中国平安 PING AN
保险·银行·投资

基于Hive/ES
金融大数据指标系统

沈百军 2016年10月 @QCon

Why

流动性不足

套利 价值 大小 相关

低风险 携带 动量

保险连接 质量

波动 事件风险

产品
(股票/基金/债券)

客户

承受能力

套利 择时 性别 购买力

年龄 相关

风险 资产

绝对收益 稳定性

可融资性 佣金

Why-目标要求

- **超级标签**系统，标签可无限制扩展，所有标签可联合Filter
- **超级Cube**，可无限制扩展维度和度量值
- **毫秒级别**的多维分析工具，用户体验好，查询快
- **水平扩展**，必须是分布式，最好开源
- **架构部署简单，维护成本低**
- 支持Sql查询，入门快

其他系统比较

	ElasticSearch	Kylin	Spark	HBase	SSAS Cognos
聚合度量值排序	√	X	√	X	√
维度&度量值扩展性	10W~50W	32个维度	10W	1维	有限个
条件明细筛选	√	X	√	X	X
水平扩展型	√	√	√	√	X
易用性	√	√	√	√	X
性能和用户体验	毫秒	毫秒	10~60s	毫秒	2~10s
高维聚合查询	X	√	√	√	X
数据实时性	流式写入	流式写入	流式写入	流式写入	X

