

# Druid和Kylin

## 在美团点评的选型与实践

高大月 2017-08-05



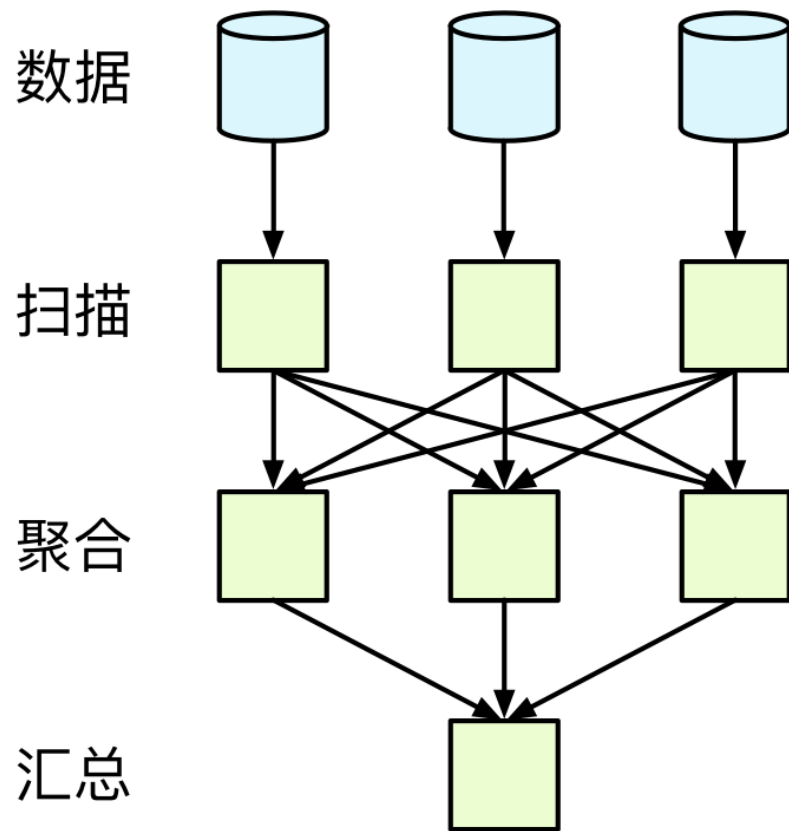
# 提纲

---

- **OLAP引擎选型**
- Druid在美团点评的实践
- Kylin在美团点评的实践

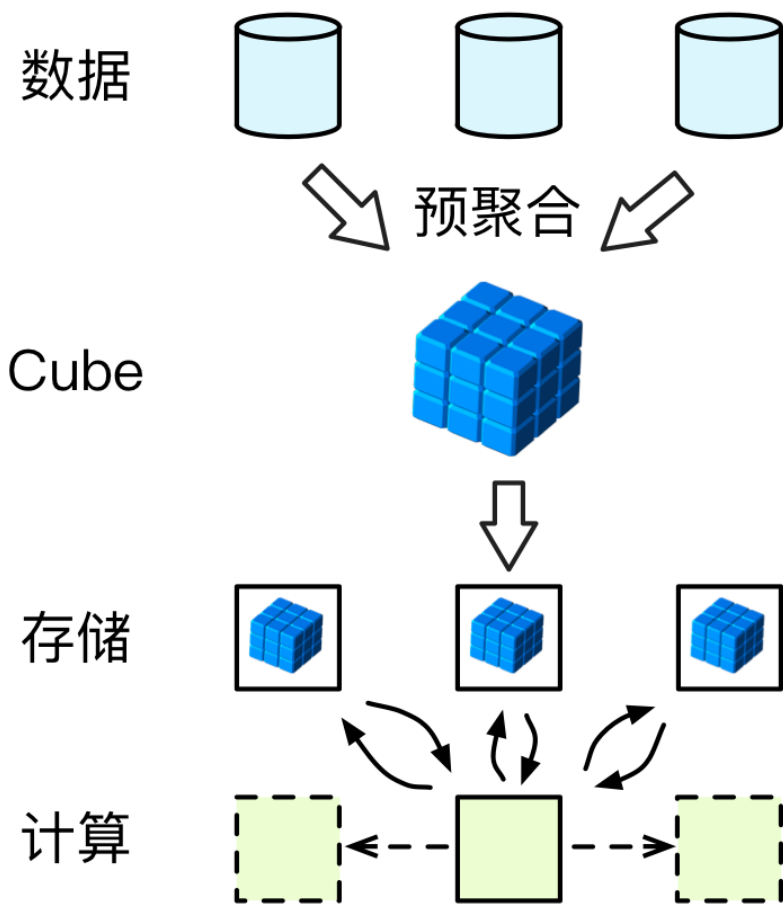
# ROLAP (Presto / SparkSQL)

