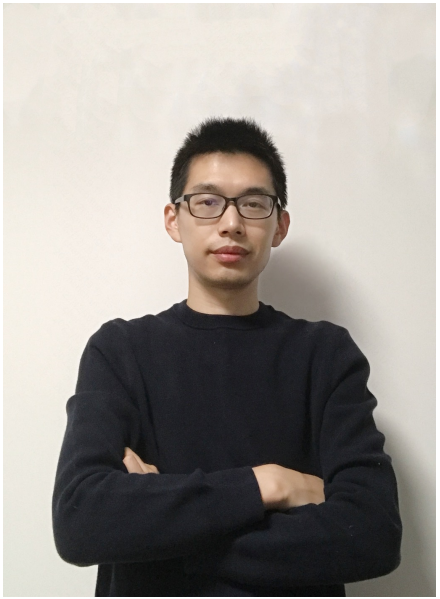
快手 | DataFun.

指标中台核心技术-开放分析语言OAX设计

陈长林 快手大数据平台技术专家

关于我



陈长林，快手大数据平台技术专家

2021年加入快手，曾在美团、阿里从事大数据领域和电商业务后端领域的研发工作。当前主要从事大数据领域BI数据产品、数据建模、分析语言等相关工作，在大数据数据分析领域建设有丰富实践经验。

目录 CONTENT

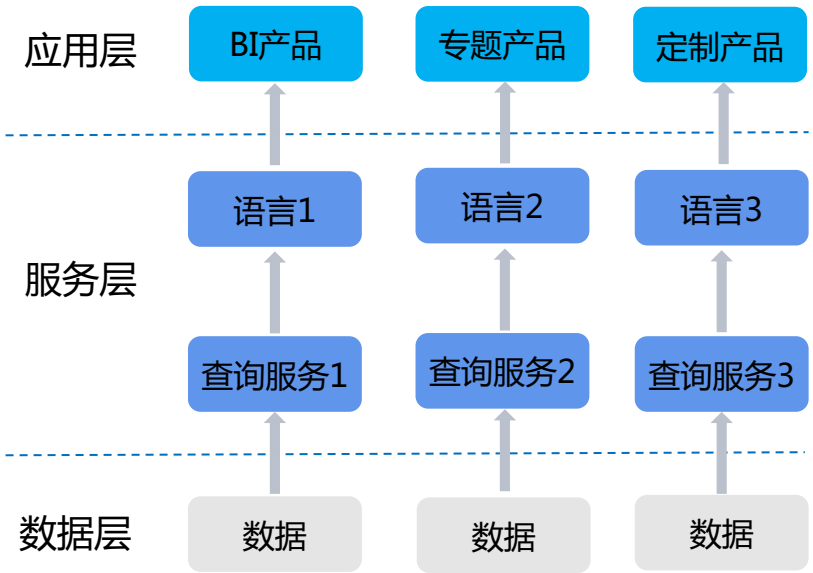
01 OAX整体介绍

什么是OAX

1. 全称：**O**pen **A**nalysis e**X**pressions, 快手开放分析表达式
2. 定位：以数据集为载体，面向分析场景的查询语言
3. 目标群体：有分析需求的用户或系统

