01.大促复盘代码

注：零售经分已统一建立BC订单分离，故C端用户、订单提取逻辑变化，本页面所有涉及C端用户订单提取的sql均参考新逻辑自行匹配更新，新逻辑详见该页面：零售经分官方BC分离用户

财务销售

http://bdp.jd.com/ide/task/job_detail.html?jobId=8544413&timeRange=primary

```
--
-- Map  True
set hive.merge.mapfiles = true;
-- Reduce  False
set hive.merge.mapredfiles = true;
--
set hive.merge.size.per.task = 256000000;
--map-reducemerge
set hive.merge.smallfiles.avgsize = 256000000;
-- spark
set spark.sql.hive.mergeFiles = true;
set hive.exec.parallel = true;
--Reduce
set hive.exec.reducers.bytes.per.reducer = 2000000000;
--
set hive.map.aggr = true;
set hive.groupby.mapaggr.checkinterval = 100000;
set hive.auto.convert.join = true;

set hive.exec.parallel = true;
--Reduce
set hive.exec.reducers.bytes.per.reducer = 2000000000;
--
set hive.map.aggr = true;
set hive.groupby.mapaggr.checkinterval = 100000;
set hive.auto.convert.join = true;
SELECT
        statdate ,
        type ,
        double_type ,
        pur_first_dept_name ,
        pur_second_dept_name ,
        pur_third_dept_name ,
        item_first_cate_name ,
        item_second_cate_name ,
        item_third_cate_name ,
        user_flag ,
        order_type ,
        sum(gmv) gmv,
        sum(net_revenue) ,
        sum(sale_gross)
FROM
        app.v_app_pur_morning_report_gmv_rev_sum_arc_dsc_xfpchenbao
where
        dt in('2021-03-01', '2020-03-01')  --dt
        and
        (
                statdate between '2020-03-01' and '2020-03-08'  --statdate
        or statdate between '2021-03-01' and '2021-03-08'
        )
group by
        statdate,
        type,
        double_type,
        item_first_cate_name,
        item_second_cate_name,
        item_third_cate_name,
        pur_first_dept_name,
        pur_second_dept_name,
        pur_third_dept_name,
        order_type,
        user_flag
```

TOC零售GMV剔除商仓

```
```

```
SELECT
        statdate,
        bu_name,
        dept_name_1,
        dept_name_2,
        dept_name_3,
        SUM(item_qtty) AS cw_quantity,
        SUM(gmv_sum) cw_gmv
FROM
        (
                SELECT
                        bu_name,
                        dept_id_1,
                        dept_id_2,
                        dept_id_3,
                        dept_name_1,
                        dept_name_2,
                        dept_name_3,
                        barndname_full,
                        item_sku_id,
                        item_first_cate_cd,
                        item_first_cate_name,
                        item_second_cate_cd,
                        item_second_cate_name,
                        item_third_cate_cd,
                        item_third_cate_name,
                        major_supp_brevity_code,
                        major_supp_name,
                        shop_id,
                        regexp_replace(regexp_replace(trim(shop_name), '"', ''), ',', '') shop_name,
                        main_brand_code,
                        regexp_replace(regexp_replace(trim(main_barndname_full), '"', ''), ',', '')
main_barndname_full,
                        pop_vender_id,
                        pop_vender_name,
                        data_type,
                        pop_operator_erp_acct,
                        CASE
                                WHEN data_type = 1
                                THEN ''
                                WHEN data_type = 3
                                THEN 'POP'
                        END MODE
                FROM
                        gdm.gdm_m03_sold_item_sku_da
                WHERE
                        dt = sysdate( - 1)
                        AND bu_id = '1727'
                        AND data_type IN('1', '3')
                        -- AND dept_id_2 IN('46', '48', '1868', '5011', '4268') ------
        )
        t
JOIN
        (
                SELECT
                        dt,
                        statdate,
                        fin_ord_type_cd,
                        sale_order_id,
                        item_sku_id,
                        gmv_sum,
                        item_qtty,
                        parent_sale_ord_id
                FROM
                        app.v_app_s12_fin_fullkpis_income_gmv_det_sum_dsc
                WHERE
                        dt IN('2021-08-01', '2020-08-01', '2019-08-01')
                        AND user_flag = 'TOC_'
        )
        a
ON
```

```
        t.item_sku_id = a.item_sku_id
GROUP BY
        statdate,
        bu_name,
        dept_name_1,
