ES衍生系统介绍以及索引技术在数据部门的应用

CSDN数据支撑部 架构&算法
祝威廉

ES 真的很棒

1. 部署方便，下载，启动，完事！
2. 功能太全了！而且都是RESTFull 化的
3. 有大的应用案例(GitHub)
4. 文档丰富，使用者多，插件也丰富了
5. 多年发展，稳定！

为什么我们要开发CS

简化，简化，剥离，剥离

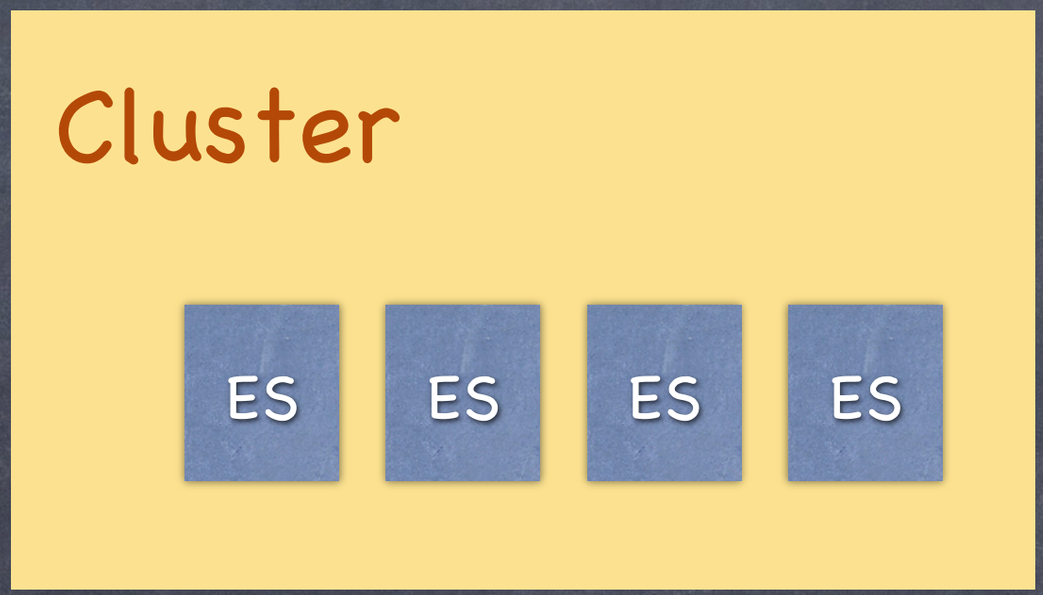
CS简化了什么

- Netty Socket /HTTP => AKKA/HTTP
- 集群状态维护 => zookeeper
- 引入新的MVC框架(ServiceFramework),而不是自己开发
- Plugins => Strategies
- 剥离其他功能，只提供索引 增删改查功能。

