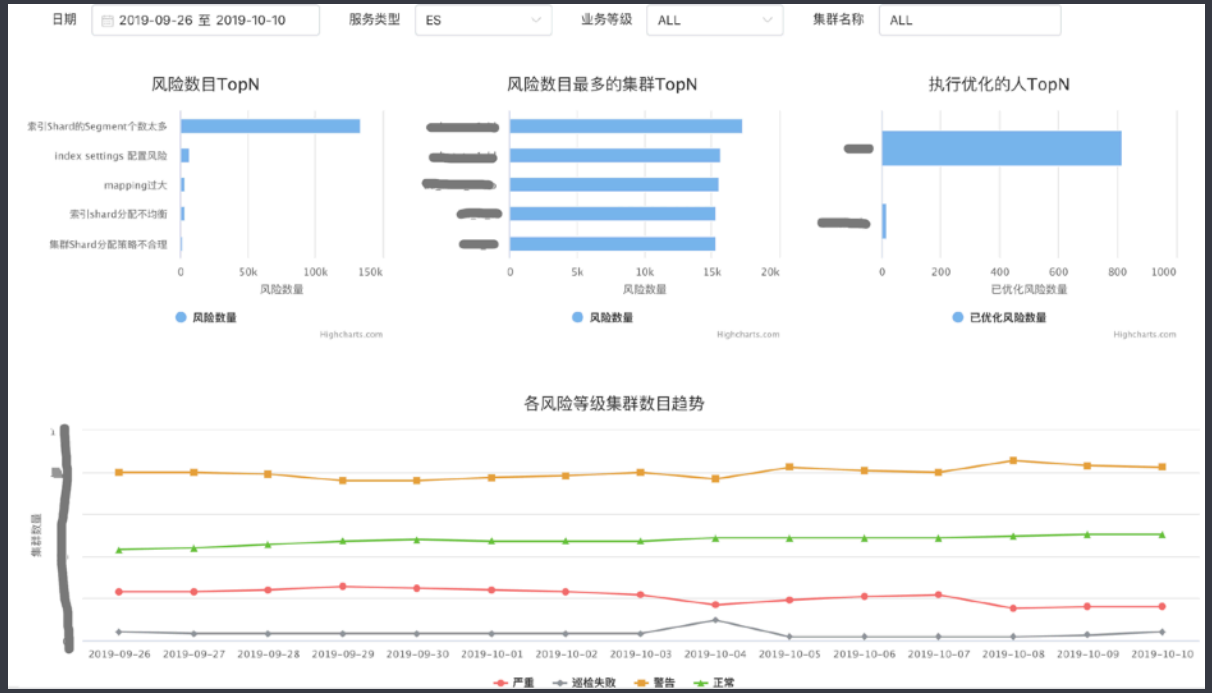
- 集群存储容量不足风险
- 集群Settings配置风险[可自动优化]
- 集群Shard分配策略不合理[可自动优化]
- 集群Shard个数分配不均衡
- 集群节点磁盘占用不均衡
- 集群Index Template Order 校验不合规
- 索引Template 配置风险[可自动优化]
- 索引settings 配置风险[可自动优化]
- 索引Mapping过大
- 索引shard分配不均衡[可自动优化]
- 时序型索引Shard size不合理[可自动优化]
- 非时序型索引Shard size不合理
- 索引Shard的Segment个数太多[可自动优化]
- 索引Shard数据损坏无法配[可自动优化]

