DTCC

数，造，未，来

第十二届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2021

2021 年 10 月 18 日 - 20 日 | 北京国际会议中心

Spark
Hadoop
OLAP
HANA
MPP
Oracle
SQL

IT168.com ChinaUnix.net ITPUB

Elasticsearch基于对象存储的冷热分离架构与NLP增强实践

百度资深研发工程师 武云峰

01 基于对象存储(BOS)的冷热数据分离架构

02 基于对象存储(BOS)冷热分离架构的应用实践

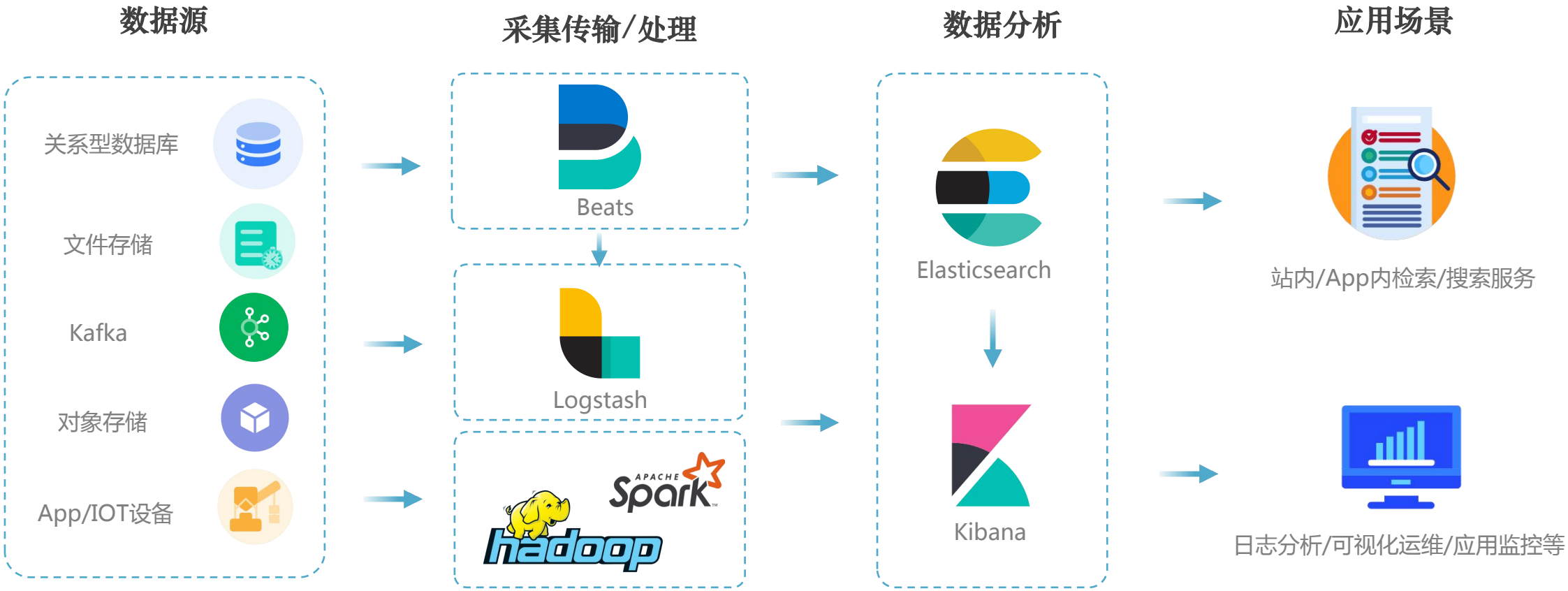
03 Elasticsearch与NLP特性增强实践

04 未来展望

01

基于对象存储(BOS)的冷热数据分离架构

ELK经典应用架构


数据源
采集传输/处理
数据分析
应用场景
关系型数据库
文件存储
Kafka
对象存储
App/IOT设备
Beats
Logstash
APACHE Spark
hadoop
Elasticsearch
Kibana
站内/App内检索/搜索服务
日志分析/可视化运维/应用监控等

公有云存储成本对比

存储类型	类型	元/GB/月	100T数据成本
通用型SSD	NVME	0.9000	9.2万
高性能云磁盘	HDD	0.3500	3.58万
对象存储(BOS)	**标准存储**	**0.119**	**1.22万**

来源参考：百度智能云相关产品目录价格

在公有云上，对象存储的成本是高性能云磁盘的 **34%** ，是通用性SSD磁盘的 **13%**

当前业务遇到的问题

以互联网/制造业/IOT等领域为例

- **数据体量庞大**：日增数据量大，数据每多保留一天都意味着成本的线性增加
- **冷热数据价值差异**：“冷” “热”区分度大，最感兴趣的往往是最近几天，保存的数据周期越久越好
- **冷数据响应敏度低**：“冷”数据可以被正常检索、能出结果就好，对速度并无特别要求，能快最好

