final as(

        select

            -- ledger\_id, x\_terr\_id)

            cust\_account\_id as cust\_acct\_id,

            org\_id,

            ledger\_id,

            -- x\_terr\_id

            fund\_id,

            nvl(

                lag(ending\_accrual) over (

                    partition by fund\_id, cust\_account\_id, org\_id, ledger\_id

                    order by fund\_id

                ),

                0

            )

            beginning\_accrual,

            accrual\_amount,

            adjustment\_amount,

            paid\_amount,

            nvl(ending\_accrual, 0) ending\_accrual,

            unposted\_accruals,

            datasource\_num\_id,

            integration\_id,

            {{ env\_var("DBT\_ETL\_PROC\_ID") }}  etl\_proc\_id

        from

            (

                select

                    cust\_account\_id,

                    org\_id,

                    ledger\_id,

                    fund\_id,

                    -- x\_terr\_id,

                    sum(nvl(accrual\_amount, 0)) accrual\_amount,

                    sum(nvl(adjustment\_amount, 0)) adjustment\_amount,

                    sum(nvl(paid\_amount, 0)) paid\_amount,

                    sum(

                        sum(nvl(accrual\_amount, 0))

                        + sum(nvl(adjustment\_amount, 0))

                        - sum(nvl(paid\_amount, 0))

                    -- x\_terr\_id,ledger\_id

                    ) over (

                        partition by fund\_id, cust\_account\_id, org\_id, ledger\_id

                        order by fund\_id

                    )

                    ending\_accrual,

                    sum(unposted\_accruals) unposted\_accruals,

                    datasource\_num\_id,

                    integration\_id,

                    {{ env\_var("DBT\_ETL\_PROC\_ID") }}  etl\_proc\_id

                from

                    (

                        select

                            fund\_periods.fund\_id,

                            fund\_periods.cust\_account\_id,

                            fund\_periods.org\_id,

                            fund\_periods.ledger\_id,

                            -- fund\_periods.x\_terr\_id,

                            nvl(tm.accrual\_amount, 0) accrual\_amount,

                            nvl(tm.adjustment\_amount, 0) adjustment\_amount,

                            nvl(tm.unposted\_accruals, 0) unposted\_accruals,

                            nvl(tm.paid\_amount, 0) paid\_amount,

                            tm.datasource\_num\_id,

                            tm.integration\_id,

                            tm.etl\_proc\_id

                        from

                            (

                                select distinct

                                    fun.fund\_id::varchar(16777216) fund\_ids,

                                    fun.cust\_account\_id::varchar(

                                        16777216

                                    ) cust\_account\_ids,

                                    fun.org\_id::varchar(16777216) org\_ids,

                                    fun.ledger\_id::varchar(16777216) ledger\_ids,

                                    -- x\_terr\_id x\_terr\_ids,

                                    nvl(fun.gl\_date, fun.adjustment\_date) gl\_date,

                                    sum(

                                        case

                                            when

                                                fun.utilization\_type = 'ACCRUAL'

                                                and fun.gl\_posted\_flag = 'Y'

                                                and fun.gl\_date is not null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    accrual\_amount,

                                    sum(

                                        case

                                            when

                                                fun.utilization\_type = 'ADJUSTMENT'

                                                and fun.gl\_posted\_flag = 'Y'

                                                and fun.gl\_date is not null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    adjustment\_amount,

                                    sum(

                                        case

                                            when

                                                fun.gl\_posted\_flag <> 'Y'

                                                and fun.gl\_date is null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    unposted\_accruals,

                                    0 paid\_amount,

                                    fun.\_source\_id datasource\_num\_id,

                                    coalesce(to\_char(fun.cust\_account\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(fun.fund\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(fun.org\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(fun.ledger\_id), '')

                                    || '~'

                                    || to\_char(fun.\_source\_id) integration\_id ,

                                    {{ env\_var("DBT\_ETL\_PROC\_ID") }} etl\_proc\_id

                                from wc\_trade\_funds\_f fun

                                where utilization\_type in ('ACCRUAL', 'ADJUSTMENT')

                                group by

                                    ledger\_ids,

                                    fund\_ids,

                                    cust\_account\_ids,

                                    org\_ids,

                                    -- x\_terr\_id,

                                    nvl(fun.gl\_date, fun.adjustment\_date),

                                    \_source\_id

                                union all

                                select distinct

                                    cl.fund\_id::varchar(16777216) fund\_id1,

                                    cl.cust\_account\_id::varchar(

                                        16777216

                                    ) cust\_account\_id1,

                                    cl.org\_id::varchar(16777216) org\_id1,

                                    wc\_fund\_d.ledger\_id::varchar(16777216) ledger\_id1,

                                    -- x\_terr\_id,

                                    cl.gl\_date gl\_date,

                                    0 accrual\_amt,

                                    0 adjust\_amt,

                                    0 unposted\_amt,

                                    sum(cl.settled\_doc\_amt) paid\_amount,

                                    cl.\_source\_id datasource\_num\_id,

                                    coalesce(to\_char(cl.cust\_account\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(cl.fund\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(cl.org\_id), '')

                                    || '~'

                                    ||coalesce(to\_char(wc\_fund\_d.ledger\_id), '')

                                    || '~'

                                    || to\_char(cl.\_source\_id) integration\_id,

                                    {{ env\_var("DBT\_ETL\_PROC\_ID") }} etl\_proc\_id

                                from wc\_claims\_f cl

                                left outer join

                                    wc\_fund\_d on fund\_id1 = wc\_fund\_d.fund\_id

                                where cl.payment\_status = 'PAID'

                                -- and fund\_id = 490692

                                -- x\_terr\_id,ledger\_id

                                group by

                                    fund\_id1,

                                    cust\_account\_id1,

                                    org\_id1,

                                    ledger\_id1,

                                    gl\_date,

                                    \_source\_id

                            ) tm

                        left outer join

                            (

                                -- , x\_terr\_id, ledger\_id

                                select fund\_id, cust\_account\_id, org\_id, ledger\_id

                                from

                                    (

                                        select distinct

                                            fund\_id,

                                            cust\_account\_id,

                                            org\_id,

                                            ledger\_id,

                                            -- x\_terr\_id,

                                            a\_gl\_date,

                                            min(a\_gl\_date)

                                            min\_date,

                                            max(a\_gl\_date)

                                            max\_date

                                        from

                                            (

                                                select

                                                    a.fund\_id::varchar(

                                                        16777216

                                                    ) fund\_id,

                                                    a.cust\_account\_id::varchar(

                                                        16777216

                                                    ) cust\_account\_id,

                                                    a.org\_id::varchar(16777216) org\_id,

                                                    a.ledger\_id::varchar(

                                                        16777216

                                                    ) ledger\_id,

                                                    a.gl\_date a\_gl\_date,

                                                    a.\_source\_id datasource\_num\_id,

                                                    coalesce(to\_char(a.cust\_account\_id), '')

                                                    || '~'

                                                    ||coalesce(to\_char(a.fund\_id), '')

                                                    || '~'

                                                    ||coalesce(to\_char(a.org\_id), '')

                                                    || '~'

                                                    ||coalesce(to\_char(a.ledger\_id), '')

                                                    || '~'

                                                    || to\_char(a.\_source\_id) integration\_id,

                                                    {{ env\_var("DBT\_ETL\_PROC\_ID") }} etl\_proc\_id

                                                    -- x\_terr\_id

                                                from wc\_trade\_funds\_f a

                                                where

                                                    a.utilization\_type

                                                    in ('ACCRUAL', 'ADJUSTMENT')

