SQL-on-Hadoop在FreeWheel的实践

FreeWheel数据平台部 林明

Geekbang.
极客邦科技

整合全球最优质学习资源，帮助技术人和企业成长

Growing Technicians, Growing Companies

InfoQ

专注中高端技术人员的
技术媒体

EGO EXTRA GEEKS' ORGANIZATION NETWORKS

高端技术人员
学习型社交网络

StuQ

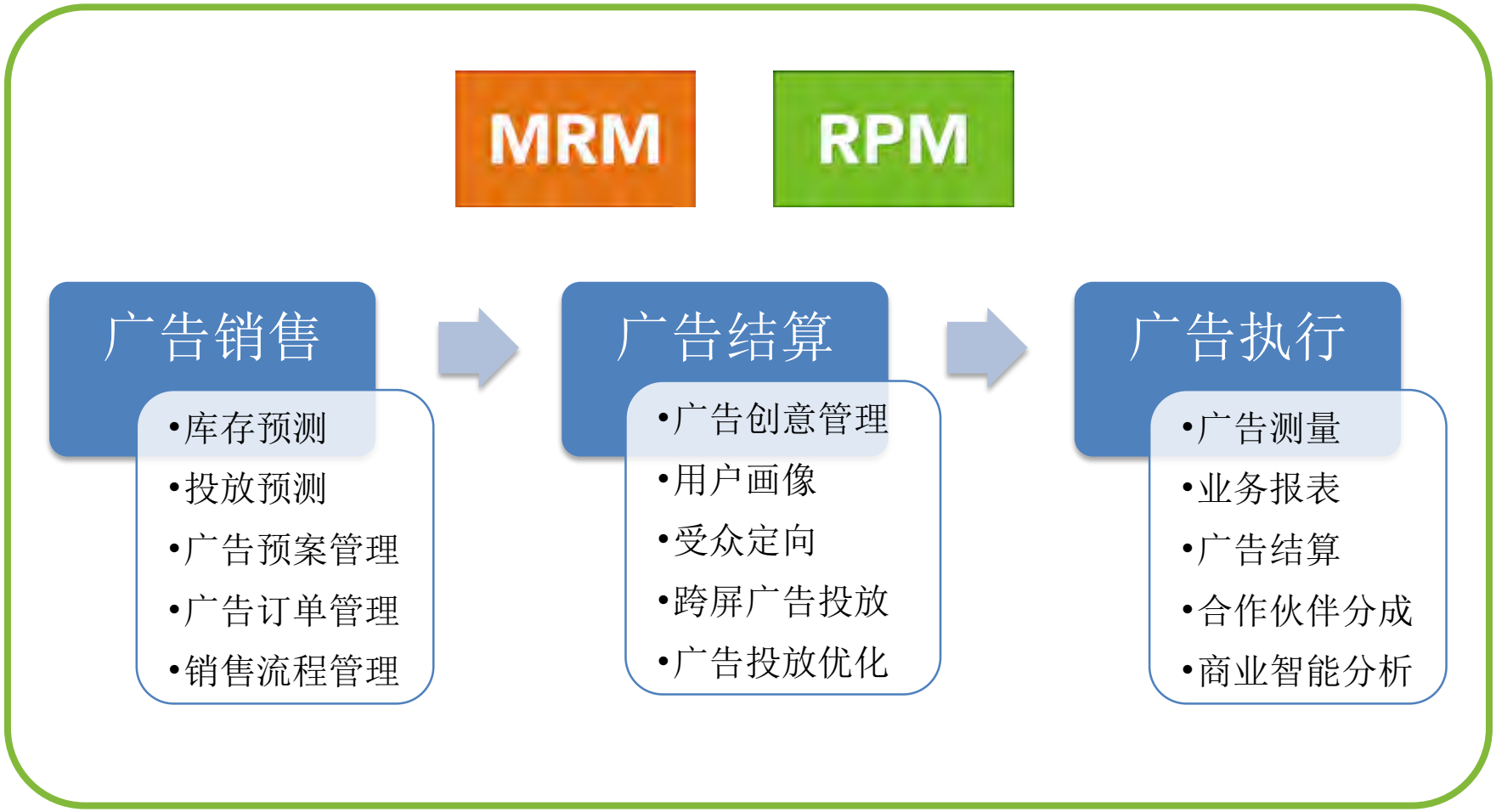
实践驱动的
IT职业学习和服务平台

GiT GEEKBANG INTERNATIONAL TRAINING 极客邦培训

一线专家驱动的
企业培训服务

地址：北京市朝阳区洛娃大厦C座6层1607室　　网址：www.geekbang.org

