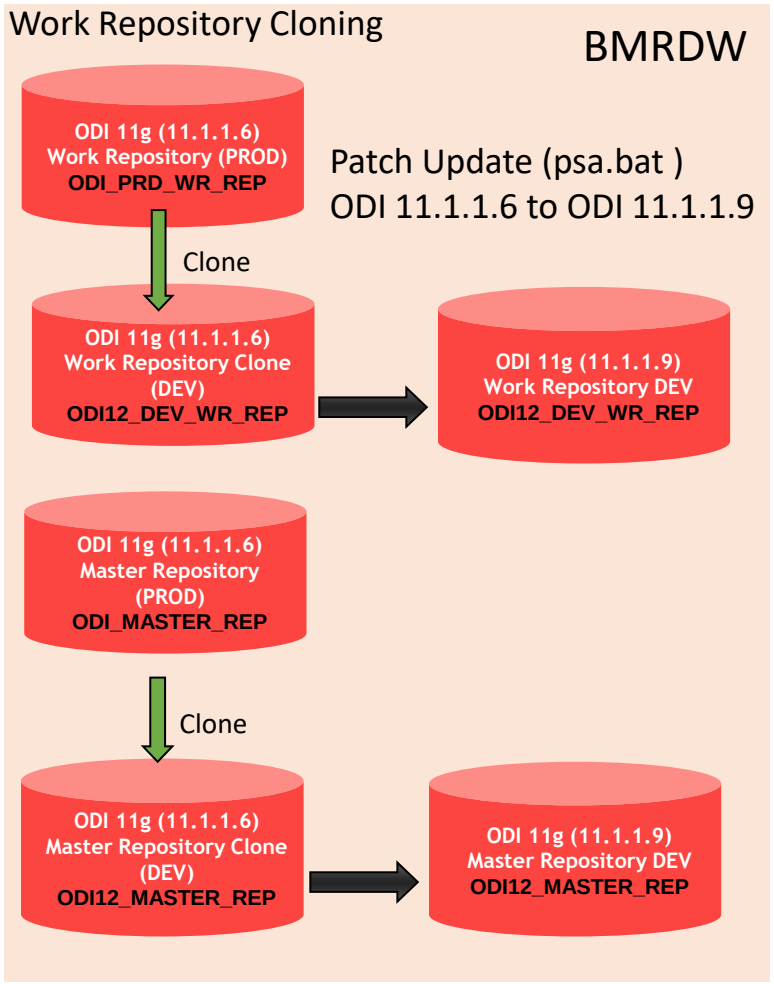
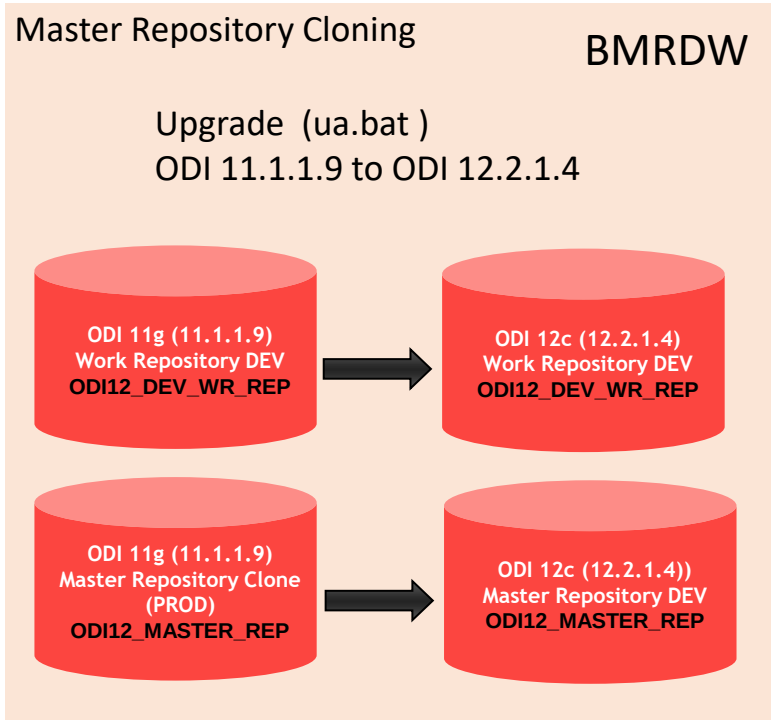
ODI 12c Upgrade KT Session