CS 延续了什么

- ES的 JSON IN JSON OUT 以及 RESTFull 风格
- 是 ES API集合 一个子集
- 集群形态，数据分片

ES 架构



CS 架构

connected
BY AKKA

Cluster

Strategies

CS | CS | CS | CS

CS

Strategies

Indexing Module

Strategies介绍

- 组件化
- 链路化
- 配置化
- 取代插件机制
- 独立组件，广泛应用在数据平台其他服务中

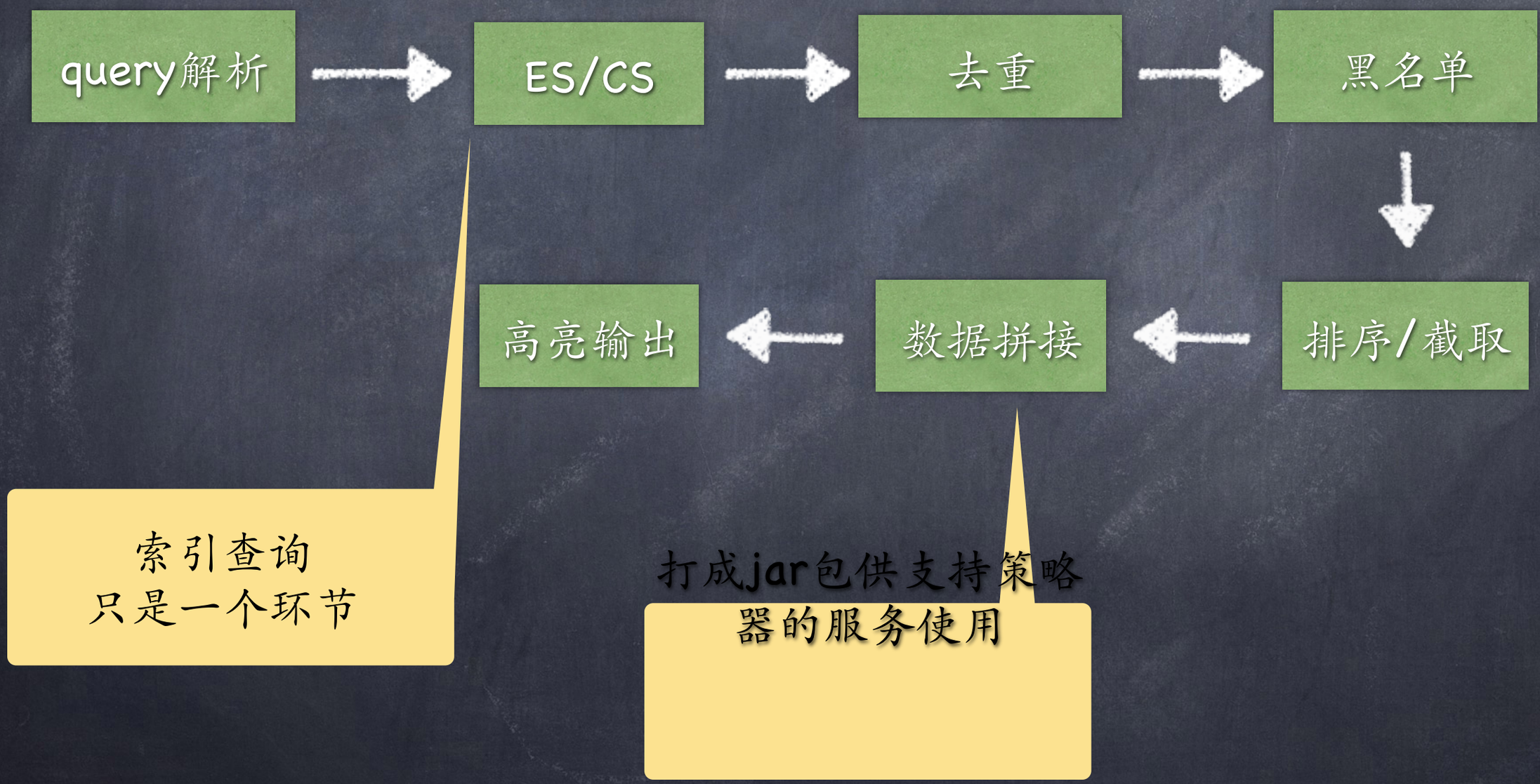
Strageties概念组成

- 策略(stragety)
- 处理器(Processor)
- 组合器(Compositor)

```
"so_search": {
    "desc": "so.csdn.net使用的搜索策略",
    "strategy": "net.csdn.service.strategy.SoStrategy",
    "processor": [
        {
            "name": "net.csdn.service.strategy.SearchProcessor"
        }
    ],
    "ref": □,
    "compositor": [
        {
            "name": "net.csdn.service.strategy.DeduplicateCompositor"
        },
        {
            "name": "net.csdn.service.strategy.BlackListCompositor",
            "params":[{}]
        },
        {
            "name": "net.csdn.service.strategy.SortAndCutCompositor"
        },
        {
            "name": "net.csdn.service.strategy.DataGatewayCompositor",
            "params": [
                {
                    "download": "title,body"
                }
            ]
        },

        {
            "name": "net.csdn.service.strategy.RenderCompositor"
        }
    ],
    "configParams": {}
},
```