我们的业务



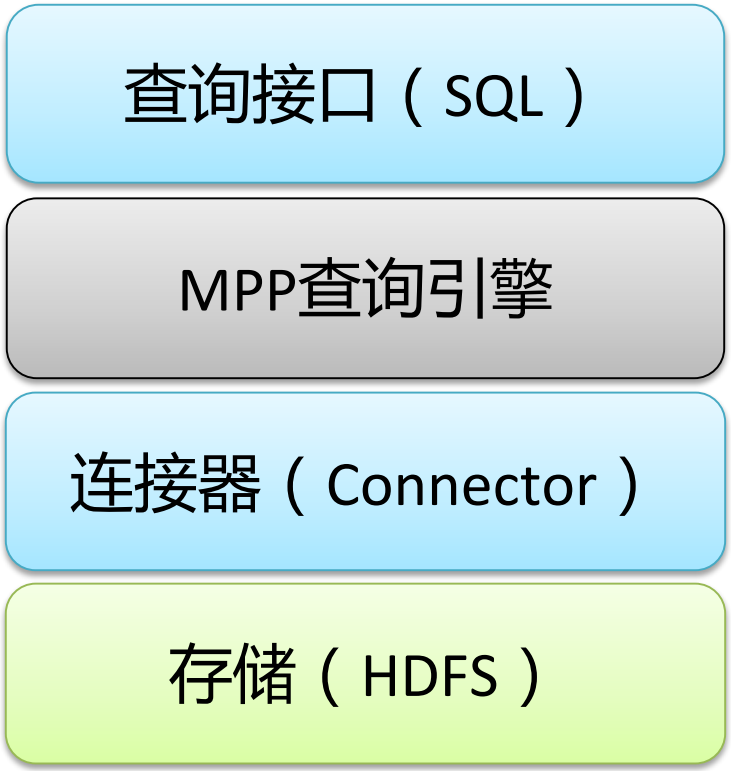
单日投放近10亿次广告，生成2TB广告投放数据

Ad-hoc数据分析

- 应用场景
  - 咨询团队分析客户业务
  - 客户服务团队解决客户问题
  - 工程师团队分析线上问题

- 业务需求
  - 可以同时分析多个数据源
  - 获取多维度和多时间跨度的分析结果
  - 在几分钟甚至几秒内完成

SQL-on-Hadoop解决方案



候选方案

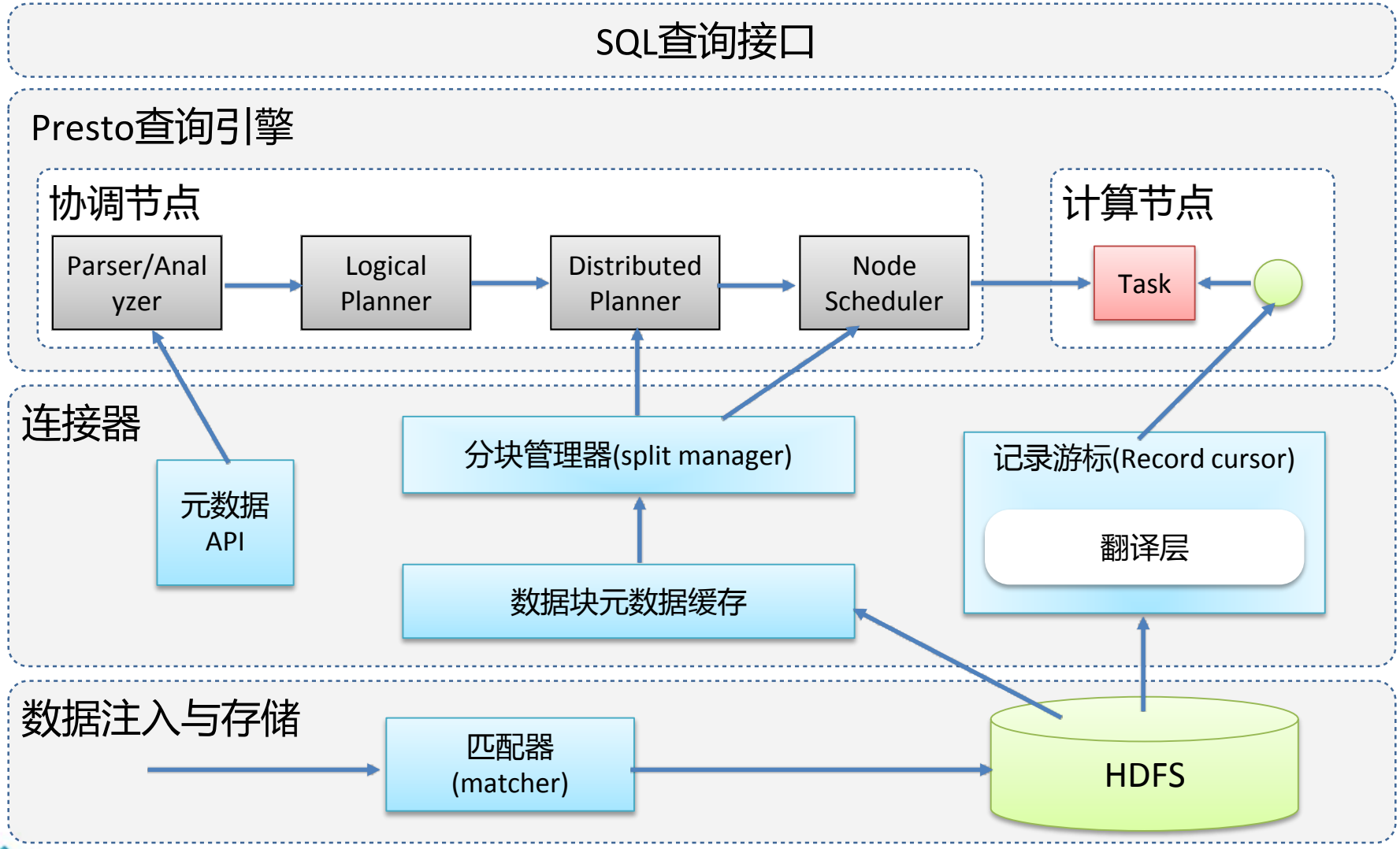
- MPP查询引擎
  - Impala/Stinger/Drill/Hive/Presto等
- 存储格式
  - 列组/ORC/Parquet等

