Elasticsearch
在大数据可视化分析中的应用

苏淦

2019.03

目录

CONTENTS

关于我

个人简介：

- 任职： 大数据中心/平台部/负责人
- 部门目标： 以基础建设提高项目研发效率，用技术驱动业务！
- 联系方式： sugan@datastory.com.cn （欢迎简历！）

主要经历：

- 数说中台业务线系列PaaS平台研发
  - 数说方舟、数说工场、数说海纳、数说罗盘
- 数说分布式互联网数据爬虫平台建设
- 曾负责多个企业级大数据平台、高并发系统项目架构和实施

大数据可视化分析工具

- ✓ BI 报表工具： PowerBI、Tableau、QuickBi 等（支持多数据源）
- ✓ 擅长可视化工具： ds.js 、 kibana（Es only）
- ✓ 数据分析/挖掘工具： jupyter

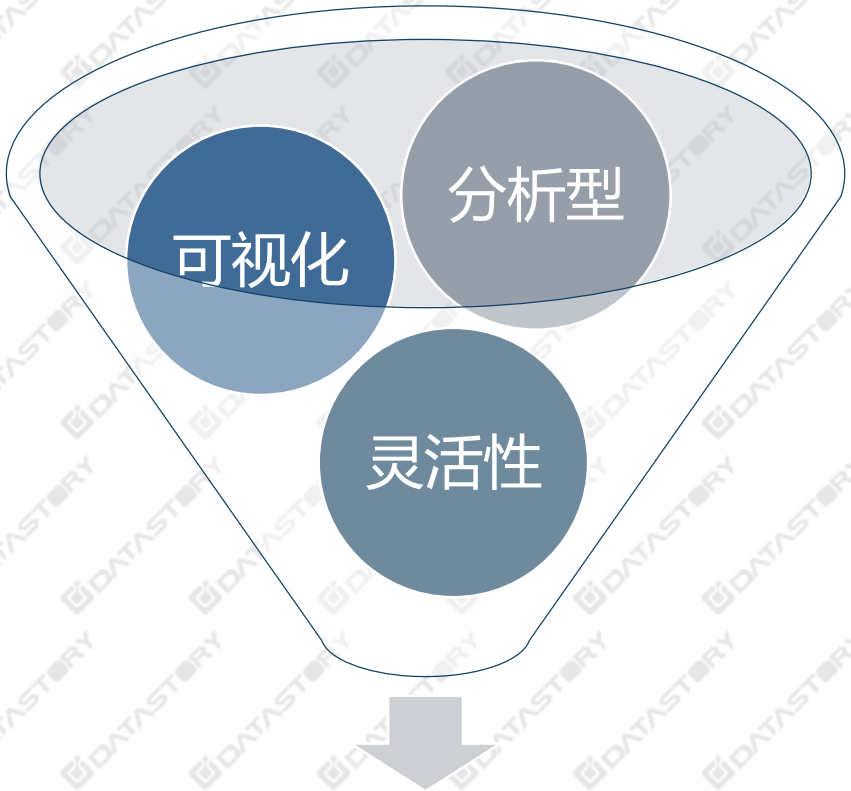


tableau　Power BI　QuickBi

D3　kibana　jupyter

数据可视化分析- 相关需求场景

角色多样

- 运营人员
- 业务主管
- 管理层

- 数据分析师
- 算法工程师

- 程序猿(其他)

场景多样

- 图表： 转化率、趋势图、度量指标等
- 报表： 多种业务流程、绩效业绩等
- 经营驾驶舱： 管理决策经营报表，业绩总览等

- 分析功能：数据源检索、筛选、透视、交叉联合分析等
- 数据处理： 数据集过滤筛选、 数据处理转换

- 其他： 加酷炫自定义图表、 做数据API、 导数据等等...

大数据可视化分析- 功能需求总结

- ✓ 多数据源对接能力
- ✓ 丰富的图表展现方式
- ✓ 数据分析的实时可交互性（ UI交互即数据分析）
- ✓ 多样的数据结果集过滤/转换能力
- ✓ 一定的可编程能力 （高级玩家）
- ✓ 满足多人协作分析（分析师团队）
- ✓ 数据结果集对接其他系统能力

数说可视化分析产品 – Why ES

ES的特性，很好地匹配 “数据可视化分析 功能需求” 的大部分关键点：

1. 集群弹性可线性扩展， 可处理数据量级范围广
2. 支持高实时响应，用户体验佳
3. 灵活的数据结构 （RDBMS所不能满足）
4. 支持全文检索和多种聚合操作， 满足OLAP分析场景
5. 多种Restful API： 最简单的方式提供了更多开发者能力