                                                union all

                                                select distinct

                                                    k.fund\_id::varchar fund\_id2,

                                                    k.cust\_account\_id

                                                    ::varchar cust\_account\_id2,

                                                    k.org\_id::varchar org\_id2,

                                                    wc\_fund\_d.ledger\_id

                                                    ::varchar ledger\_id2,

                                                    k.gl\_date k\_gl\_date,

                                                    k.\_source\_id datasource\_num\_id,

                                                     coalesce(to\_char(k.cust\_account\_id), '')

                                                     || '~'

                                                     ||coalesce(to\_char(k.fund\_id), '')

                                                     || '~'

                                                     ||coalesce(to\_char(k.org\_id), '')

                                                     || '~'

                                                     ||coalesce(to\_char(wc\_fund\_d.ledger\_id), '')

                                                     || '~'

                                                     || to\_char(k.\_source\_id) integration\_id,

                                                     {{ env\_var("DBT\_ETL\_PROC\_ID") }}  etl\_proc\_id

                                                -- x\_terr\_id

                                                from wc\_claims\_f k

                                                left outer join

                                                    wc\_fund\_d

                                                    on k.fund\_id::varchar

                                                    = wc\_fund\_d.integration\_id::varchar

                                                where

                                                    k.payment\_status = 'PAID'

                                                    and k.fund\_id::varchar > '0'

                                                group by

                                                    k.fund\_id::varchar,

                                                    k.cust\_account\_id::varchar,

                                                    k.org\_id::varchar,

                                                    wc\_fund\_d.ledger\_id::varchar,

                                                    k\_gl\_date,

                                                    k.\_source\_id

                                            -- and a.fund\_wid = 490692

                                            ) h

                                        group by

                                            fund\_id,

                                            cust\_account\_id,

                                            org\_id,

                                            ledger\_id,

                                            a\_gl\_date

                                    -- x\_terr\_id,ledger\_id

                                    ) fund

                                -- having

                                --

                                having

                                    fund.a\_gl\_date <= trunc(current\_date(), 'MM')

                                    and fund.a\_gl\_date

                                    between trunc(min\_date, 'MM') and trunc(

                                        max\_date, 'MM'

                                    )

                                order by fund.a\_gl\_date

                            ) fund\_periods

                            on fund\_periods.fund\_id = tm.fund\_ids

                            and fund\_periods.cust\_account\_id = tm.cust\_account\_ids

                            and fund\_periods.ledger\_id = tm.ledger\_ids

                            and fund\_periods.org\_id = tm.org\_ids

                    -- and fund\_periods.x\_terr\_id = tm.x\_terr\_ids

                    ) b

                -- where b.fund\_id = 496226 and cust\_account\_id =

                -- 664467,ledger\_id,x\_terr\_id

                group by cust\_account\_id, org\_id, fund\_id, ledger\_id,datasource\_num\_id,integration\_id

            )

   )

select \*, $batch\_start\_time w\_insert\_dt, $batch\_start\_time w\_update\_dt

from final

select

    SQ\_SALES\_DEMAND\_FS.ORDER\_TYPE      C1\_DEMAND\_TYPE,

    SQ\_SALES\_DEMAND\_FS.DEMAND\_CLASS    C2\_DEMAND\_CLASS,

    SQ\_SALES\_DEMAND\_FS.QUANTITY    C3\_DEMAND\_QTY,

    SQ\_SALES\_DEMAND\_FS.W\_UPDATE\_DT     C4\_W\_UPDATE\_DT,

    SQ\_SALES\_DEMAND\_FS.DUE\_DATE    C5\_DUE\_DATE,

    SQ\_SALES\_DEMAND\_FS.SALES\_ORDER\_NUMBER      C6\_SALES\_ORDER\_NUMBER,

    SQ\_SALES\_DEMAND\_FS.COMPILE\_DESIGNATOR      C7\_PLAN\_NAME,

    SQ\_SALES\_DEMAND\_FS.DEMAND\_FLAG     C8\_DEMAND\_FLAG,

    SQ\_SALES\_DEMAND\_FS.WEEKLY\_FLAG     C9\_WEEKLY\_FLAG,

    SQ\_SALES\_DEMAND\_FS.DAILY\_CUTOFF\_BUCKET     C10\_DAILY\_CUTOFF\_BUCKET,

    SQ\_SALES\_DEMAND\_FS.CUSTOMER\_ID     C11\_CUSTOMER\_ID,

    SQ\_SALES\_DEMAND\_FS.ORGANIZATION\_ID     C12\_ORGANIZATION\_ID,

    SQ\_SALES\_DEMAND\_FS.INVENTORY\_ITEM\_ID       C13\_INVENTORY\_ITEM\_ID

from

( /\* Subselect from CST\_SALES\_DEMAND\_F.SQ\_SALES\_DEM\_FS

\*/

 (SELECT MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID INVENTORY\_ITEM\_ID,

      MSC\_DEMANDS.CUSTOMER\_ID CUSTOMER\_ID,

      TRUNC(MSC\_DEMANDS.USING\_ASSEMBLY\_DEMAND\_DATE) DUE\_DATE,

      NVL(MSC\_DEMANDS.DEMAND\_CLASS,'NA') DEMAND\_CLASS,

      SUM(NVL(MSC\_DEMANDS.DAILY\_DEMAND\_RATE,MSC\_DEMANDS.USING\_REQUIREMENT\_QUANTITY)) QUANTITY,

      TRUNC(MSC\_DEMANDS.LAST\_UPDATE\_DATE) W\_UPDATE\_DT,

      MSC\_DEMANDS.ORGANIZATION\_ID ORGANIZATION\_ID,

      MFG\_LOOKUPS.MEANING ORDER\_TYPE,

      TO\_CHAR(SUBSTR(MSC\_DEMANDS.ORDER\_NUMBER,1,instr(MSC\_DEMANDS.ORDER\_NUMBER,'.',1)-1)) SALES\_ORDER\_NUMBER,

      MSC\_PLANS.COMPILE\_DESIGNATOR COMPILE\_DESIGNATOR,

      CASE

        WHEN ((NVL(case when msc\_demands.organization\_id = 1200 and msc\_demands.source\_organization\_id = 1202 then null else msc\_demands.source\_organization\_id end,0) > 0

        AND MSC\_DEMANDS.source\_organization\_id           <> MSC\_DEMANDS.organization\_id)

        OR msc\_demands.inventory\_item\_id                 <> msc\_demands.using\_assembly\_item\_id)

        THEN 'N'

        ELSE 'Y'

      END demand\_flag , -- added to exlcude component item demand used for make items, and do not apply org dependent logic for dop items in dtx

      CASE

        WHEN MFG\_LOOKUPS.MEANING         ='Forecast'

        AND MSC\_DEMANDS.ORGANIZATION\_ID IN

          (SELECT sr\_organization\_id

          FROM apps.msd\_dp\_scn\_entries\_denorm

          WHERE start\_time >=(sysdate-365)

          AND BUCKET\_TYPE   =2

          )

        THEN 'Y'

        ELSE 'N'

      END WEEKLY\_FLAG ,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    FROM APPS.MSC\_DEMANDS MSC\_DEMANDS,

      APPS.MSC\_PLANS MSC\_PLANS,

      APPS.MSC\_SYSTEM\_ITEMS MSC\_SYSTEM\_ITEMS,

      APPS.MFG\_LOOKUPS MFG\_LOOKUPS,

      APPS.MSC\_PLAN\_ORGANIZATIONS mpo

      /\*

    WHERE(1                                     =1)