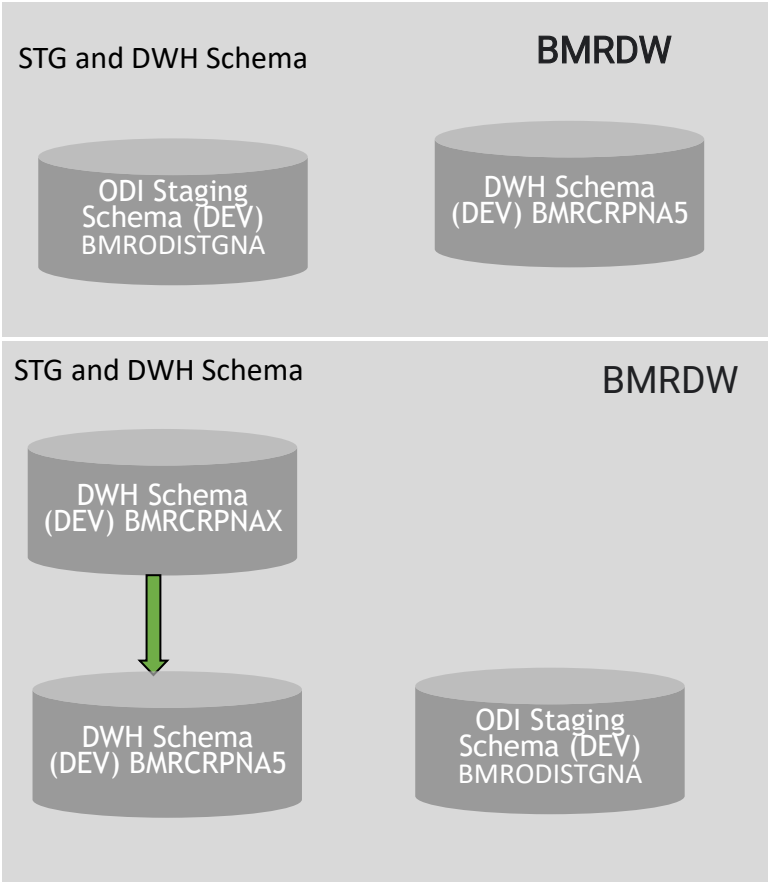
Agenda

- ODI 11g to ODI 12c Upgrade Steps
- ODI 12c new features overview
- ODI 12c features – Example
- ODI 11g to ODI 12c Migration Issues – Example
- Demo
- Q & A