        dept_name_2,
        dept_name_3 grouping sets((statdate, bu_name, dept_name_1, dept_name_2, dept_name_3),(statdate,
bu_name, dept_name_1, dept_name_2),(statdate, bu_name, dept_name_1),(statdate, bu_name))
```

用户

http://bdp.jd.com/ide/task/job_detail.html?jobId=8545046&timeRange=primary

```
set hive.exec.parallel = true;
--Reduce
set hive.exec.reducers.bytes.per.reducer = 2000000000;
--
set hive.map.aggr = true;
set hive.groupby.mapaggr.checkinterval = 100000;
set hive.auto.convert.join = true;

SELECT
        m.period ,
    case spite_user_flag when '1' then '' else '' end ,
    mode ,
        m.dept_id_1 1id,
        m.dept_name_1 1,
        m.dept_id_2 id,
        m.dept_name_2 ,
        CASE fst_all_yn
                when '1'
                then ''
                when '0'
                then ''
                else ''
        end ,
        COUNT(DISTINCT m.user_log_acct) ,
        COUNT(
                CASE
                        WHEN par_ord_num >= 2
                        THEN m.user_log_acct
                        ELSE NULL
                END) ,
        SUM(par_ord_num) ,
        SUM(ord_num) ,
        SUM(sale_qtty) ,
        SUM(amount)
FROM
        (
                SELECT
                        period,
                        COALESCE(mode, '') mode,
                        COALESCE(d.user_log_acct, a.user_log_acct) user_log_acct,
                        dept_id_1,
                        dept_name_1,
                        COALESCE(dept_id_2, 33) dept_id_2,
                        COALESCE(dept_name_2, '') dept_name_2,
                        COUNT(DISTINCT a.sale_ord_id) ord_num,
                        COUNT(DISTINCT a.parent_sale_ord_id) par_ord_num,
                        SUM(after_prefr_amount_1) amount,
                        SUM(sale_qtty) sale_qtty
                FROM
                        (
                                SELECT
```

```
                        dt period,
                        item_sku_id,
                        lower(trim(user_log_acct)) user_log_acct,
                        sale_ord_id,
                        parent_sale_ord_id,
                        after_prefr_amount_1,
                        sale_qtty
                FROM
                        app.v_adm_d04_trade_ord_det_sku_snapshot_xfp
                WHERE
                        (
                                dt between '2021-03-01' and '2021-03-08'
                                or dt between '2020-03-01' and '2020-03-08'
                        )
                        AND intraday_ord_deal_flag = '1' ----
                        AND split_status_cd NOT IN('1') --
                        AND valid_flag = '1' --
                        AND biz_flag_collect['int_pur_ord_flag'] <> 1 ----
                        AND biz_flag_collect['dist_ord_flag'] <> 1---
                        AND biz_flag_collect['corp_ord_flag'] <> 1-----
                        AND biz_flag_collect['xtl_ord_flag'] <> 1 ----
                        --AND virtual_ord_flag <> '1'---
                        and substr(ord_flag,60,3) <>'040' ----   -------------(
                        and biz_flag_collect['yhd_ord_flag'] <> 1 ----
        )
        a
JOIN
        (
                SELECT
                        dept_id_1,
                        dept_name_1,
                        dept_id_2,
                        dept_name_2,
                        dept_id_3,
                        dept_name_3,
                        item_sku_id,
                        CASE
                                WHEN data_type = 1
                                THEN ''
                                WHEN data_type = 3
                                THEN 'POP'
                        END mode
                FROM
                        gdm.gdm_m03_sold_item_sku_da
                WHERE
                        dt = sysdate( - 1)
                        AND dept_id_1 = 33
        )
        b
ON
        a.item_sku_id = b.item_sku_id
LEFT JOIN
        (
                SELECT
                        lower(trim(unif_user_log_acct)) user_log_acct,
                        lower(trim(user_acct_name)) pin
                from
                        gdm.gdm_m01_userinfo_basic_da
                where
                        dt = sysdate( - 1)
        ) ----pin
        d
ON
        a.user_log_acct = d.pin
GROUP BY
        period,
        mode,
        COALESCE(d.user_log_acct, a.user_log_acct),
        dept_id_1,