    AND mpo.plan\_id                             = msc\_demands.plan\_id

    AND mpo.organization\_id                     = msc\_demands.organization\_id

    AND mpo.sr\_instance\_id                      = msc\_demands.sr\_instance\_id

    AND (MFG\_LOOKUPS.LOOKUP\_TYPE                ='MSC\_DEMAND\_ORIGINATION')

    AND (MSC\_DEMANDS.USING\_REQUIREMENT\_QUANTITY >0)

    AND (MFG\_LOOKUPS.MEANING                   IN('Forecast','Payback Demand','Planned order demand','Interorganization Demand','Sales Orders','Work order demand','Safety Stock Demand'))

    AND (MSC\_PLANS.PLAN\_ID                      =MSC\_DEMANDS.PLAN\_ID)

    AND (MSC\_PLANS.COMPILE\_DESIGNATOR          IN

      (SELECT DESIGNATOR FROM APPS.MSC\_DESIGNATORS WHERE DISABLE\_DATE IS NULL

      ))

    AND (MSC\_SYSTEM\_ITEMS.PLAN\_ID       =MSC\_PLANS.PLAN\_ID)

    AND (MSC\_SYSTEM\_ITEMS.SR\_INSTANCE\_ID=MSC\_DEMANDS.SR\_INSTANCE\_ID

    AND MSC\_DEMANDS.INVENTORY\_ITEM\_ID   =MSC\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID

    AND MSC\_SYSTEM\_ITEMS.ORGANIZATION\_ID=MSC\_DEMANDS.ORGANIZATION\_ID

    AND MSC\_SYSTEM\_ITEMS.PLAN\_ID        =MSC\_DEMANDS.PLAN\_ID)

    AND (MSC\_DEMANDS.ORIGINATION\_TYPE   =MFG\_LOOKUPS.LOOKUP\_CODE)

    GROUP BY MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID,

      MSC\_DEMANDS.CUSTOMER\_ID,

      TRUNC(MSC\_DEMANDS.USING\_ASSEMBLY\_DEMAND\_DATE),

      NVL(MSC\_DEMANDS.DEMAND\_CLASS,'NA'),

      TRUNC(MSC\_DEMANDS.LAST\_UPDATE\_DATE),

      MSC\_DEMANDS.ORGANIZATION\_ID,

      MFG\_LOOKUPS.MEANING,

      SUBSTR(MSC\_DEMANDS.ORDER\_NUMBER,1,instr(MSC\_DEMANDS.ORDER\_NUMBER,'.',1)-1),

      MSC\_PLANS.COMPILE\_DESIGNATOR,

      CASE

        WHEN ((NVL(case when msc\_demands.organization\_id = 1200 and msc\_demands.source\_organization\_id = 1202 then null else msc\_demands.source\_organization\_id end,0) > 0

        AND MSC\_DEMANDS.source\_organization\_id           <> MSC\_DEMANDS.organization\_id)

        OR msc\_demands.inventory\_item\_id                 <> msc\_demands.using\_assembly\_item\_id)

        THEN 'N'

        ELSE 'Y'

      END,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    )

  UNION ALL

    (SELECT MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID C11\_INVENTORY\_ITEM\_ID,

      MSC\_FORECAST\_UPDATES.CUSTOMER\_ID C12\_CUSTOMER\_ID,

      TRUNC(MSC\_FORECAST\_UPDATES.CONSUMPTION\_DATE) C13\_DUE\_DATE,

      NVL(MSC\_FORECAST\_UPDATES.DEMAND\_CLASS,'NA') C14\_DEMAND\_CLASS,

      SUM(MSC\_FORECAST\_UPDATES.CONSUMED\_QTY) C15\_QUANTITY,

      TRUNC(MSC\_FORECAST\_UPDATES.LAST\_UPDATE\_DATE) C16\_W\_UPDATE\_DT,

      MSC\_FORECAST\_UPDATES.ORGANIZATION\_ID C17\_ORGANIZATION\_ID,

      'Consumption' C8\_ORDER\_TYPE,

      TO\_CHAR(SUBSTR(MSC\_FORECAST\_UPDATES.SALES\_ORDER\_NUMBER,1,instr(MSC\_FORECAST\_UPDATES.SALES\_ORDER\_NUMBER,'.',1)-1)) C18\_SALES\_ORDER\_NUMBER,

      MSC\_PLANS.COMPILE\_DESIGNATOR C19\_COMPILE\_DESIGNATOR,

      'Y' AS demand\_flag,

      'N' AS weekly\_flag,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    FROM APPS.MSC\_PLANS MSC\_PLANS,

      APPS.MSC\_SYSTEM\_ITEMS MSC\_SYSTEM\_ITEMS,

      APPS.MSC\_FORECAST\_UPDATES MSC\_FORECAST\_UPDATES,

      APPS.MSC\_PLAN\_ORGANIZATIONS mpo

    WHERE(1                               =1)

    AND mpo.plan\_id                       = msc\_forecast\_updates.plan\_id

    AND mpo.organization\_id               = msc\_forecast\_updates.organization\_id

    AND mpo.sr\_instance\_id                = msc\_forecast\_updates.sr\_instance\_id

    AND (MSC\_FORECAST\_UPDATES.CONSUMED\_QTY>0)

    AND (MSC\_PLANS.PLAN\_ID                =MSC\_SYSTEM\_ITEMS.PLAN\_ID)

    AND (MSC\_PLANS.COMPILE\_DESIGNATOR    IN

      (SELECT DESIGNATOR FROM APPS.MSC\_DESIGNATORS WHERE DISABLE\_DATE IS NULL

      ))

    AND (MSC\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID=MSC\_FORECAST\_UPDATES.INVENTORY\_ITEM\_ID

    AND MSC\_SYSTEM\_ITEMS.ORGANIZATION\_ID   =MSC\_FORECAST\_UPDATES.ORGANIZATION\_ID)

    AND (MSC\_PLANS.PLAN\_ID                 =MSC\_FORECAST\_UPDATES.PLAN\_ID)

    GROUP BY MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID,

      MSC\_FORECAST\_UPDATES.CUSTOMER\_ID,

      TRUNC(MSC\_FORECAST\_UPDATES.CONSUMPTION\_DATE),

      NVL(MSC\_FORECAST\_UPDATES.DEMAND\_CLASS,'NA'),

      TRUNC(MSC\_FORECAST\_UPDATES.LAST\_UPDATE\_DATE),

      MSC\_FORECAST\_UPDATES.ORGANIZATION\_ID,

      SUBSTR(MSC\_FORECAST\_UPDATES.SALES\_ORDER\_NUMBER,1,instr(MSC\_FORECAST\_UPDATES.SALES\_ORDER\_NUMBER,'.',1)-1),

      MSC\_PLANS.COMPILE\_DESIGNATOR,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    )

  UNION ALL

    (SELECT MTL\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID C20\_INVENTORY\_ITEM\_ID,