OAX设计背景

烟囱式建设



面临的问题：

- 烟囱系统
- 规范缺乏
- 接入效率低

问题的解决：

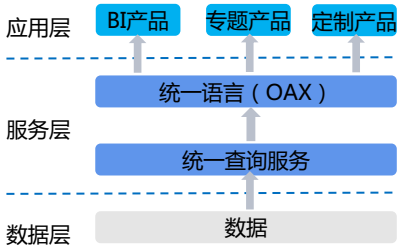
- 统一，开放
- 建立统一规范

OAX设计指导原则

规范

- 分析模型
- 分析语法
- 访问协议

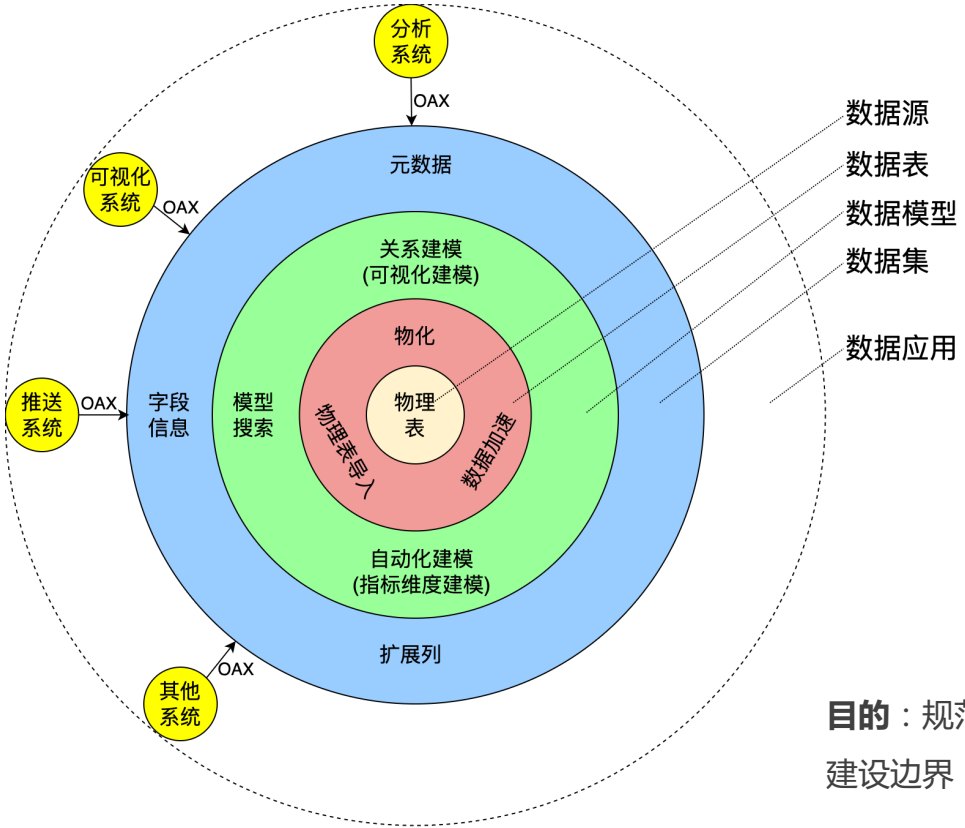
统一



开放

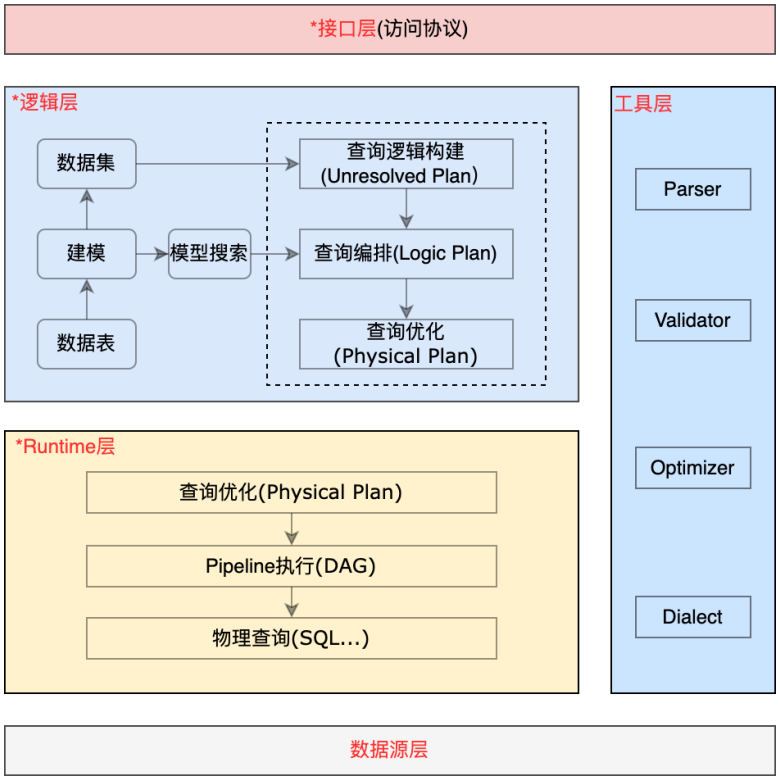
- 功能扩展
- 外部使用

分析模型



目的：规范系统开发，明确分析服务各模块的建设边界

OAX语言架构



1 **接口层**

用户访问分析服务入口，OAX提供了规范的访问协议

2 **逻辑层**

分析查询逻辑的构建，以执行计划树表示

3 **Runtime层**