        dept_name_1,
        dept_id_2,
```

```
                        dept_name_2 grouping sets((period, COALESCE(d.user_log_acct, a.user_log_acct),
dept_id_1, dept_name_1, dept_id_2, dept_name_2),(period, COALESCE(d.user_log_acct, a.user_log_acct), dept_id_1,
dept_name_1),(mode,period, COALESCE(d.user_log_acct, a.user_log_acct), dept_id_1, dept_name_1, dept_id_2,
dept_name_2),(mode,period, COALESCE(d.user_log_acct, a.user_log_acct), dept_id_1, dept_name_1))
        )
        m
LEFT OUTER JOIN
        (
                SELECT
                        dept_id_1,
                        dept_id_2,
                        user_log_acct,
                        fst_ord_dt period,
                        fst_all_yn
                from
                        (
                                SELECT
                                        dept_id_1,
                                        COALESCE(dept_id_2, '33') dept_id_2,
                                        lower(trim(unif_user_log_acct)) user_log_acct,
                                        MIN(fst_ord_dt) fst_ord_dt, --
                                        MAX(fst_all_yn) fst_all_yn
                                FROM
                                        app.v_adm_s01_user_new_or_old_flag_detail_xfp
                                WHERE
                                        dt = sysdate( - 2)
                                        AND dept_id_1 = '33'
                                        AND tp = 'dept'
                                GROUP BY
                                        dept_id_1,
                                        dept_id_2,
                                        lower(trim(unif_user_log_acct)) grouping sets((dept_id_1, lower(trim
(unif_user_log_acct))),(dept_id_1, dept_id_2, lower(trim(unif_user_log_acct))))
                        )
                        f
                where
                        fst_ord_dt between '2021-03-01' and '2021-03-08'
                        or fst_ord_dt between '2020-03-01' and '2020-03-08'
        )
        t1
ON
        m.user_log_acct = t1.user_log_acct
        AND m.dept_id_2 = t1.dept_id_2
        AND m.dept_id_1 = t1.dept_id_1
        AND m.period = t1.period
        --poppoppoppoppop
left join
    (
        select
            lower(trim(unif_user_log_acct)) user_log_acct,
            spite_user_flag
        from
            app.v_adm_s01_user_new_or_old_flag_detail_xfp
        where
            dt = sysdate(-2)
            and spite_user_flag = 1
            and tp = 'dept'
        group by
            lower(trim(unif_user_log_acct)),
            spite_user_flag
    )
    c
on
    m.user_log_acct = c.user_log_acct
GROUP BY
        m.period,
        mode,
        case spite_user_flag when '1' then '' else '' end,
        CASE fst_all_yn
                when '1'
                then ''
```

```
                when '0'
                then ''
                else ''
        end,
        m.dept_id_1,
        m.dept_name_1,
        m.dept_id_2,
        m.dept_name_2;
```

PLUS用户

http://bdp.jd.com/ide/task/job_detail.html?jobId=8539991&timeRange=primary

```
--
-- Map  True
set hive.merge.mapfiles = true;
-- Reduce  False
set hive.merge.mapredfiles = true;
--
set hive.merge.size.per.task = 256000000;
--map-reducemerge
set hive.merge.smallfiles.avgsize = 256000000;
-- spark
set spark.sql.hive.mergeFiles = true;
set hive.exec.parallel = true;
--Reduce
set hive.exec.reducers.bytes.per.reducer = 2000000000;
--
set hive.map.aggr = true;
set hive.groupby.mapaggr.checkinterval = 100000;
set hive.auto.convert.join = true;


SELECT
        year(sale_ord_tm) ,
        dept_name_1 1,
        coalesce(dept_name_2, '') ,
        CASE
                when identity_cnt >= 1
                then 'plus'
                ELSE 'noplus'
        end as plus,
        count(distinct pin) ,
        count(distinct parent_sale_ord_id) ,
        --  sum(sale_qtty) as sale_qtty,
        sum(amount) as
from
        (
                SELECT
                        coalesce(d.user_log_acct, ord.pin) pin,
                        sale_ord_tm,
                        parent_sale_ord_id,
                        b.item_sku_id,
                        sale_qtty,
                        amount,
                        dept_name_1,
                        dept_name_2,
                        sum(
                                CASE
                                        WHEN ord.sale_ord_tm >= pluser.begin_date