      OE\_ORDER\_LINES\_ALL.SOLD\_TO\_ORG\_ID C21\_CUSTOMER\_ID,

      TRUNC(OE\_ORDER\_LINES\_ALL.REQUEST\_DATE) C22\_DUE\_DATE,

      NVL(OE\_ORDER\_LINES\_ALL.DEMAND\_CLASS\_CODE,'NA') C23\_DEMAND\_CLASS,

      SUM(OE\_ORDER\_LINES\_ALL.ORDERED\_QUANTITY\*NVL(MTL\_UOM\_CONVERSIONS.CONVERSION\_RATE,1)) C24\_QUANTITY,

      TRUNC(OE\_ORDER\_LINES\_ALL.LAST\_UPDATE\_DATE) C25\_W\_UPDATE\_DT,

      MTL\_SYSTEM\_ITEMS.ORGANIZATION\_ID C26\_ORGANIZATION\_ID,

      'Unschedule' C8\_ORDER\_TYPE,

      TO\_CHAR(OE\_ORDER\_HEADERS\_ALL.ORDER\_NUMBER) C27\_SALES\_ORDER\_NUMBER,

      'NA' C19\_COMPILE\_DESIGNATOR,

      'Y'AS demand\_flag,

      'N'AS weekly\_flag,

      0

    FROM APPS.MTL\_UOM\_CONVERSIONS MTL\_UOM\_CONVERSIONS

    RIGHT OUTER JOIN(((APPS.OE\_ORDER\_LINES\_ALL OE\_ORDER\_LINES\_ALL

    INNER JOIN APPS.MTL\_SYSTEM\_ITEMS MTL\_SYSTEM\_ITEMS

    ON OE\_ORDER\_LINES\_ALL.INVENTORY\_ITEM\_ID=MTL\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID

    AND OE\_ORDER\_LINES\_ALL.SHIP\_FROM\_ORG\_ID=MTL\_SYSTEM\_ITEMS.ORGANIZATION\_ID)

    INNER JOIN APPS.OE\_ORDER\_HEADERS\_ALL OE\_ORDER\_HEADERS\_ALL

    ON OE\_ORDER\_LINES\_ALL.HEADER\_ID=OE\_ORDER\_HEADERS\_ALL.HEADER\_ID)

    INNER JOIN APPS.MTL\_PARAMETERS MTL\_PARAMETERS

    ON OE\_ORDER\_LINES\_ALL.SHIP\_FROM\_ORG\_ID    =MTL\_PARAMETERS.ORGANIZATION\_ID)

    ON MTL\_UOM\_CONVERSIONS.INVENTORY\_ITEM\_ID  =OE\_ORDER\_LINES\_ALL.INVENTORY\_ITEM\_ID

    AND MTL\_UOM\_CONVERSIONS.UOM\_CODE          =OE\_ORDER\_LINES\_ALL.ORDER\_QUANTITY\_UOM

    WHERE(1                                   =1)

    AND (OE\_ORDER\_LINES\_ALL.FLOW\_STATUS\_CODE IN('BOOKED','ENTERED')

    AND OE\_ORDER\_LINES\_ALL.OPEN\_FLAG          ='Y'

    AND OE\_ORDER\_LINES\_ALL.LINE\_CATEGORY\_CODE ='ORDER')

    GROUP BY MTL\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID,

      OE\_ORDER\_LINES\_ALL.SOLD\_TO\_ORG\_ID,

      TRUNC(OE\_ORDER\_LINES\_ALL.REQUEST\_DATE),

      NVL(OE\_ORDER\_LINES\_ALL.DEMAND\_CLASS\_CODE,'NA'),

      TRUNC(OE\_ORDER\_LINES\_ALL.LAST\_UPDATE\_DATE),

      MTL\_SYSTEM\_ITEMS.ORGANIZATION\_ID,

      TO\_CHAR(OE\_ORDER\_HEADERS\_ALL.ORDER\_NUMBER)

    )

  UNION ALL

    (SELECT MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID C28\_INVENTORY\_ITEM\_ID,

      MSC\_SALES\_ORDERS.CUSTOMER\_ID C29\_CUSTOMER\_ID,

      --     TRUNC(MSC\_SALES\_ORDERS.REQUIREMENT\_DATE)         C30\_DUE\_DATE,

      TRUNC(OLA.ACTUAL\_SHIPMENT\_DATE) C30\_DUE\_DATE,

      NVL(MSC\_SALES\_ORDERS.DEMAND\_CLASS,'NA') C31\_DEMAND\_CLASS,

      SUM(MSC\_SALES\_ORDERS.COMPLETED\_QUANTITY) C32\_QUANTITY,

      TRUNC(MSC\_SALES\_ORDERS.LAST\_UPDATE\_DATE) C33\_W\_UPDATE\_DT,

      MSC\_SALES\_ORDERS.ORGANIZATION\_ID C34\_ORGANIZATION\_ID,

      'Shipped' C8\_ORDER\_TYPE,

      TO\_CHAR(SUBSTR(MSC\_SALES\_ORDERS.SALES\_ORDER\_NUMBER,1,instr(MSC\_SALES\_ORDERS.SALES\_ORDER\_NUMBER,'.',1)-1)) C35\_SALES\_ORDER\_NUMBER,

      MSC\_PLANS.COMPILE\_DESIGNATOR C19\_COMPILE\_DESIGNATOR,

      'Y' AS demand\_flag,

      'N' AS weekly\_flag,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    FROM APPS.MSC\_SYSTEM\_ITEMS MSC\_SYSTEM\_ITEMS,

      APPS.MSC\_SALES\_ORDERS MSC\_SALES\_ORDERS,

      APPS.MSC\_PLANS MSC\_PLANS,

      APPS.OE\_ORDER\_LINES\_ALL OLA

    WHERE(1                          =1)

    AND (MSC\_SALES\_ORDERS.DEMAND\_ID IN

      (SELECT DISTINCT SALES\_ORDER\_ID

      FROM APPS.MSC\_FORECAST\_UPDATES M,

        APPS.MSC\_SYSTEM\_ITEMS I

      WHERE M.INVENTORY\_ITEM\_ID=I.INVENTORY\_ITEM\_ID

      AND M.ORGANIZATION\_ID    = I.ORGANIZATION\_ID

      AND M.PLAN\_ID            = I.PLAN\_ID

      ))

    AND MSC\_SALES\_ORDERS.DEMAND\_SOURCE\_LINE = OLA.LINE\_ID

    AND (MSC\_SALES\_ORDERS.COMPLETED\_QUANTITY>0)

    AND (MSC\_SYSTEM\_ITEMS.INVENTORY\_ITEM\_ID =MSC\_SALES\_ORDERS.INVENTORY\_ITEM\_ID

    AND MSC\_SYSTEM\_ITEMS.ORGANIZATION\_ID    =MSC\_SALES\_ORDERS.ORGANIZATION\_ID)

    AND (MSC\_PLANS.PLAN\_ID                  =MSC\_SYSTEM\_ITEMS.PLAN\_ID

    AND MSC\_SYSTEM\_ITEMS.PLAN\_ID           <>-1)

    AND (MSC\_PLANS.COMPILE\_DESIGNATOR       IN

      (SELECT DESIGNATOR FROM APPS.MSC\_DESIGNATORS WHERE DISABLE\_DATE IS NULL

      ))