我们的选择

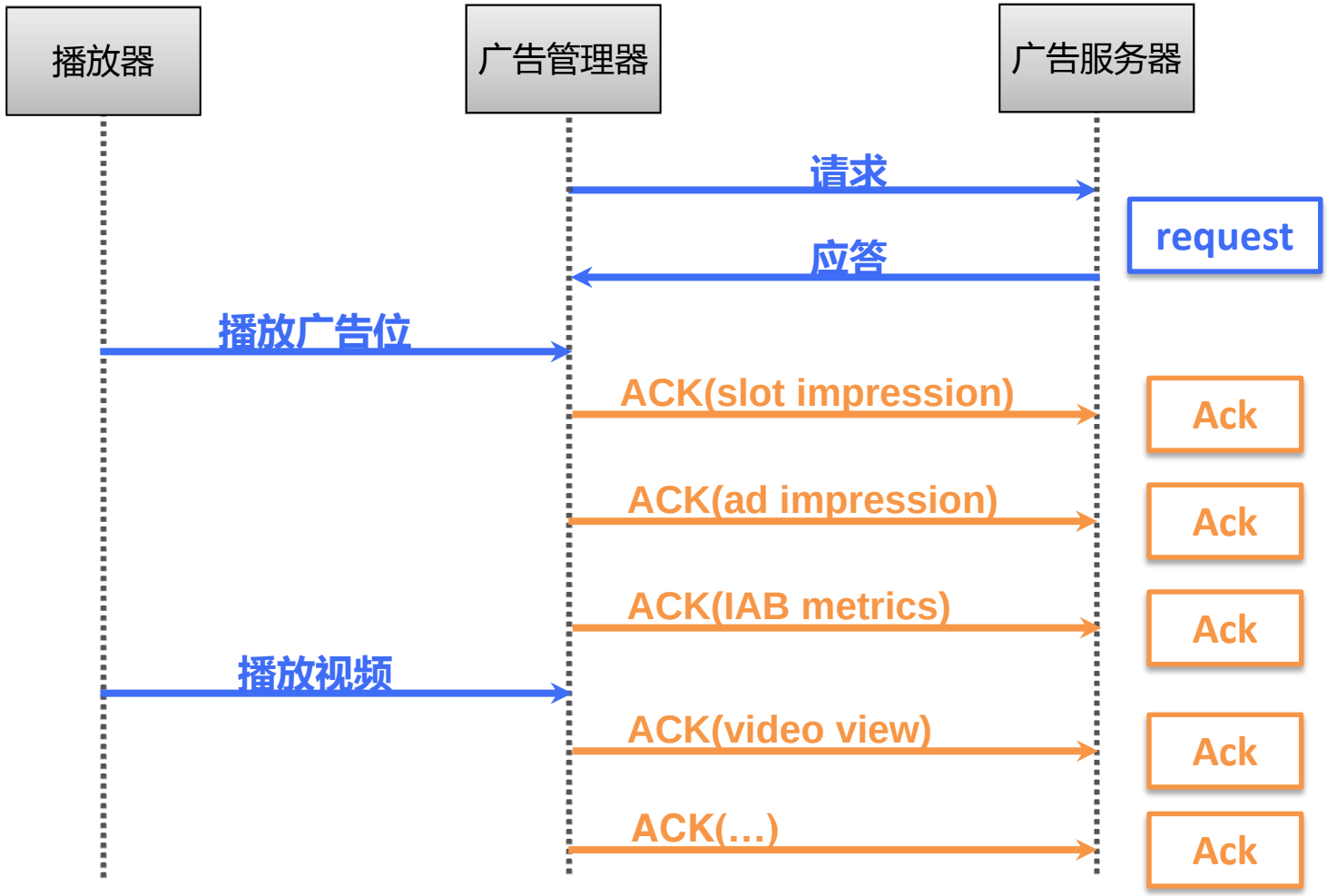
- MPP查询引擎： Presto
  - 非常快的查询速度
  - 支持对不同的数据源进行join操作
  - 方便二次开发

- 存储格式：Parquet
  - 支持复杂的嵌套数据结构
  - 高效的记录碎片化（shred）与装配（assembly）
  - 高效的压缩
  - 开发社区活跃