查询逻辑的执行，执行计划的调度和执行，包含本地计算和分布式计算

4 **数据源层**

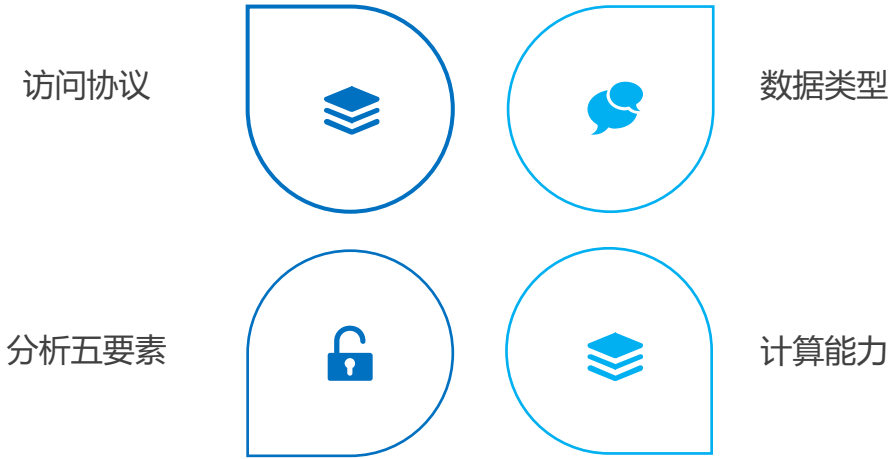
管理和适配分析服务依赖的数据源

5 **工具层**

提供对OAX语言的处理工具，包含对OAX的解析、验证、优化、方言处理，基于Apache Calcite扩展

02 OAX语法设计

OAX表达分析需求

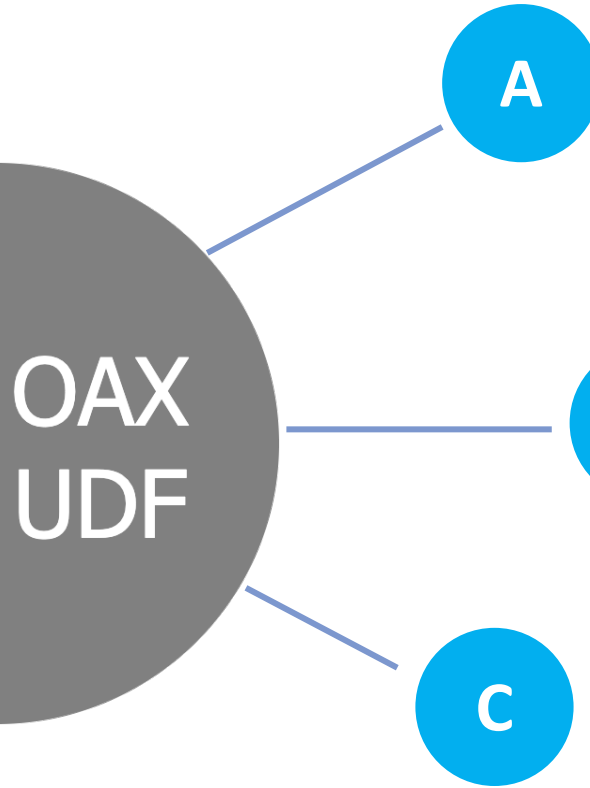


四种数据类型

数据类型

- 文本（String）
  - 表达文本内容的数据类型
- 数值（Numeric）
  - 表达整数、浮点数的数据类型。
- 日期（DateTime）
  - 精确到毫秒的13位时间戳、表达时间语义的数据类型。
- 布尔（Boolean）
  - “真”(1)或“假” (0) 中的一个。