方案

Hive

ElasticSearch

Schema

系统介绍

- 什么是指标系统
- 条件筛选 & 多维统计分析
- 指标定义和维护

➢ 什么是指标系统（3W）

WHO

- 对象：给谁打指标，必须有一个主题对象（如客户、股票、视频等）

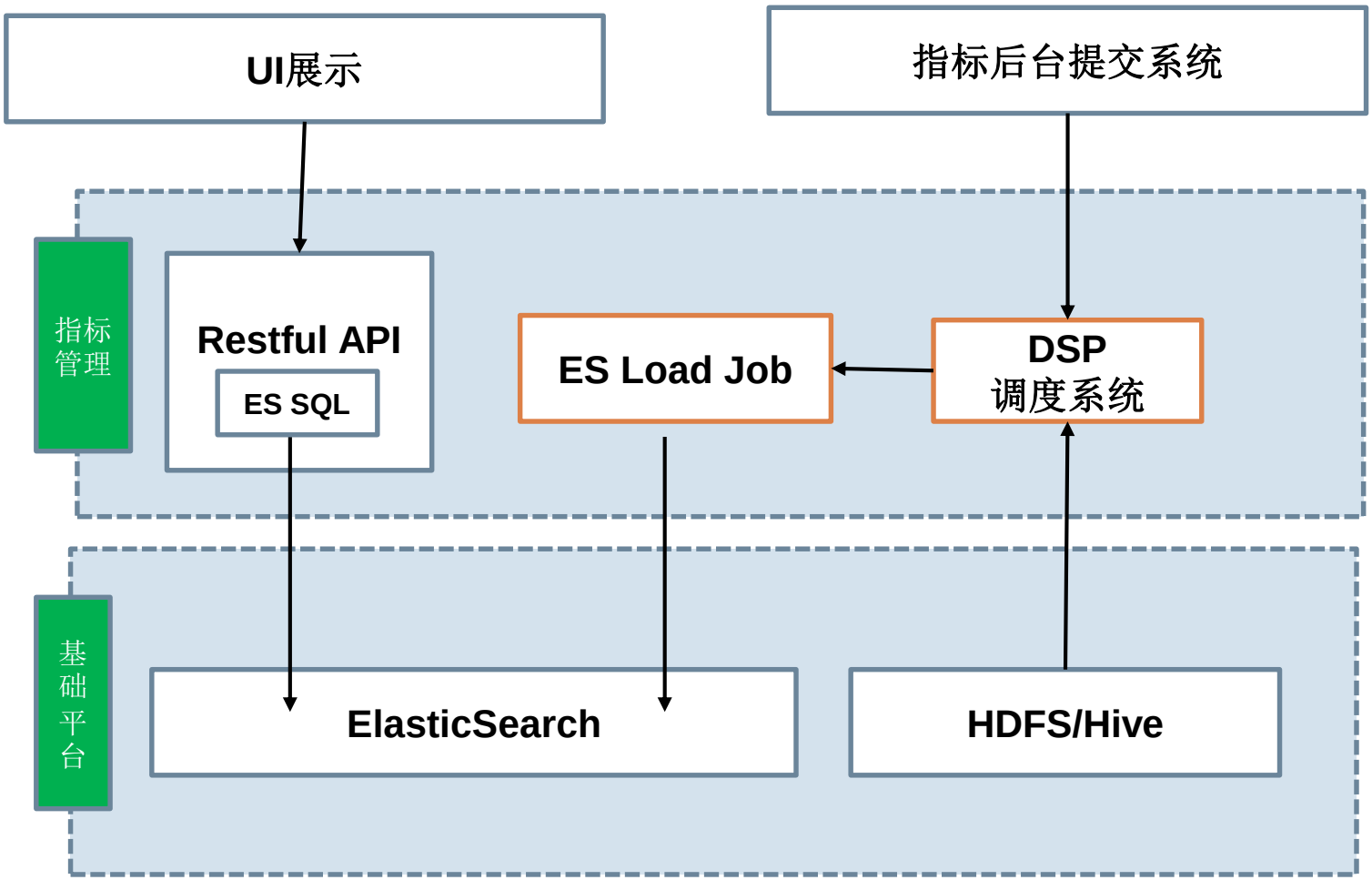
WHAT

- 给对象的各个属性，标签，度量值等