……

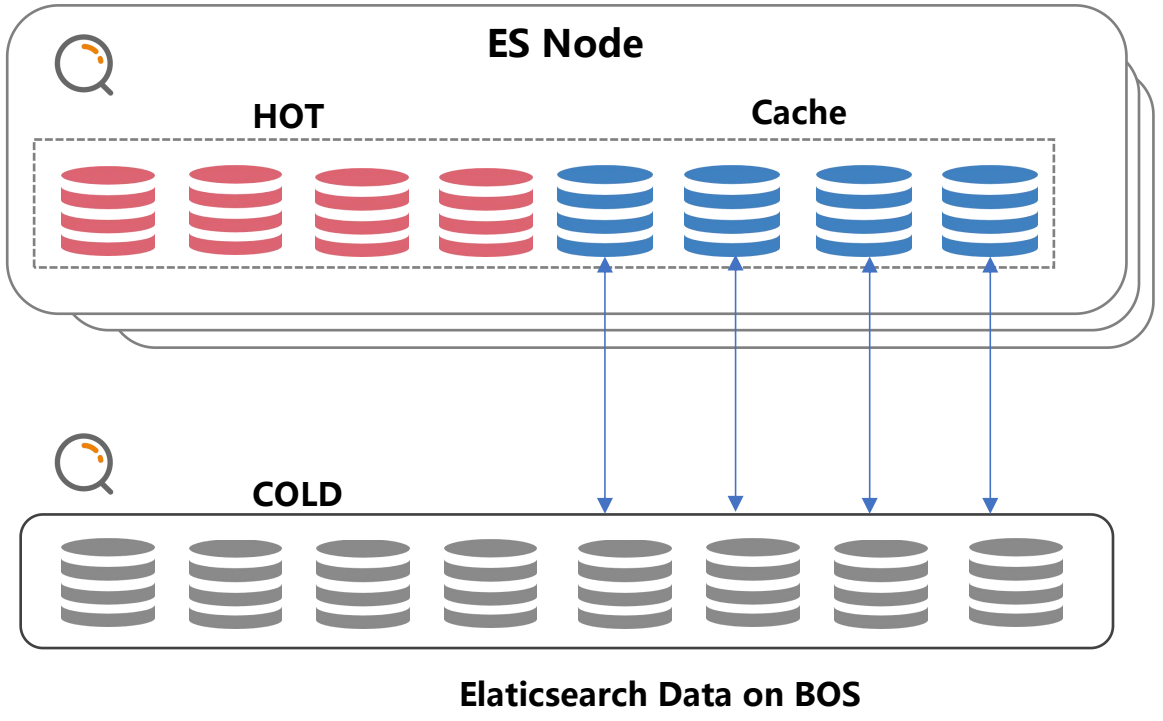
如果能利用公有云的对象存储，存储成本可以降低 **90%！**

基于BOS的冷热分离架构

百度智能云ES(BES)可以使用对象存储BOS来存放索引，并且可以直接查询BOS上的索引数据

- **分级**：冷数据存BOS，热数据存SSD，大幅降低成本
- **扩展**：单节点容纳数倍磁盘存储空间数据（1:5 – 1:20）
- **兼容**： 查询API完全兼容，直接查询远端BOS数据