Ad-hoc数据分析系统

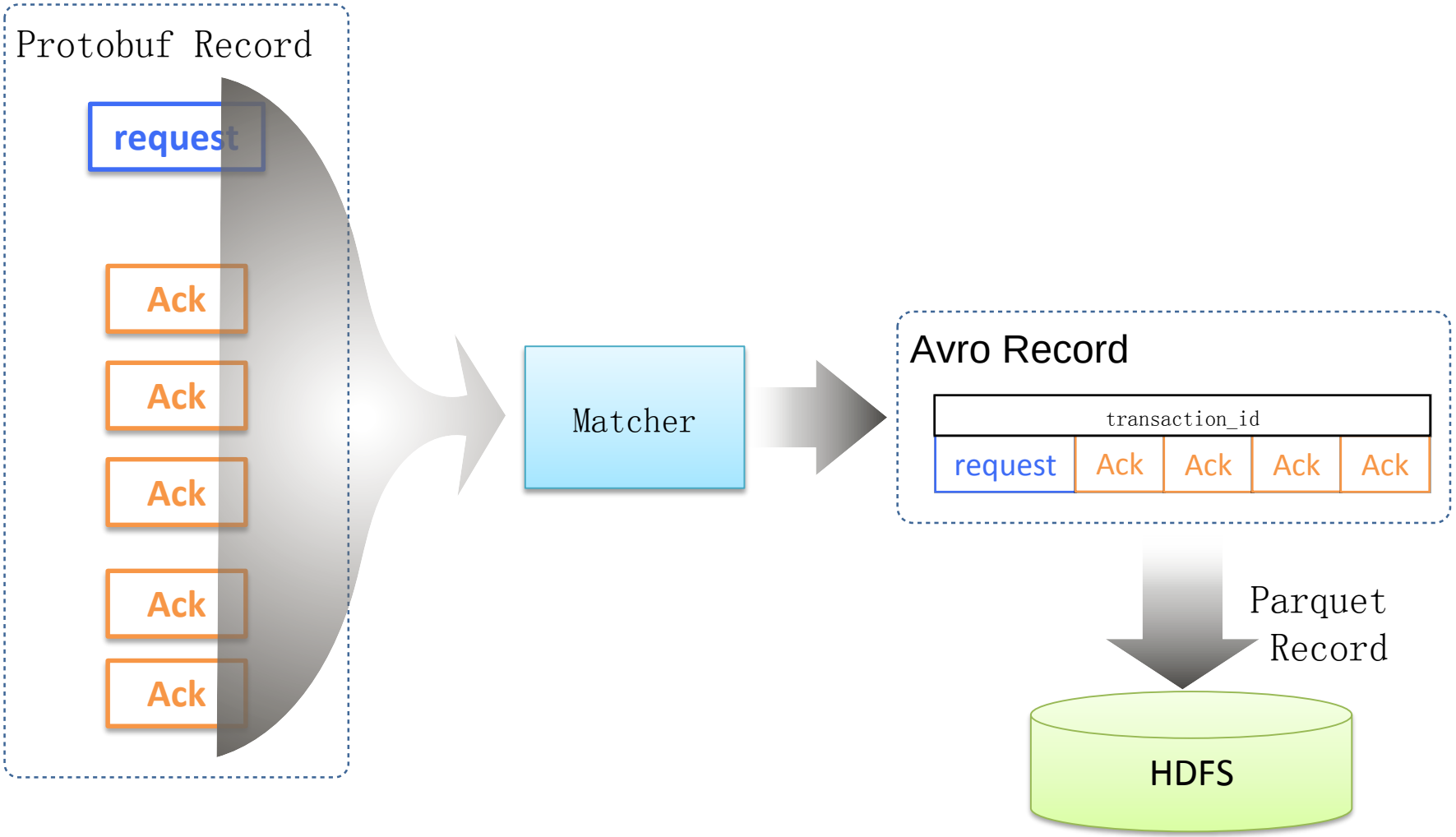


数据注入



广告请求与广告服务器交互示意图

数据注入

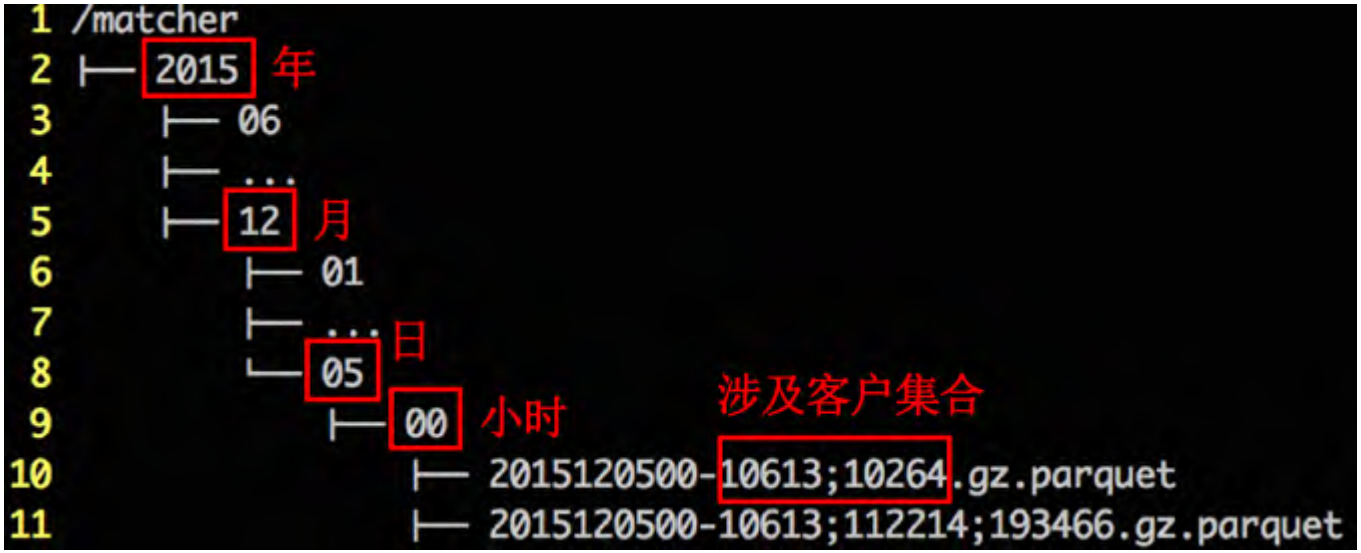

Protobuf Record
request
Ack
Ack
Ack
Ack
Ack
Matcher
Avro Record
transaction_id
request
Ack
Ack
Ack
Ack
Parquet
Record
HDFS

