

营收-商城订单事实表[完成]

由 景涛创建, 最终由 侯宪贺修改于 十一月 14, 2022

2020年12月末，上线新版商城；相比老版商城，业务逻辑，表结构等存在诸多变化。
 该文档记录了业务变更日志，以及新版数仓商城相关事实表的设计、使用、变迁等。

- 1. 使用说明
 - 1.1 概述
 - 1.2 表简介
 - 表详细设计
 - 1.3 修改方法
 - 1.4 常用指标统计
 - 1.4.1 下单用户
 - 1.4.2 订单数，下单用户数，实付金额指标
 - 1.4.3 C币、优惠券、企业储值、兑换码金额
 - 1.4.4 IOS账户余额，人民币等支付金额
 - 1.5 注意事项
- 2. 变更日志
- 3. 订单相关事实表设计
 - 3.1 高层模型
 - 3.2 ETL说明
- 4. 案例
 - 4.1 获得最近30天曾购买电子书会员的用户
 - 4.2 统计最近一周曾购买大课、非大课、VIP商品的下单用户，订单数，以及实付款金额
 - 4.3 统计正版资源带来的订单量

1. 使用说明

1.1 概述

dwd_mall相关表，只保存老版商场数据。
 而dwd_mallv2相关表，融合了新版和老版商场数据。
 因此设计到dwd_mall_orders_info表的SQL统计脚本，都需要进行改动。

1.2 表简介

现包含订单，下单，支付，退款4张事实表。

表名	粒度	说明
dwd_mallv2_order_info	分期订单粒度，一行代表一个分期支付订单。	对于拆单情况，只保存主订单。 对于分期订单，为了兼容老商城，因此将保存多条记录；
dwd_mallv2_order_detail_info	商品粒度，一行代表一个商品。	一个订单可以包含购买多个商品，1对多关系。 商品针对子订单，因此该表只保留子订单。