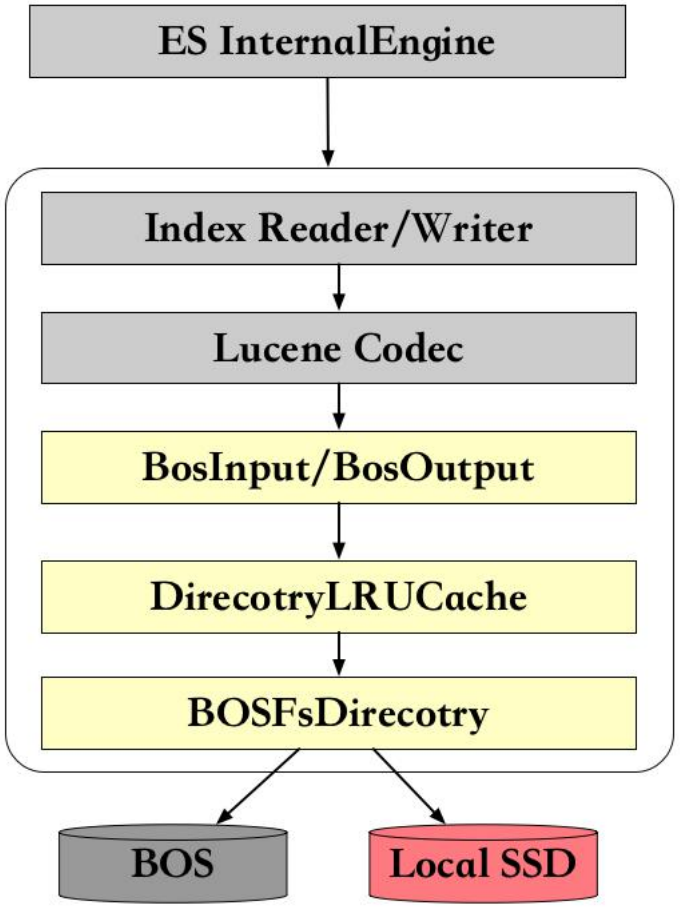
管控平台调度



基于BOS的冷热分离架构——实现路径

实现路径

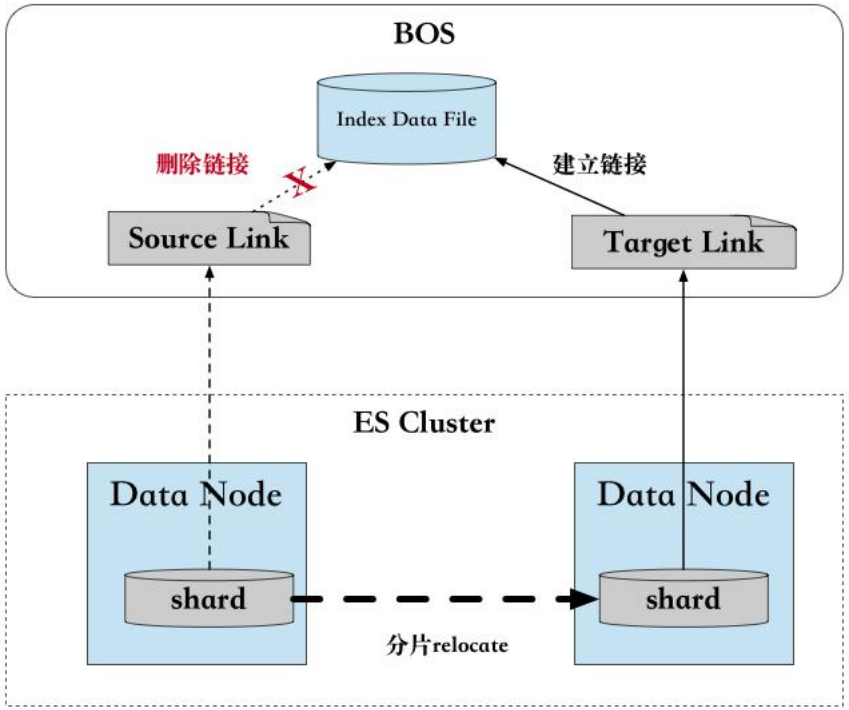
- **BosFsDirectory**：实现对 BOS 和 Local SSD 的管理
- **DirectoryLRUCache** ：缓存索引，加速查询
- **BosInput/BosOutput**： 屏蔽异构存储介质



基于BOS的冷热分离架构——Relocate

Relocate改进

- **热数据迁移**： ES正常的迁移逻辑
- **冷数据迁移**：采用建立“硬链”的方式实现
- **临界数据**：数据先上传至BOS，然后“硬链”方式



基于BOS的冷热分离架构——冷热观测

GET /_bpack/migration/stats

GET /_bpack/migration/{index}/stats

- **hot.data**： 热数据大小，本地
- **cold.data**：冷数据大小，远端BOS
- **cache.data**：cache大小，查询的cache
- …

uuid	status	active.shard	finished.shard	waiting.shard	running.shard	hot.data	cold.data	cache.data
[redacted]	COLD	24	24	0	0	0b	1.5tb	0b
[redacted]	COLD	12	12	0	0	0b	556.8gb	0b
[redacted]	COOLDOWN	12	0	12	0	0b	461.4gb	0b
[redacted]	COLD	12	12	0	0	0b	397.6gb	3.3gb
[redacted]	COLD	12	12	0	0	0b	284.2gb	0b
[redacted]	COLD	12	12	0	0	0b	217.4gb	0b
[redacted]	COLD	18	18	0	0	0b	199.4gb	0b
[redacted]	COLD	12	12	0	0	0b	164gb	0b
[redacted]	COLD	4	4	0	0	0b	62.8gb	0b

基于BOS的冷热分离架构——迁移限速

- 索引文件上传速度控制

```
PUT /_cluster/settings
{
   "persistent": {
       “bpack.migrate.max_bytes_per_sec” : "40mb"
   }
}
```

- 上传线程数和队列大小

```
PUT _cluster/settings
{
   "persistent": {
      "thread_pool.bos_upload.size": 10,
      "thread_pool. bos_upload.queue_size": 1000
   }
}
```

基于BOS的冷热分离架构——使用方式

方式一 创建基于BOS的冷热分离架构的索引

Step 1：集群关联远端对象存储bucket

```
PUT /_cluster/settings
{
   "persistent": {
      "bpack.remote_storage.bos.access_key": "xxxxxxxx",
      "bpack.remote_storage.bos.secret_key": "xxxxxxxx",
      "bpack.remote_storage.bos.bucket": "es_remote_bucket",
      "bpack.remote_storage.bos.endpoint": "bj.bcebos.com",
      "bpack.remote_storage.bos.base_path": "es_index_data"
   }
}
```

Step 2：创建索引

```
PUT /index
{
   "settings": {
      "number_of_replicas": 0,
      "number_of_shards": 1,
       "index.store.type": "bosfs"
   }
}
```

Step 3：索引置冷，启用远端BOS存储

POST /_bpack/migrate/{*index_name*}/cold

基于BOS的冷热分离架构——使用方式

方式二 对于已经存在索引启用基于BOS的冷热分离架构

Step 1 : 关闭索引

```
POST /{index_name}/_close
```

Step 2 : 设置 index.store.type

```
PUT /{index_name}
{
  "settings": {
        "index.store.type":"bosfs"
  }
}
```