    GROUP BY MSC\_SYSTEM\_ITEMS.SR\_INVENTORY\_ITEM\_ID,

      MSC\_SALES\_ORDERS.CUSTOMER\_ID,

      TRUNC(OLA.ACTUAL\_SHIPMENT\_DATE),

      --TRUNC(MSC\_SALES\_ORDERS.REQUIREMENT\_DATE),

      NVL(MSC\_SALES\_ORDERS.DEMAND\_CLASS,'NA'),

      TRUNC(MSC\_SALES\_ORDERS.LAST\_UPDATE\_DATE),

      MSC\_SALES\_ORDERS.ORGANIZATION\_ID,

      SUBSTR(MSC\_SALES\_ORDERS.SALES\_ORDER\_NUMBER,1,instr(MSC\_SALES\_ORDERS.SALES\_ORDER\_NUMBER,'.',1)-1),

      MSC\_PLANS.COMPILE\_DESIGNATOR,

      MSC\_PLANS.DAILY\_CUTOFF\_BUCKET

    )

  UNION ALL

    (SELECT MSI.sr\_INVENTORY\_ITEM\_ID INVENTORY\_ITEM\_ID,

      NULL CUSTOMER\_ID,

      MSS.PERIOD\_START\_DATE DUE\_DATE,

      'NA' DEMAND\_CLASS,

      MSS.SAFETY\_STOCK\_QUANTITY QUANTITY,

      SYSDATE W\_UPDATE\_DT,

      MSS.ORGANIZATION\_ID,

      'Safety Stock' ORDER\_TYPE,

      NULL SALES\_ORDER\_NUMBER,

      MSC.COMPILE\_DESIGNATOR COMPILE\_DESIGNATOR,

      'N' DEMAND\_FLAG,

      'N' WEEKLY\_FLAG,

      MSC.DAILY\_CUTOFF\_BUCKET

    FROM apps.MSC\_SAFETY\_STOCKS MSS,

      apps.MSC\_SYSTEM\_ITEMS MSI,

      APPS.MSC\_PLANS MSC

    WHERE MSS.SR\_INSTANCE\_ID    = MSI.SR\_INSTANCE\_ID

    AND MSS.INVENTORY\_ITEM\_ID   = MSI.INVENTORY\_ITEM\_ID

    AND MSS.ORGANIZATION\_ID     = MSI.ORGANIZATION\_ID

    AND MSS.PLAN\_ID             = MSI.PLAN\_ID

    AND MSI.PLAN\_ID             =MSC.PLAN\_ID

    AND(MSC.COMPILE\_DESIGNATOR IN

      (SELECT DESIGNATOR FROM APPS.MSC\_DESIGNATORS WHERE DISABLE\_DATE IS NULL

      ))

    )

  UNION ALL

    (SELECT OLA.INVENTORY\_ITEM\_ID C28\_INVENTORY\_ITEM\_ID,

      OLA.SOLD\_TO\_ORG\_ID C29\_CUSTOMER\_ID,

      TRUNC(NVL(OLA.LATEST\_ACCEPTABLE\_DATE,OLA.SCHEDULE\_SHIP\_DATE)) C30\_DUE\_DATE,

      NVL(OLA.DEMAND\_CLASS\_CODE,'NA') C31\_DEMAND\_CLASS,

      SUM(OLA.ORDERED\_QUANTITY\*NVL(MUC.CONVERSION\_RATE,1)) C32\_QUANTITY, --UOM Conversion required

      TRUNC(OLA.LAST\_UPDATE\_DATE) C33\_W\_UPDATE\_DT,

      OLA.SHIP\_FROM\_ORG\_ID C34\_ORGANIZATION\_ID,

      'Sales Orders - LAD' C8\_ORDER\_TYPE,

      TO\_CHAR(OHA.ORDER\_NUMBER) C35\_SALES\_ORDER\_NUMBER,

      'GLOBAL' C19\_COMPILE\_DESIGNATOR,

      'Y' AS demand\_flag,

      'N' AS weekly\_flag,

      0 DAILY\_CUTOFF\_BUCKET

    FROM APPS.OE\_ORDER\_LINES\_ALL OLA,

      APPS.OE\_ORDER\_HEADERS\_ALL OHA,

      APPS.MTL\_UOM\_CONVERSIONS MUC

    WHERE(1                     =1)

    AND OLA.HEADER\_ID           = OHA.HEADER\_ID

    AND OLA.OPEN\_FLAG           = 'Y'

    AND OHA.OPEN\_FLAG           = 'Y'

    AND OLA.ORDER\_SOURCE\_ID    <>10

    AND OLA.LINE\_CATEGORY\_CODE  ='ORDER'

    AND MUC.INVENTORY\_ITEM\_ID(+)=OLA.INVENTORY\_ITEM\_ID

    AND MUC.UOM\_CODE(+)         =OLA.ORDER\_QUANTITY\_UOM

    GROUP BY OLA.INVENTORY\_ITEM\_ID,

      OLA.SOLD\_TO\_ORG\_ID,

      TRUNC(NVL(OLA.LATEST\_ACCEPTABLE\_DATE,OLA.SCHEDULE\_SHIP\_DATE)),

      NVL(OLA.DEMAND\_CLASS\_CODE,'NA'),

      TRUNC(OLA.LAST\_UPDATE\_DATE),

      OLA.SHIP\_FROM\_ORG\_ID,

      OHA.ORDER\_NUMBER

    )

)   SQ\_SALES\_DEMAND\_FS

where   (1=1)

with

    wc\_trade\_funds\_f as (select \* from {{ ref("stg\_oracle\_\_trade\_funds\_fs") }}),

    wc\_claims\_f as (select \* from {{ ref("stg\_oracle\_\_wc\_claims") }}),

    wc\_fund\_d as (select \* from dev\_edw.edw\_omni.wc\_fund\_d),

    wc\_user\_d as (select \* from dev\_edw.edw\_omni.w\_user\_d),

    wc\_claims\_sq as (

        select \*

        from wc\_claims\_f

        where

            (1 = 1)

            {% if is\_incremental() %}

            and (

                (

                    wc\_claims\_f.\_batch\_update\_date >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date1 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date1 <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date2 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date2 <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date3 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date3 <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date4 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date4 <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date5 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date5 <= $batch\_end\_time

                )

                or (

                    wc\_claims\_f.\_batch\_update\_date6 >= $batch\_start\_time

                    and wc\_claims\_f.\_batch\_update\_date16 <= $batch\_end\_time

                )

            )

            {% endif %}

    ),

    wc\_trade\_sq as (

        select \*

        from wc\_trade\_funds\_f

        where

            (1 = 1)

            {% if is\_incremental() %}

            and (

                wc\_trade\_funds\_f.\_batch\_update\_date >= $batch\_start\_time

                and wc\_trade\_funds\_f.\_batch\_update\_date <= $batch\_end\_time

            )

            {% endif %}

    ),

    final as (

        select

            -- ledger\_id, x\_terr\_id)

            cust\_account\_id as cust\_acct\_id,

            org\_id,

            ledger\_id,

            x\_terr\_id

            fund\_id,

            nvl(

                lag(ending\_accrual) over (

                    partition by fund\_id, cust\_account\_id, org\_id, ledger\_id, x\_terr\_id

                    order by fund\_id

                ),

                0

            )

            beginning\_accrual,

            accrual\_amount,

            adjustment\_amount,

            paid\_amount,

            nvl(ending\_accrual, 0) ending\_accrual,

            unposted\_accruals,

            datasource\_num\_id,

            integration\_id,

            9999 etl\_proc\_id

        from

            (

                select

                    cust\_account\_id,

                    org\_id,

                    ledger\_id,

                    fund\_id,

                    x\_terr\_id,

                    sum(nvl(accrual\_amount, 0)) accrual\_amount,

                    sum(nvl(adjustment\_amount, 0)) adjustment\_amount,

                    sum(nvl(paid\_amount, 0)) paid\_amount,

                    sum(

                        sum(nvl(accrual\_amount, 0))