一个典型的搜索请求


query解析
ES/CS
去重
黑名单
排序/截取
数据拼接
高亮输出
索引查询
只是一个环节
打成jar包供支持策略
器的服务使用

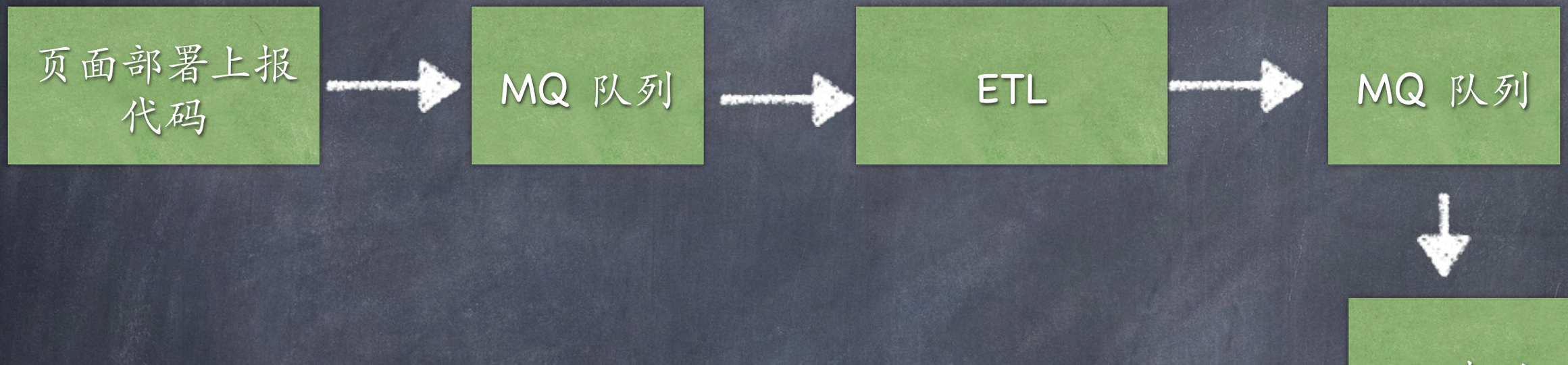
CS 架构的优势

- Reboot 索引服务不可承受之痛
- Strategies 灵活，轻量，可随时更改
- 索引服务轻易不重启，大部分调整只需要重启 strategies 服务

CS用途 - 站内搜索

- 数据库抓取或推送/web抓取 进行索引构建
- 都存在索引更新问题(数据更新，删除比较难办)
- 如何解决?

CS用途 - 站内搜索



- 新建：用户预览或者有其他用户访问也可
- 更新：ETL 负责更新周期
- 删除：用户访问404 或者更新周期发现内容已经删除

CS用途 - 站内搜索

- 对业务系统透明
- 一并解决了索引同步问题
- 复用了上报(凡是个站点，都改有自己的上报吧)

CS用途 - 标签系统

- 内容/用户 都会被打上标签
- 传统使用数据库存储
- 1000w内容*平均5个tag* = 5000w
- 数据挖掘会从各个维度给内容/人 打标签，所以标签数目 >>> 5
- 分表分库。。。。很麻烦。。。。

CS用途 - 标签系统

index = f(key => List)

CS用途 - 标签系统

```
curl -XPUT "127.0.0.1:9500/func_query_extern/csdn/_mapping" -d '{"csdn": {
        "_source": {
            "enabled": false
        },
        "properties": {
            "id":           {"type": "integer","index": "not_analyzed"},
            "tag":          {"type": "string","term_vector":"with_positions_offsets","analyzer":"CommaAnalyzer"},
            "system_tag":   {"type": "string","term_vector":"with_positions_offsets","analyzer":"CommaAnalyzer"},
            "query_tag":    {"type": "string","term_vector":"with_positions_offsets","analyzer":"CommaAnalyzer"},
            "created_at":   {"type": "string","index": "not_analyzed"}
        }
    }
}'
```

CS用途 - HBase 辅助查询

- 数据团队内部只使用： 索引，HBase，文件，Redis
- HBase 各种过滤条件查询繁琐且慢
- 使用索引可以很好解决类似问题

感谢大家

个人简介:

现在CSDN数据支撑部门从事架构和算法方面的研究

微博: http://weibo.com/princecharmingj

GitHub/Code 项目

- ServiceFramework:https://github.com/allwefantasy/ServiceFramework
- ActiveORM:https://github.com/allwefantasy/active_orm
- MongoMongo:https://github.com/allwefantasy/mongomongo
- QuickSand(流沙): https://github.com/allwefantasy/QuickSand