以上事实表遵守以下规则：

- 时间字段
将10位、13位时间戳统一转化为10位时间戳类型。
- 金额字段
统一以 '元' 为单位。

表详细设计

对应表结构如下。

```
----- 订单维度表 -----
CREATE TABLE IF NOT EXISTS dwd_mallv2_order_info
(
    order_id          BIGINT COMMENT '订单id',
    order_number      STRING COMMENT '订单号',
    total_price       DOUBLE COMMENT '原价',
    discount_price    DOUBLE COMMENT '身份折扣后价格',
    deduction_price   DOUBLE COMMENT '抵扣价格(C币、余额等)',
    freight           DOUBLE COMMENT '运费',
    real_price        DOUBLE COMMENT '实付金额, 不包含余额支付',
    paid_price        DOUBLE COMMENT '已付金额',
    refund_price      DOUBLE COMMENT '退款金额',
    csdn_username     STRING COMMENT '订单所有人',
    order_status      BIGINT COMMENT '订单状态1、等待付款, 2、取消订单, 3支付成功, 4已经发货, 5订单完成, 6订单关闭, 7已付首付, 8正在审核, 9等待退款, 10退款成功, 11退款失败',
    order_type        BIGINT COMMENT '订单类型',
    is_instalment     BIGINT COMMENT '是否分期付款1否2是',
    flag              BIGINT COMMENT '订单业务线 1 C币商城 2 普通商城 3 学院 4 活动 5 C币充值 6 vip下载 7 程序员杂志 8电子杂志 9 图文课 10 兑换码 11 企业储值 12 问答 13 其他',
    session_id        STRING COMMENT 'ssid',
    order_source      BIGINT COMMENT '订单来源: 0为请求时为设置来源, 其他数字值与flag一致, 比如从学院下的订单购买的是商城的商品值为3',
    anonymous_purchase BIGINT COMMENT '是否匿名购买订单, 1、是, 2、否',
    is_comment        BIGINT COMMENT '是否评价',
    create_time       BIGINT COMMENT '订单创建时间',
    update_time       BIGINT COMMENT '最后修改时间',
    pay_time          BIGINT COMMENT '支付时间',
    express_time      BIGINT COMMENT '邮寄时间',
    confirm_time      BIGINT COMMENT '邮寄时间',
    finish_time       BIGINT COMMENT '完成时间',
    expire_time       BIGINT COMMENT '过期时间',
    comment_time      BIGINT COMMENT '评价时间',
    create_ip         STRING COMMENT '创建IP',
    contact_id        BIGINT COMMENT '联系人信息',
    real_name         STRING COMMENT '收货人真实姓名',
    mobile            STRING COMMENT '收货人手机号',
    email             STRING COMMENT '收货人邮箱',
    province          STRING COMMENT '收货人省份',
    city              STRING COMMENT '收货人城市',
    company           STRING COMMENT '收货人公司',
    position          STRING COMMENT '收货人职位',
    address           STRING COMMENT '收货人地址',
    alias             STRING COMMENT '收货人地址别名',
    order_source_id   BIGINT COMMENT '订单来源表ID',
    request_domain    STRING COMMENT '下单页域名',
```

```

request_url      STRING COMMENT '下单页地址',
referer_domain  STRING COMMENT '下单页ref域名',
referer_url     STRING COMMENT '下单页ref地址',
device_id       STRING COMMENT '设备唯一ID',
device_type     STRING COMMENT '设备分类',
event_type      STRING COMMENT '活动类型',
event_id        STRING COMMENT '活动ID',
share_username  STRING COMMENT '活动受益者csdn账号',
order_extend_id BIGINT COMMENT '订单拓展表ID',
c_first_ref     STRING COMMENT '落地页refer',
c_first_page    STRING COMMENT '落地页url',
dc_sid          STRING COMMENT 'SESSION id',
c_utm_source    STRING COMMENT '媒体平台参数，一般用于标识广告投放的广告主、网站等信息，该参数为必填的物料信息',
c_utm_medium    STRING COMMENT '计划名称参数，一般用于标识广告所属的推广计划信息，只有设置了推广计划信息，才可以设置推广单元信息',
c_utm_campaign  STRING COMMENT '单元名称参数，一般用于标识广告所属的推广单元信息，只有设置了推广单元信息，才可以设置关键词和创意信息',
c_utm_term      STRING COMMENT '关键词参数，一般用于标识触发广告的关键词信息',
c_utm_content   STRING COMMENT '创意参数，一般用于标识广告的创意形式信息',
c_page_id       STRING COMMENT '页面类型ID',
c_mod           STRING COMMENT '点击事件ID',
spm             STRING COMMENT 'SPM页面跟踪码',
c_ref           STRING COMMENT '订单来源前一页',
order_resource  STRING COMMENT '订单来源',
use_balance_price DOUBLE COMMENT '我的余额'
)
COMMENT '订单汇总信息表';

---- 订单明细事实表 ----
CREATE TABLE IF NOT EXISTS dwd_mallv2_order_detail_info
(
    order_detail_id BIGINT COMMENT '主键',
    biz_no           STRING COMMENT '业务编号',
    order_id         BIGINT COMMENT '订单ID',
    order_number     STRING COMMENT '订单编号',
    parent_order_id  BIGINT COMMENT '主订单ID 默认主订单',
    parent_order_number STRING COMMENT '主订单号',
    csdn_username    STRING COMMENT '订单所有人',
    order_status     BIGINT COMMENT '订单状态1、等待付款，2、取消订单，3支付成功，4已经发货，5订单完成，6订单关闭，7已付首付，8正在审核，9等待退款，10退款成功，11退款失',
    order_type       BIGINT COMMENT '订单类型',
    is_instalment    BIGINT COMMENT '是否分期付款1否2是',
    session_id       STRING COMMENT 'ssid',
    order_source     BIGINT COMMENT '订单来源：0为请求时为设置来源，其他数字值与flag一致，比如从学院下的订单购买的是商城的商品值为3',
    anonymous_purchase BIGINT COMMENT '是否匿名购买订单，1、是，2、否',
    is_comment       BIGINT COMMENT '是否评价',
    create_time      BIGINT COMMENT '订单创建时间',
    update_time      BIGINT COMMENT '最后修改时间',
    pay_time         BIGINT COMMENT '支付时间',
    express_time     BIGINT COMMENT '邮寄时间',
    confirm_time     BIGINT COMMENT '邮寄时间',
    finish_time      BIGINT COMMENT '完成时间',
    expire_time      BIGINT COMMENT '过期时间',
    comment_time     BIGINT COMMENT '评价时间',
    create_ip        STRING COMMENT '创建IP',
    contact_id       BIGINT COMMENT '联系人信息',