三类计算能力



A 基本计算

允许用户在原始值或计算结果值粒度进行计算，分为：数字函数、字符串函数、日期函数、类型转换、逻辑函数、聚合函数、指标函数、高级计算

例如：SUM([消耗])，CONCAT([姓], [名]), DATE_ADD(‘day’ , NOW(), 1),
计算周同比：YOY(SUM([消耗]), 1, 'week', 'value')

B 动态粒度计算

允许计算过程中改变数据计算粒度，可以在较高粒度(EXCLUDE)、较低粒度(INCLUDE)、独立粒度(FIXED)进行计算

例如计算各省的消耗占比：SUM([消耗]) / {EXCLUDE [省份]: SUM([消耗])}

C 表计算

允许用户对结果值再进行计算，例如：滚动类函数、窗口类函数、偏移类函数等

例如要计算MTD消耗 (本月1号至本月当前日期的消耗累计)：RUNNING_SUM(SUM([消耗])) ALONG([日])

五个分析要素

分析需求：2020~2021年湖北省各市的GDP及占比

维度

数据的观察角度，即从哪个角度去分析问题，看待问题；

例如：年份、省份、城市

指标

业务事实的度量；

例如：GDP、占比

数据集

数据源和可视化展示的中间环节，承接数据源的输入，并为用户输出模型

假设：国民经济数据集

过滤条件

观察局部角度的结果值，包括原始值过滤和结果值过滤

例如：省份 = ‘湖北’

时间范围

观察指标数据的时间范围

例如：年份 >= ‘2020’ AND 年份 <= ‘2021’

五要素的两种形式

PB形式

```
message OaxQuery {
    repeated Select select = 1;          //指标、维度
    From from = 2;                       //数据集
    Filter where = 3;                    //纬度值过滤
    Filter context_where = 4;            //上下文过滤器
    Filter having = 5;                   //结果值过滤
    repeated string group_by = 6;
    repeated OrderBy order_by = 7;
    LimitSpec limit = 8;
}
```

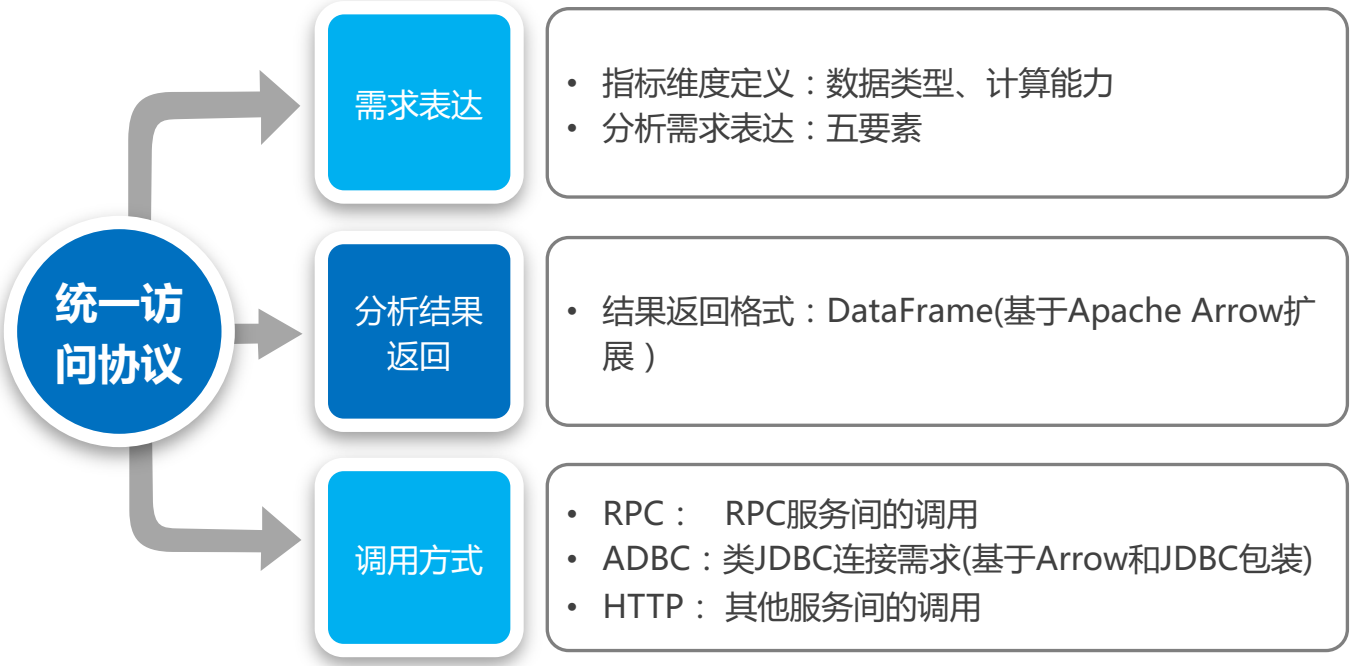
类SQL形式

```
SELECT select_expr [, …]                                --指标、维度
FROM [ dataset_urn | sub_query ]                        --数据集
[ WHERE where_condition ]                               --纬度值过滤
[ CONTEXT_WHERE context_where_condition ]               --上下文过滤
[ HAVING having_condition]                              --结果值过滤
[ GROUP BY grouping_element [, …] ]
[ ORDER BY expression [ ASC | DESC ][, …] ]
[ LIMIT offset,size ]
```