数据存储

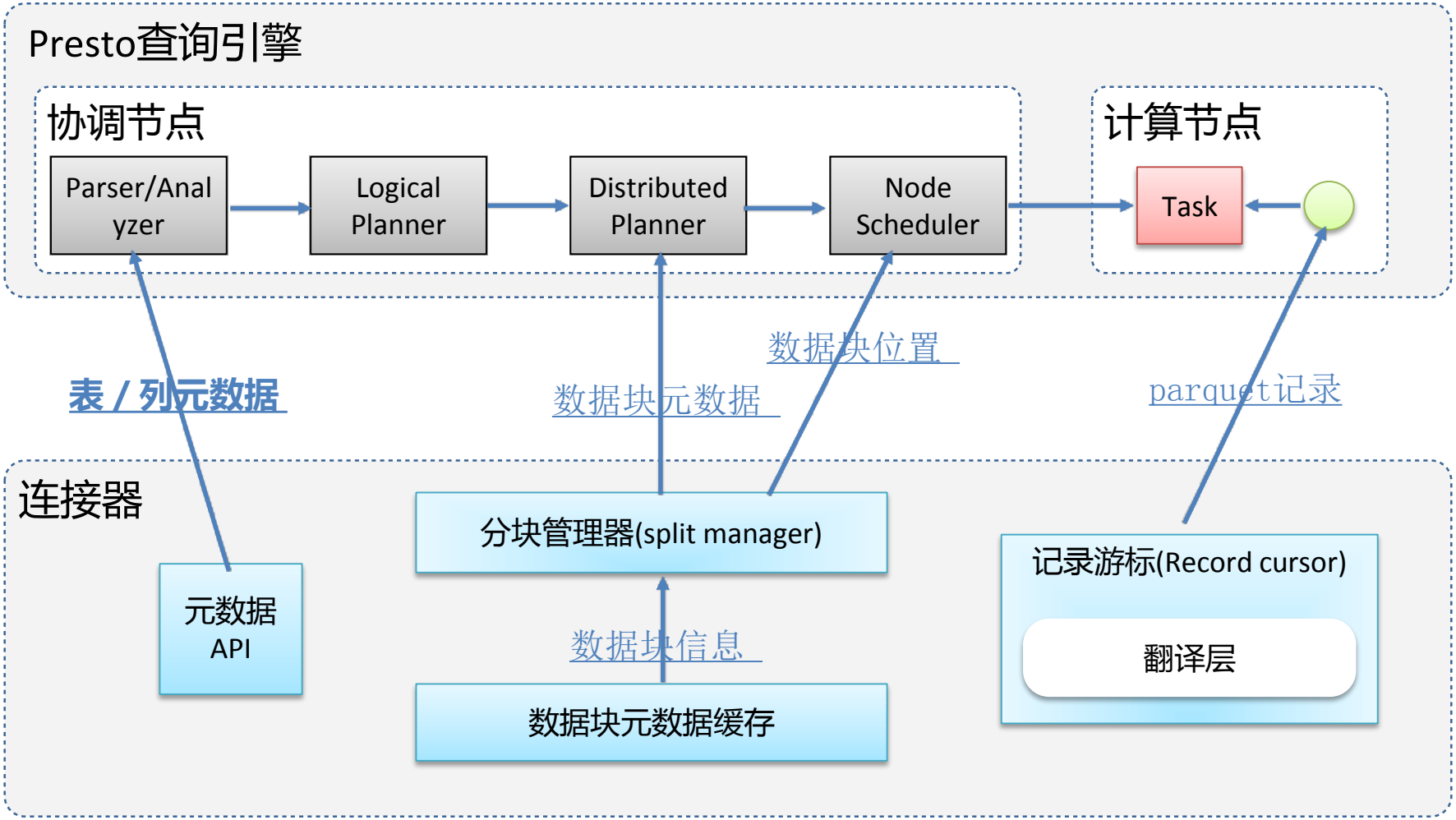
- 查询Pattern

```
SELECT c1, …
FROM transaction
WHERE event_date >= [start]
    and event_date < [end]
    and network_id = [id]
```

- 数据的目录组织结构

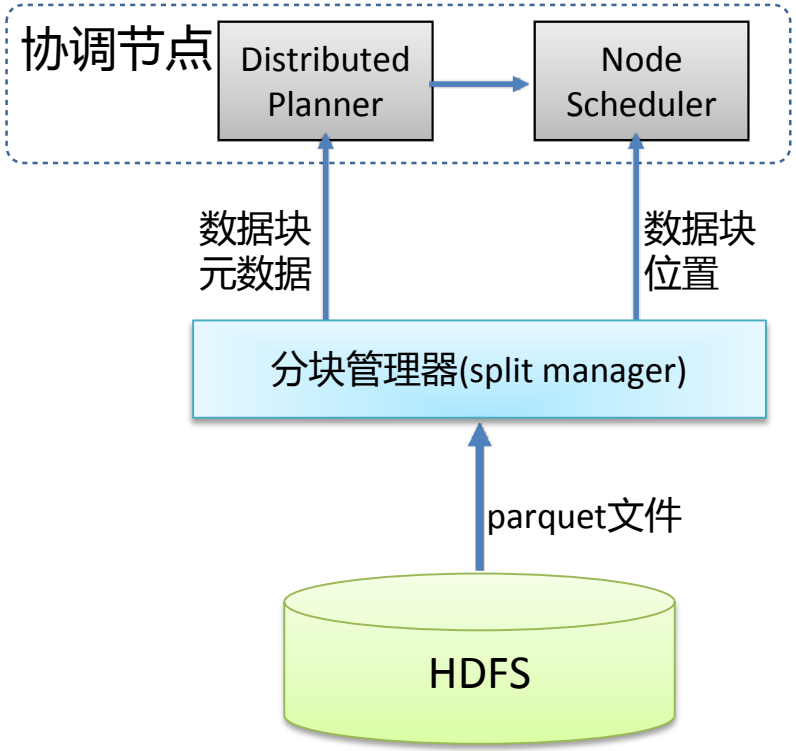


连接器


Presto查询引擎
协调节点
Parser/Analyzer
Logical Planner
Distributed Planner
Node Scheduler
计算节点
Task
表 / 列元数据
数据块元数据
数据块位置
parquet记录
连接器
元数据 API
分块管理器(split manager)
数据块信息
数据块元数据缓存
记录游标(Record cursor)
翻译层