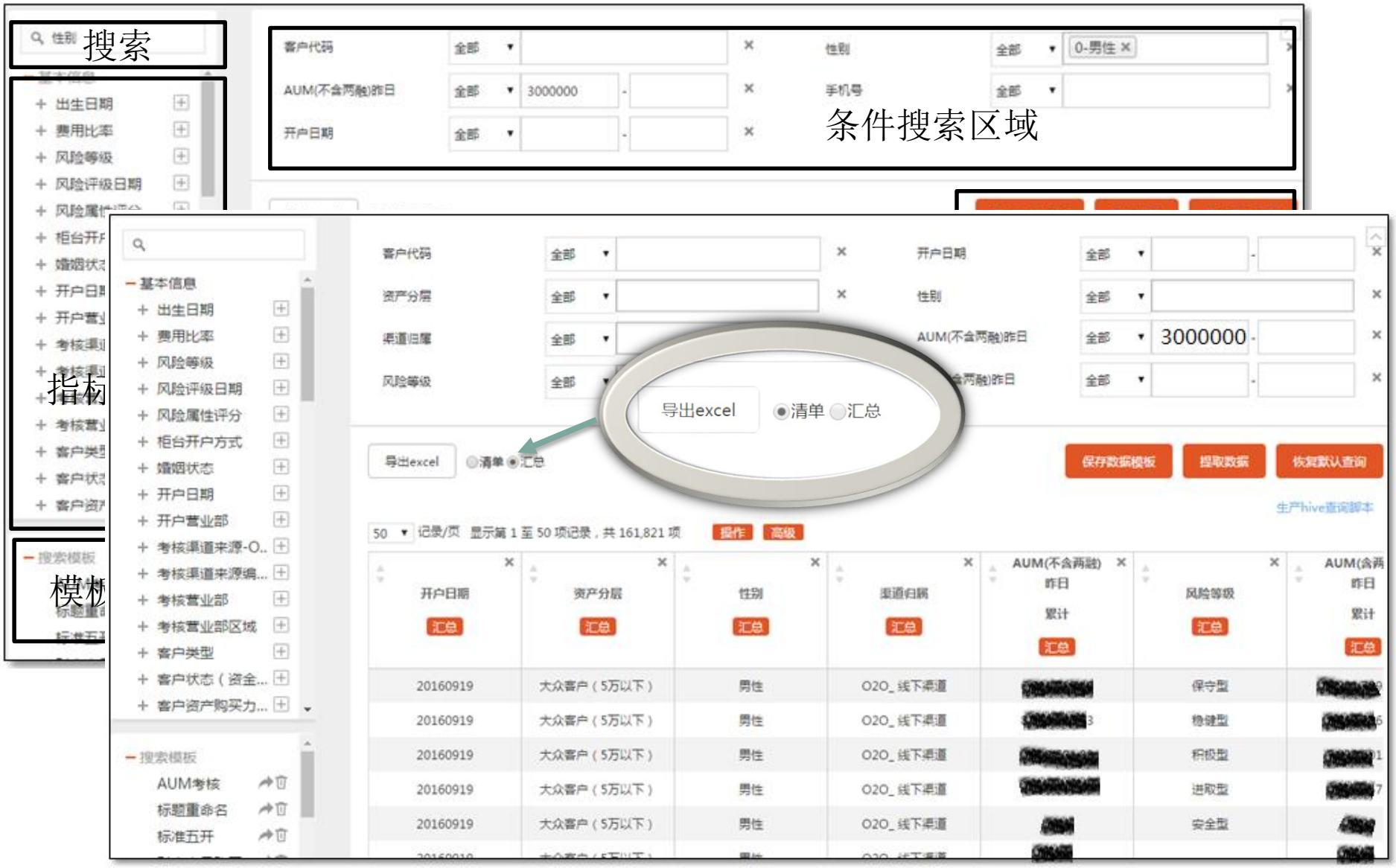
HOW

- 精准化营销
- 多维统计分析
- 千人千面

指标整体架构图

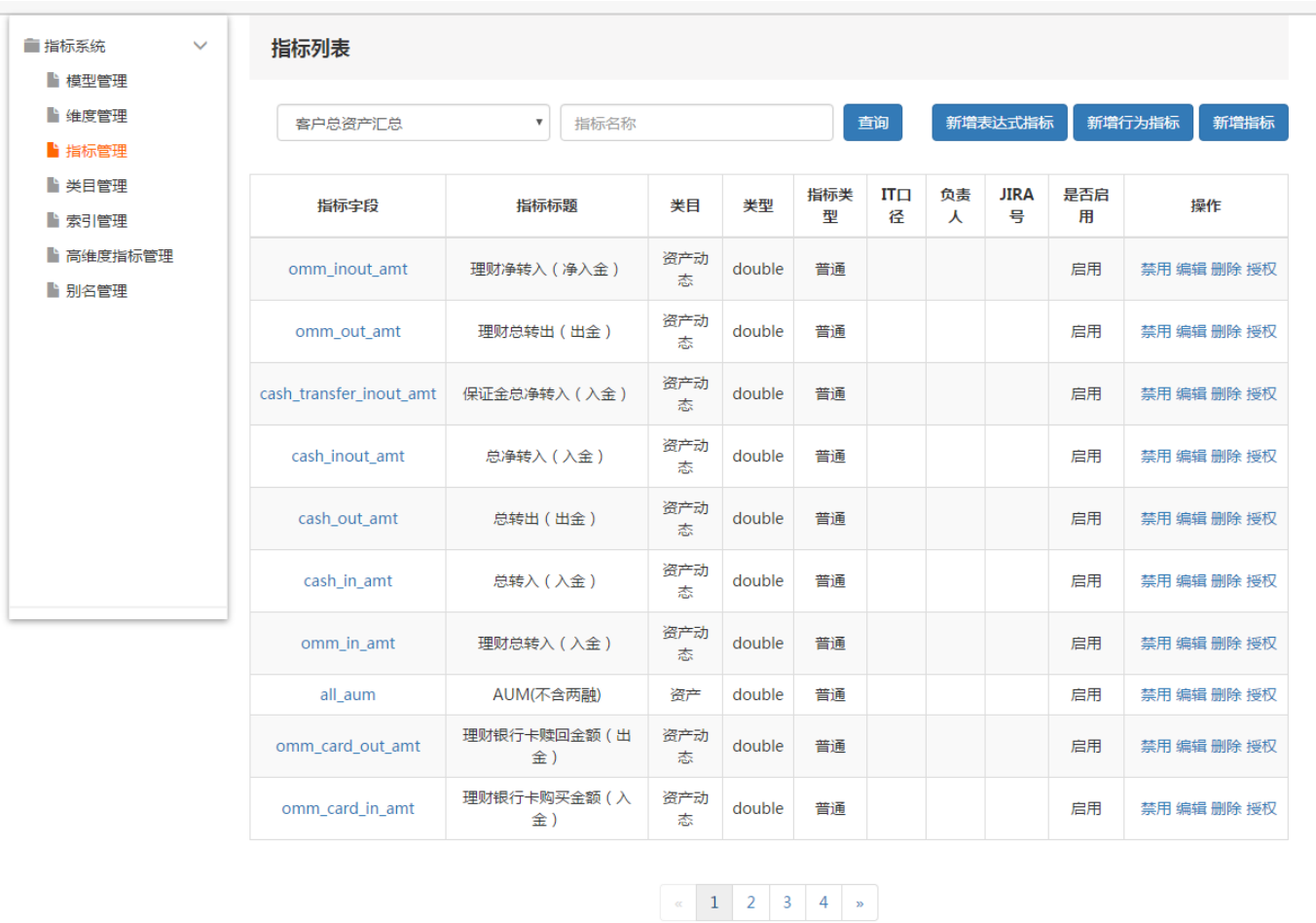

UI展示
指标后台提交系统
指标管理
Restful API
ES SQL
ES Load Job
DSP
调度系统
基础平台
ElasticSearch
HDFS/Hive