分析需求：
2020~2021年湖北省各市的GDP及占比

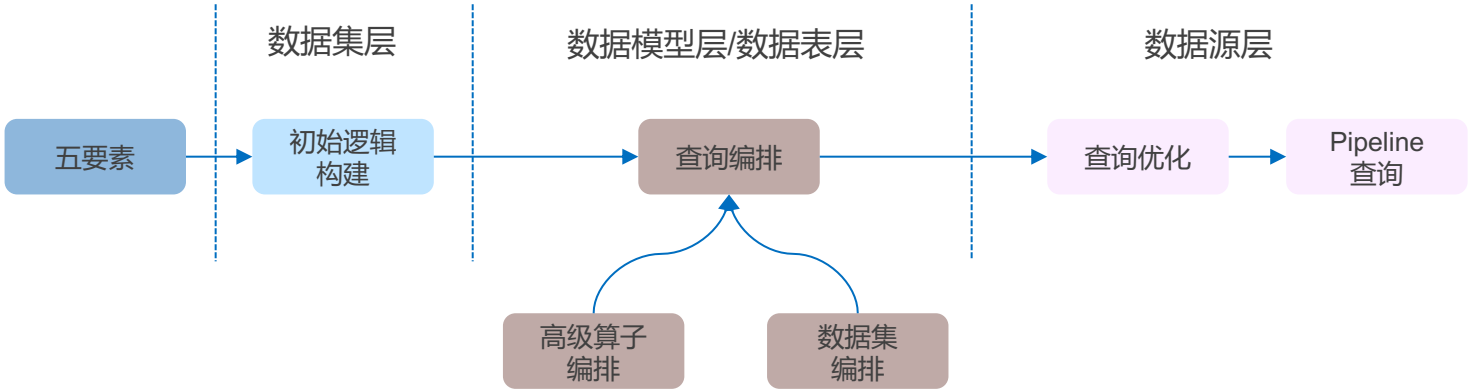
```
SELECT [年份],                                                  --维度
       [省份],                                                  --维度
       [城市],                                                  --维度
       SUM([GDP]) AS gdp,                                       --指标
       SUM([GDP]) / {EXCLUDE [城市]: SUM([GDP])} AS gdp_rate    --指标
FROM [国民经济数据集]                                           --数据集
WHERE [年份] >= '2020' and [年份] <= '2021'                     --时间范围
      AND [省份] IN ('湖北')                                    --过滤条件
GROUP BY [年份], [省份], [城市]
ORDER BY [年份] ASC
LIMIT 0,  100
```

OAX访问协议

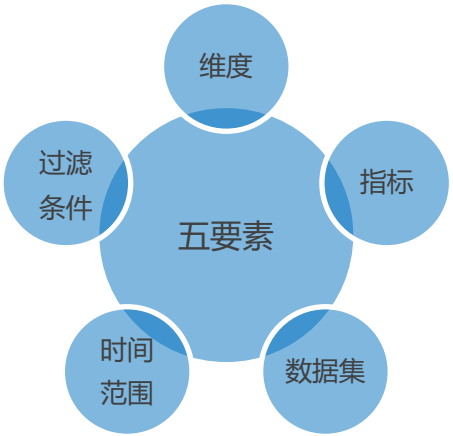

统一访问协议
需求表达
指标维度定义：数据类型、计算能力
分析需求表达：五要素
分析结果返回
结果返回格式：DataFrame(基于Apache Arrow扩展）
调用方式
RPC： RPC服务间的调用
ADBC：类JDBC连接需求(基于Arrow和JDBC包装)
HTTP： 其他服务间的调用