                                                AND ord.sale_ord_tm <= pluser.end_real_date
                                        THEN 1
                                        ELSE 0
                                end) AS identity_cnt
                FROM
                        (
                                SELECT
```

```
                        item_sku_id,
                        lower(trim(user_log_acct)) pin,
                        sale_ord_tm,
                        parent_sale_ord_id,
                        after_prefr_amount_1 amount,
                        sale_qtty
                FROM
                        app.v_adm_d04_trade_ord_det_sku_snapshot_xfp
                WHERE
                        (
                                dt between '2021-03-01' and '2021-03-08'
                                or dt between '2020-03-01' and '2020-03-08'
                        )
                        AND intraday_ord_deal_flag = '1' ----
                        AND split_status_cd NOT IN('1') --
                        AND valid_flag = '1' --
                        AND biz_flag_collect['int_pur_ord_flag'] <> 1 ----
                        AND biz_flag_collect['dist_ord_flag'] <> 1---
                        AND biz_flag_collect['corp_ord_flag'] <> 1-----
                        AND biz_flag_collect['xtl_ord_flag'] <> 1 ----
                        --AND virtual_ord_flag <> '1'---
                        and substr(ord_flag,60,3) <>'040' ----   -------------(
                        and biz_flag_collect['yhd_ord_flag'] <> 1 ----
        )
        ord
join
        (
                select
                        item_sku_id,
                        dept_name_1,
                        dept_name_2
                from
                        gdm.gdm_m03_sold_item_sku_da
                where
                        dt = sysdate( - 1)
                        and dept_name_1 = ''
        )
        b
on
        ord.item_sku_id = b.item_sku_id
left join
        (
                select
                        lower(trim(unif_user_log_acct)) user_log_acct,
                        lower(trim(user_acct_name)) pin
                from
                        gdm.gdm_m01_userinfo_basic_da
                where
                        dt = sysdate( - 1)
        ) ----pin
        d
on
        ord.pin = d.pin
left join
        (
                SELECT
                        lower(trim(pin)) as pin,
                        begin_date,
                        end_real_date
                FROM
                        fdm.fdm_plus_n_plus_pins_stage__chain
                WHERE
                        dp = 'ACTIVE'
                        AND
                        (
                                stage IN(4000)
                                or sku_no = '100012971100'
                        )
                        AND flag <> '-1'
                        and (
                            to_date(end_real_date) >= '2021-03-01'
```

```
                                        and to_date(begin_date) <= '2021-03-08'
                                        or to_date(end_real_date) >= '2020-03-01'
                                        and to_date(begin_date) <= '2020-03-08'
                                )
                        GROUP BY
                                lower(trim(pin)),
                                begin_date,
                                end_real_date
                )
                pluser
        ON
                ord.pin = pluser.pin
        left join
                (
                        select
                                lower(trim(unif_user_log_acct)) user_log_acct
                        from
                                app.v_adm_s01_user_new_or_old_flag_detail_xfp
                        where
                                dt = sysdate( - 2)
                                and spite_user_flag = 1
                                and tp = 'dept'
                        group BY
                                lower(trim(unif_user_log_acct))
                )
                c
        on
                d.user_log_acct = c.user_log_acct
        where
                c.user_log_acct is null
        GROUP BY
                coalesce(d.user_log_acct, ord.pin),
                sale_ord_tm,
                parent_sale_ord_id,
                b.item_sku_id,
                sale_qtty,
                amount,
                dept_name_1,
