



# clickhouse数仓应用实践

演讲人：朱元

日期：2019-10-20

# 目录

## CONTENTS



现状背景



应用实践



所遇问题



01

## 现状背景

# 现状

即席查询性能差



数据链路长



数据压缩率低



需求响应慢





02

应用实践

# 数据架构

应用能力层

数据展示能力

多维分析能力

数据共享能力

数据层

oracle数据平台

结果表

低度, 中度, 高度汇总表

ods

clickhouse 数 仓

商品维度表

商品维度表

配送主题清单表

订单主题清单表

供应商维度表

区域维度表

验收主题清单表

库存主题清单表

仓库维度表

供应商商品价格变动维度表

销售主题清单表

收入主题清单表

获取层

数据采集工具kettle

日志收集工具

数据源

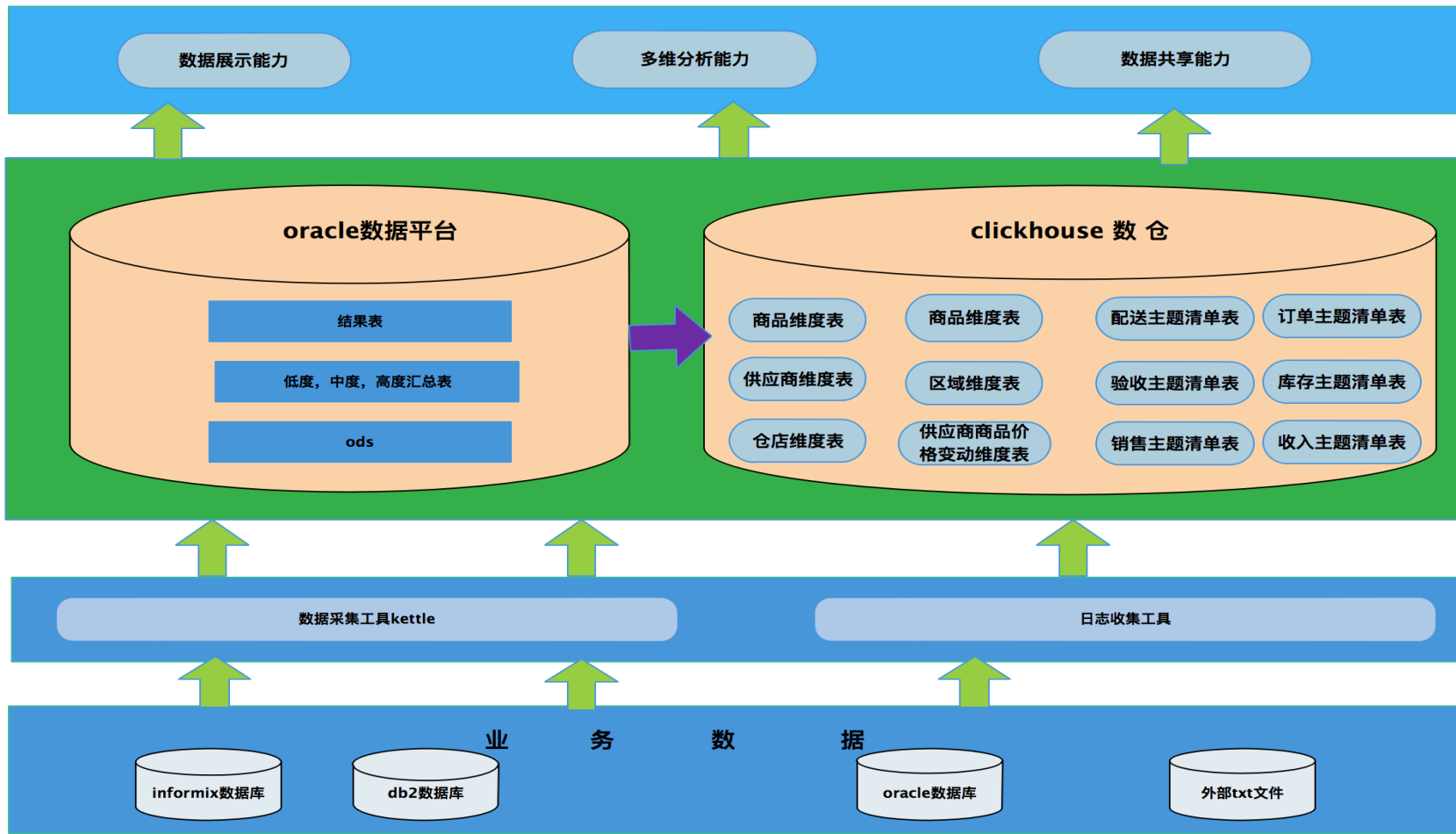
informix数据库

db2数据库

业 务 数 据

oracle数据库

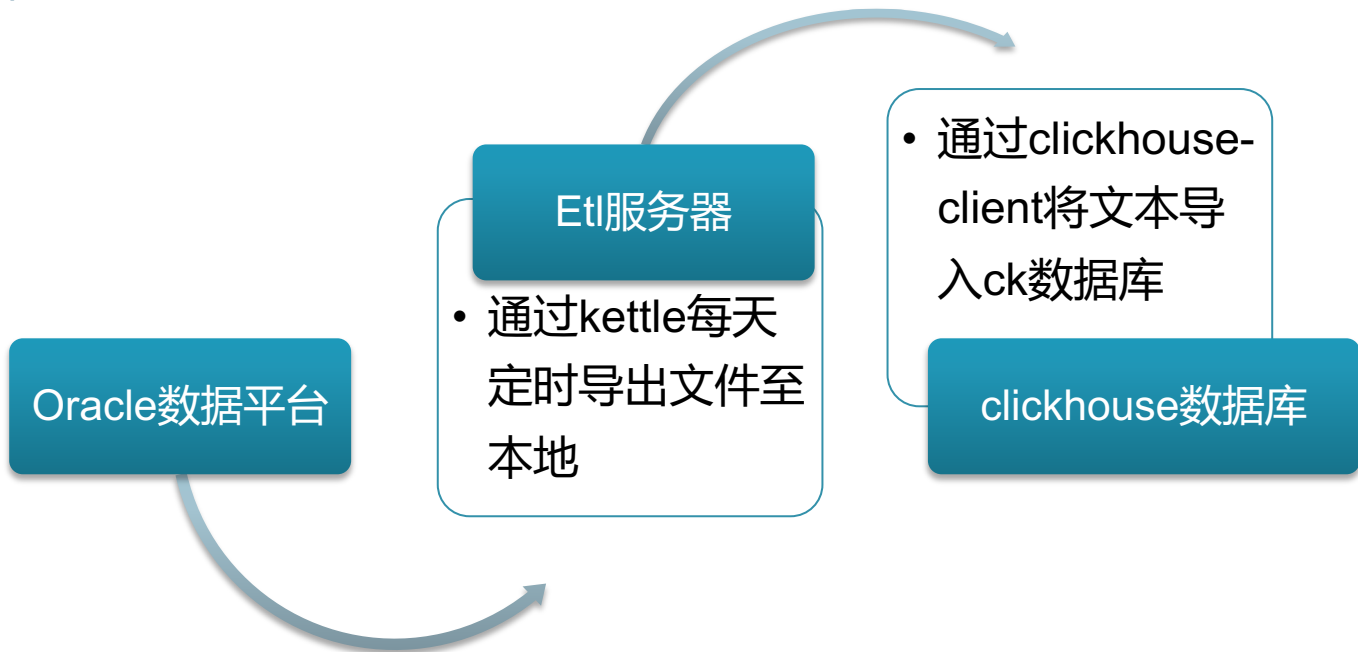
外部txt文件



# 数据同步ck

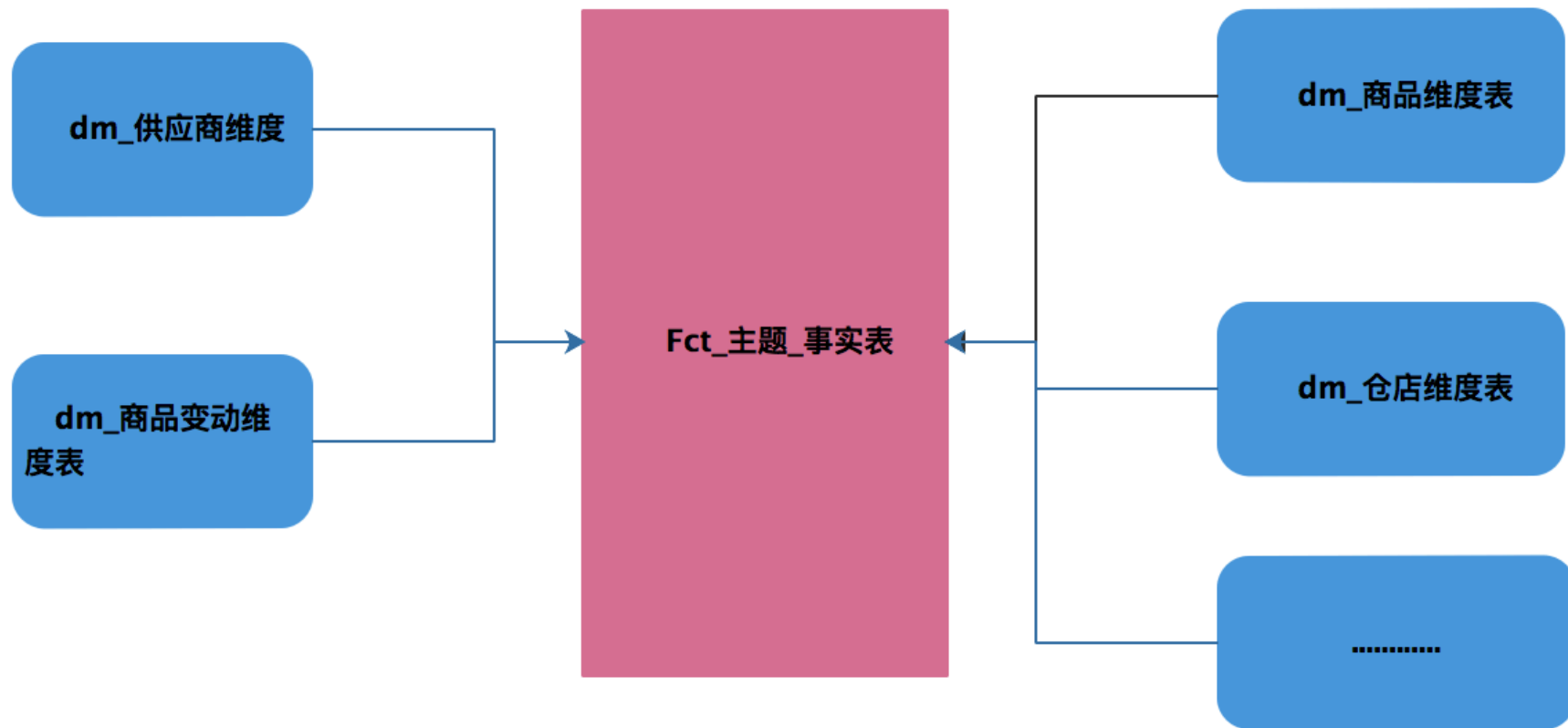
- 1, 基于公司对数据要求为T+1
2. 基于现有开发人员水平及成本

因此采用可视化同步工具kettle. 先将oracle数据平台维度信息以及相关主题清单数据同步至clickhouse数据库



# 数仓建设

ck数仓数据模型采用星型模型搭建