Step 3 : 打开索引

```
POST /{index_name}/_open
```

Step 4 : 索引置冷

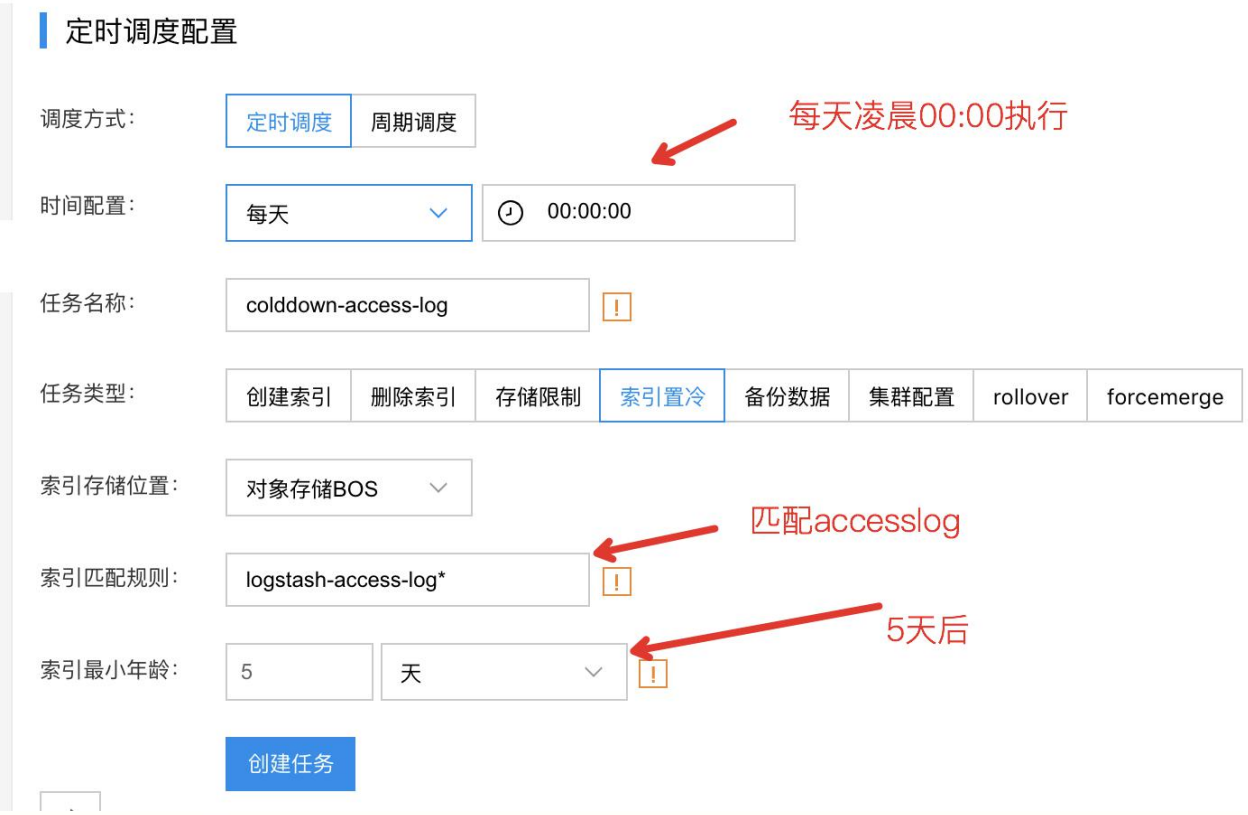
```
POST /_bpack/migrate/{index_name}/cold
```

基于BOS的冷热分离架构——使用推荐

- 索引按 月/天/小时 等创建, 按时间做分区

logstash-access-log-2021-10-01
logstash-access-log-2021-10-02
logstash-access-log-2021-10-03
…