                        + sum(nvl(adjustment\_amount, 0))

                        - sum(nvl(paid\_amount, 0))

                    -- x\_terr\_id,ledger\_id

                    ) over (

                        partition by

                            fund\_id, cust\_account\_id, org\_id, ledger\_id, x\_terr\_id

                        order by fund\_id

                    )

                    ending\_accrual,

                    sum(unposted\_accruals) unposted\_accruals,

                    datasource\_num\_id,

                    integration\_id,

                    9999 etl\_proc\_id

                from

                    (

                        select

                            fund\_periods.fund\_id,

                            fund\_periods.cust\_account\_id,

                            fund\_periods.org\_id,

                            fund\_periods.ledger\_id,

                            fund\_periods.x\_terr\_id,

                            nvl(tm.accrual\_amount, 0) accrual\_amount,

                            nvl(tm.adjustment\_amount, 0) adjustment\_amount,

                            nvl(tm.unposted\_accruals, 0) unposted\_accruals,

                            nvl(tm.paid\_amount, 0) paid\_amount,

                            tm.datasource\_num\_id,

                            tm.integration\_id,

                            tm.etl\_proc\_id

                        from

                            (

                                select distinct

                                    fun.fund\_id::varchar(16777216) fund\_ids,

                                    fun.cust\_account\_id::varchar(

                                        16777216

                                    ) cust\_account\_ids,

                                    fun.org\_id::varchar(16777216) org\_ids,

                                    fun.ledger\_id::varchar(16777216) ledger\_ids,

                                    COALESCE(

                                        edw\_omni.hot\_get\_rep\_key(

                                            fun.ship\_to\_site\_use\_id,

                                            'ALL',

                                            'ALL',

                                            fun.cust\_account\_id

                                        ),

                                        '0'

                                    ) x\_terr\_ids,

                                    nvl(fun.gl\_date, fun.adjustment\_date) gl\_date,

                                    sum(

                                        case

                                            when

                                                fun.utilization\_type = 'ACCRUAL'

                                                and fun.gl\_posted\_flag = 'Y'

                                                and fun.gl\_date is not null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    accrual\_amount,

                                    sum(

                                        case

                                            when

                                                fun.utilization\_type = 'ADJUSTMENT'

                                                and fun.gl\_posted\_flag = 'Y'

                                                and fun.gl\_date is not null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    adjustment\_amount,

                                    sum(

                                        case

                                            when

                                                fun.gl\_posted\_flag <> 'Y'

                                                and fun.gl\_date is null

                                            then accrual\_loc\_amt

                                        end

                                    )

                                    unposted\_accruals,

                                    0 paid\_amount,

                                    fun.\_source\_id datasource\_num\_id,

                                    coalesce(to\_char(fun.cust\_account\_id), '')

                                    || '~'

                                    || coalesce(to\_char(fun.fund\_id), '')

                                    || '~'

                                    || coalesce(to\_char(fun.org\_id), '')

                                    || '~'

                                    || coalesce(to\_char(fun.ledger\_id), '')

                                    || '~'

                                    || coalesce(to\_char(fun.ship\_to\_site\_use\_id), '')

                                    || '~'

                                    || to\_char(fun.\_source\_id) integration\_id,

                                    9999 etl\_proc\_id

                                from wc\_trade\_funds\_f fun

                                where utilization\_type in ('ACCRUAL', 'ADJUSTMENT')

                                group by

                                    ledger\_ids,

                                    fund\_ids,

                                    cust\_account\_ids,

                                    org\_ids,

                                    x\_terr\_ids,

                                    nvl(fun.gl\_date, fun.adjustment\_date),

                                    \_source\_id

                                union all

                                select distinct

                                    cl.fund\_id::varchar(16777216) fund\_id1,

                                    cl.cust\_account\_id::varchar(

                                        16777216

                                    ) cust\_account\_id1,

                                    cl.org\_id::varchar(16777216) org\_id1,

                                    wc\_fund\_d.ledger\_id::varchar(16777216) ledger\_id1,

                                    COALESCE(

                                        edw\_omni.hot\_get\_rep\_key(

                                            cl.cust\_shipto\_acct\_site\_id,

                                            'ALL',

                                            'ALL',

                                            cl.cust\_account\_id

                                        ),

                                        '0'

                                    ) x\_terr\_id1,

                                    cl.gl\_date gl\_date,

                                    0 accrual\_amt,

                                    0 adjust\_amt,

                                    0 unposted\_amt,

                                    sum(cl.settled\_doc\_amt) paid\_amount,

                                    cl.\_source\_id datasource\_num\_id,

                                    coalesce(to\_char(cl.cust\_account\_id), '')

                                    || '~'

                                    || coalesce(to\_char(cl.fund\_id), '')

                                    || '~'

                                    || coalesce(to\_char(cl.org\_id), '')

                                    || '~'

                                    || coalesce(to\_char(wc\_fund\_d.ledger\_id), '')

                                    || '~'

                                    || coalesce(

                                        to\_char(cl.cust\_shipto\_acct\_site\_id), ''

                                    )

                                    || '~'

                                    || to\_char(cl.\_source\_id) integration\_id,

                                    9999 etl\_proc\_id

                                from wc\_claims\_f cl

                                left outer join

                                    wc\_fund\_d on fund\_id1 = wc\_fund\_d.fund\_id

                                where cl.payment\_status = 'PAID'

                                -- and fund\_id = 490692

                                -- x\_terr\_id,ledger\_id

                                group by

                                    fund\_id1,

                                    cust\_account\_id1,

                                    org\_id1,

                                    ledger\_id1,

                                    x\_terr\_id1,

                                    gl\_date,

                                    \_source\_id

                            ) tm

                        left outer join

                            (

                                -- , x\_terr\_id, ledger\_id

                                select

                                    fund\_id,

                                    cust\_account\_id,

                                    org\_id,

                                    ledger\_id,

                                    x\_terr\_id

                                from

                                    (

                                        select distinct

                                            fund\_id,

                                            cust\_account\_id,

                                            org\_id,

                                            ledger\_id,

                                            x\_terr\_id,

                                            a\_gl\_date,

                                            min(a\_gl\_date)

                                            min\_date,

                                            max(a\_gl\_date)

                                            max\_date

                                        from

                                            (

                                                select

                                                    a.fund\_id::varchar(

                                                        16777216

                                                    ) fund\_id,

                                                    a.cust\_account\_id::varchar(

                                                        16777216

                                                    ) cust\_account\_id,

                                                    a.org\_id::varchar(16777216) org\_id,

                                                    a.ledger\_id::varchar(

                                                        16777216

                                                    ) ledger\_id,

                                                    COALESCE(

                                                        edw\_omni.hot\_get\_rep\_key(

                                                            a.ship\_to\_site\_use\_id,

                                                            'ALL',

                                                            'ALL',

                                                            a.cust\_account\_id

                                                        ),

                                                        '0'

                                                    ) x\_terr\_id,

                                                    a.gl\_date a\_gl\_date,

                                                    a.\_source\_id datasource\_num\_id,

                                                    coalesce(

                                                        to\_char(a.cust\_account\_id), ''

                                                    )

                                                    || '~'

                                                    || coalesce(to\_char(a.fund\_id), '')

                                                    || '~'

                                                    || coalesce(to\_char(a.org\_id), '')

                                                    || '~'