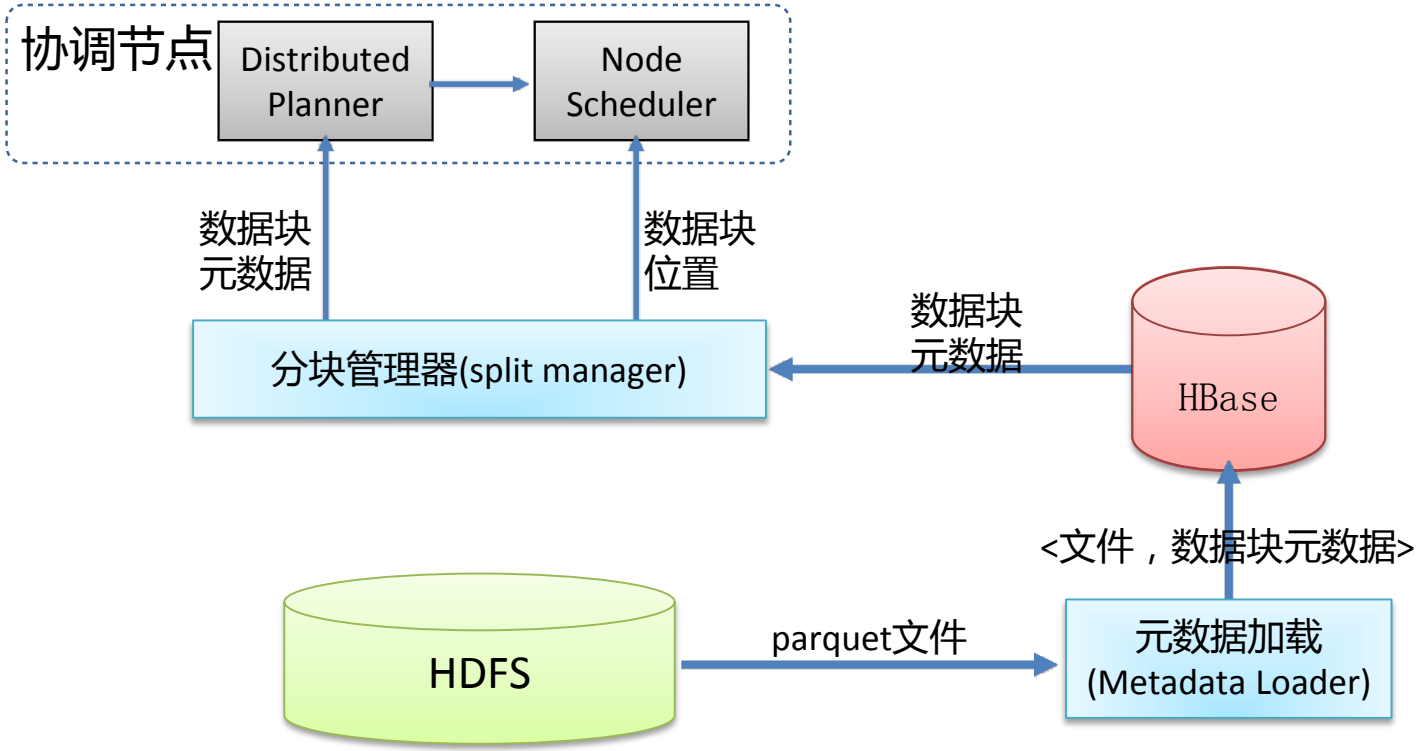
分块管理器（Split Manager）

- 将表分块由Presto协调节点（coordinator）分发到计算节点（worker）进行处理



分块管理器（Split Manager）

- 将表分块由Presto协调节点（coordinator）分发到计算节点（worker）进行处理



记录游标（Record Cursor）

- 读取parquet记录供Presto计算（worker）节点处理
  - 结合Predicate Pushdown及ParquetFilter避免反序列化无关的records

查询接口

- 表 transaction

```
{
    “request” : {
        “user_id” : “string”
    },
    “acks” : [
    {
        “event_type” : “string”,
        “event_name” : “string”
    }]
}
```

request.user_id	acks
user_1	[{“i”, “vv”}, {“i”, “adImp”}]
user_2	[{"i”, “vv”}, {“i”, “adImp”}, {“I”, “adImp”}]
user_3	[“i”, “vv”]

如何查询每个用户观看广告的次数？

- 找出每个transaction包含的满足如下条件的ack的个数

```
event_type = ‘i’ && event_name = ‘adImp’
```

- 尝试？

```
SELECT request.user_id,
   count(1)
FROM transaction
WHERE acks.event_type = ‘i’
   and acks.event_name = ‘adImp’
GROUP BY request.user_id;
```

Hive方法

```
SELECT request.user_id,
      count(1)
FROM transaction
LATERAL VIEW explode(acks) acklist AS event
WHERE event.event_type = ‘i’
      and event.event_name = ’adImp'
GROUP BY request.user_id;
```