## 数仓建设-维度表

一般维度表数据量不大. 目前采用的是引擎Log+字典表(dictionary)

```
create table dw_hr.dw_shop_info
(
  shopid String,
  shopcode String,
  shopname String,
  address String,
  buid UInt8,
  inbuid UInt8,
  shopformid UInt8,
  shofortype UInt8,
  shopstatus UInt8,
  buname String,
  categorytreeid UInt8,
  payshopid String,
  regionid UInt64,
  provinces_regions String,
  propertytype String,
  dctype UInt8,
  prov_name String,
  city_name String
)
engine = Log
```

```
<syndex>
<dictionary>
  <name>dw_shop</name>
  <source>
    <clickhouse>
      <host>192.168.1.103</host>
      <port>9000</port>
      <user>default</user>
      <password>1q2w3e</password>
      <db>dw_hr</db>
      <table>dw_shop_info</table>
    </clickhouse>
  </source>
  <lifetime>
    <min>200</min>
    <max>260</max>
  </lifetime>
  <layout>
    <complex_key_hashed/>
  </layout>
  <structure>
    <key>
      <attribute>
        <name>shopid</name>
        <type>String</type>
      </attribute>
    </key>
    <attribute>
      <name>buid</name>
      <type>UInt8</type>
      <null_value>0</null_value>
    </attribute>
    <attribute>
      <name>buname</name>
      <type>String</type>
      <null_value>未知</null_value>
    </attribute>
    <attribute>
      <name>categorytreeid</name>
      <type>UInt8</type>
      <null_value>0</null_value>
    </attribute>
    <attribute>
      <name>shopname</name>
```