ES 智能诊断系统

概要介绍

目标：

- 量化核心KPI
- 明确优化方向
- 快速定位问题



ES 智能诊断系统

概要介绍

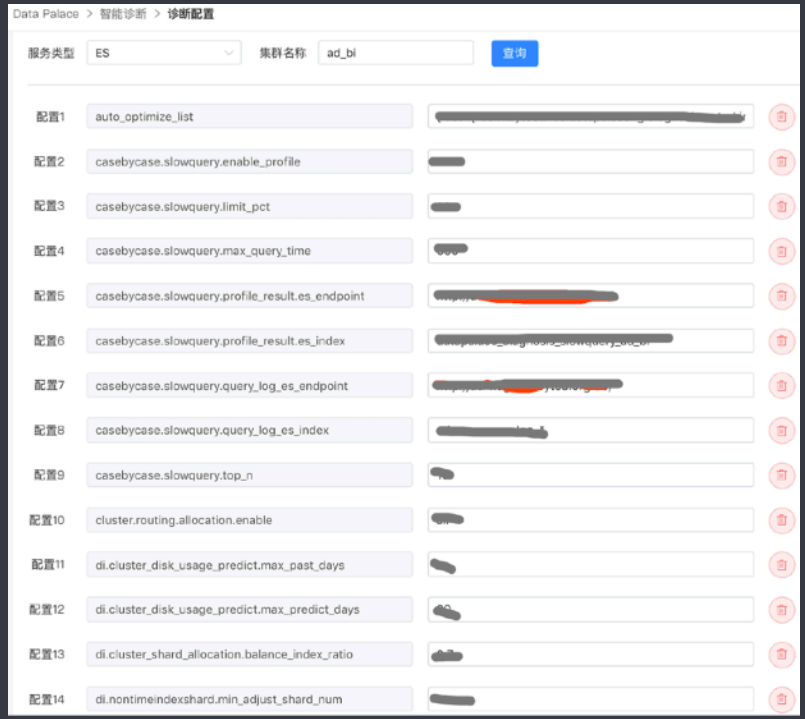
特点：

- 诊断项目明确
- 建议有细节，可量化
- 多种优化方式，支持批量操作
- 数据披露详尽
- 监控信息入口

elastic

ES 智能诊断系统

概要介绍



集群、索引级别的诊断配置

elastic

ES 智能诊断系统

诊断项：索引Shard的Segment个数太多[可自动优化]

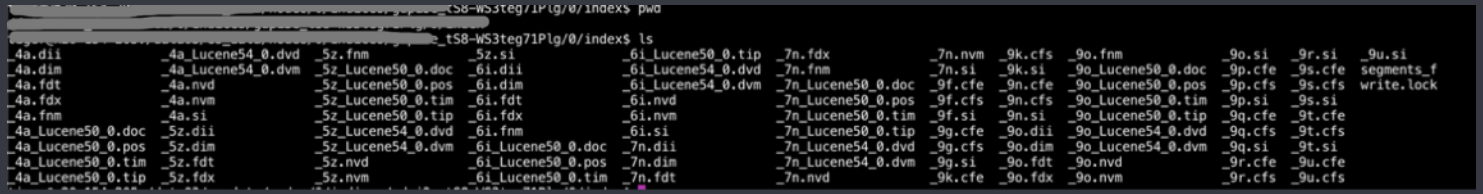
集群诊断报告

elastic

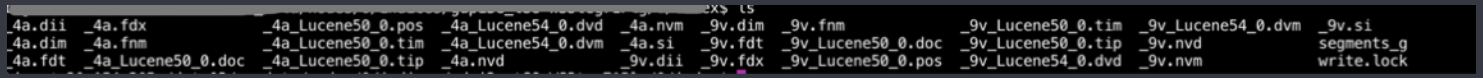
ES 智能诊断系统

诊断项：索引Shard的Segment个数太多[可自动优化]