我们的方法

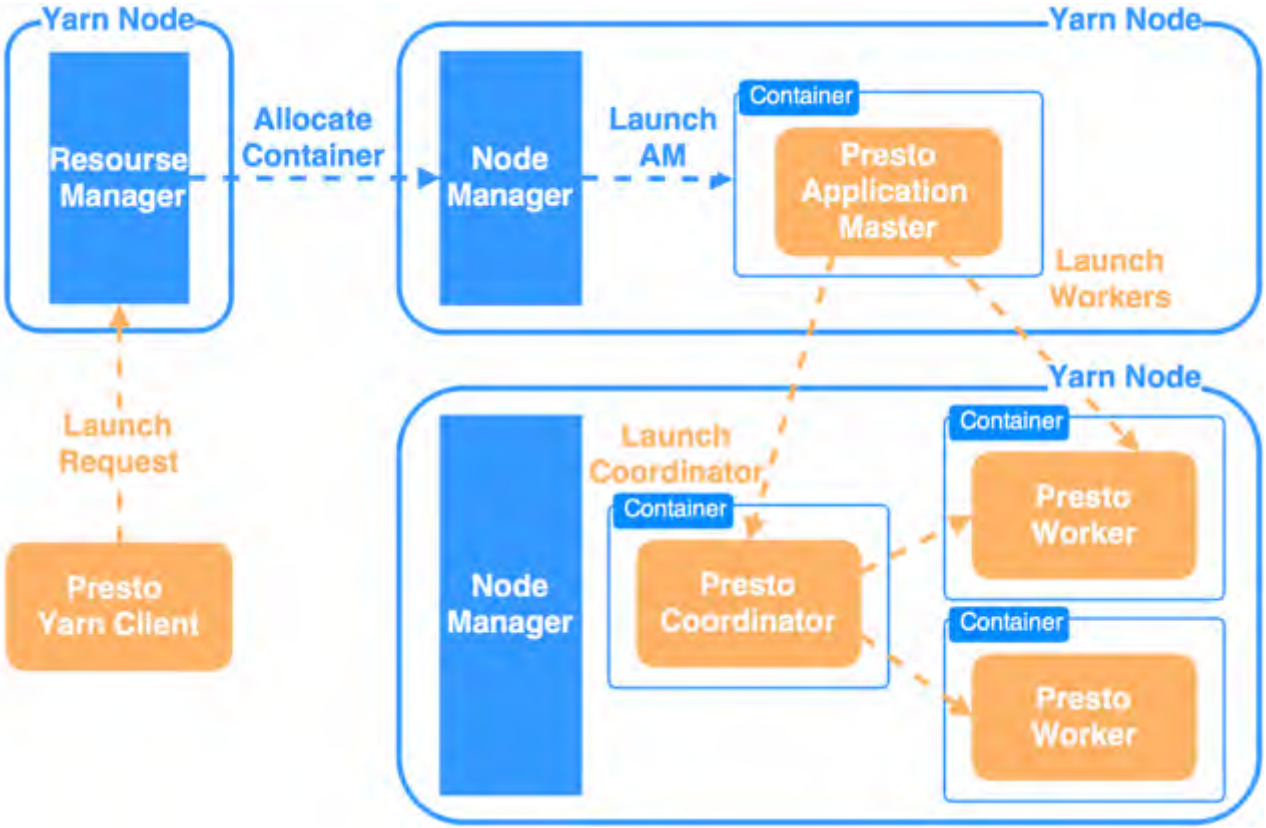
- 逻辑表到物理表的映射
  - 物理表：transaction
  - 逻辑表：transaction, ack

event_type	event_name	request.user_id

```
SELECT request.user_id,
       count(1)
FROM ack
WHERE event_type = ‘i’
       and event_name = ‘adImp’
GROUP BY request.user_id;
```