```

```

real_name      STRING COMMENT '收货人真实姓名',
mobile         STRING COMMENT '收货人手机号',
email          STRING COMMENT '收货人邮箱',
province       STRING COMMENT '收货人省份',
city           STRING COMMENT '收货人城市',
company        STRING COMMENT '收货人公司',
position        STRING COMMENT '收货人职位',
address        STRING COMMENT '收货人地址',
alias          STRING COMMENT '收货人地址别名',
request_domain STRING COMMENT '下单页域名',
request_url    STRING COMMENT '下单页地址',
referrer_domain STRING COMMENT '下单页ref域名',
referrer_url   STRING COMMENT '下单页ref地址',
device_id      STRING COMMENT '设备唯一ID',
device_type    STRING COMMENT '设备分类',
event_type     STRING COMMENT '活动类型',
event_id       STRING COMMENT '活动ID',
share_username STRING COMMENT '活动受益者csdn账号',
c_first_ref    STRING COMMENT '落地页refer',
c_first_page   STRING COMMENT '落地页url',
dc_sid         STRING COMMENT 'SESSION id',
c_utm_source   STRING COMMENT '媒体平台参数, 一般用于标识广告投放的广告主、网站等信息, 该参数为必填的物料信息',
c_utm_medium   STRING COMMENT '计划名称参数, 一般用于标识广告所属的推广计划信息, 只有设置了推广计划信息, 才可以设置推广单元信息',
c_utm_campaign STRING COMMENT '单元名称参数, 一般用于标识广告所属的推广单元信息, 只有设置了推广单元信息, 才可以设置关键词和创意信息',
c_utm_term     STRING COMMENT '关键词参数, 一般用于标识触发广告的关键词信息',
c_utm_content  STRING COMMENT '创意参数, 一般用于标识广告的创意形式信息',
c_page_id      STRING COMMENT '页面类型ID',
c_mod          STRING COMMENT '点击事件ID',
spm            STRING COMMENT 'SPM页面跟踪码',
c_ref          STRING COMMENT '订单来源前一页',
order_resource STRING COMMENT '订单来源',
flag           BIGINT COMMENT '业务线',
product_id     STRING COMMENT '产品ID',
product_type   BIGINT COMMENT '产品类型',
product_name   STRING COMMENT '产品名称',
card_type      STRING COMMENT '会员卡类型',
goods_id       STRING COMMENT '条目ID',
goods_type     BIGINT COMMENT '商品类型',
goods_category_id STRING COMMENT '商品分类',
goods_name     STRING COMMENT '商品名称',
goods_owner    STRING COMMENT '商品作者',
shop_id        STRING COMMENT '店铺ID',
delivery_way   BIGINT COMMENT '交付方式 1:线上 2:线下',
goods_buy_num  BIGINT COMMENT '商品购买数量',
goods_single_price DOUBLE COMMENT '商品单价',
total_price    DOUBLE COMMENT '原价',
discount_price DOUBLE COMMENT '身份折扣后价格',
deduction_price DOUBLE COMMENT '抵扣价格(C币、余额等)',
real_price     DOUBLE COMMENT '真实价格',
can_refund_price DOUBLE COMMENT '可退款金额',
use_balance_price DOUBLE COMMENT '我的余额'
)
COMMENT '订单明细事实表';

```

1.3 修改方法

场景	修改方式
获取 下单用户 统计 下单用户数、订单数、 实付金额、实付金额+我的余额	可以直接使用dwd_mallv2_order_info替换dwd_mall_orders_info表。 (dwd_mallv2_order_info字段少，但是存在的字段其含义与老表保持一致)
统计 商品相关指标 使用 商品字段 过滤数据	使用dwd_mallv2_order_detail_info替换dwd_mall_orders_info表。 (dwd_mallv2_order_detail_info保存商品相关下单明细)

1.4 常用指标统计

1.4.1 下单用户

1) 获取下单用户。

```
SELECT
    distinct tolower(csdn_username) as username
FROM
    csdn.dwd_mallv2_order_info  --
WHERE
    order_status in (3,4,5)
AND
    flag = 6
```