                                                    || coalesce(

                                                        to\_char(a.ledger\_id), ''

                                                    )

                                                    || '~'

                                                    || coalesce(

                                                        to\_char(a.ship\_to\_site\_use\_id),

                                                        ''

                                                    )

                                                    || '~'

                                                    || to\_char(

                                                        a.\_source\_id

                                                    ) integration\_id,

                                                    9999 etl\_proc\_id

                                                -- x\_terr\_id

                                                from wc\_trade\_funds\_f a

                                                where

                                                    a.utilization\_type

                                                    in ('ACCRUAL', 'ADJUSTMENT')

                                                union all

                                                select distinct

                                                    k.fund\_id::varchar fund\_id2,

                                                    k.cust\_account\_id

                                                    ::varchar cust\_account\_id2,

                                                    k.org\_id::varchar org\_id2,

                                                    wc\_fund\_d.ledger\_id

                                                    ::varchar ledger\_id2,

                                                    COALESCE(

                                                        edw\_omni.hot\_get\_rep\_key(

                                                            k.cust\_shipto\_acct\_site\_id

                                                            ::varchar,

                                                            'ALL',

                                                            'ALL',

                                                            k.cust\_account\_id::varchar

                                                        ),

                                                        '0'

                                                    ) x\_terr\_id,

                                                    k.gl\_date k\_gl\_date,

                                                    k.\_source\_id datasource\_num\_id,

                                                    coalesce(

                                                        to\_char(k.cust\_account\_id), ''

                                                    )

                                                    || '~'

                                                    || coalesce(to\_char(k.fund\_id), '')

                                                    || '~'

                                                    || coalesce(to\_char(k.org\_id), '')

                                                    || '~'

                                                    || coalesce(

                                                        to\_char(wc\_fund\_d.ledger\_id), ''

                                                    )

                                                    || '~'

                                                    || coalesce(

                                                        to\_char(

                                                            k.cust\_shipto\_acct\_site\_id

                                                        ),

                                                        ''

                                                    )

                                                    || '~'

                                                    || to\_char(

                                                        k.\_source\_id

                                                    ) integration\_id,

                                                    9999 etl\_proc\_id

                                                -- x\_terr\_id

                                                from wc\_claims\_f k

                                                left outer join

                                                    wc\_fund\_d

                                                    on k.fund\_id::varchar

                                                    = wc\_fund\_d.integration\_id::varchar

                                                where

                                                    k.payment\_status = 'PAID'

                                                    and k.fund\_id::varchar > '0'

                                                group by

                                                    k.fund\_id::varchar,

                                                    k.cust\_account\_id::varchar,

                                                    k.org\_id::varchar,

                                                    wc\_fund\_d.ledger\_id::varchar,

                                                    k.cust\_shipto\_acct\_site\_id::varchar,

                                                    k\_gl\_date,

                                                    k.\_source\_id

                                            -- and a.fund\_wid = 490692

                                            ) h

                                        group by

                                            fund\_id,

                                            cust\_account\_id,

                                            org\_id,

                                            ledger\_id,

                                            x\_terr\_id,

                                            a\_gl\_date

                                    -- x\_terr\_id,ledger\_id

                                    ) fund

                                -- having

                                --

                                having

                                    fund.a\_gl\_date <= trunc(current\_date(), 'MM')

                                    and fund.a\_gl\_date

                                    between trunc(min\_date, 'MM') and trunc(

                                        max\_date, 'MM'

                                    )

                                order by fund.a\_gl\_date

                            ) fund\_periods

                            on fund\_periods.fund\_id = tm.fund\_ids

                            and fund\_periods.cust\_account\_id = tm.cust\_account\_ids

                            and fund\_periods.ledger\_id = tm.ledger\_ids

                            and fund\_periods.org\_id = tm.org\_ids

                            and fund\_periods.x\_terr\_id = tm.x\_terr\_ids

                    ) b

                -- where b.fund\_id = 496226 and cust\_account\_id =

                -- 664467,ledger\_id,x\_terr\_id

                group by

                    cust\_account\_id,

                    org\_id,

                    fund\_id,

                    ledger\_id,

                    x\_terr\_id,

                    datasource\_num\_id,

                    integration\_id

            )

    )

select \*

from final

* With dev\_omni\_nvl\_code

Nvl\_code

with

wc\_trade\_funds\_f as (select \* from dev\_discover.FINANCIALS.stg\_oracle\_\_trade\_funds\_fs),

wc\_claims\_f as (select \* from dev\_discover.FINANCIALS.stg\_oracle\_\_wc\_claims),

wc\_fund\_d as (select \* from dev\_edw.edw\_omni.wc\_fund\_d),

wc\_user\_d as (select \* from dev\_edw.edw\_omni.w\_user\_d),

final as(

select

-- ledger\_id, x\_terr\_id)

cust\_account\_id as cust\_acct\_id,

org\_id,

ledger\_id,

x\_terr\_id

fund\_id,

nvl(

lag(ending\_accrual) over (

partition by fund\_id, cust\_account\_id, org\_id, ledger\_id,x\_terr\_id

order by fund\_id

),

0

)

beginning\_accrual,

accrual\_amount,

adjustment\_amount,

paid\_amount,

nvl(ending\_accrual, 0) ending\_accrual,

unposted\_accruals,

datasource\_num\_id,

integration\_id,

9999 etl\_proc\_id

from

(

select

cust\_account\_id,

org\_id,

ledger\_id,

fund\_id,

x\_terr\_id,

sum(nvl(accrual\_amount, 0)) accrual\_amount,

sum(nvl(adjustment\_amount, 0)) adjustment\_amount,

sum(nvl(paid\_amount, 0)) paid\_amount,

sum(

sum(nvl(accrual\_amount, 0))

+ sum(nvl(adjustment\_amount, 0))

- sum(nvl(paid\_amount, 0))

-- x\_terr\_id,ledger\_id

) over (

partition by fund\_id, cust\_account\_id, org\_id, ledger\_id,x\_terr\_id

order by fund\_id

)

ending\_accrual,

sum(unposted\_accruals) unposted\_accruals,

datasource\_num\_id,

integration\_id,

9999 etl\_proc\_id

from

(

select

fund\_periods.fund\_id,

fund\_periods.cust\_account\_id,

fund\_periods.org\_id,

fund\_periods.ledger\_id,

fund\_periods.x\_terr\_id,

nvl(tm.accrual\_amount, 0) accrual\_amount,

nvl(tm.adjustment\_amount, 0) adjustment\_amount,

nvl(tm.unposted\_accruals, 0) unposted\_accruals,

nvl(tm.paid\_amount, 0) paid\_amount,

tm.datasource\_num\_id,

tm.integration\_id,

tm.etl\_proc\_id

from

(

select distinct

fun.fund\_id::varchar(16777216) fund\_ids,

fun.cust\_account\_id::varchar(

16777216

) cust\_account\_ids,

fun.org\_id::varchar(16777216) org\_ids,

fun.ledger\_id::varchar(16777216) ledger\_ids,

nvl(DEV\_EDW.EDW\_OMNI.HOT\_GET\_REP\_KEY(fun.ship\_to\_site\_use\_id,'ALL','ALL',fun.cust\_account\_id),'0') x\_terr\_ids,

nvl(fun.gl\_date, fun.adjustment\_date) gl\_date,

sum(

case

when

fun.utilization\_type = 'ACCRUAL'

and fun.gl\_posted\_flag = 'Y'