条件筛选和多维分析（Filter & Cube）

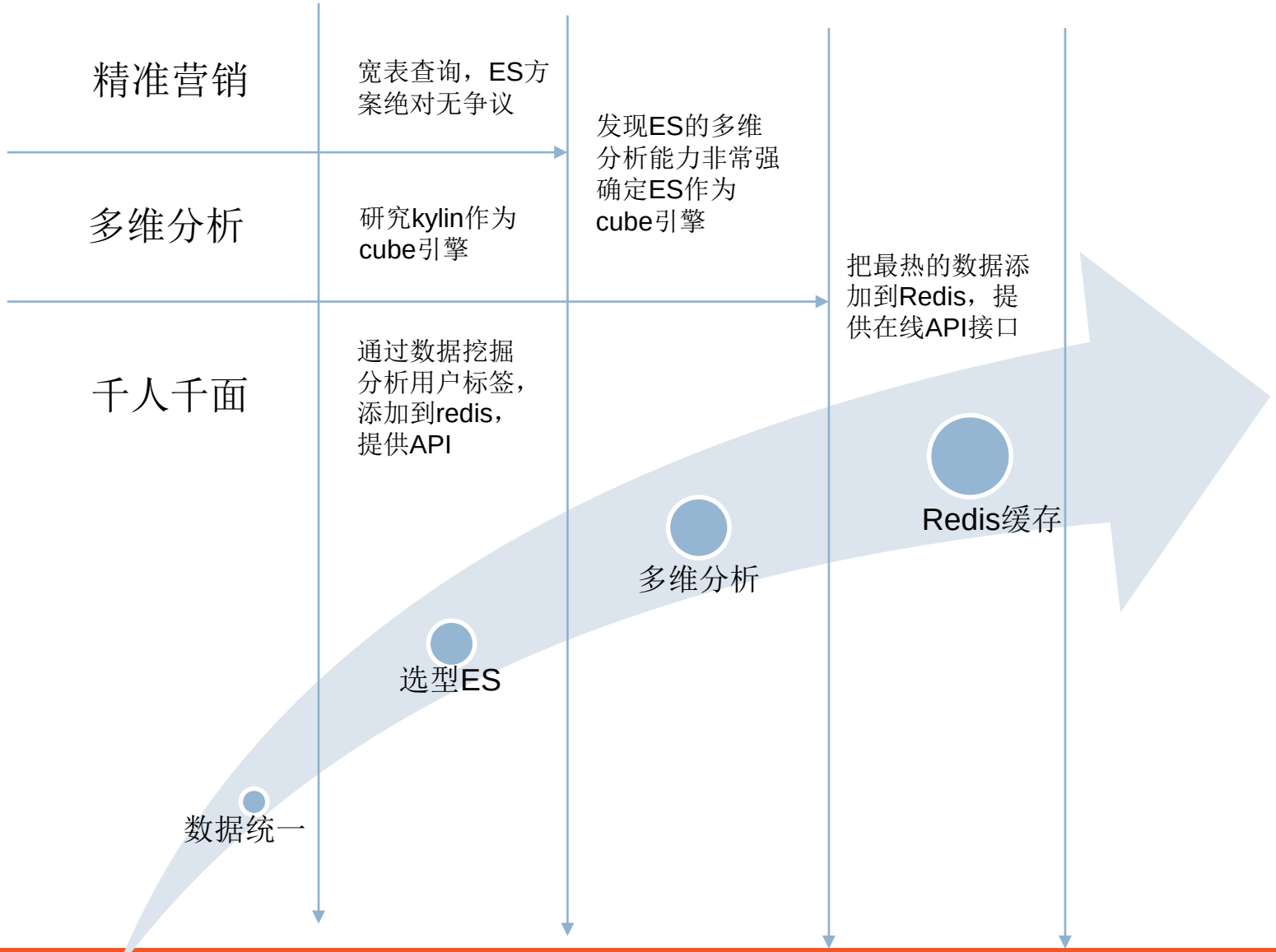


指标系统元数据定义

- 指标模型管理
- 指标管理
- 维度管理
- 其他辅助设置



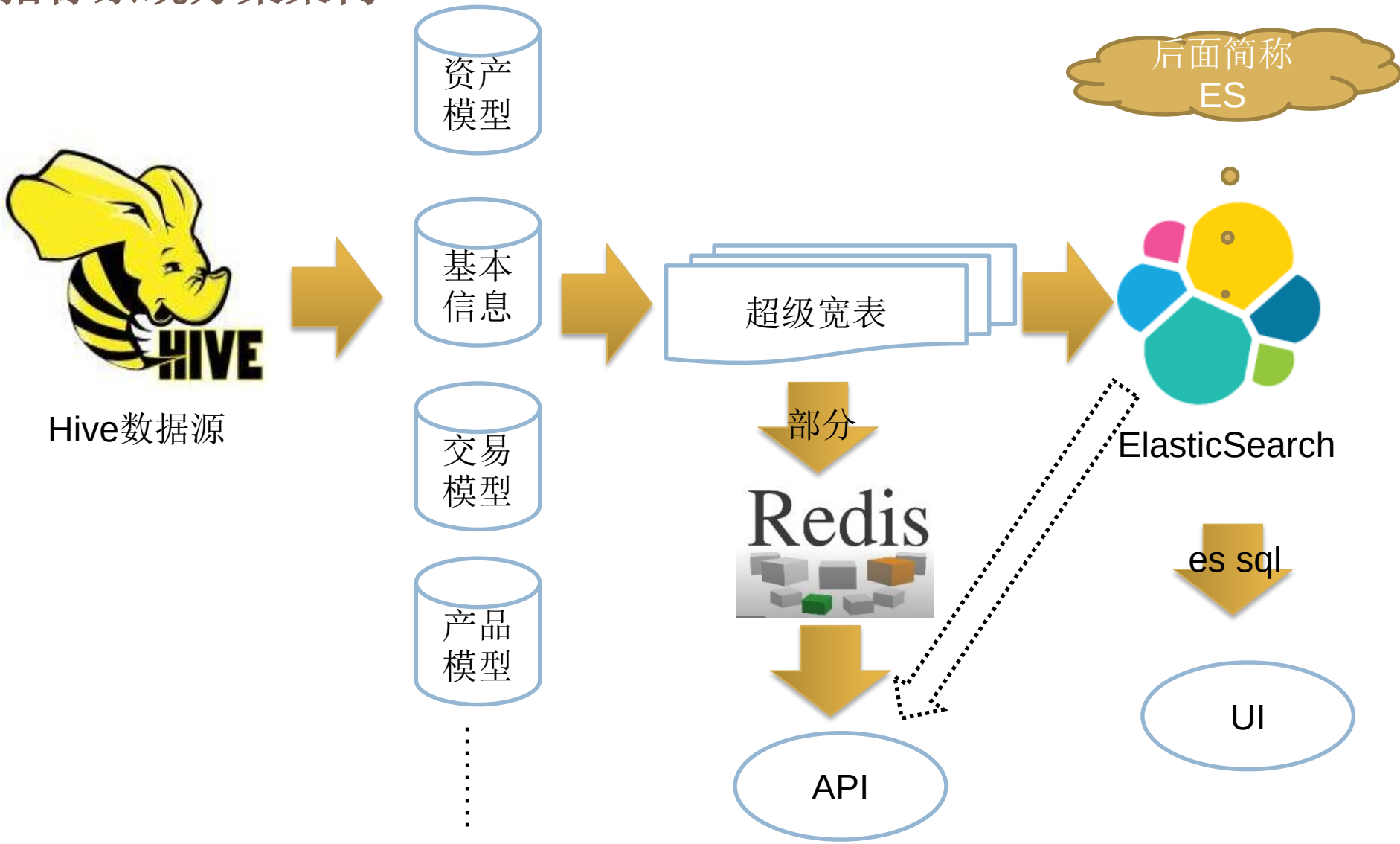
开发实践 ---- 摸石头过河


精准营销
多维分析
千人千面
宽表查询，ES方案绝对无争议
研究kylin作为cube引擎
通过数据挖掘分析用户标签，添加到redis，提供API
发现ES的多维分析能力非常强确定ES作为cube引擎
把最热的数据添加到Redis，提供在线API接口
数据统一
选型ES
多维分析
Redis缓存

系统技术架构

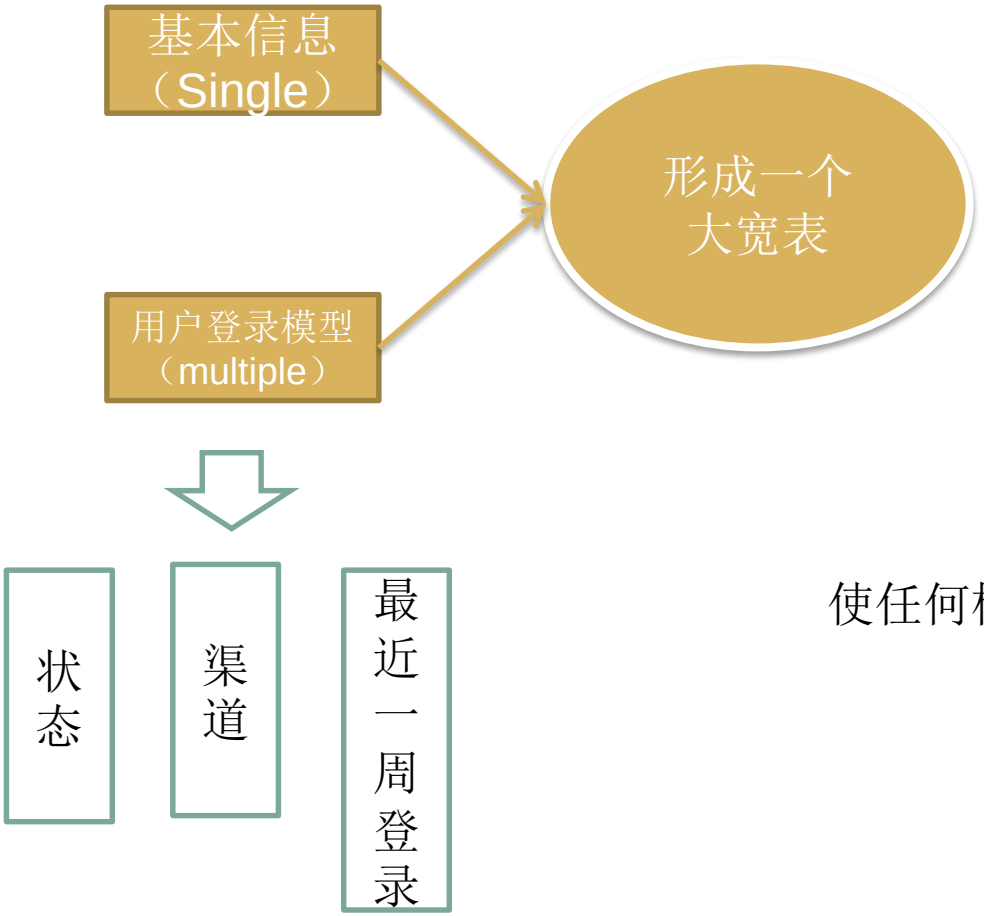
- 整体架构
- 指标元数据设计
  - 模型设计
  - 指标设计

指标系统方案架构


后面简称
ES
资产
模型
基本
信息
交易
模型
产品
模型
超级宽表
HIVE
Hive数据源
部分
Redis
API
ElasticSearch
es sql
UI