                dept_name_2
        )
        a
group by
        CASE
                when identity_cnt >= 1
                then 'plus'
                ELSE 'noplus'
        end,
        year(sale_ord_tm),
        dept_name_1,
        dept_name_2 grouping sets((
                        CASE
                                when identity_cnt >= 1
                                then 'plus'
                                ELSE 'noplus'
                        end, year(sale_ord_tm), dept_name_1),(
                        CASE
                                when identity_cnt >= 1
                                then 'plus'
                                ELSE 'noplus'
                        end, year(sale_ord_tm), dept_name_1, dept_name_2))
```

商详流量

```
SET jn_begin = '2021-03-01';
SET jn_end = '2021-03-18';
```

```
SET qn_begin = '2020-03-01';
SET qn_end = '2020-03-18';
SET hive.exec.parallel = true;
SET hive.input.format = org.apache.hadoop.hive.ql.io.CombineHiveInputFormat;
SET hive.hadoop.supports.splittable.combineinputformat = true;
SET hive.exec.parallel = true;
SET hive.optimize.cp = true;
SET mapreduce.input.fileinputformat.split.maxsize = 256000000;
SET mapreduce.input.fileinputformat.split.minsize.per.node = 256000000;
SET mapreduce.input.fileinputformat.split.minsize.per.rack = 256000000;
SET hive.merge.mapfiles = true;
SET hive.merge.mapredfiles = true;
SET hive.merge.size.per.task = 256000000;
SET hive.merge.smallfiles.avgsize = 256000000;
SET hive.input.format = org.apache.hadoop.hive.ql.io.CombineHiveInputFormat;
SET mapred.min.split.size = 1000000000;
SET mapred.max.split.size = 4000000000;
SELECT
        dt,
        b.dept_name_1 AS dept_name,
        b.mode,
        SUM(sku_pv) AS pv,
        COUNT(DISTINCT browser_uniq_id) AS uv
FROM
        (
                SELECT
                        dt,
                        sku_id,
                        sku_pv,
                        browser_uniq_id
                FROM
                        adm.adm_s14_online_log_smart_item_d
                WHERE
                        (
                                (
                                        dt BETWEEN ${hiveconf:qn_begin} AND ${hiveconf:qn_end} --
                                )
                                OR
                                (
                                        dt BETWEEN ${hiveconf:jn_begin} AND ${hiveconf:jn_end} --
                                )
                        )
                        --and bs = '311210'
        )
        a
JOIN
        (
                SELECT
                        dept_id_1,
                        dept_name_1,
                        dept_id_2,
                        dept_name_2,
                        dept_id_3,
                        dept_name_3,
                        item_sku_id,
                        item_first_cate_cd,
                        item_second_cate_cd,
                        item_third_cate_cd,
                        item_first_cate_name,
                        item_second_cate_name,
                        item_third_cate_name,
                        CASE
                                WHEN data_type = 1
                                THEN ''
                                WHEN data_type = 3
                                THEN 'POP'
                        END MODE
                FROM
                        --gdm.gdm_m03_sold_item_sku_da
                        gdm.gdm_m03_mkt_item_sku_da
                WHERE
```

```
                        dt = sysdate( - 1)
                        AND dept_id_1 = 33
                        and dept_id_2 in ('425','5012')
                        and item_first_cate_cd = '12259'
                        and item_sku_id <> '100012043978'
        )
        b
ON
        a.sku_id = b.item_sku_id
GROUP BY
        dt,
        b.dept_name_1,
        b.mode
```

商详首次来源流量

```
SET jn_begin = '2021-10-20';
SET jn_end = '2021-11-13';
SET qn_begin = '2020-10-20';
SET qn_end = '2020-11-13';
SET hive.exec.parallel = true;
SET hive.input.format = org.apache.hadoop.hive.ql.io.CombineHiveInputFormat;
SET hive.hadoop.supports.splittable.combineinputformat = true;
SET hive.exec.parallel = true;
SET hive.optimize.cp = true;
SET mapreduce.input.fileinputformat.split.maxsize = 256000000;
SET mapreduce.input.fileinputformat.split.minsize.per.node = 256000000;
SET mapreduce.input.fileinputformat.split.minsize.per.rack = 256000000;
SET hive.merge.mapfiles = true;