- 调度平台配置索引”变冷”策略



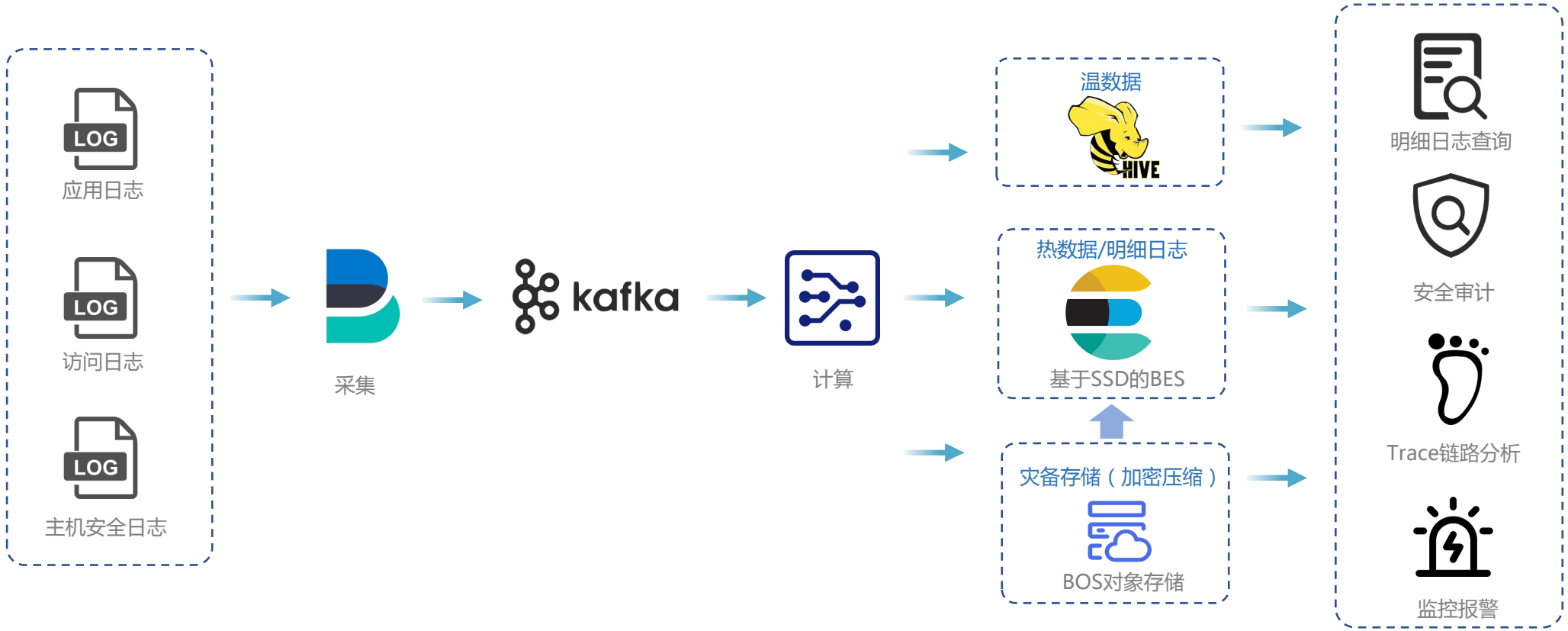
02

基于BOS冷热分离架构的应用实践

案例一：超大规模安全日志实时分析与明细查询 – 某金融公司

业务数据落地架构统一，日志保留天数增加5倍

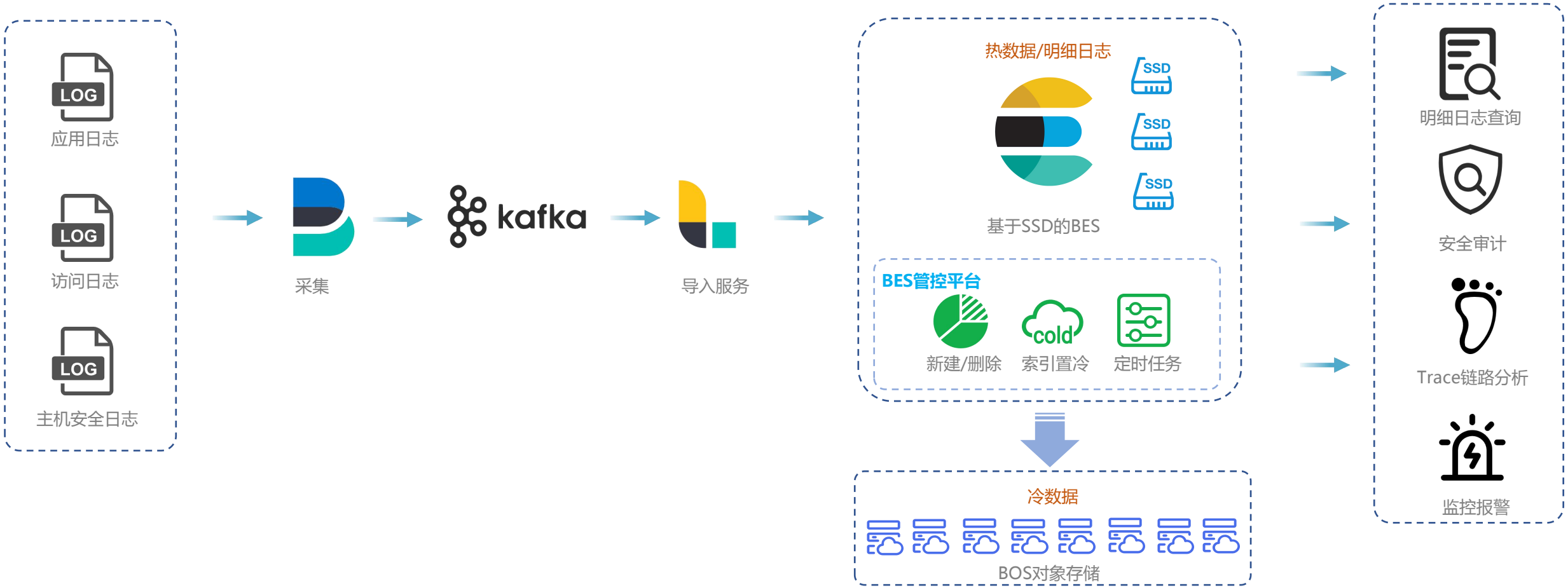
客户原有日志分析架构


LOG
应用日志
LOG
访问日志
LOG
主机安全日志
采集
kafka
计算
温数据
HIVE
热数据/明细日志
基于SSD的BES
灾备存储（加密压缩）
BOS对象存储
明细日志查询
安全审计
Trace链路分析
监控报警

客户痛点

- 明细日志如果全部依赖ES, 存储成本高昂，所以只考虑将热数据放到ES中
- 查询几乎全部是模糊搜索，Hive查询冷数据速度非常慢（半小时以上）
- 备份到 BOS的数据到ES的恢复速度无法满足业务预期
- 系统复杂度高，接口不统一，业务方使用成本高
- ……

新的日志分析架构


LOG
应用日志
LOG
访问日志
LOG
主机安全日志
采集
kafka
导入服务
热数据/明细日志
SSD
SSD
SSD
基于SSD的BES
BES管控平台
新建/删除
cold
索引置冷
定时任务
冷数据
BOS对象存储
明细日志查询
安全审计
Trace链路分析
监控报警