集群资源统一管理（Presto on YARN）

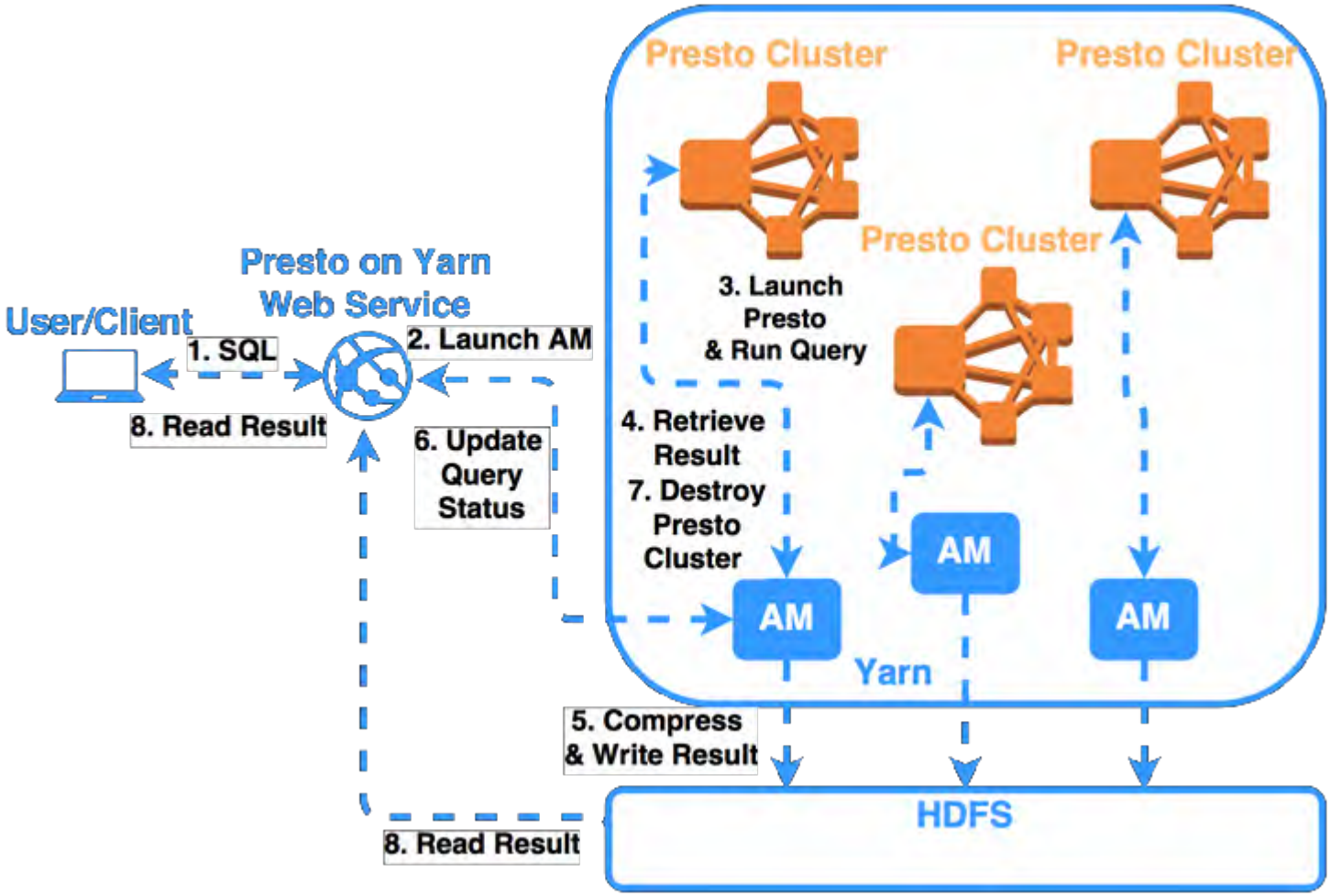
- 日志查询服务作为长期运行服务部署在YARN上



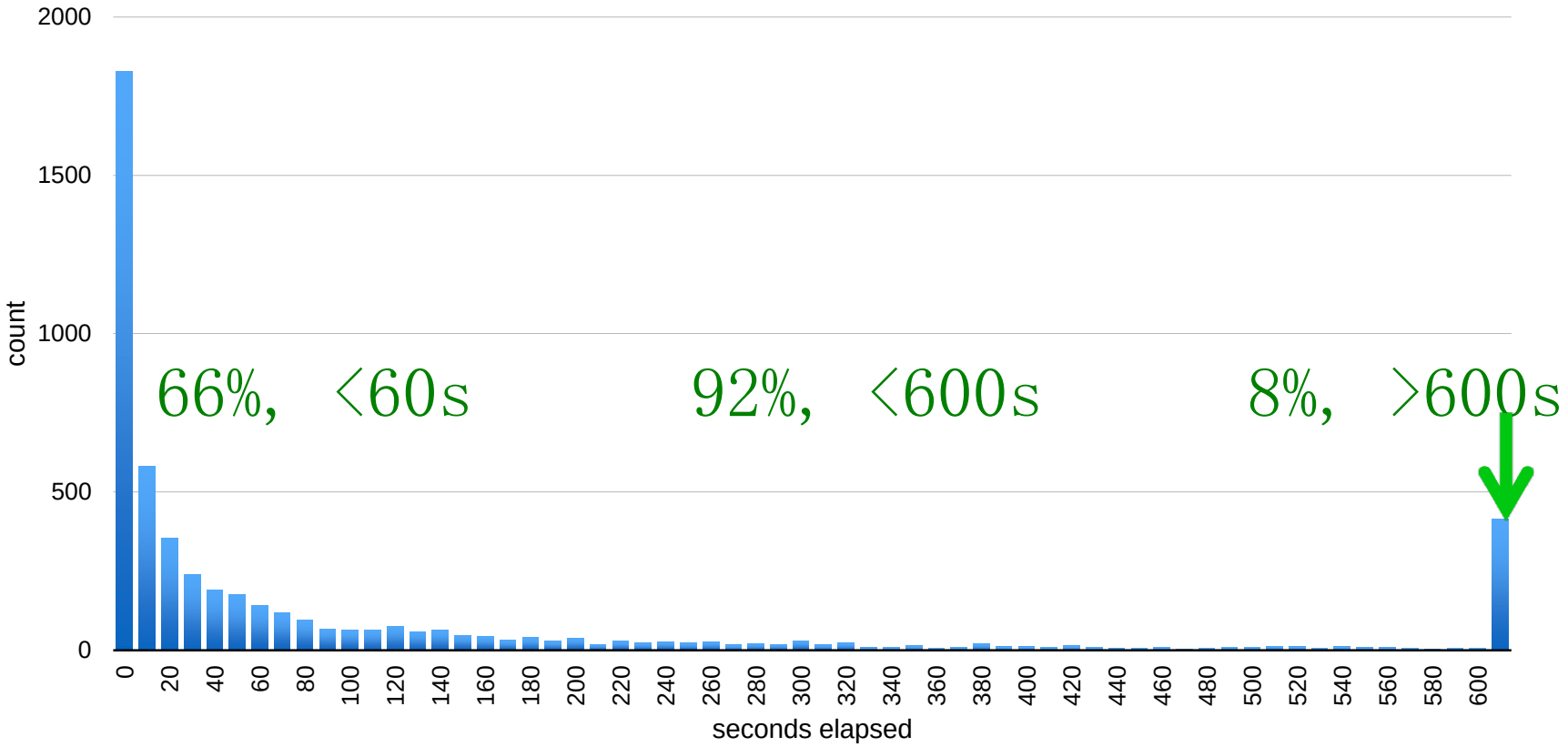
Presto as a Service

- 通过REST API调用该服务
  - 动态提交周期性执行的“长”sql
  - 与MR/Spark作业竞争资源，提高集群资源利用率
  - 统一管理线下数据流水线内嵌的查询请求

集群资源统一管理（Presto on YARN）



查询语句执行时间分布


2000
1500
1000
500
0
count
66%, <60s
92%, <600s
8%, >600s
0
20
40
60
80
100
120
140
160
180
200
220
240
260
280
300
320
340
360
380
400
420
440
460
480
500
520
540
560
580
600
seconds elapsed

Thanks!