指标元数据设计---- 模型设计

局限性：大宽表，每个用户只能一条记录

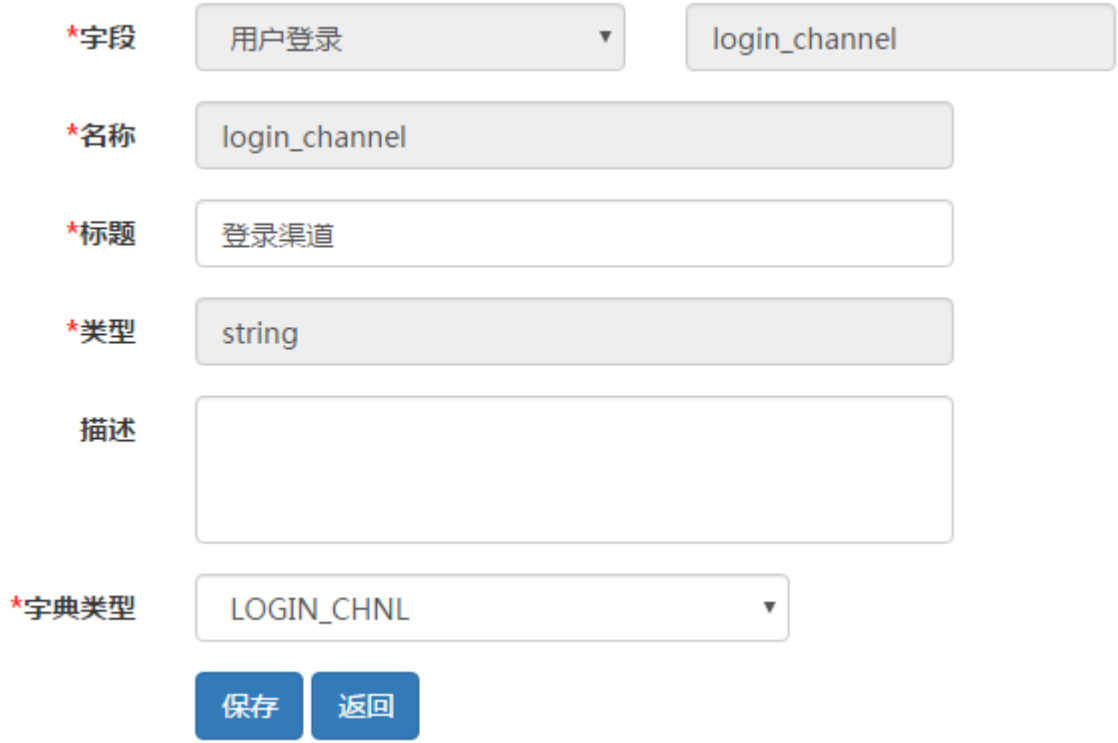


使任何模型，每个对象都变为一条记录

指标元数据设计---- 模型维度设计

- 模型设计

- 维度设计



指标元数据设计---- 指标设计

- 指标设计

- 扩展指标
  - Hive里面是Map字段
  - 设置维度后，每个渠道一个key
  - 存放在Map，ES是一个字段
- 添加指标时
  可以选择多个字段

指标元数据设计----其他设置

- **表达式指标**
  - 用户可以根据自己的逻辑添加指标
- **模板定义**
  - 根据设计的查询分享给其他用户
- **索引设置**
  - 索引是有限资源
- **指标别名设置**

多项聚合

Use group by (fieldName),(fieldName, fieldName)
SELECT * FROM account GROUP BY (gender, state, age),(state),(age)

分区间聚合

put fieldName followed by your ranges
SELECT COUNT(age) FROM bank GROUP BY range(age, 20,25,30,35,40)

地理查询

GEO_DISTANCE_RANGE(center,'1m','1km',100.5,0.50001)
GEO_DISTANCE(center,'1km',100.5,0.5)

优化和填坑

- ElasticSearch优化和填坑
  - ElasticSearch优化点
  - 填坑大行动
- 核心计算调度

ElasticSearch原理

- ✓ 时间复杂度：O(N) ->O(logN)-> O(1)
- ✓ 文档(Document) 单词(Word) 倒排索引(Inverted Index)
- ✓ Bool filter & And/Or/Not filters(Bitset & non-bitset)
- ✓ Filter execution order
- ✓ Instance & Shard

	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8
gender-男		√		√		√		
Level-高净值	√	√	√		√	√	√	√

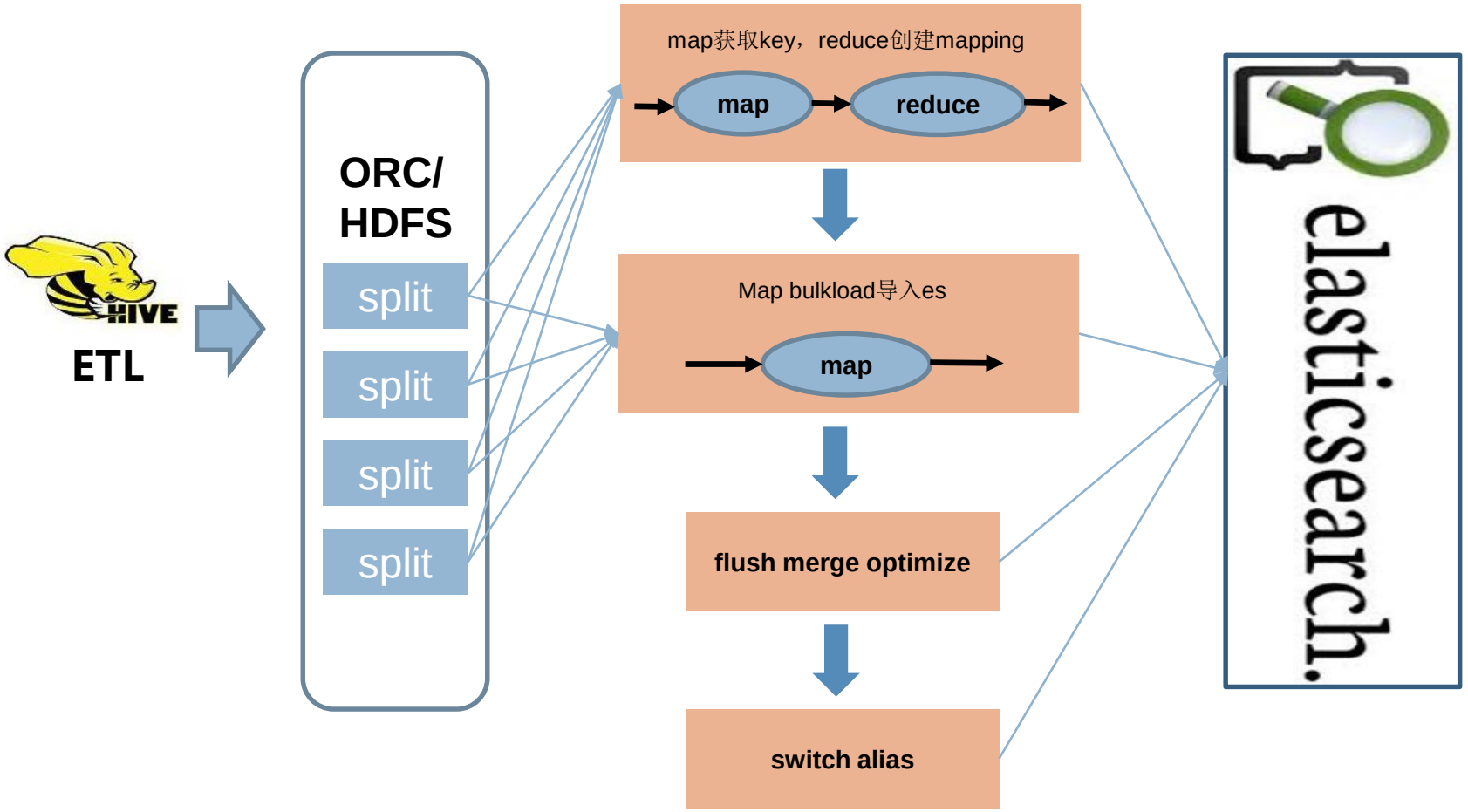
O(1)