某个Shard优化前（单Shard 102个文件）：



优化后（单Shard 29个文件）：



elastic

ES 智能诊断系统

诊断项：集群存储容量不足风险

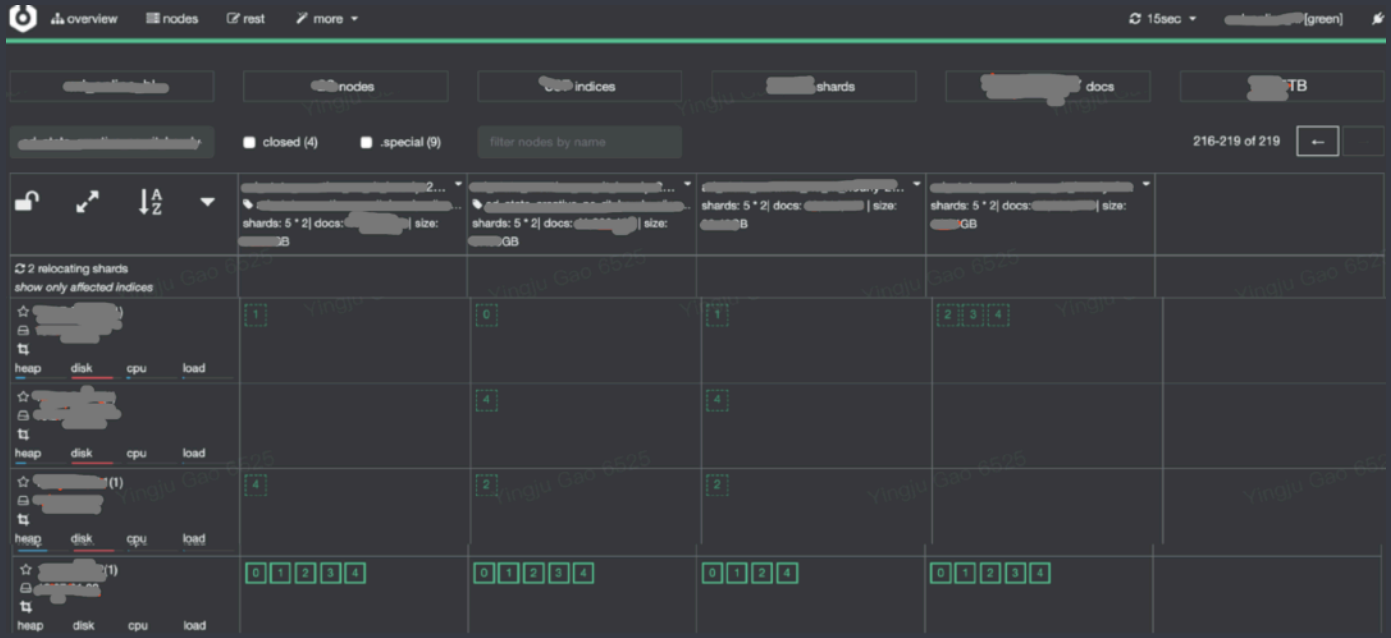
优化建议：
据此集群过去 7 天的日写入数据量预估每日写入量为 ? GB/天。**预计集群磁盘会在 8 天后容量不足，无法分配索引Shard。**请尽快 (1)扩容 (2)减少数据写入 (3) 删除历史数据。(当前数据：磁盘总大小 ? GB, 已使用 ? GB，占比 80 %， Watermark：95 %)

优化效果：
此诊断项上线后，**容量不足从事后报警转变为提前预测**，由容量不足引发的集群 RED(Primary Shard无法分配)的**电话报警从一周最多4次减少到0次。**

elastic

ES 智能诊断系统

诊断项：索引shard分配不均衡[可自动优化]



问题现象

elastic

ES 智能诊断系统

诊断项：索引shard分配不均衡[可自动优化]

优化建议：

此索引在各个节点或着在同一节点的不同分区上的shard分配不均衡。
这个索引的总shard个数为10, 允许分配的节点个数为18。
需要迁移的shard个数为5,需要迁移的数据总大小为30.60/GB, 迁移比例为0.50, 存在有多个shard分配到同一磁盘分区的情况有 0 个。
建议手动Reroute(参考 https://www.elastic.co/guide/en/elasticsearch/reference/current/cluster-reroute.html)此索引的shard。

elastic

ES 智能诊断系统

诊断项：时序型索引Shard size不合理[可自动优化]

问题：目前准备优化的Shard Size阈值是Google上查到的 2G ～60G的范围，可能不合理

优化建议：
此索引为时序型索引(按日、月分割的索引)，Shard size不合理，索引总大小 ? MB, shard 个数：26。一般每个shard合理的大小范围是2000 ~ 60000 MB。根据此索引近期的实际shard size分析并结合该索引可分配节点的磁盘分区总数：83 * 90%，建议调整此索引对应index template 中 number_of_shards = 11，来使新生成的索引shard大小更合理。