---



- 优势
  - 支持任意的SQL表达
  - 无数据冗余和预处理
- 不足
  - 大数据量、复杂查询下分钟级响应
  - 不支持实时数据
- 适用场景
  - 对灵活性非常高的即席查询场景

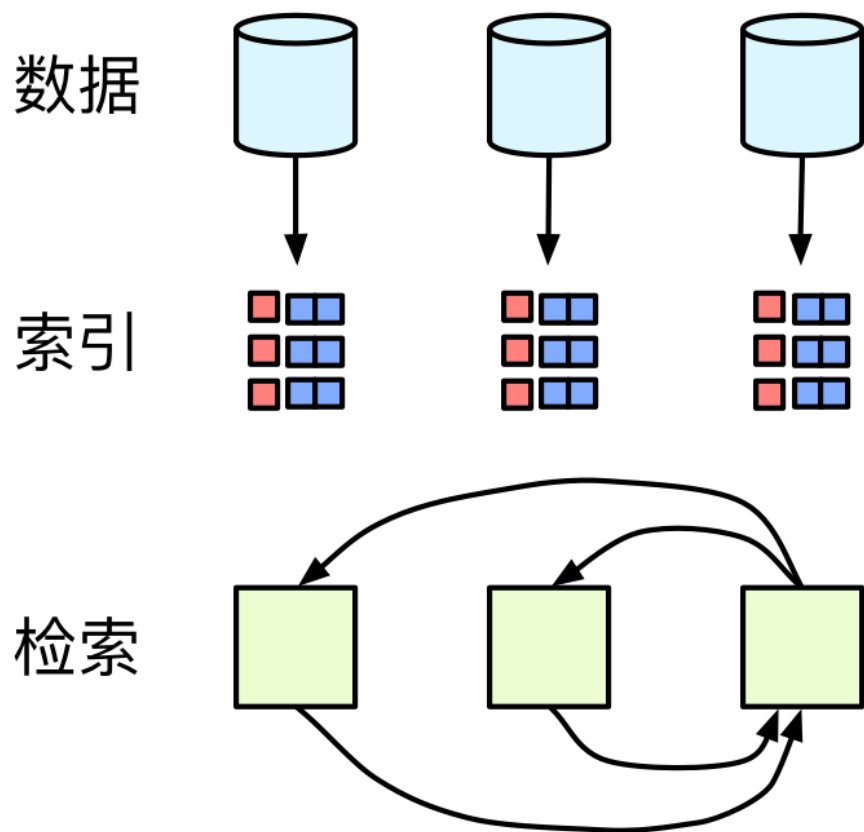
# MOLAP (Kylin, Druid)



- 优势
  - 支持超大原始数据集
  - 高性能、高并发
- 不足
  - 不支持明细数据查询
  - 需要预先定义维度、指标
- 适用场景
  - 对性能要求非常高的OLAP场景