bitset - and		√				√		

ElasticSearch--blukload

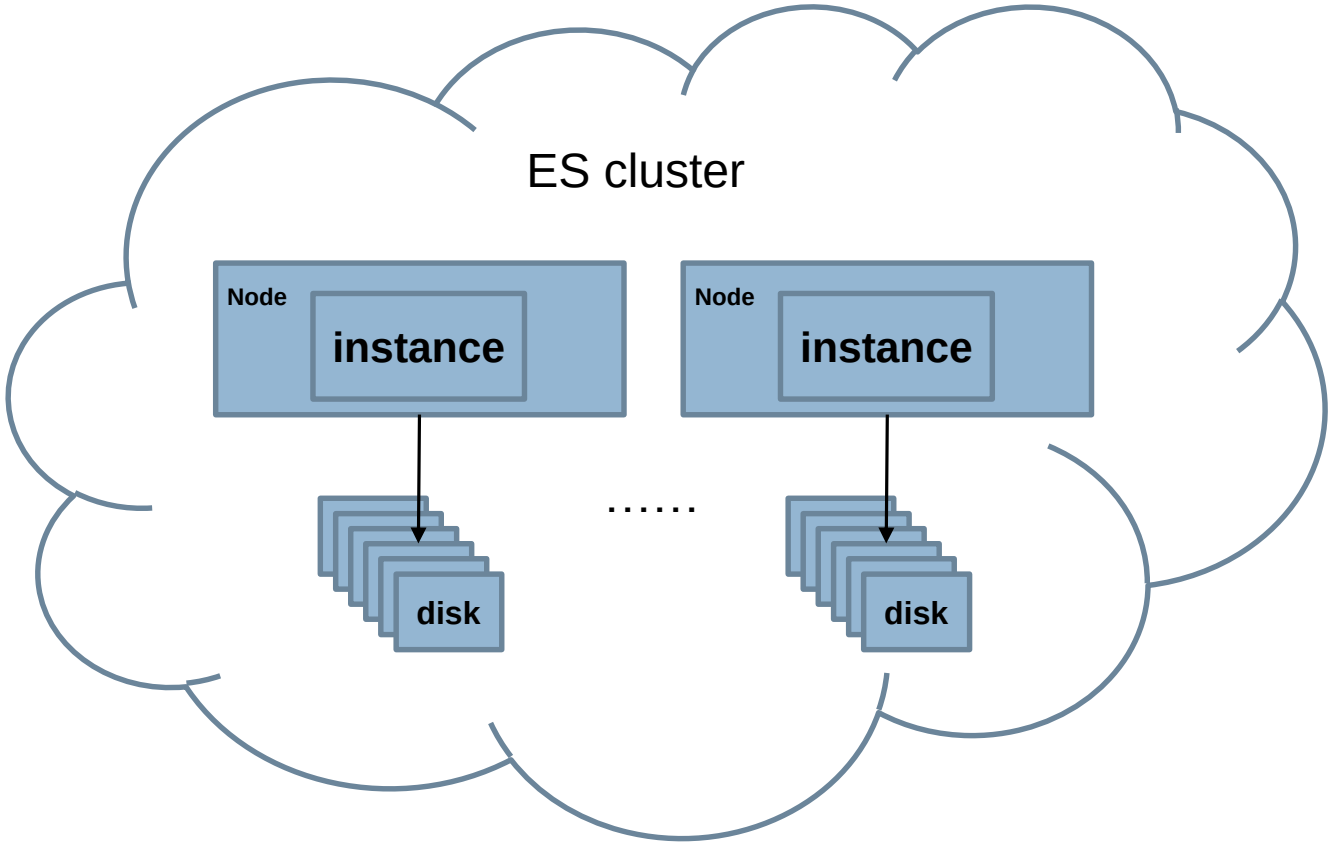
11万字段，两万个索引字段，每天要7点前导入es，核心将ORCFile里的数据导入ES