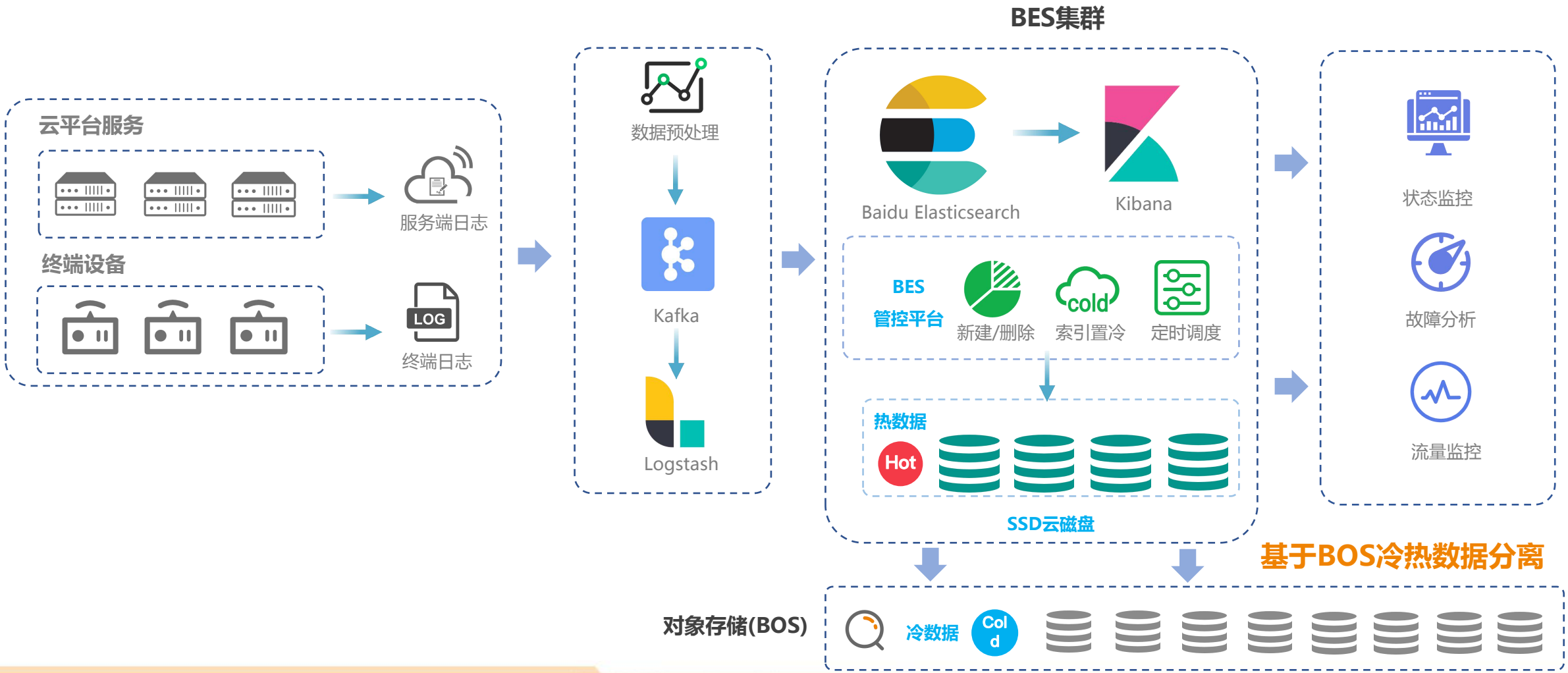
客户收益

- 突破单节点存储上限，**在SSD使用率30%的情况下（约400G），存储10TB+索引**
- 热数据写入SSD，**导入性能不受影响**
- 冷数据无需额外操作即可检索，**10TB+冷数据查询在1分钟内返回结果**
- 业务层不感知对象存储(BOS)，**查询不需要做任何改动**
- 调度平台提供**索引自动置冷，零脚本化**

案例二：超大规模IOT数据存储与可视化分析 - 制造业公司

保证IOT数据实时并发导入，冷热分离大幅降成本

客户应用架构


BES集群
云平台服务
服务端日志
终端设备
LOG
终端日志
数据预处理
Kafka
Logstash
Baidu Elasticsearch
Kibana
BES
管控平台
新建/删除
cold
索引置冷
定时调度
热数据
Hot
SSD云磁盘
状态监控
故障分析
流量监控
基于BOS冷热数据分离
对象存储(BOS)
冷数据
Cold