## 数仓建设 - 主题事实清单表

主题事实清单表采用引擎MergeTree. 同步策略: 每日从oracle数据平台增量同步到ck数仓.

```
create table dw_hr.fct_rpt_dc_shop_vender_day
(
    stat_year UInt16,
    stat_month UInt32,
    stat_day Date,
    stat_day_str String default formatDateTime(stat_day, '%F'),
    buid UInt8,
    dc_id String,
    venderid String,
    dctype UInt8,
    shop_id String,
    shopformid UInt8,
    logistics UInt8,
    datasource UInt8,
    rpt_qty Decimal(18,4),
    rpt_boxes Decimal(18,4),
    rpt_cost Decimal(18,4),
)
engine = MergeTree PARTITION BY toYYYYMM(stat_day) ORDER BY (stat_day, dc_id) SETTINGS index_granularity = 8192
;
```

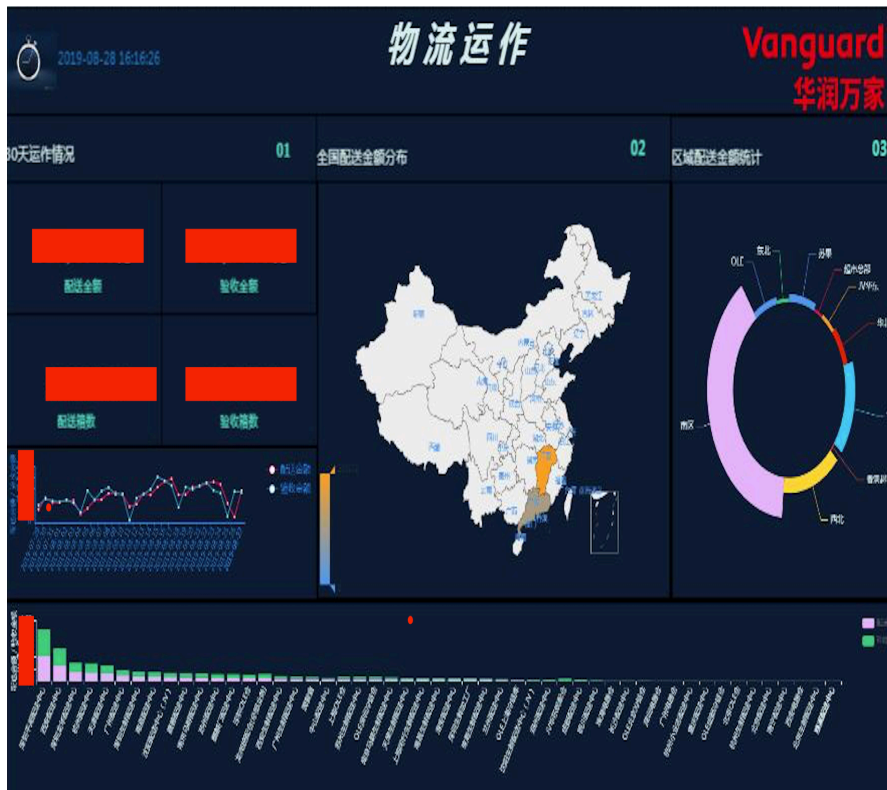
## 数仓建设-对外数据

目前对外开放是主题事实清单表+维度表 封装成一个视图,类似如下

```
create or replace view vw_fct_rpt_dc_shop_sku_vender_day as
select
    stat_year ,
    stat_month ,
    stat_day ,
    stat_day_str ,
    buid ,
    dictGet('dw_buinfo', 'buname', toUInt64(buid)) AS buid_name,
    dc_id,
    dictGet('dw_shop', 'shopname', tuple(dc_id)) AS dc_shop_name,
    dictGet('dw_shop', 'provinces_regions', tuple(dc_id)) AS dc_shop_region,
    dictGet('dw_shop', 'prov_name', tuple(dc_id)) AS dc_shop_prov,
    dictGet('dw_shop', 'city_name', tuple(dc_id)) as dc_shop_city,
    dctype ,
    shop_id ,
    dictGet('dw_shop', 'shopname', tuple(shop_id)) AS order_shop_name,
    dictGet('dw_shop', 'provinces_regions', tuple(shop_id)) AS order_shop_region,
    dictGet('dw_shop', 'prov_name', tuple(shop_id)) AS order_shop_prov,
    dictGet('dw_shop', 'city name', tuple(shop_id)) as order shop city,
```

## 数据展示+多维分析

采用开源报表系统davinci 地址: <https://github.com/edp963/davinci>



数据类型