优化目标：
- 自动调整不合理的索引Shard个数

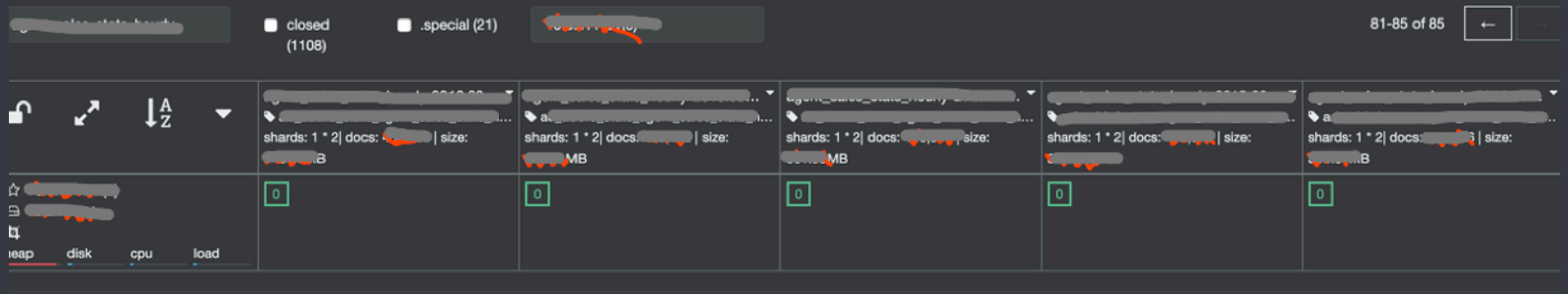
典型案例：
- 一个只有几MB的索引，有100+个Shard
- 一个几TB的索引，只有2个Shard
- 按日划分的索引，其大小会随着时间变化，如果shard个数始终不变是不合理的。

elastic

ES 智能诊断系统

诊断项：集群Shard个数分配不均衡

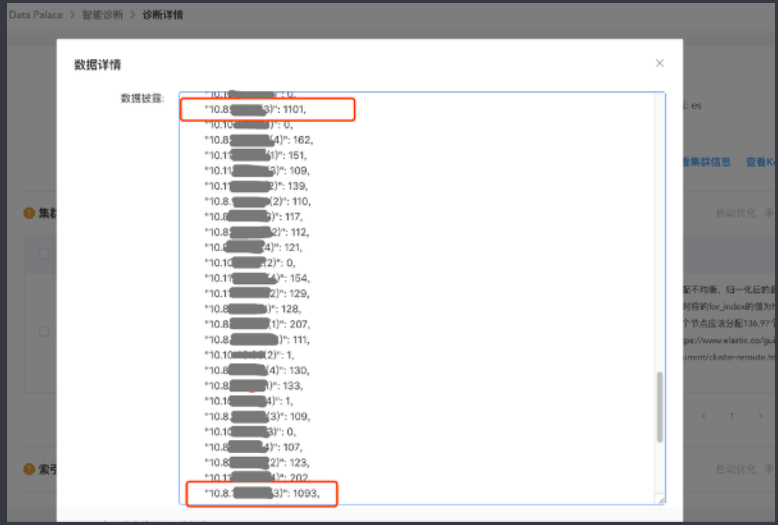
举例，ad_bi集群，现象：部分分日写入的索引，shard集中分配到了同一个节点上。



elastic

ES 智能诊断系统

诊断项：集群Shard个数分配不均衡



诊断结果显示：有两个节点分配的shard个数过多，达到1100 + ，是其他节点的6倍以上。

优化步骤：

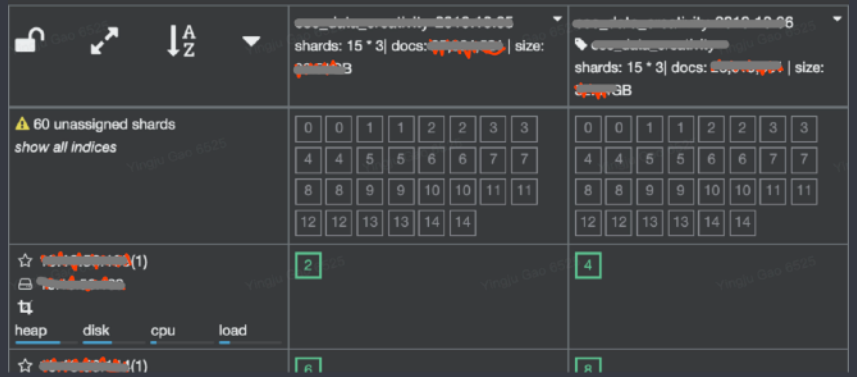
1. 下线业务
2. 调整c.r.a.balance.index，c.r.a.balance.shard两个参数
3. 等待集群完成自动relocating
4. 上线业务