客户收益

✓ **性能**

热数据的导入/查询无影响， P99查询延时增加 → 导入性能 **保持不变**

✓ **成本**

基于对象存储冷热数据分离，存储成本大幅降低 → 存储成本降低 **80%+**

✓ **易用性**

Lucene级别扩展，业务端无需适配 → 业务逻辑改造 **0%**

03

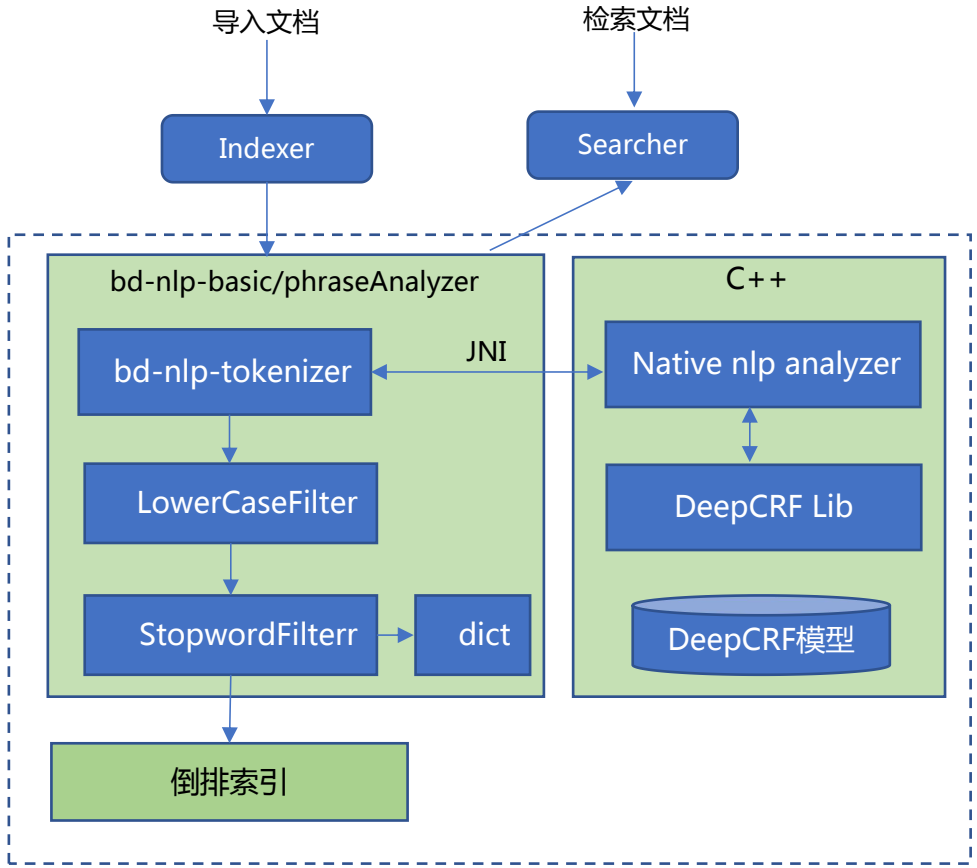
Elasticsearch与NLP特性增强实践

百度智能云ES-NLP中文分词

- 依托百度NLP在中文搜索分词领域十几年的海量数据积累
- 深耕锤炼分词模型，支持个性化/垂直行业领域定制
- 两种分词力度：基础力度和短语力度

(bd-nlp-basic 和 bd-nlp-phrase)

https://cloud.baidu.com/doc/BES/s/Lke3o72jg



NLP中文分词——效果对比

ik VS 百度NLP

- **语义理解对比**

ik：明天国家公祭日 >（明，天国，家公，祭日）

百度NLP：明天国家公祭日 >（明天，国家，公祭日）

- **分词冗余度对比**

ik： 维修基金 >（维修基金，维修， 维， 修， 基金， 基， 金）

百度NLP： 维修基金 >（维修， 基金）

NLP中文分词——模型编辑

原始文本：　云原生数仓团队和数仓团队

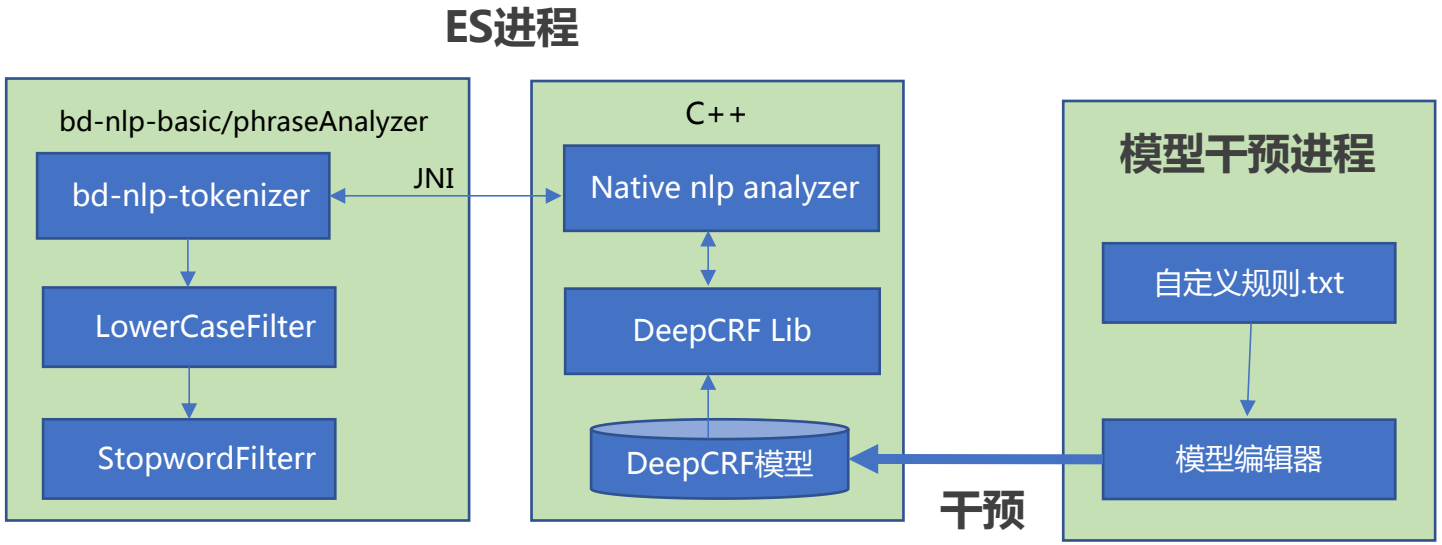
basic:　[云原生, 数, 仓, 团队, 和, 数, 仓, 团队]

phrase:　[云原生, 数, 仓, 团队, 和, 数, 仓, 团队]