# Search Engine (ES)

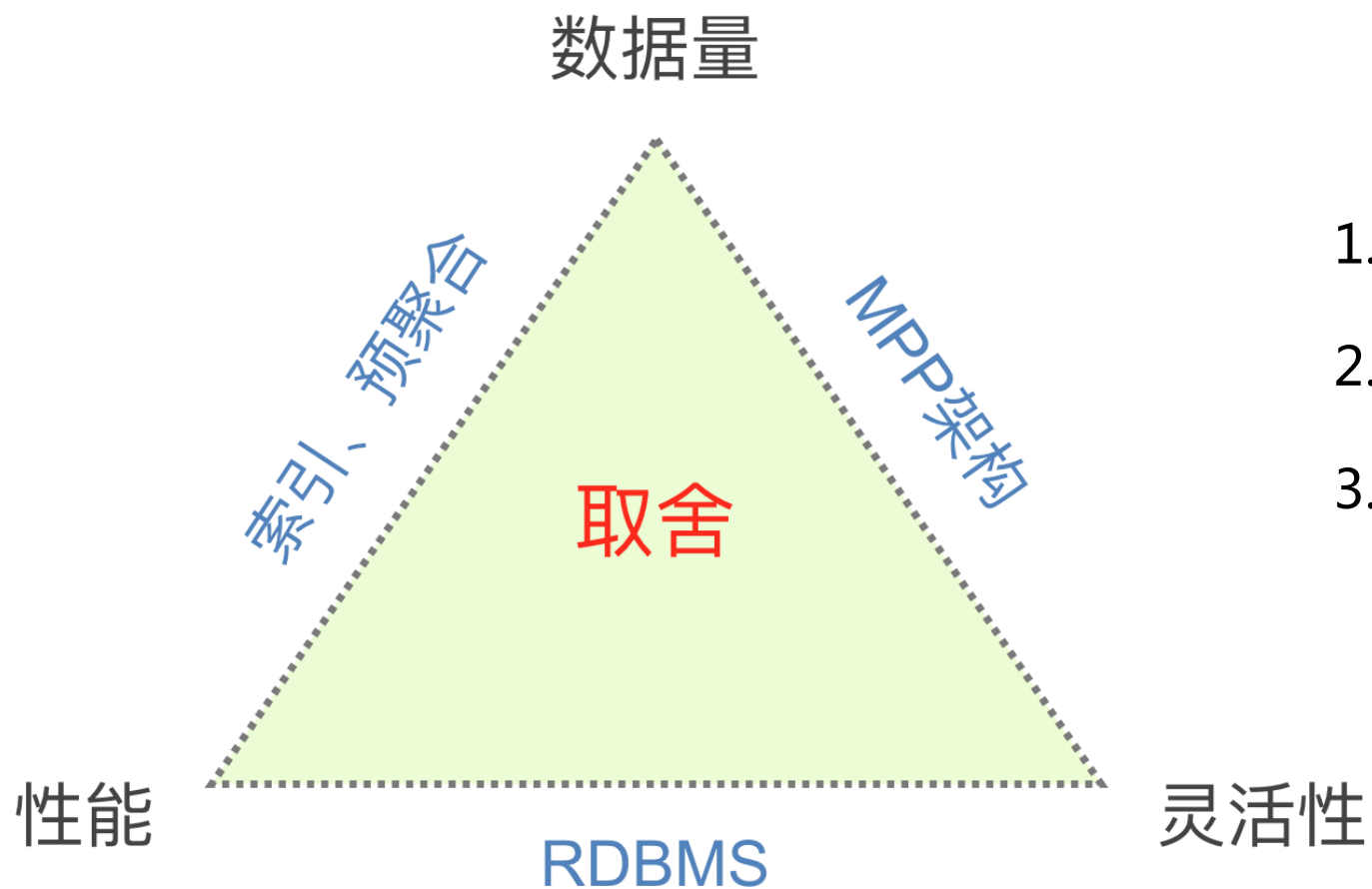
---



- 优势
  - 强大的明细检索功能
  - 同时支持实时与离线数据
- 不足
  - 大数据量、复杂查询下分钟级响应
  - 不支持Join、子查询等
- 适用场景
  - 中小数据规模的简单OLAP分析的场景

# 选型建议

---



1. 没有银弹
2. 了解不同架构/技术的取舍
3. 根据业务特点进行选择

# 美团点评OLAP场景的特点

---



# 为什么选择Kylin和Druid?

---

	Presto / Spark	Kylin	Druid	ES
亚秒级响应				
高并发				
百亿数据集				
精确去重计算				
SQL接口				
离线				
实时				



# 提纲

---

- OLAP引擎选型
- **Druid**在美团点评的实践
- Kylin在美团点评的实践

# Druid使用概况

---

- 定位：实时OLAP引擎
- 支撑业务：广告、风控、算法等
- 单集群40台物理机，100个Datasource，索引存储20 TB
- 每日从Kafka摄入百亿条消息
- 每日查询量超150万次，TP99时延~1秒