elastic

ES 智能诊断系统

诊断项：Index Template 校验不合规

问题：



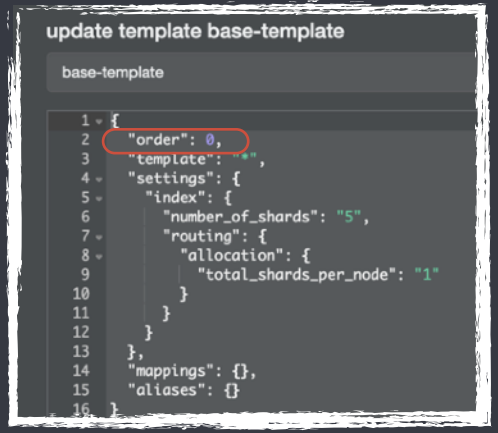
诊断结果：

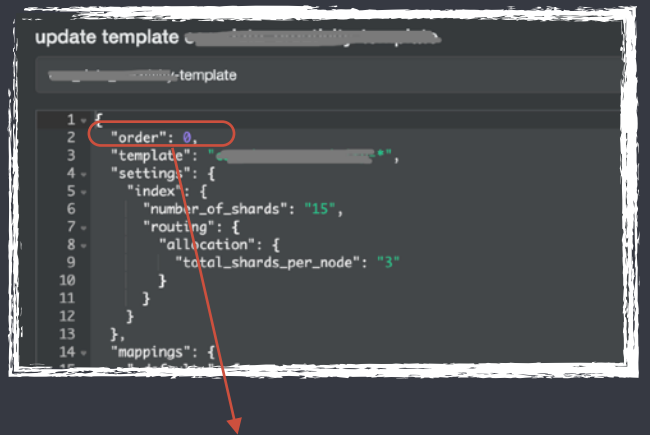
elastic

ES 智能诊断系统

诊断项：Index Template 校验不合规

原因 & 解决：





更新为 order: 10

elastic

ES 智能诊断系统

诊断项：索引Template 配置风险[可自动优化]

目标：能够自动适应集群节点数量变化

- 集群扩容，缩容
- 异常下线节点后
- 设置node attribute

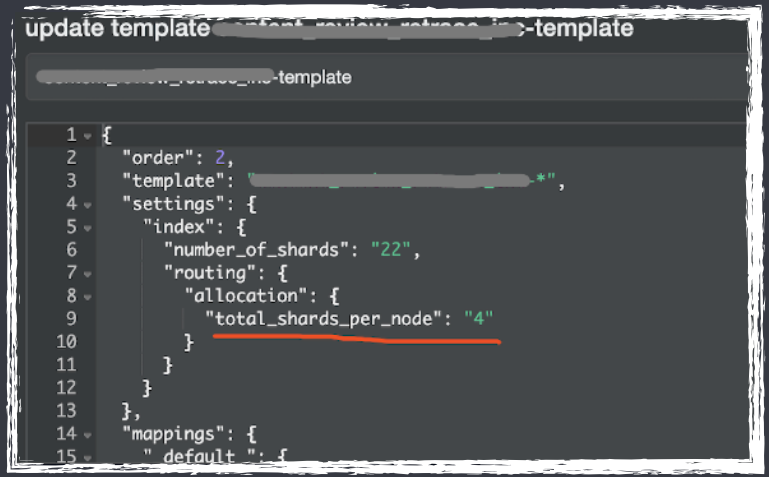
elastic

ES 智能诊断系统

诊断项：索引Template 配置风险[可自动优化]