虽然ES可以自动创建mapping，但创建的效率非常低，所以希望能一次性的把mapping创建



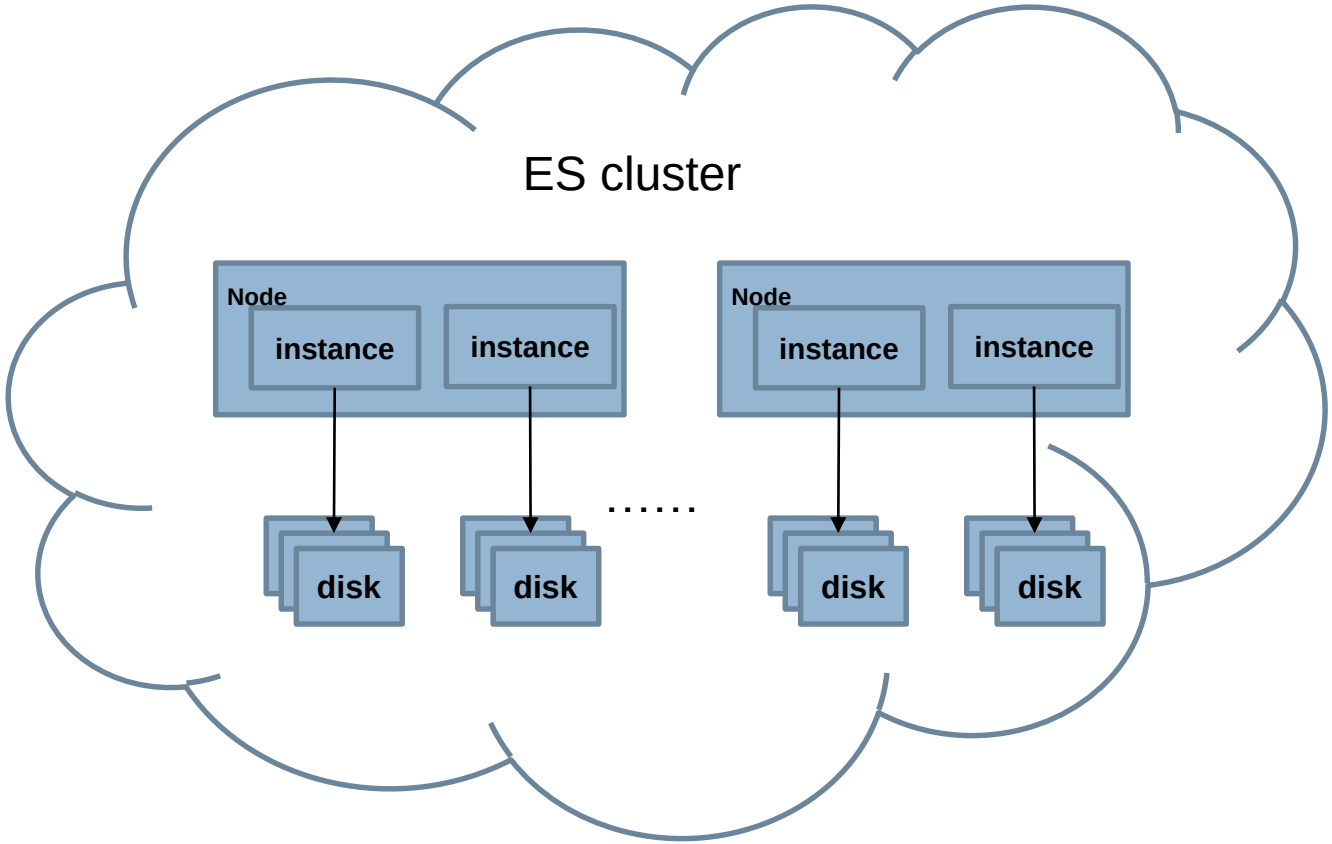
ElasticSearch优化—IO优化

- 一台机子多个实例，硬盘裸盘
- 多硬盘提高IO，实例之间硬盘不共享
- SHARD个数等于硬盘个数，最大化利用硬盘IO



ElasticSearch优化—IO优化

- 一台机子多个实例，硬盘为裸盘
- 多硬盘提高IO，实例之间硬盘不共享
- SHARD个数等于硬盘个数，最大化利用硬盘IO



ElasticSearch优化—IO优化

- Bulkload 的瓶颈在IO
  - 购买32块SSD硬盘和96块普通一盘比较

（以下数据为1000w记录，11万字段，1w索引，索引500G）

	32块SSD硬盘	96块普通硬盘
Import time	80分钟	90分钟
Merge time	20分钟	30分钟
Hard Disk Load	40%	80%

- 结论：时间上面没有明显提高，但load提高一倍
  - 进一步优化方法，购买更多SSD硬盘
  - 增加机器数量
  - 减少索引个数，优化索引

ElasticSearch优化

- 尝试G1 GC，提高吞吐能力
- 一台机子多个实例
- 多硬盘提高IO，实例之间硬盘不共享
- SHARD个数等于硬盘个数，最大化利用硬盘IO
- bootstrap.mlockall设置true，防止swap
- 使用bulk load，加大threadpool.bulk.queue_size=1024 避免数据丢失
  - Map=90
- Index settings disable _all
- 使用index alias代替index name

填坑：ElasticSearch TopN—慎用terms size

ElasticSearch terms size聚合的时候，如果维度基数大于size，聚合结果求TopN可能是近似值

shard A	shard B	shard C
A 30	B 12	E 15
B 25	C 10	B 10
C 4	D 8	D 8
D 3	E 7	C 6
E 2	A 2	A 1

求top3

shard A	shard B	shard C
A 30	B 12	E 15
B 25	C 10	B 10
C 4	D 8	D 8

结果

B	47
A	30
D	16

正确结果：

B	47
A	33
E	24

结论：

- terms size & shard size 必须超过维度元素个数

填坑：禁止高维组合查询 & distinct 不精确

- 高维查询
  - 当高维组合查询时，es的heap & load 就会增高，会拒绝服务
  - 高维在业务应用中其实不多，无业务含义，应该禁用（kylin 对高维支持很好）
- Distinct 使用 Hyper loglog 算法，有一定误差