1.4.2 订单数，下单用户数，实付金额指标

该3个原子指标，需要从新表dwd_mallv2_order_detail_info或者dwd_mallv2_order_info获取数据，进行聚合统计。
dwd_mallv2_order_detail_info表只保留子订单，而dwd_mallv2_order_info只保留主订单。
相比老表dwd_mall_orders_info表而言，基本没有变化，常用过滤字段，比如flag, order_status, card_type, pay_time, goods_type都维持原状不变。
1) 统计订单数。
这与老商城不同，由于新商城存在拆单情况，**通常情况下都是统计父订单数，因此采用parent_order_number。**

```
-- 方式一：从dwd_mallv2_order_info统计
SELECT COUNT(DISTINCT order_number) FROM dwd_mallv2_order_info

-- 方式二：从dwd_mallv2_order_detail_info统计
SELECT COUNT(DISTINCT parent_order_number) FROM dwd_mallv2_order_detail_info
```

2) 统计下单用户数。

```
-- 方式一：从dwd_mallv2_order_info统计
SELECT COUNT(DISTINCT csdn_username) FROM dwd_mallv2_order_info

-- 方式二：从dwd_mallv2_order_detail_info统计
SELECT COUNT(DISTINCT csdn_username) FROM dwd_mallv2_order_detail_info
```

3) 统计实付金额，直接对real_price进行求和。

```
-- 方式一：从dwd_mallv2_order_info统计.
SELECT SUM(real_price) FROM dwd_mallv2_order_info  -- 由于该表只保存主订单，从主订单粒度累加real_price。

-- 方式二：从dwd_mallv2_order_detail_info统计
SELECT SUM(real_price) FROM dwd_mallv2_order_detail_info  -- 由于该表只保存子订单，从子订单粒度累加real_price。
```

4) 统计实付金额 + ‘我的余额’。

```
-- 方式一：从dwd_mallv2_order_info统计.
SELECT SUM(real_price + use_balance_price) FROM dwd_mallv2_order_info  -- 由于该表只保存主订单，从主订单粒度累加real_price。

-- 方式二：从dwd_mallv2_order_detail_info统计
SELECT SUM(real_price + use_balance_price) FROM dwd_mallv2_order_detail_info  -- 由于该表只保存子订单，从子订单粒度累加real_price。
```

1.4.3 C币、优惠券、企业储值、兑换码金额

以上5种支付信息，从dwd_mall_orders_info表迁移至dwd_mallv2_order_pay_info表，对应不同的currency币种取值。
也就是说，dwd_mall_orders_info表中不存在以上支付相关字段，比如cb_price、use_stored_value_amount、coupon_price等字段，
如想获取这些信息，需要关联dwd_mallv2_order_pay_info表。（use_balance_price除外，该字段已冗余）

currency	币种	币种面值	替换dwd_mall_orders_info字段
01	C币	CAST(currency_value AS DOUBLE)	cb_price
03	企业储值	CAST(currency_value AS DOUBLE)	use_stored_value_amount 乘以 stored_value_discount
21	优惠券	currency_value保存的不是金额，而是 优惠券 key值，需要关联优惠券表获取金额	coupon_key
23	兑换码	currency_value保存的不是金额，而是 兑换码 key值，需要关联兑换码表获取金额	redeem_code
7000	我的余额	CAST(currency_value AS DOUBLE)	use_balance_price

1) 获取 ‘我的余额’ 支付金额。

等同于dwd_mall_orders_info表中的use_balance_price，更方便的方式是直接dwd_mallv2_order_detail_info表获取。

```
SELECT order_number, CAST(currency_value AS DOUBLE) AS use_banlance_price
```

```
FROM csdn.dwd_mallv2_order_pay_info
where currency = '7000' -- 币种为'我的余额'
```