雷达系列产品	数说立方	数说方舟
仪表盘/联动图表	分析型工具	可视化+ 分析 + 编程

数据方舟 –功能简介

数说方舟

1 数据源接入与集成

- 可接入本地数据文件及远程数据库，能力覆盖关系型数据库及NoSQL数据库
- Elasticsearch为内置首先数据源！

2 多种数据分析交互方式

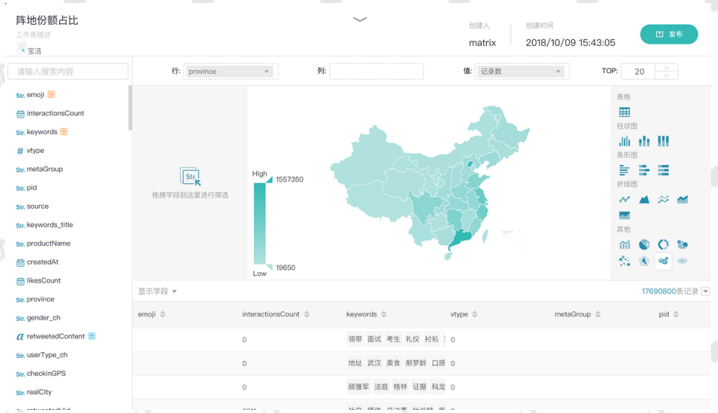
- 支持拖拽式数据分析交互，系统内置分析方式覆盖大部分分析场景。
- 支持编写自定义分析脚本，能满足任意分析场景。