# Druid硬件/JVM配置

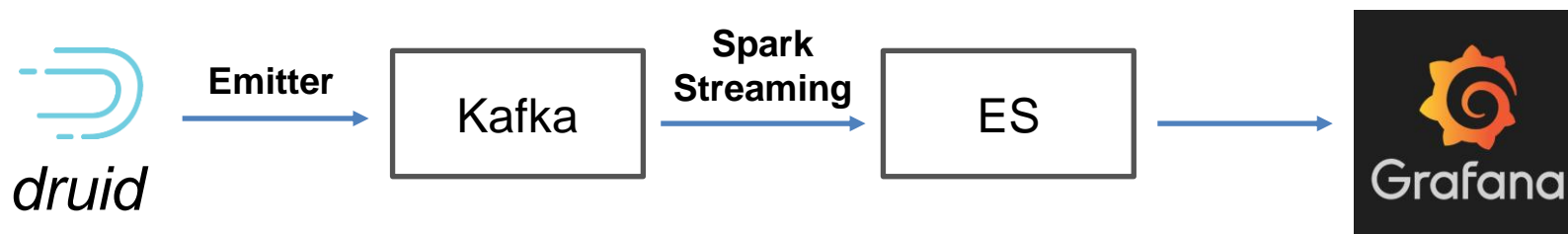
---

角色	硬件配置	JVM配置
Coordinator/Overlord	2 x 8g VMs	6g Heap
Broker	8 x 8core 16g VMs	11g Heap, 3g Non-Heap
Historical	19 x 40core 128g 12disk 物理机	12g Heap, 10g Non-Heap
MiddleManager	19 x 40core 128g 物理机	8 x 6g Heap Peons
Tranquility Clients	2 x 40core 128g 物理机	2~3g per JVM

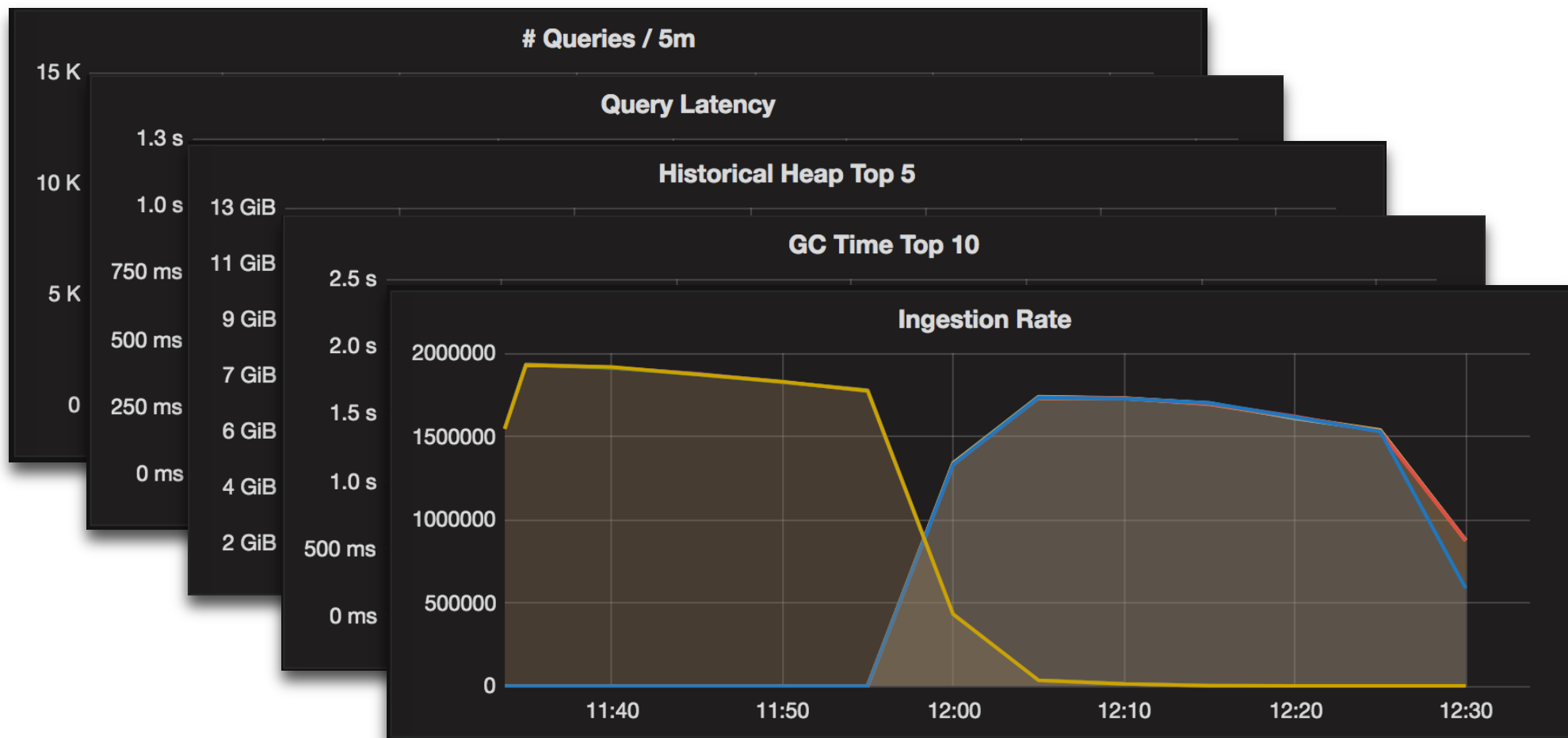
# Druid监控

---

- 需要监控哪些指标？
  - 业务侧：DataSource粒度，例如QPS、Latency、Ingestion Rate等
  - 平台侧：集群/节点粒度，例如CPU，I/O，JVM等
- 监控数据如何使用？
  - Dashboard、多维分析（OLAP）、慢查询分析（明细）
- 方案