SET hive.merge.mapredfiles = true;
SET hive.merge.size.per.task = 256000000;
SET hive.merge.smallfiles.avgsize = 256000000;
SET hive.input.format = org.apache.hadoop.hive.ql.io.CombineHiveInputFormat;
SET mapred.min.split.size = 1000000000;
SET mapred.max.split.size = 4000000000;
SET hive.map.aggr = true;
SET hive.groupby.mapaggr.checkinterval = 100000;
SET hive.auto.convert.join = true;
SET mapreduce.map.memory.mb=4096;
SET mapreduce.reduce.memory.mb=4096;
SELECT
        '' type,
        a.*
FROM
        (
                SELECT
                        YEAR(dt) YEAR,
                        b.bu_id,
                        b.bu_name,
                        COALESCE(b.dept_id_1, b.bu_id) dept_id_1,
                        COALESCE(b.dept_name_1, b.bu_name) dept_name_1,
                        COALESCE(b.dept_id_2, b.dept_id_1, b.bu_id) dept_id_2,
                        COALESCE(b.dept_name_2, b.dept_name_1, b.bu_name) dept_name_2,
                        first_src_url_first_cate_name AS v1,
                        first_src_url_second_cate_name AS v2,
                        --first_src_url_third_cate_name AS v3,
                        COUNT(1) pv,
                        COUNT(DISTINCT browser_uniq_id) uv
                FROM
                        (
                                SELECT
                                        dt,
                                        sku_id,
                                        user_log_acct,
```

```
                                        browser_uniq_id,
                                        first_src_url_first_cate_name,
                                        first_src_url_second_cate_name
                                        --first_src_url_third_cate_name
                                FROM
                                        adm.adm_d14_traffic_item_src_next_d
                                WHERE
                                        (
                                                (
                                                        dt BETWEEN ${hiveconf:qn_begin} AND ${hiveconf:qn_end}
                                                )
                                                OR
                                                (
                                                        dt BETWEEN ${hiveconf:jn_begin} AND ${hiveconf:jn_end}
                                                )
                                        )
                                        -- and dc = 'app'
                        )
                        a
                JOIN
                        (
                                SELECT
                                        item_sku_id,
                                        bu_id,
                                        bu_name,
                                        dept_name_1,
                                        dept_id_1,
                                        dept_name_2,
                                        dept_id_2
                                FROM
                                        gdm.gdm_m03_mkt_item_sku_da
                                WHERE
                                        dt = sysdate( - 1)
                                        AND bu_id = '1727'
                        )
                        b
                ON
                        a.sku_id = b.item_sku_id
                GROUP BY
                        YEAR(dt),
                        b.bu_id,
                        b.bu_name,
                        b.dept_id_1,
                        b.dept_name_1,
                        b.dept_id_2,
                        b.dept_name_2,
                        first_src_url_first_cate_name,
                        first_src_url_second_cate_name grouping sets((YEAR(dt), b.bu_id, b.bu_name,
first_src_url_first_cate_name),(YEAR(dt), b.bu_id, b.bu_name, first_src_url_first_cate_name,
first_src_url_second_cate_name),(YEAR(dt), b.bu_id, b.bu_name, b.dept_id_1, b.dept_name_1,
first_src_url_first_cate_name, first_src_url_second_cate_name),(YEAR(dt), b.bu_id, b.bu_name, b.dept_id_1, b.
dept_name_1, first_src_url_first_cate_name),(YEAR(dt), b.bu_id, b.bu_name, b.dept_id_1, b.dept_name_1, b.
dept_id_2,
                        b.dept_name_2, first_src_url_first_cate_name, first_src_url_second_cate_name),(YEAR
(dt), b.bu_id, b.bu_name, b.dept_id_1, b.dept_name_1, b.dept_id_2, b.dept_name_2,
first_src_url_first_cate_name))
        )
        a
```