此外，在进行join的时候，需要使用parent_order_number进行关联；

获取每个订单的实付金额 + ‘我的余额’（替换dwd_mall_orders_info表中的use_banlance_price + real_price）。

```
SELECT
    '成单金额' as itype,
    case
        when sum(a.real_price + nvl(cast(b.currency_value as double), 0)) is null then 0
        else sum(a.real_price + nvl(cast(b.currency_value as double), 0))
    end as icount
FROM (
    SELECT parent_order_number, SUM(real_price) as real_price
    FROM csdn.dwd_mallv2_order_detail_info
    WHERE order_status IN (3, 4,5)
    AND flag = '26'
    AND to_char(from_unixtime(pay_time), 'YYYY-MM-DD') = '${pt}'
    GROUP BY parent_order_number -- 注意：考虑拆单情况，因此需要通过parent_order_number进行聚合
) a
LEFT JOIN (
    SELECT order_number, currency_value
    FROM csdn.dwd_mallv2_order_pay_info
    where currency = '7000' -- 币种为'我的余额'
) b
ON a.parent_order_number = b.order_number
```

1.4.4 IOS账户余额，人民币等支付金额

新版pay_type只有当币种为人民币时，才有效。

1) 获取使用 'ISO账户余额' 支付的订单。

```
SELECT * FROM dwd_mallv2_order_pay_info WHERE currency = '100'
```

1.5 注意事项

公式一： $real_price = discount_price - discount_price + freight$

公式二：主订单金额 = 所有子订单对应金额之和

2. 变更日志

- 购物车
新商城引入购物车特性，单个订单可以购买多个商品。
- 币种
新商城将优惠券、兑换码、现金、余额、企业储值、其他收益统一抽象为币种。
一个订单可以使用多种币种进行支付。
具体取值参考下列截图。

ods_mall_mp_pay_tb_wallet_code

查询时间: 0.003128 sec 展期时间: sec				
id	pay_code	name	wallet_code	type
1	01	C币	100	coin
2	02	UCS账户余额	1000	coin
3	03	企业储值	2000	coin
4	21	优惠券		coupon
5	22	优惠券		coupon
6	23	兑换码		coupon
7	4001	博悦VIP可见文章账户	4001	rmb
8	3000	@/收账户	3000	rmb
9	4002	博悦付费专栏账户	4002	rmb
10	4003	学院收益账户	4003	rmb
11	6000	分销平台收益账户	6000	rmb
12	6001	分销平台-学院账户	6001	rmb
13	6002	分销平台-会员账户	6002	rmb
14	6003	分销平台-gitee账户	6003	rmb
15	7000	我的余额	7000	rmb
16	6004	分销博悦专栏	6004	rmb
17	7001	我的收益	7001	rmb
18	4004	红包收益	4004	rmb
19	4005	问答收益	4005	rmb
20	4006	博悦打赏收益	4006	rmb
21	4007	下载vip分成收益	4007	rmb
22	4999	待提现收益	4999	rmb
23	4008	下载付费资源收益	4008	rmb
24	8000	人民币贷款账户	8000	rmb
25	8001	虚拟货币贷款账户	8001	coin

- id
新商城order_id，以及其他id，统一从1开始自增；
这导致在合并过程中，与老版商城数据冲突。因此需要对新商城id进行特殊处理。
目前针对新版商城采用1000000000+id方式，避免id冲突。
- order_number
订单编号位数同老版商城不同；老版17位，新版23位。
建议关联订单相关业务表时，通过order_number进行关联。
- 金额单位
老版统一采用‘元’为单位；新版统一采用‘分’为单位。
- 时间精度
老版时间戳为10位，精确到秒。
新版时间戳为13位，精确到毫秒。
- 拆单
新版商城存在拆单情况，一个父订单按照店铺会拆分为若干个子订单；
子订单的order_id和order_number同父订单不同。
在统计订单个数的时候，基于父订单进行统计。
- 枚举值
大部分枚举值无变化，部分变化如下。
is_instalment: 新版同老版取值正好相反，新版 1是 2否；老版 2是 1否。
flag: 增加其他取值。