# Druid监控 (Dashboard)



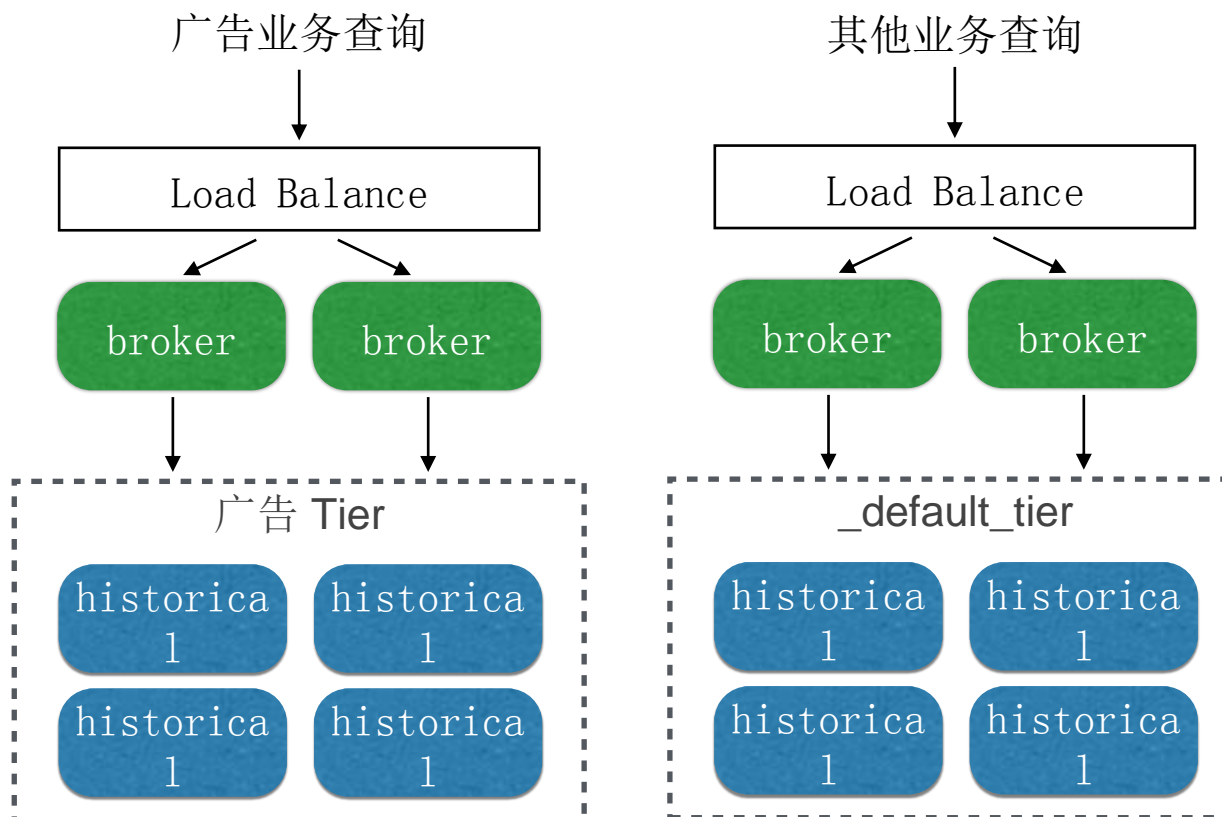
# Druid监控（多维分析）

---

Filter	→	Query	metric:"query/time" AND service:broker		
Agg	→	Metric 	Count		
Grouping	↗	Group by	Terms	remoteAddress	▸ Top 10, Order by: Doc Count
	↘	Then by	Date Histogram	es_timestamp	▸ Interval: auto