优化前：

自动优化后：



集群扩容、缩容，索引冷热节点迁移都需要重新计算 total_shards_per_nodes

elastic

ES 智能诊断系统

CaseByCase

1.自动保存慢查询Profile

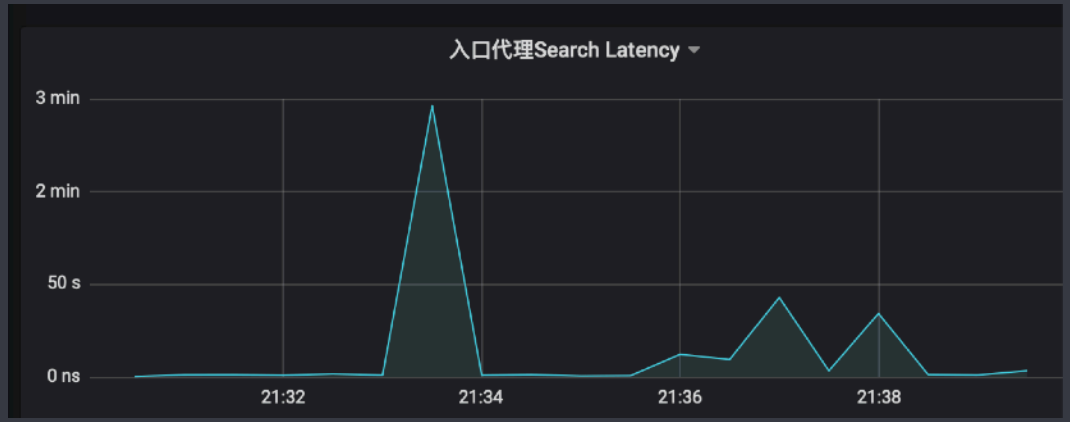
- 自动根据Search Latency PCT99 检测慢查询。
- 及时执行Profile，方便慢查询复现。
- 避免侵入业务。
- 兜底限速，防止影响正常业务。

2. 监控数据关联分析（皮尔逊系数）

- 分析监控数据的关联性（使用皮尔逊系数寻找监控时序数据的相关性）。
- 快速缩小排查问题的范围。
- 为工程师定位根因节约时间。

ES 智能诊断系统

CaseByCase: Search PCT99 慢查询关联指标分析



从3000+指标中找到线上查询时延毛刺的可能原因

指标数值：

- jvm.mem.heap_used
- bulk_queue
- search_queue
- pending_tasks
- load.1min
- cpu.busy
- cpu.iowait
- disk.io.write_bytes/device=+
- disk.io.read_bytes/device=+
- 。。。