干预规则：　云原生　[数仓　团队]

basic:　[云原生, 数仓, 团队, 和, 数, 仓, 团队]

phrase:　[云原生, 数仓团队, 和, 数, 仓, 团队]



百度智能云ES——Query权重分析

搜索的一个例子:


```
POST _search
{
    "query": {
        "match": {
            "content": "滑雪怎么样"
        }
    }
}
```

搜索解释：查询文档包含 `滑雪` 或 `怎么样`的文档

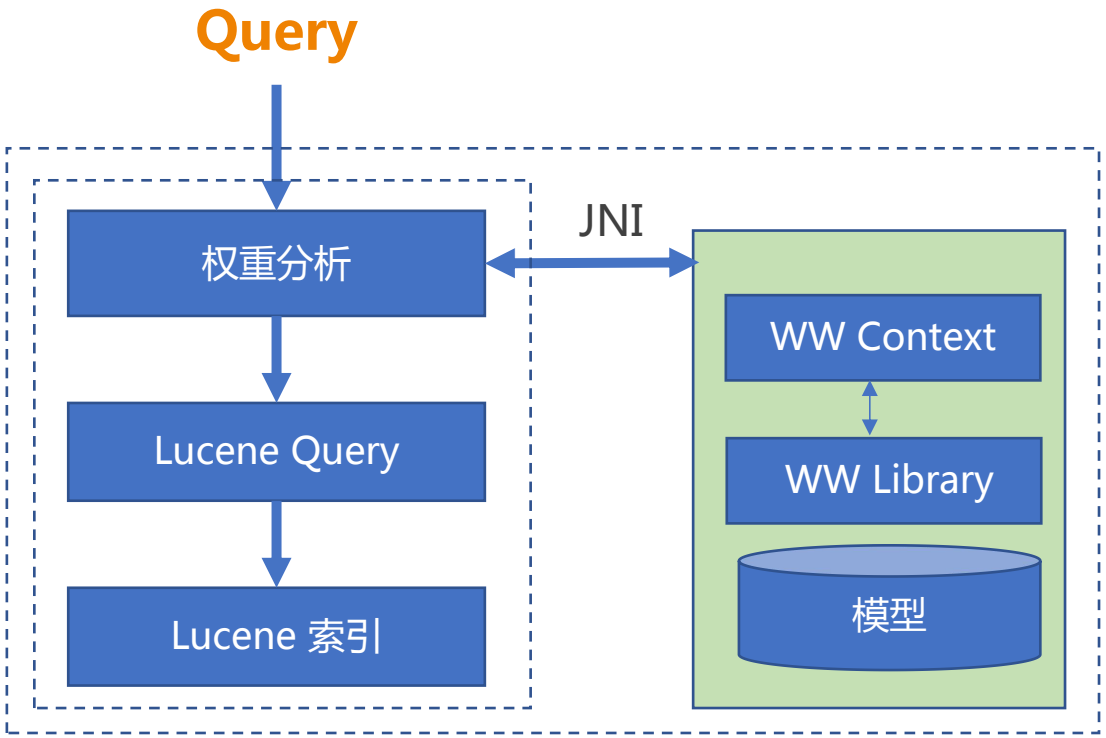
这里有两个问题：

1. 用户关注 的是`滑雪`相关内容
2. 仅包含 `怎么样` 的文档可能出现在TopN中

百度智能云ES——Query权重分析

WordWeight

- 基于双向LSTM+FC模型，模型效果和性能均处于业界领先地位
- 数百亿量级的数据上预训练
- 每个词汇的权重分为四档，从低到高分别为0，1，2，3



名次解释：Query文本中 **词汇的重要度** 即 **权重**

百度智能云ES——Query权重分析

用法一：所有词汇按照权重大小进行提升(boost)

```
POST index/_search
{
    "query": {
        "weighted_match": {
            "content": "滑雪怎么样"
        }
    }
}
```

Score = 3 * TermQuery<滑雪> + 1 * TermQuery<怎么样>

权重识别：滑雪-权重3，怎么样-权重1

召回结果：包含了“滑雪”的文档会被召回；包含“滑雪”和“怎么样”，文档召回且排序靠前；仅包含怎么样的文档权重会比包含了滑雪的文档权重低

百度智能云ES——Query权重分析

用法二：权重高的词参与过滤，权重低的词仅参与分数加成

```
GET  {index}/_search
{
   "query": {
     "weighted_match": {
            "content": {
                 "query": "滑雪怎么样",
                  "cutoff_weight": "2"
             }
         }
     }
}
```

权重识别：滑雪-权重3，怎么样-权重1

召回结果：包含了“**滑雪**”的文档会**被召回**；包含“滑雪”和“怎么样”，文档召回且排序靠前；仅包含**怎么样**的文档不会召回

04

未来展望

未来展望

- BOS的冷数据置热功能、单副本容灾考虑等
- 基于BOS的冷热数据分离架构 cache-block查询优化，感知不同的索引文件、不同的查询意图等
- 更多的NLP算子的集成，比如Query改写、Query扩展、反黄反暴力、信息摘要提取等
- ……


THANKS
时序
开源
自研
分布式
文档
Spark
HANA
SQL
OLAP
Oracle
Hadoop
MPP
DB2