3. 订单相关事实表设计

3.1 高层模型

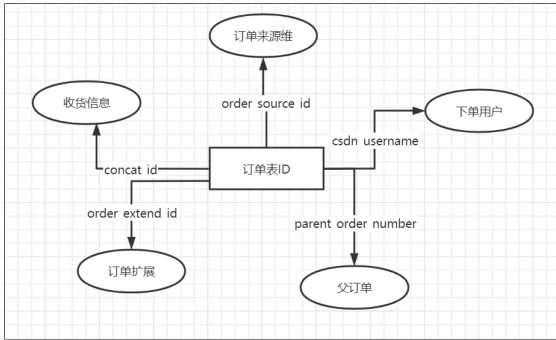


图2.2.1 订单高层模型图

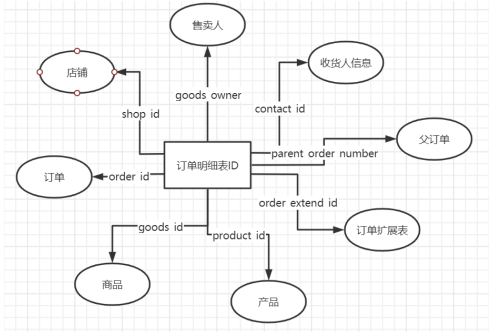


图2.2.2 订单明细高层模型图

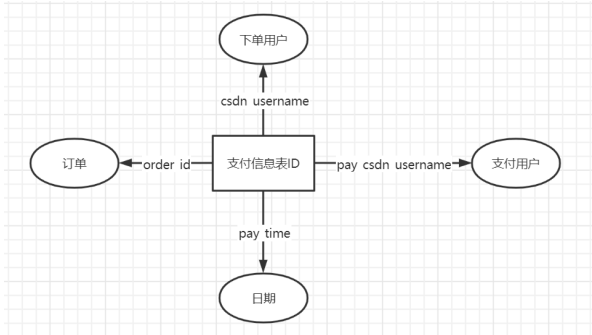


图2.2.3 支付日志高层模型图

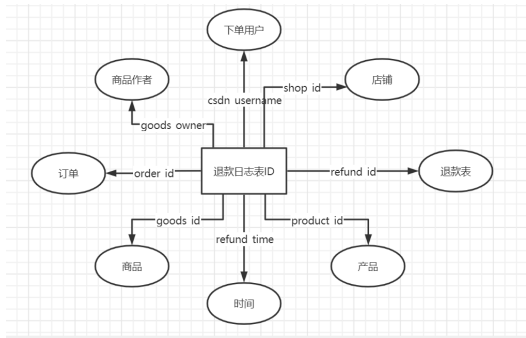


图2.2.4 退款日志高层模型图

3.2 ETL说明

ETL脚本同表名一致，更加详细的信息，可参考maxcompute内对应ETL脚本。

ETL脚本名	字段	说明
dwd_mallv2_order_info	order_id, contact_id, order_extend_id	V1 ID保持不变，V2 ID需要增加1000000000。 老版：不变 新版：1000000000 + 原ID
	create_time update_time refund_time pay_time comment_time expire_time finish_time confirm_time	时间字段，V1 10位时间戳，V2 13位时间戳。 老版：不变 新版：时间戳 / 1000, 转化为10位
	total_price discount_price deduction_price freight real_price paid_price refund_price	金额字段，V1 以‘元’为单位；V2 以‘分’为单位。 老版：不变 新版：金额 / 100. 转化为‘元’
	referer_url	处理v2版本的app端订单对应的referer_url，按照v1版本规则进行处理： 1、把不规范数据置为null referer_url='(null)' OR referer_url LIKE '%Activity' OR referer_url LIKE '%Controller' OR referer_url="" 2、关联dim_app_order_sourcepage_type表，根据is_match进行数据处理： CASE WHEN b.is_match=1 THEN b.match_url WHEN b.is_match=2 THEN a.refer ELSE a.referer_url END AS referer_url