指标纬度：

- 集群
- 节点
- 索引
- 宿主机
- 磁盘分区
- 网卡
- 。。。

elastic

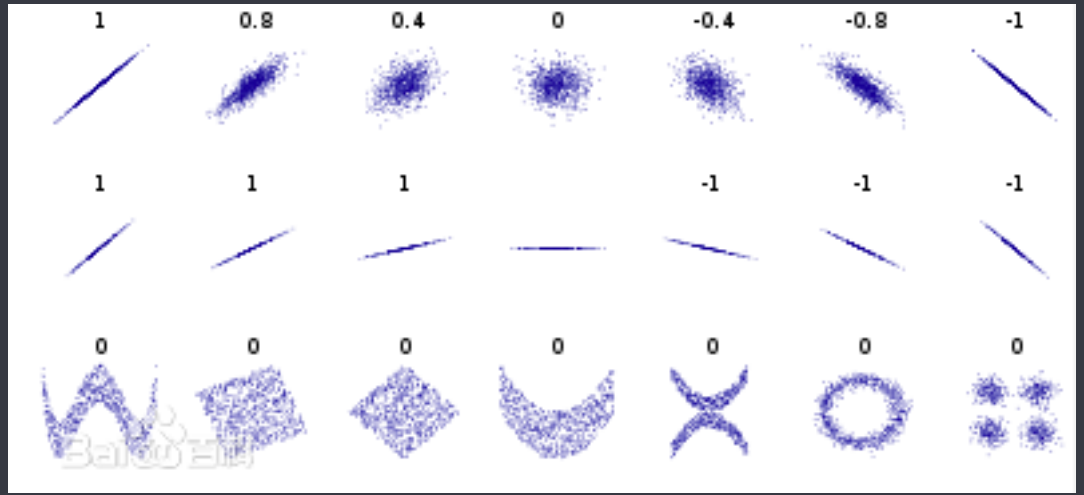
ES 智能诊断系统

CaseByCase: Search PCT99 慢查询关联指标分析

皮尔逊相关系数：

- 皮尔逊相关系数可用于度量两个变量的相关性。
- 值处于-1到1之间，1表示正相关，-1表示负相关，0表示无关。

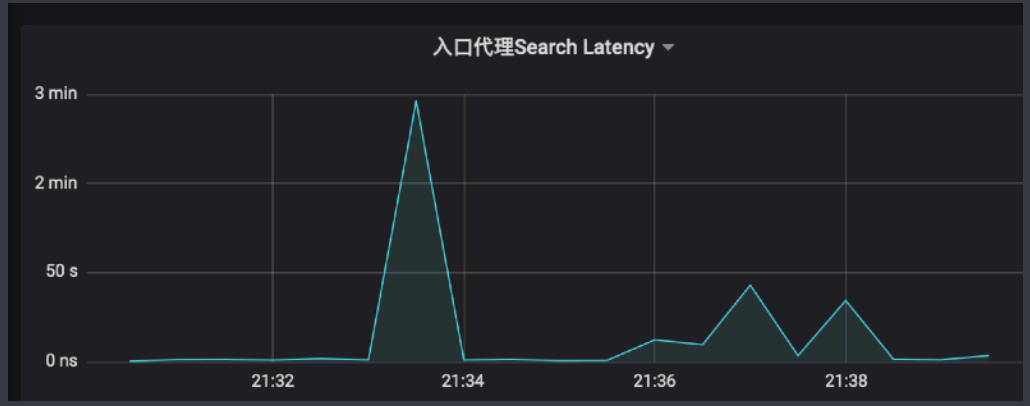
$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y},$$



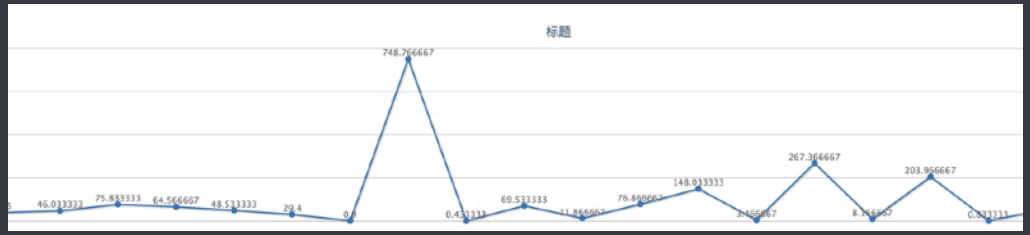
elastic

ES 智能诊断系统

CaseByCase: Search PCT99 慢查询关联指标分析



Search Latency Pct99趋势图



10.10.xx.yy节点的disk.io.read_requests/device=sdl的趋势图

结论：

- 进行慢查询定位分析，发现关联度最高的是 10.10.xx.yy 下的 disk.io.read_requests/device=sdl 指标关联度最高。
- 数据计算时间 < 10s，远远小于人工定位时间