3 多种数据可视化方案

- 一键式数据可视化操作，覆盖分布、趋势、地理、对比等多种场景。
- 多年快消数据可视化经验积累，图表涵盖行业多种销售场景、多类指标。

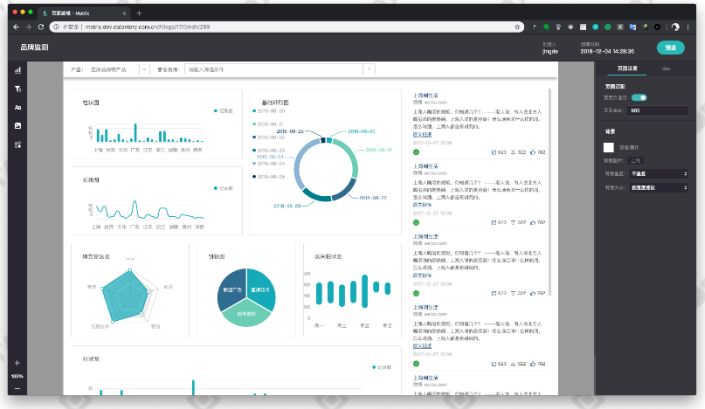
4 分析结果一键发布API/页面

- 支持用户将分析结果发布为web API
- 支持多种格式的数据结构返回，对前端支持友好，降低联调成本。



5 大数据门户自主搭建

- 快速将分析结果搭建为数据门户应用，允许自定义菜单、页面、权限分配。
- 完善的页面联动、图表联动支持。



6 搭配ELK监控与告警机制

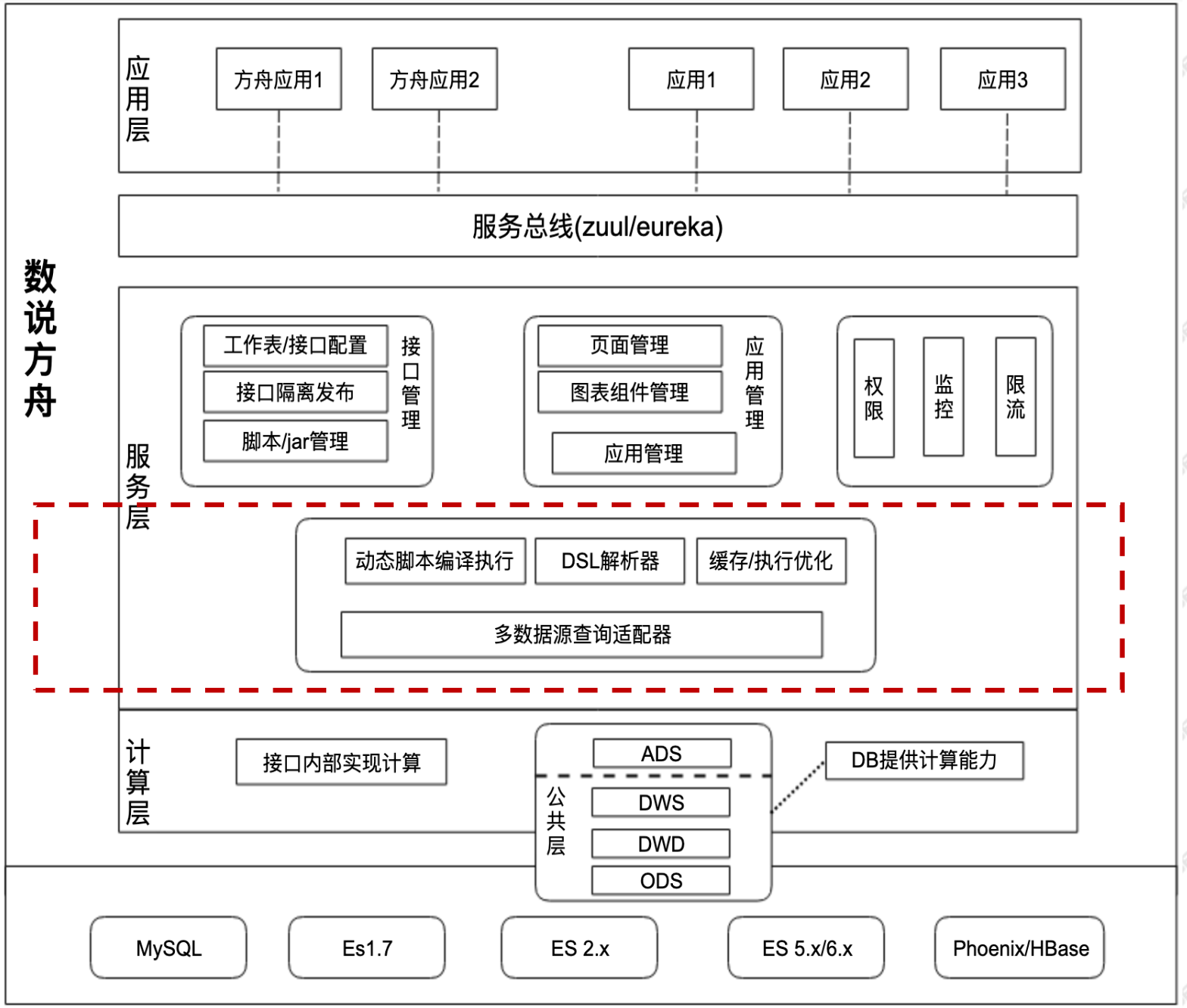
- 自定义配置数据异常告警机制。
- 完善的日志记录，确保数据门户应用的响应速度与稳定。

数据方舟： 提供可视化分析查询能力的应用开发平台

数据方舟 – 系统架构

技术栈说明

- 存储：
  - 主要数据源为Elasticsearch
  - 支持主流关系型/非关系数据库
- 服务端：
  - Spring Cloud 微服务套件
- 前端：
  - Vue.js/ Echarts框架