and fun.gl\_date is not null

then accrual\_loc\_amt

end

)

accrual\_amount,

sum(

case

when

fun.utilization\_type = 'ADJUSTMENT'

and fun.gl\_posted\_flag = 'Y'

and fun.gl\_date is not null

then accrual\_loc\_amt

end

)

adjustment\_amount,

sum(

case

when

fun.gl\_posted\_flag <> 'Y'

and fun.gl\_date is null

then accrual\_loc\_amt

end

)

unposted\_accruals,

0 paid\_amount,

fun.\_source\_id datasource\_num\_id,

coalesce(to\_char(fun.cust\_account\_id), '')

|| '~'

||coalesce(to\_char(fun.fund\_id), '')

|| '~'

||coalesce(to\_char(fun.org\_id), '')

|| '~'

||coalesce(to\_char(fun.ledger\_id), '')

|| '~'

||coalesce(to\_char(fun.ship\_to\_site\_use\_id), '')

|| '~'

|| to\_char(fun.\_source\_id) integration\_id ,

9999 etl\_proc\_id

from wc\_trade\_funds\_f fun

where utilization\_type in ('ACCRUAL', 'ADJUSTMENT')

group by

ledger\_ids,

fund\_ids,

cust\_account\_ids,

org\_ids,

x\_terr\_ids,

nvl(fun.gl\_date, fun.adjustment\_date),

\_source\_id,

ship\_to\_site\_use\_id

union all

select distinct

cl.fund\_id::varchar(16777216) fund\_id1,

cl.cust\_account\_id::varchar(

16777216

) cust\_account\_id1,

cl.org\_id::varchar(16777216) org\_id1,

wc\_fund\_d.ledger\_id::varchar(16777216) ledger\_id1,

nvl(DEV\_EDW.EDW\_OMNI.HOT\_GET\_REP\_KEY(cl.cust\_shipto\_acct\_site\_id,'ALL','ALL',cl.cust\_account\_id),'0') x\_terr\_id,

cl.gl\_date gl\_date,

0 accrual\_amt,

0 adjust\_amt,

0 unposted\_amt,

sum(cl.settled\_doc\_amt) paid\_amount,

cl.\_source\_id datasource\_num\_id,

coalesce(to\_char(cl.cust\_account\_id), '')

|| '~'

||coalesce(to\_char(cl.fund\_id), '')

|| '~'

||coalesce(to\_char(cl.org\_id), '')

|| '~'

||coalesce(to\_char(wc\_fund\_d.ledger\_id), '')

|| '~'

||coalesce(to\_char(cl.cust\_shipto\_acct\_site\_id), '')

|| '~'

|| to\_char(cl.\_source\_id) integration\_id,

9999 etl\_proc\_id

from wc\_claims\_f cl

left outer join

wc\_fund\_d on fund\_id1 = wc\_fund\_d.fund\_id

where cl.payment\_status = 'PAID'

-- and fund\_id = 490692

-- x\_terr\_id,ledger\_id

group by

fund\_id1,

cust\_account\_id1,

org\_id1,

ledger\_id1,

x\_terr\_id,

gl\_date,

\_source\_id,

cust\_shipto\_acct\_site\_id

) tm

left outer join

(

-- , x\_terr\_id, ledger\_id

select fund\_id, cust\_account\_id, org\_id, ledger\_id, x\_terr\_id

from

(

select distinct

fund\_id,

cust\_account\_id,

org\_id,

ledger\_id,

x\_terr\_id,

a\_gl\_date,

min(a\_gl\_date)

min\_date,

max(a\_gl\_date)

max\_date

from

(

select

a.fund\_id::varchar(

16777216

) fund\_id,

a.cust\_account\_id::varchar(

16777216

) cust\_account\_id,

a.org\_id::varchar(16777216) org\_id,

a.ledger\_id::varchar(

16777216

) ledger\_id,

nvl(DEV\_EDW.EDW\_OMNI.HOT\_GET\_REP\_KEY(a.ship\_to\_site\_use\_id,'ALL','ALL',a.cust\_account\_id),'0') x\_terr\_id,

a.gl\_date a\_gl\_date,

a.\_source\_id datasource\_num\_id,

coalesce(to\_char(a.cust\_account\_id), '')

|| '~'

||coalesce(to\_char(a.fund\_id), '')

|| '~'

||coalesce(to\_char(a.org\_id), '')

|| '~'

||coalesce(to\_char(a.ledger\_id), '')

|| '~'

||coalesce(to\_char(a.ship\_to\_site\_use\_id), '')

|| '~'

|| to\_char(a.\_source\_id) integration\_id,

9999 etl\_proc\_id

-- x\_terr\_id

from wc\_trade\_funds\_f a

where

a.utilization\_type

in ('ACCRUAL', 'ADJUSTMENT')

union all

select distinct

k.fund\_id::varchar fund\_id2,

k.cust\_account\_id ::varchar cust\_account\_id2,

k.org\_id::varchar org\_id2,

wc\_fund\_d.ledger\_id

::varchar ledger\_id2,

nvl(DEV\_EDW.EDW\_OMNI.HOT\_GET\_REP\_KEY(k.cust\_shipto\_acct\_site\_id::varchar,'ALL','ALL',k.cust\_account\_id::varchar),'0') x\_terr\_id,

k.gl\_date k\_gl\_date,

k.\_source\_id datasource\_num\_id,

coalesce(to\_char(k.cust\_account\_id), '')

|| '~'

||coalesce(to\_char(k.fund\_id), '')

|| '~'

||coalesce(to\_char(k.org\_id), '')

|| '~'

||coalesce(to\_char(wc\_fund\_d.ledger\_id), '')

|| '~'

||coalesce(to\_char(k.cust\_shipto\_acct\_site\_id), '')

|| '~'

|| to\_char(k.\_source\_id) integration\_id,

9999 etl\_proc\_id

-- x\_terr\_id

from wc\_claims\_f k

left outer join

wc\_fund\_d

on k.fund\_id::varchar

= wc\_fund\_d.integration\_id::varchar

where

k.payment\_status = 'PAID'

and k.fund\_id::varchar > '0'

group by

k.fund\_id::varchar,

k.cust\_account\_id::varchar,

k.cust\_shipto\_acct\_site\_id::varchar,

k.org\_id::varchar,

wc\_fund\_d.ledger\_id::varchar,

k\_gl\_date,

k.\_source\_id

-- and a.fund\_wid = 490692

) h

group by

fund\_id,

cust\_account\_id,

org\_id,

ledger\_id,

x\_terr\_id,

a\_gl\_date

-- x\_terr\_id,ledger\_id

) fund

-- having

--

having

fund.a\_gl\_date <= trunc(current\_date(), 'MM')

and fund.a\_gl\_date

between trunc(min\_date, 'MM') and trunc(

max\_date, 'MM'

)

order by fund.a\_gl\_date

) fund\_periods

on fund\_periods.fund\_id = tm.fund\_ids

and fund\_periods.cust\_account\_id = tm.cust\_account\_ids

and fund\_periods.ledger\_id = tm.ledger\_ids

and fund\_periods.org\_id = tm.org\_ids

and fund\_periods.x\_terr\_id = tm.x\_terr\_ids

) b

-- where b.fund\_id = 496226 and cust\_account\_id =

-- 664467,ledger\_id,x\_terr\_id

group by cust\_account\_id, org\_id, fund\_id, ledger\_id,x\_terr\_id,datasource\_num\_id,integration\_id

)

)

select \*

from final