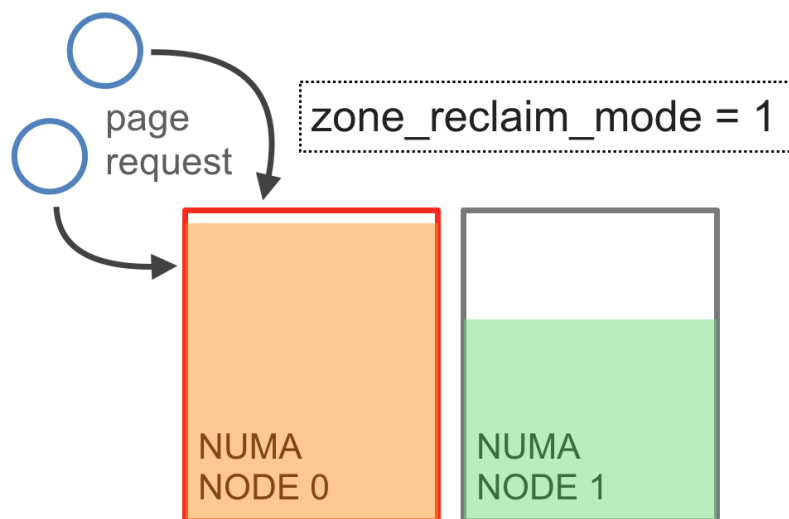
# 资源隔离

- 背景：不同业务的workload和稳定性要求不一样
- 目标：实现业务线粒度的隔离
- 可选方案
  - 多集群
  - 单集群分组隔离



# 踩过的一些坑 (1/2)

- 现象：Historical节点sys cpu飙高，集体掉线
- 原因：NUMA架构启用了zone reclaim mode，造成direct page scan



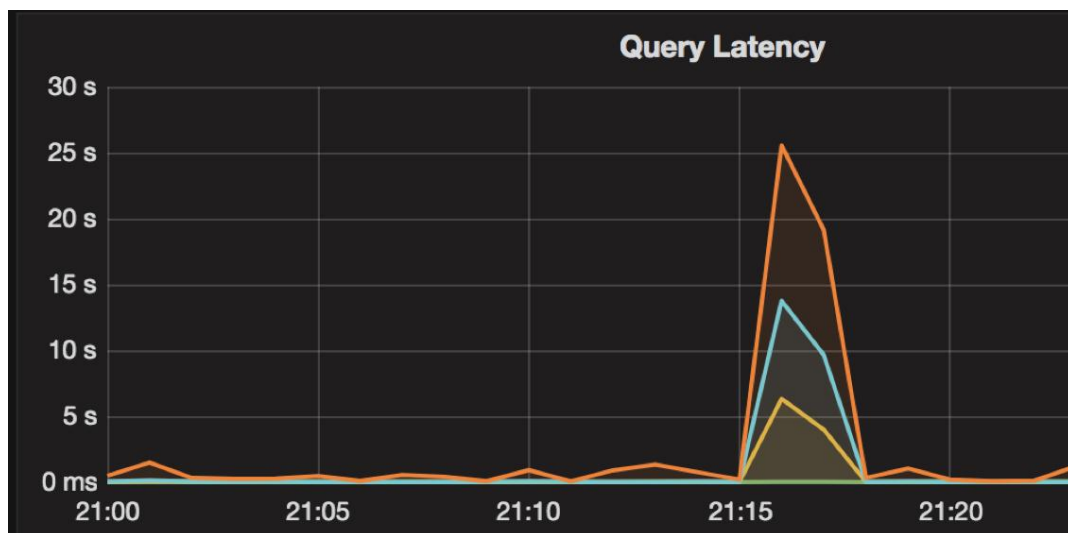
pgfree/s	pgscank/s	pgscand/s	pgsteal/s	%vmeff
89.00	0.00	2781632.00	0.00	0.00
87.00	0.00	2793440.00	0.00	0.00
86.00	0.00	2780384.00	2.00	0.00
84.85	0.00	2826957.58	0.00	0.00
84.00	0.00	2756192.00	0.00	0.00
299.01	0.00	2750859.41	0.00	0.00
227.00	0.00	2789184.00	0.00	0.00

- 解决方法：echo 0 > /proc/sys/vm/zone\_reclaim\_mode



# 踩过的一些坑 (2/2)

- 现象：查询性能不稳定，偶尔会出现尖刺
- 原因：CentOS 6.5默认启用了“透明大页”功能，可能造成内存分配延迟



- 解决方法：禁用“透明大页”
  - `echo "never" > /sys/kernel/mm/transparent_hugepage/enabled`
  - `echo "never" > /sys/kernel/mm/transparent_hugepage/defrag`