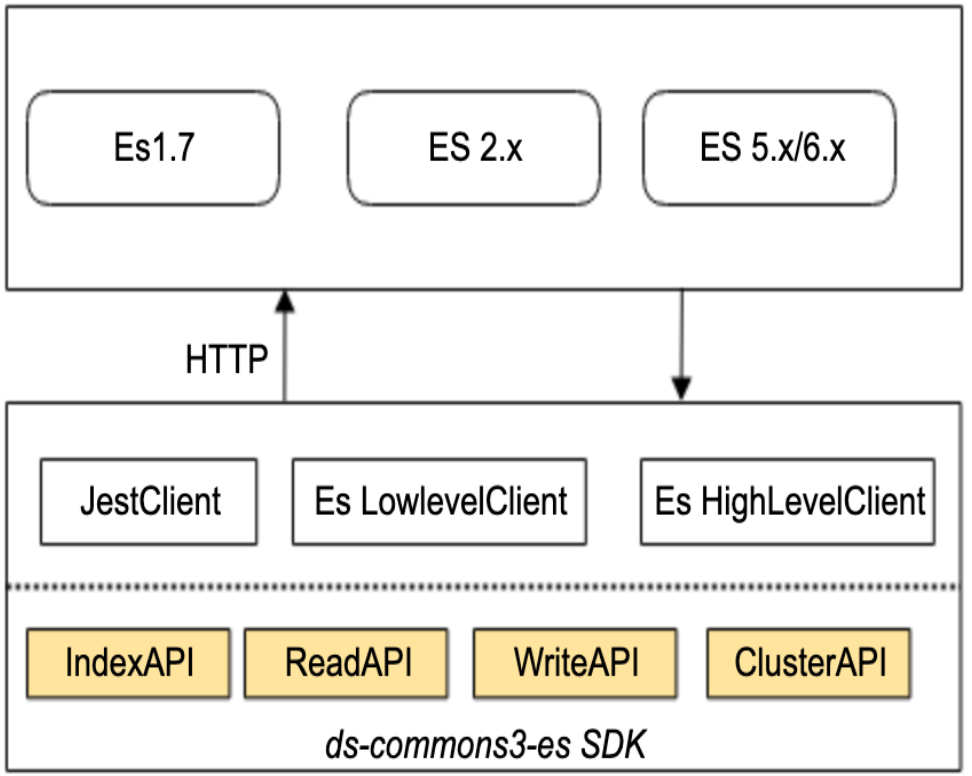
问题1- 多版本ES 读写支持

业务系统现状：

1. 多个应用系统ES版本差异大，从1.7、2.x、5.x都有
2. 旧业务系统改造升级难度大，部分系统处于暂停维护状态
3. 但是部分系统模块和数据仍然在发光发热~

解决方法：

1. Transport相关接口跨版本兼容性问题太多，也难以通过接口适配来兼容， 只走Http协议的Restful接口 ，也为做后面异常请求监控打好基础
2. 为了保持代码级别大部分兼容 （QueryBuilder类的姿势），优选官方5.6出的High-Level RestClient
3. 封装ds-commons3-es SDK，改造ES 官方High-Level RestClient ，在Http request / response 侧添加钩子，对不同版本的请求/返回 json 格式进行适配。



问题1- 多版本ES 读写支持

实施总结：

1. ds-commons3-es SDK开发难度小， 只在请求拦截，在json级别产生的兼容冲突也跟预期一样并不多

收益：

1. 旧业务系统的代码容易迁移， 改动冲突远小于升级代码包依赖， 为升级业务系统的ES提供了前提条件
2. 支持了多版本读写的，满足数说方舟查询引擎对ES的能力支持

```
@Override
protected String compatibleForV23(String sourceJson) {
    String[] deletePaths = new String[]{
            "$.aggregations..date_histogram.offset",
            "$.query..ignore_unmapped",
    };
    sourceJson = JsonPathUtil.del(sourceJson, deletePaths);
    return sourceJson;
}

@Override
protected String compatibleForV17(String sourceJson) {
    //1. 如果json包含agg，则查找在v1.7有异常的字段，进行修正
    String[] deletePaths = new String[]{
            "$.aggregations..date_histogram.offset",
            "$.query..bool..boost",
            "$.query..bool..disable_coord",
            "$.query..bool..adjust_pure_negative",
            "$.query..ignore_unmapped",
    };

    sourceJson = JsonPathUtil.del(sourceJson, deletePaths);
    return sourceJson;
}
```

问题2- 大计算量请求干挂ES

问题原因：

1. 业务所在ES集群容量接入方舟前， 容量设计未经过重新评估，对index 的聚合查询让业务集群计算能力超过原有设计负荷 。
   - 例如 拖拉了某些基数值很多的字段做agg + sort给集群造成巨大压力
2. 自定义分析的模式过于灵活，部分操作在小白用户手上，可能玩出很多杀手花样

解决方法：

1. 产品层面：
   - 规划高级字段设置选项， 允许有限地设置对各字段能进行的操作
   - 查询引擎对查询请求进行队列控制 （简单策略）

2. 业务层面：
   - 评估ES集群容量，对计算密集型场景做好优化。如调整内存磁盘容量配比， 1 比10以响应更多的agg请求场景
   - 合理预估业务集群能承受的负载（监控+压测）
   - 适当控制业务集群 search threadpool size

下阶段

- prestoDB on Elasticsearch （On the way）：
  - 基于提供跨数据源交叉查询分析功能
  - 基于外部MPP计算能力，减缓ES计算压力

- （方舟）查询引擎层调度机制完善：
  - 支持更多样的查询调度策略
  - 对后端DB进行可配置化熔断、限流

Q&A

DATASTORY