社区贡献

- 原因：

  如果terms字段为null值，group by无法进行统计显示

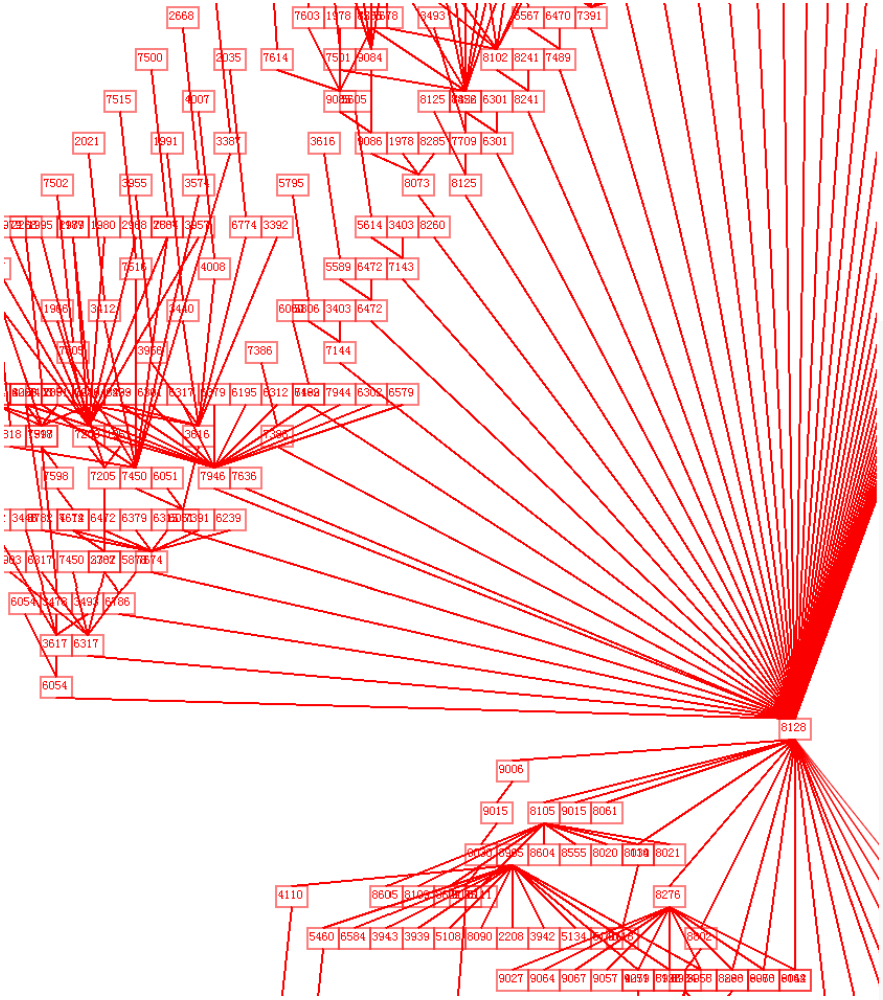
- ElasticSearch-sql: pull-267

  对terms聚合添加missing和order by功能

未来发展

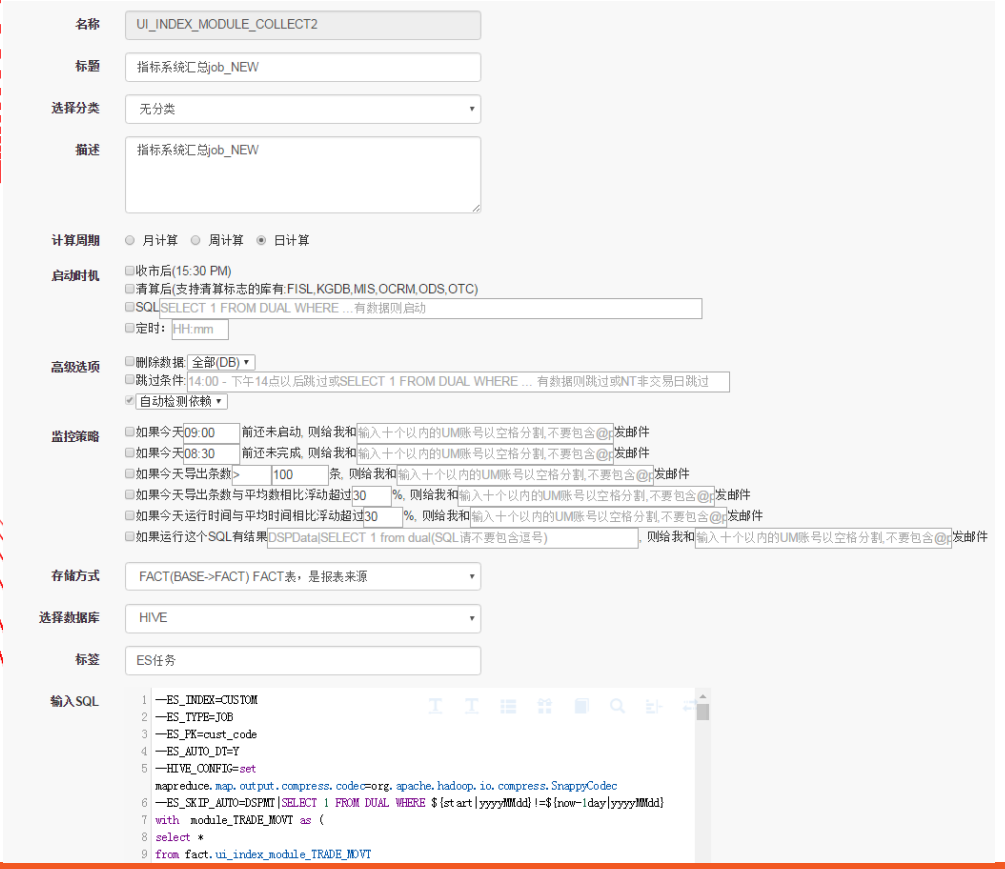
- 优化ElasticSearch bulkload，缩短导入时间
- 对象记录唯一性的缺陷，支持多记录和跨索引的Join
- 指标系统抽象化，可以任何对象；目前是客户指标系统
- 能够和多维分析UI工具打通
- 支持实时数据查询和分析
- 有望推向社区，开源

核心计算调度（主题外补充）



➢ 自动依赖

- ✓ 通过Sql分析，形成一个有向无环图



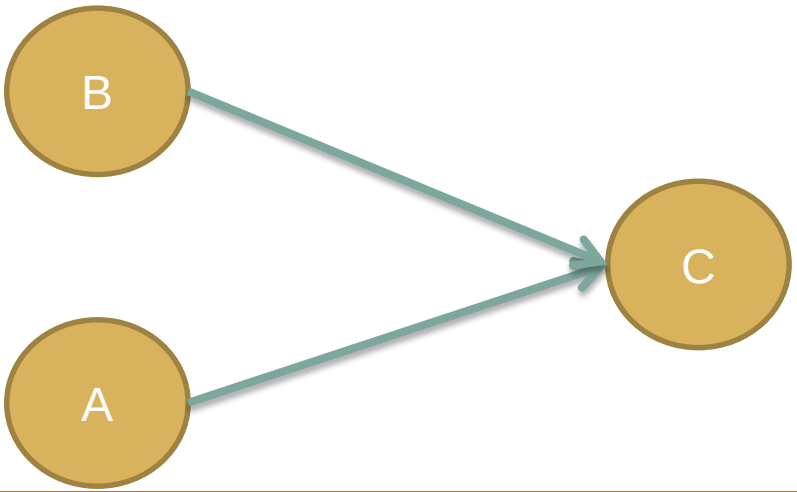
核心计算调度----sql dependent

- 平安95% 的job使用 sql 表示
- Sql 语义分析Job 依赖

A: select f1 from table1

B: select f2 from table2

C: select a.f1 + b.f2 from a join b



中国平安 PING AN
保险·银行·投资

谢谢