问题：

- 会遇到集群80%节点的磁盘读写都与毛刺关联度较高的情况，关联分析没有意义。

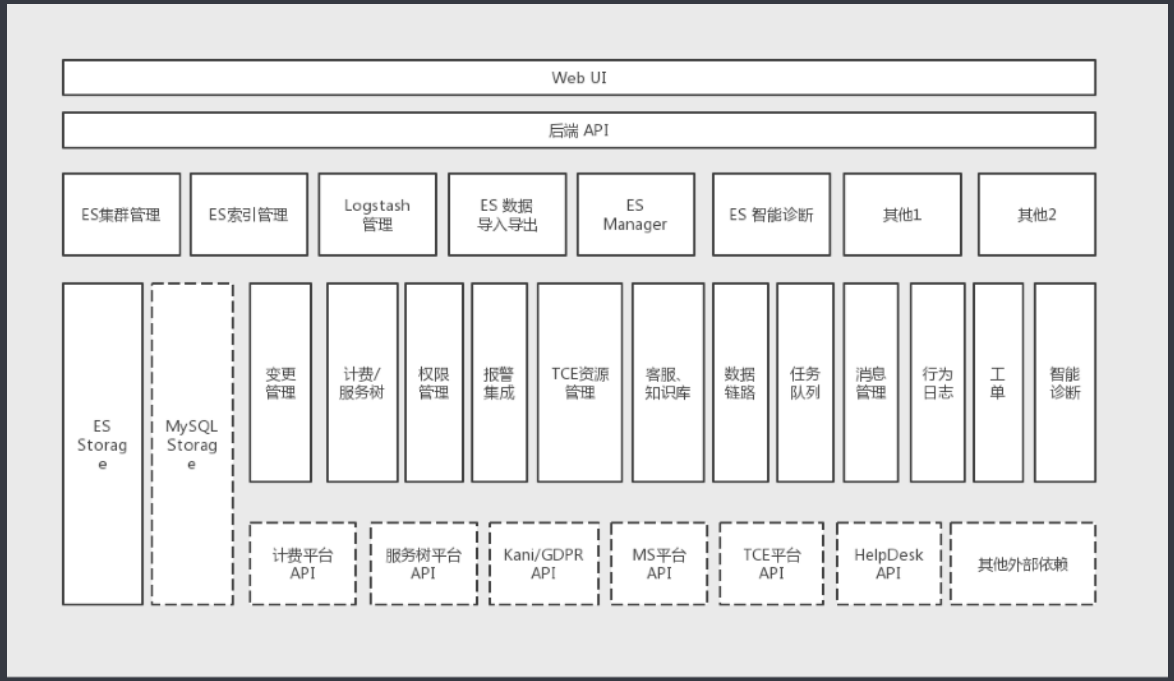
elastic

Datapalace 数据管理中台与ES智能诊断系统的设计与实现

elastic

Datapalace 数据管理中台的技术设计与实现

分层、分模块架构



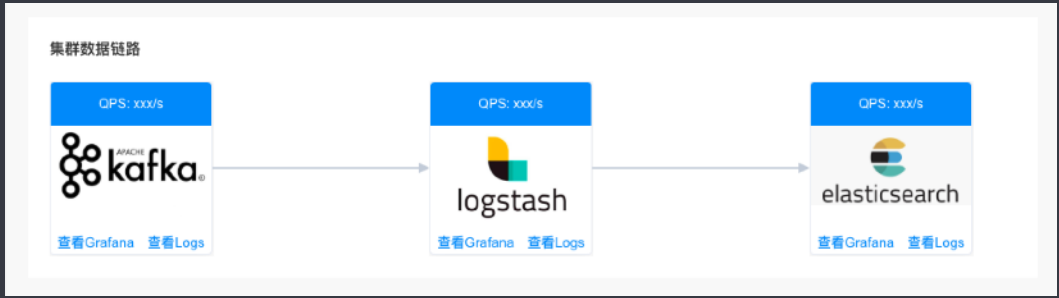
分层、分模块设计原则：

- 学习业务、理解业务后，从业务模型到软件架构的映射。
- 整体划分为基础通用功能、业务功能两层。
- 多个单一职责的模块，便于多人高效协作。
- 流程、业务分离；流程统一实现，业务插件化接入。
- 模块内高内聚，模块间低耦合。
- 部分适用：上层不依赖底层，两者依赖抽象(DIP)。

DIP(Dependence Inversion Principle) 依赖倒置/控制反转

Datapalace 数据管理中台的技术设计与实现

核心模块的设计与实现：数据链路



核心实体：

- Graph: 数据链路
- Vertex: 链路上的节点
- Edge: 节点的连接关系

```
public class Graph {

  private long id;

  private String name;

  private String description;

  private List<Vertex> vertexs;

  private List<Edge> edges;
}
```

```
/**
 * 创建数据链路
 */
public List<Vertex> createGraph(Graph graph);

/**
 * 连接节点，要求至少有一个vertex已存在于graph中
 */
public List<Vertex> connect(Vertex source, Vertex target);

/*
 * 删除链路节点
 */
public Vertex deleteVertex(Long vertexId);

/*
 * 通过链路中的任一节点，查询链路
 */
public Graph queryGraphByVertexId(long vertexId);
```

未来计划

未来计划

- 提高ES写入速度（很迫切）
  - Spark直写Lucene，再Restore到ES中
  - 修改Lucene源码
  - 对于不需要倒排索引的场景，尝试用RocksDB替换Lucene
- 提高ES查询性能
  - PCT99 慢查询，节点OOM根因定位
  - 查询流控
  - 修改查询源码，实现类似Spark的推测执行（多副本同时执行，看谁先执行完）
- 改进Master单点性能问题
  - 升级ES 版本 >= 6.8.0
  - 修改Master源码
- AIOps + ES
  - Query执行时长，资源消耗预测(神经网络)
  - 慢查询根因定位(决策树，贝叶斯，神经网络)

Thanks

个人微信

Contributors: 张海雷 高英举 汤明 伏开宇 郭峰 宋红 范文棋 袁宇豪 文炫达 卜衡 方品 龚凌磊 胡东林

elastic