dwd_mallv2_order_detail_info	order_detail_id	V1 ID保持不变, V2 ID需要增加1000000000。 老版: 不变 新版: 1000000000 + 原ID
	goods_single_price total_price discount_price deduction_price real_price can_refund_price	(1) 金额字段, V1 以 '元' 为单位; V2 以 '分' 为单位。 老版: 不变 新版: 金额 / 100. 转化为 '元' (2) real_price, V1 不存在需进行计算, 计算逻辑如下: 该商品价格 / 该订单所有商品价格 (单价*个数) * 该订单实付价格 (3) discount_price, V1 中的total_price即为折扣价格。 (4) deduction_price, 计算逻辑如下: 该商品价格 / 该订单所有商品价格 (单价*个数) * 总抵扣价格 (折扣价格+ 运费 - 实付价格)
	card_type product_type product_name	ods_mall_orders_order_detail_base表通过product_id关联ods_mall_mall_product_base表。 ods_mall_mall_product_base通过tag_id关联ods_mall_mall_tags_base表。 card_type取值逻辑如下: CASE WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%年卡%' then '年卡' WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%月卡%' then '月卡' WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%季卡%' then '季卡' WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%半年卡%' then '半年卡' WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%超级年卡%' then '超级年卡' WHEN CONCAT_WS(',', COLLECT_SET(tags.tags_name)) like '%两年卡%' then '两年卡' ELSE '否' END AS card_type
	goods_id	需要对v2库里goods_id做清洗, 统一增加前缀1000000000, v1库里的 flag=24,25,27,28,29,30,31,33 的商品信息, 需要把goods_id更新为goods_attr中的goods_id, 并 增加前缀1000000000, 如果获取商品信息, 需要关联mp_mall_goods库里的 tb_mall_goods_item表, 用detail.goods_id=item.id。mp_mall_goods库里的tb_mall_goods_item 表的id字段也需要增加统一前缀1000000000 使用说明: mallv2_tb_order_detail中获取商品更多信息, 关联mp_mall_goods库中的 tb_mall_goods_item v1版本 orders库中的order_details中获取更多商品信息, 需要根据flag进行区分, 并用goods_id 直接关联对应业务表: 1、学院课程: flag=3关联dm_edu_course中的goods_id 2、付费专栏: flag=17 关联ods_blog_blog_column_subscribe_base中的column_id 3、其他flag需要补充。
	request_url	1、自动续费和企业支付订单, 置为null 2、app下单, url不完整, 补全url 3、20220913兼容支付前置订单, 判定utm_source_ext字段中是否存在loginPay属性, 如果存在, 取scanLoginReferer字段的值替换request_url
	过滤	对于套餐包商品。 只保留

4. 案例

4.1 获得最近30天曾购买电子书会员的用户

```
SELECT
    tolower(csdn_username) csdn_username,
    to_char(min(pay_time), 'yyyy-mm-dd') pay_time
FROM
    csdn.dwd_mallv2_order_info
WHERE
    flag = 28
    AND to_char(from_unixtime(pay_time), 'yyyy-mm-dd') <= '2020-01-03' -- pay_time为BIGINT类型
GROUP BY tolower(csdn_username)
HAVING pay_time>='2020-12-03'
```

4.2 统计最近一周曾购买大课、非大课、VIP商品的下单用户，订单数，以及实付款金额

```
SELECT
    nvl(case when flag = 3 then case when goods_type = 10 then 1 else 3 end when flag in (5,6) then 2 else 0 end,-1) flag,
    count(distinct csdn_username) paid_user,
    count(distinct parent_order_number) paid_order, -- parent_order_number统计父订单数
    sum(real_price) total_paid -- 实付金额，已经进行处理，可以直接进行sum
FROM
    csdn.dwd_mallv2_order_detail_info
WHERE
    to_char(from_unixtime(pay_time), 'YYYY-MM-DD') between '2020-02-27' and '2020-01-03' -- pay_time为BIGINT类型
    AND
        order_status >= 3
    AND
        order_status != 6
    AND
        real_price > 0
    AND
        flag != 19
GROUP BY
    case when flag = 3 then case when goods_type = 10 then 1 else 3 end when flag in (5,6) then 2 else 0 end
WITH
    CUBE
```

4.3 统计正版资源带来的订单量

```
SELECT
    '成单金额' as itype,
    case
        when sum(a.real_price + nvl(cast(b.currency_value as double), 0)) is null then 0
        else sum(a.real_price + nvl(cast(b.currency_value as double), 0))
    end as icount
FROM (
```

```
SELECT parent_order_number, SUM(real_price) as real_price
FROM csdn.dwd_mallv2_order_detail_info
WHERE order_status IN (3, 4,5)
AND flag = '26'
AND to_char(from_unixtime(pay_time), 'YYYY-MM-DD') = '${pt}'
GROUP BY parent_order_number -- 注意: 考虑拆单情况, 因此需要通过parent_order_number进行聚合
) a
LEFT JOIN (
SELECT order_number, currency_value
FROM csdn.dwd_mallv2_order_pay_info
where currency = '7000' -- 币种为'我的余额'
) b
ON a.parent_order_number = b.order_number
```

无标签