# 面临的挑战

---

- 功能挑战
  - 精确去重计数
  - 实时摄入的窗口限制
  - SQL支持
- 管理挑战
  - 任务接入效率
  - 索引服务的资源利用率

# 提纲

---

- OLAP引擎选型
- Druid在美团点评的实践
- **Kylin**在美团点评的实践

# Kylin简介

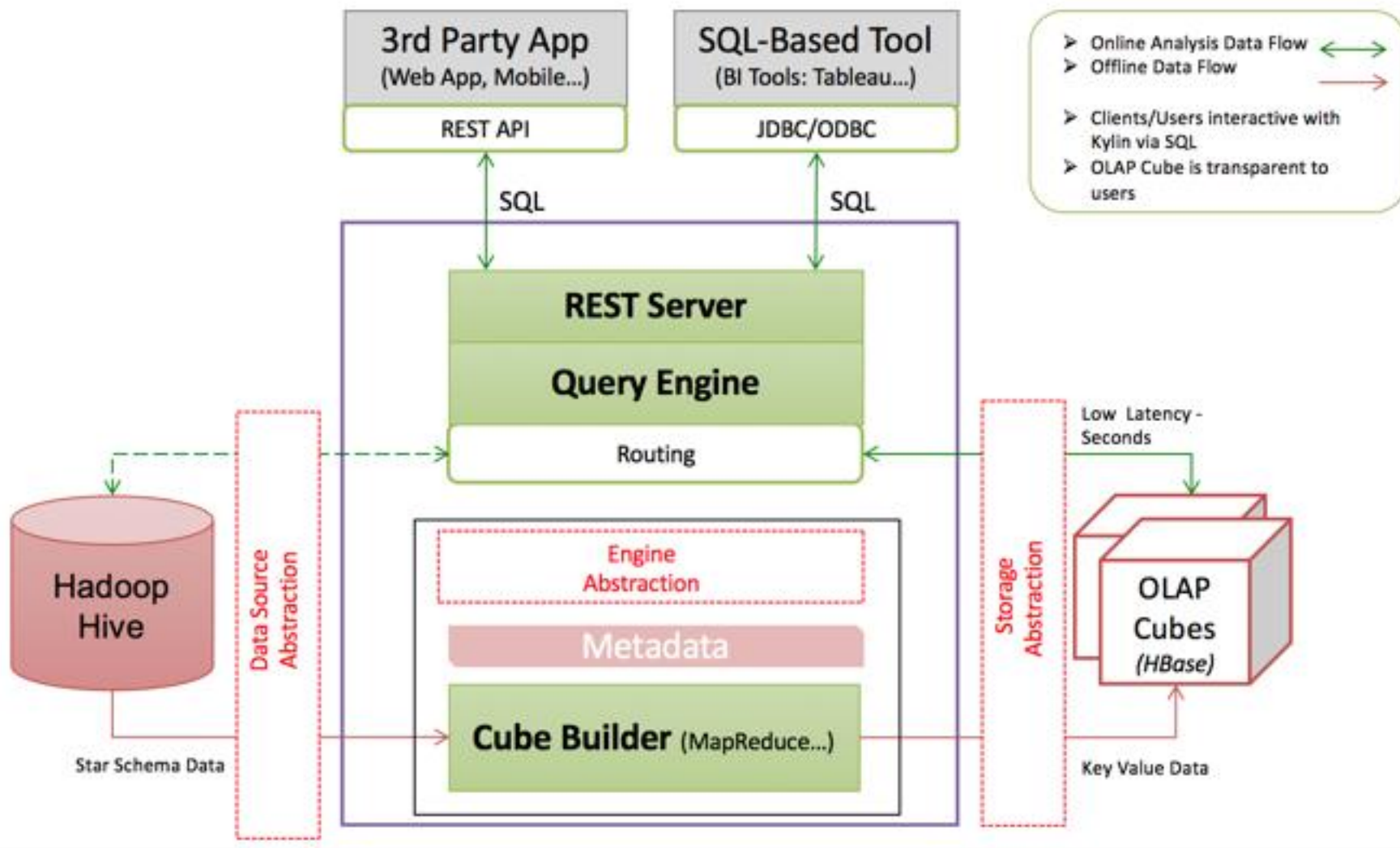
---

Kylin是一个开源的、基于Hadoop的OLAP查询引擎，能够通过标准SQL接口对超大数据集实现秒级的多维分析查询。

Kylin属于MOLAP解决方案，其核心思想是**预计算Cube**

- 预先定义维度和指标
- 预先构建Cube，Cube包含了预计算的结果
- 查询时自动从Cube中获取结果

# Kylin架构

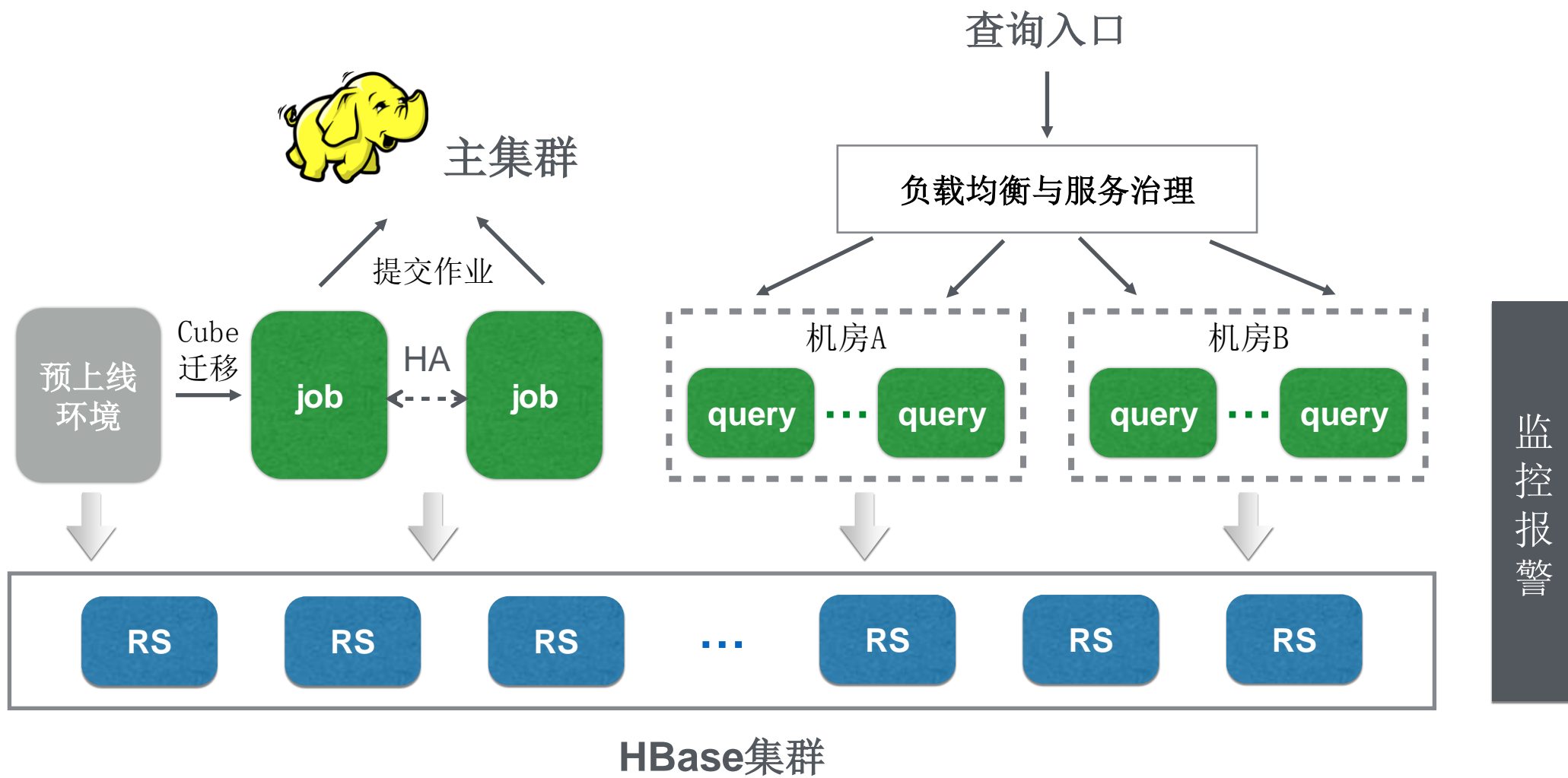


# Kylin服务概况

---

- 定位：离线OLAP引擎
- 20+个Project，350个Cube，覆盖所有业务线
- 数据总量8000亿行，Cube存储总量140TB
- 每日查询量130万次，TP99时延~1秒

# Kylin平台架构



# 主要工作

---



## 新特性

1. 精确去重计数
2. 全局字典
3. 构建服务分布式化
4. 窗口函数、Union

.....



## 功能改进

1. 大查询限制
2. 支持HBase集群HA
3. 构建性能优化
4. 前端页面加载优化

.....



## 平台化建设

1. 调度系统集成
2. 计算队列拆分
3. 权限控制与审计
4. JMX监控

.....

我们有3位Kylin Committer !



# 面临的挑战

---

- 业务隔离
- 降低Cube调优门槛
- 明细查询支持
- 低成本、高扩展的精确去重方案

谢谢大家