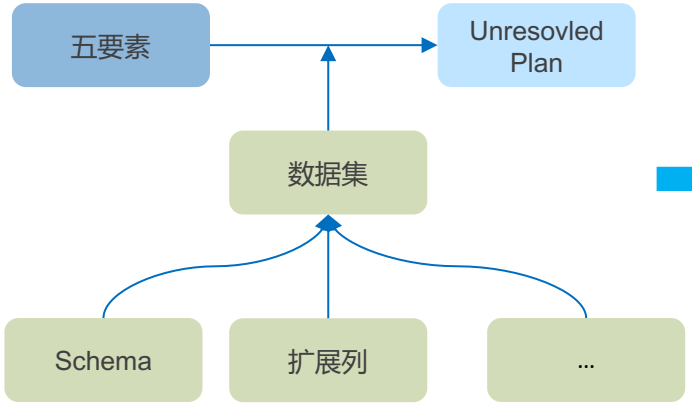
03 OAX Runtime设计

Runtime概览

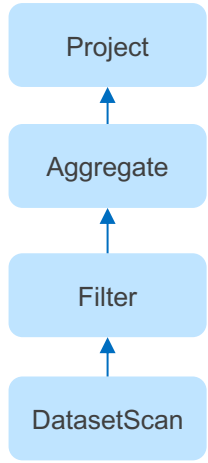


初始逻辑构建





原始查询计划

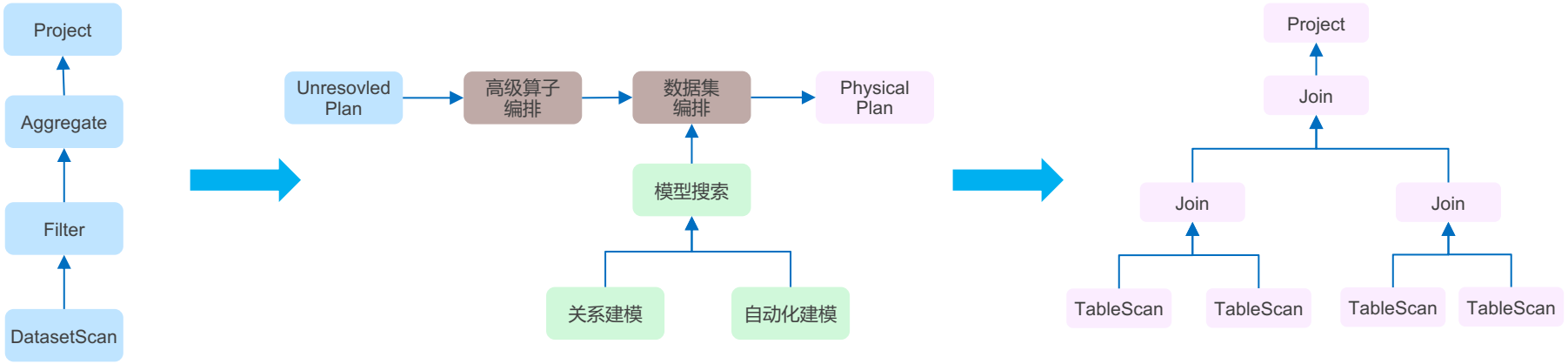


分析需求：2020~2021年湖北省各市的GDP及占比

```sql
SELECT [年份],                                          --维度
       [省份],                                          --维度
       [城市],                                          --维度
       SUM([GDP]) AS gdp,                               --指标
       SUM([GDP]) / {EXCLUDE [城市]: SUM([GDP])} AS gdp_rate  --指标
FROM [国民经济数据集]                                    --数据集
WHERE [年份] >= '2020' and [年份] <= '2021'              --时间范围
      AND [省份] IN ('湖北')                             --过滤条件
GROUP BY [年份], [省份], [城市]
ORDER BY [年份] ASC
LIMIT 0, 100
```

```
LogicalSort(sort0=[$0], dir0=[ASC], offset=[0], fetch=[100])
  LogicalProject(年份=[$0], 省份=[$1], 城市=[$2], gdp=[$3], gdp_rate=[/($3, {EXCLUDE $2 : $3})])
    LogicalAggregate(group=[{0, 1, 2}], gdp=[SUM($3)])
      LogicalFilter(condition=[AND(>=($0, '2020'), <=($0, '2021'), =($1, '湖北'))])