ABC dc\_provinces\_regions

ABC dc\_prov\_name

ABC dc\_city\_name

ABC dc\_id

ABC dc\_name

ABC dc\_shop\_typeid

ABC dc\_shop\_typename

ABC dctype

ABC dctype\_name

ABC shop\_id

ABC shop\_name

ABC shop\_provinces\_regions

数值型

123 rpt\_qty

123 rpt\_boxes

123 rpt\_cost

123 rpt\_taxcost

123 rpt\_zt\_boxes

123 rpt\_zt\_cost

123 rpt\_zt\_qty

123 rpt\_zt\_taxcost

123 rpt\_zs\_boxes

123 rpt\_zs\_cost

123 rpt\_zs\_qty

123 rpt\_zs\_taxcost

123 rpt\_zs\_boxname

数据源选择

数据源名称

数据源类型

数据源描述

数据源状态

数据源操作

数据源备注

数据源时间

数据源位置

数据源权限

数据源配置

数据源日志

数据源监控

数据源报警

数据源审计

数据源备份

数据源恢复

数据源迁移

数据源删除

数据源重置

数据源初始化

数据源更新

数据源同步

数据源分发

数据源集成

数据源接口

数据源服务

数据源应用

数据源平台

数据源生态

数据源社区

数据源市场

数据源产业

数据源经济

数据源文化

数据源社会

数据源政治

数据源法律

数据源道德

数据源宗教

数据源哲学

数据源科学

数据源技术

数据源艺术

数据源体育

数据源娱乐

数据源教育

数据源医疗

数据源交通

数据源通信

数据源能源

数据源环境

数据源农业

数据源工业

数据源商业

数据源金融

数据源媒体

数据源网络

数据源信息

数据源知识

数据源智慧

数据源未来

数据源梦想

数据源希望

数据源信念

数据源理想

数据源目标

数据源追求

数据源奋斗

数据源努力

数据源坚持

数据源毅力

数据源勇气

数据源力量

数据源信心

数据源决心

数据源恒心

数据源耐心

数据源细心

数据源专心

数据源诚心

数据源虚心

数据源爱心

数据源信心

数据源决心

数据源恒心

数据源耐心

数据源细心

数据源专心

数据源诚心

数据源虚心

数据源爱心

查询

数据

样式

配置

维度

stat\_month

stat\_day

stat\_day\_num

dc\_buid

dc\_buname

dc\_provinces\_regions

dc\_prov\_name

dc\_city\_name

dc\_id

dc\_name

dc\_shop\_typeid

dc\_shop\_typename

指标

[总计] rpt\_boxes

[总计] rpt\_cost

[总计] rpt\_taxcost

[总计] rpt\_zt\_boxes

[总计] rpt\_zt\_cost

stat\_month

stat\_day

暂无数据



03

实践遇到的问题

## 1. Memory limit (for query) exceeded

解决：通过在users.xml 配置

max\_bytes\_before\_external\_sort

max\_bytes\_before\_external\_group\_by

## 2. 用户并发量一上来,负载太高

解决：目前是在中间加redis缓存