        LogicalDatasetScan(table=[[国民经济数据集]]) --自定义LogicalDatasetScan节点
```

查询编排



高级算子编排示例：

```
SUM([GDP])/{EXCLUDE [城市]: SUM([GDP])}
```

```
LogicalAggregate(group=[{0, 1, 2}], numerator=[SUM($3)])
  LogicalProject(年份=[$0], 省份=[$1], 城市=[$2], GDP=[$4])
    LogicalFilter(condition=[AND(>=($0, '2020'), <=($0, '2021'), =($1, '湖北'))])
      LogicalTableScan(table=[[国民经济数据集]])
```

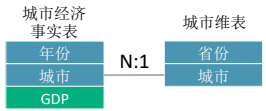
LEFT JOIN

```
LogicalAggregate(group=[{0, 1}], denominator=[SUM($2)])
  LogicalProject(年份=[$0], 省份=[$1], GDP=[$4])
    LogicalFilter(condition=[AND(>=($0, '2020'), <=($0, '2021'), =($1, '湖北'))])
      LogicalTableScan(table=[[国民经济数据集]])
```

数据集编排示例：

```
LogicalTableScan(table=[[国民经济数据集]])
```

模型关系



数据集编排

```
Join(condition=[AND(=($1, $4))], joinType=[left])
  Project(年份=[$0], 城市=[$1], GDP=[$2])
    TableScan(table=[[城市经济事实表]])
  Project(省份=[$0], 城市=[$1])
    TableScan(table=[[城市维表]])
```

查询优化

常规优化

- RBO优化
- CBO优化

计算路由

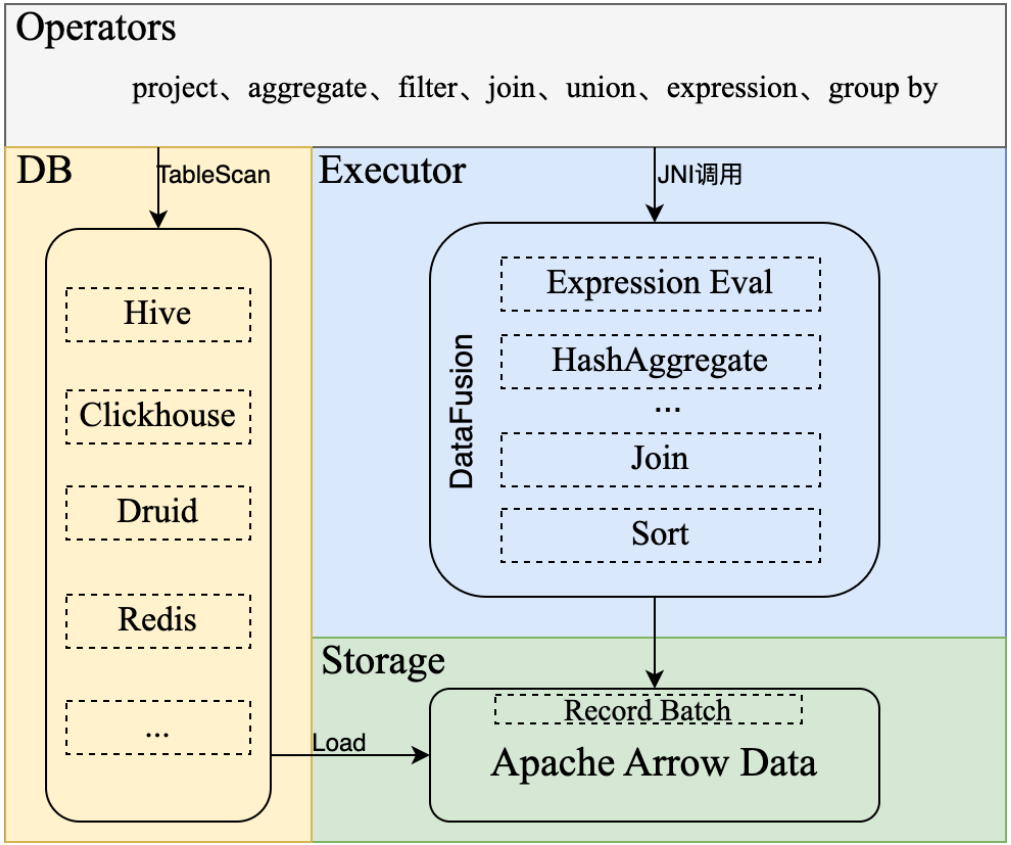
决策哪些节点在本地计算，哪些下推到引擎执行:

- 查询尽可能下推到物理引擎；
- 基于数据量级和查询代价进行二次判定是否都下推到物理引擎；

执行计划变换

通过自定义节点，实现某些引擎特有的查询能力，比如Clickhouse的LocalQuery等;

Pipeline查询



04 总结

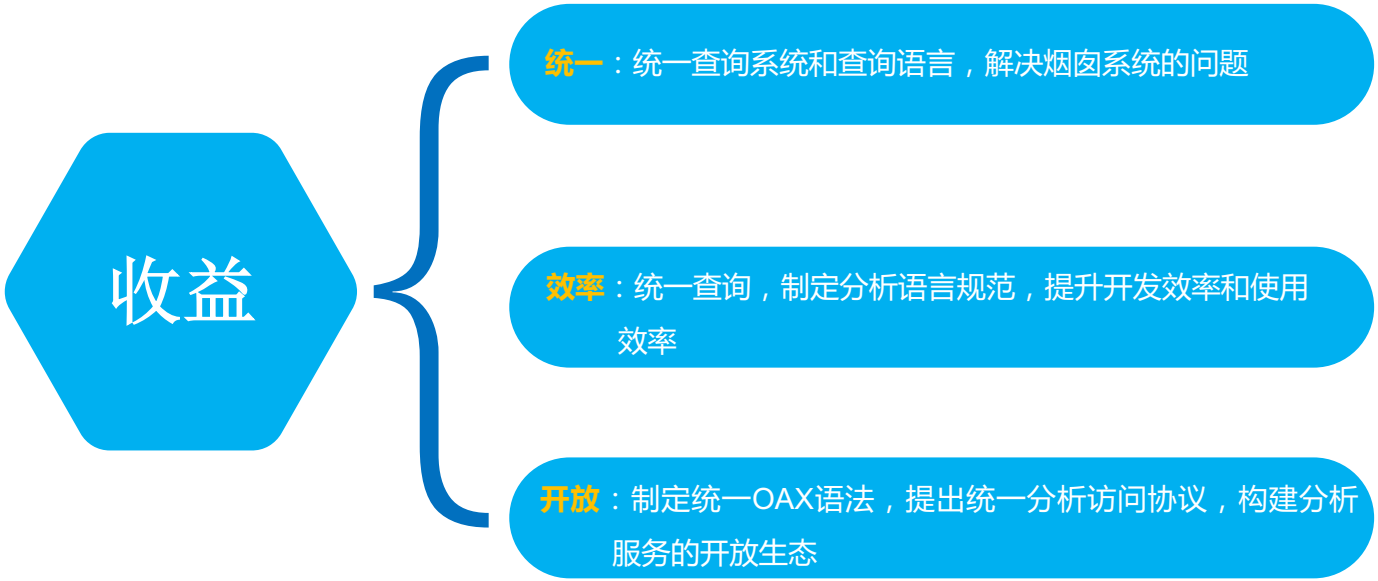
OAX设计回顾

- 设计背景
- 语言模型
- 语言架构

- 4种类型
- 3类能力
- 5个要素
- 访问协议

- 逻辑构建
- 逻辑编排
- 查询运行

OAX的收益



欢迎技术交流

快手大数据公众号

非常感谢您的观看

快手 | DataFun.