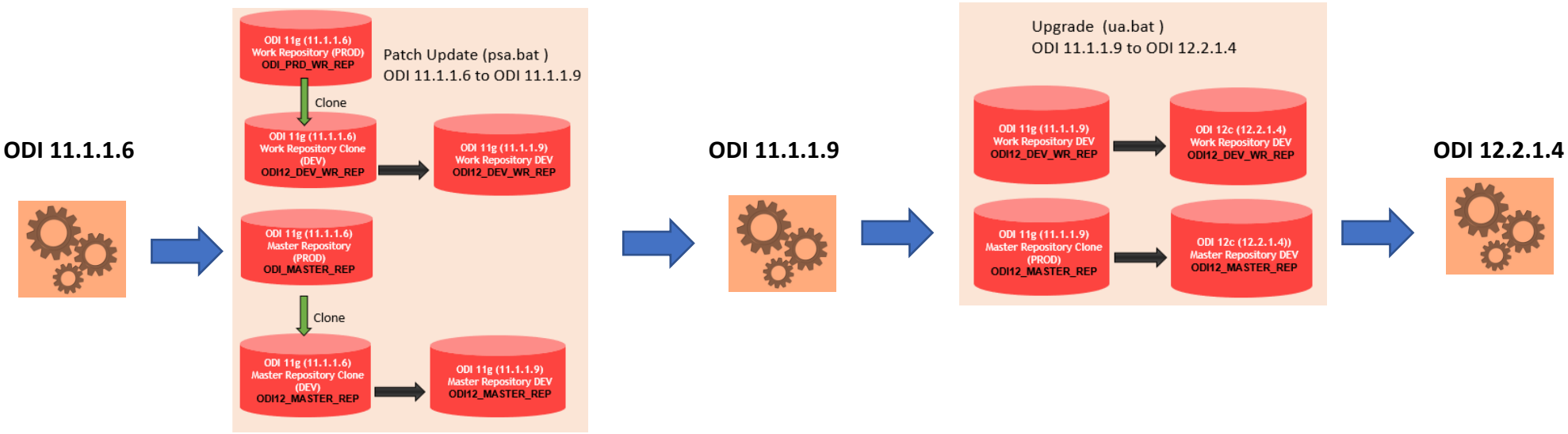
High level Upgrade Steps



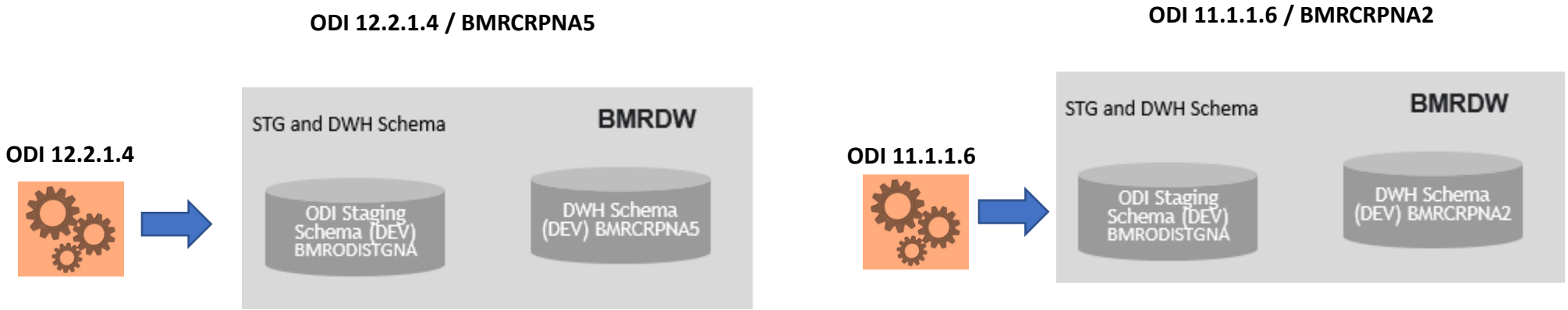




ODI Repository Upgrade Steps



Warehouse Schema



New Features of ODI 12c - Overview

- **Mappings** in place of Interface in Oracle Data Integrator 11g
- **Reusable Mappings** in place of Yellow interface
- **Flow based user interface**, connect sources to targets through a flow of components such as: Join, Filter, Aggregate, Set, Split, etc.
- Multiple Target Support
- **Knowledge Module Architecture**
  - Component-Style KM : provides an extensible component framework that improves the overall mapping design, where for example users are able to declare the transformation order
  - Template-Style KM (11g)
- Wallet Password in ODI 12c
- **Agents Configuration in ODI 12C Standalone Agent** vs **ODI 11g Agents.**
  - Manual intervention is required to configure odiparams.bat file in ODI 11g.
  - UI driven configuration through configuration wizard in ODI 12c.

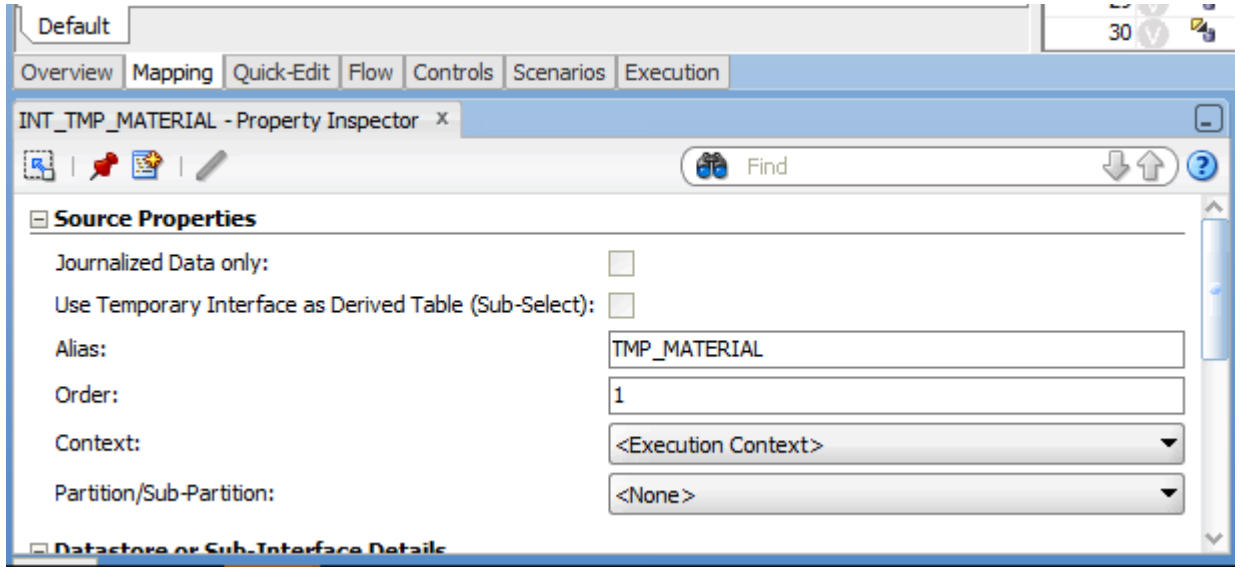
Yellow Interface to Reusable Mapping

- During upgrade from Oracle Data Integrator ODI 11g to ODI 12c, Temporary Interfaces are converted into Reusable Mappings.
- This results in creation of _RMT and _RMS Reusable Mappings.
- A temporary table gets created during execution.

ODI 11g Temporary Interface Upgrade to ODI 12c Reusable Mappings

In ODI 11g, a sub-interface component has a Use Temporary Interface as Derived Table (Sub-Select) property.
The value of this property determines which of _RMS or _RMT Reusable Mapping will be used after upgrading to ODI 12c:
•In case where the property "Use Temporary Interface as Derived Table (Sub-Select)" *is unchecked* prior to the upgrade or import from 11g, the _RMT Reusable Mapping is used.

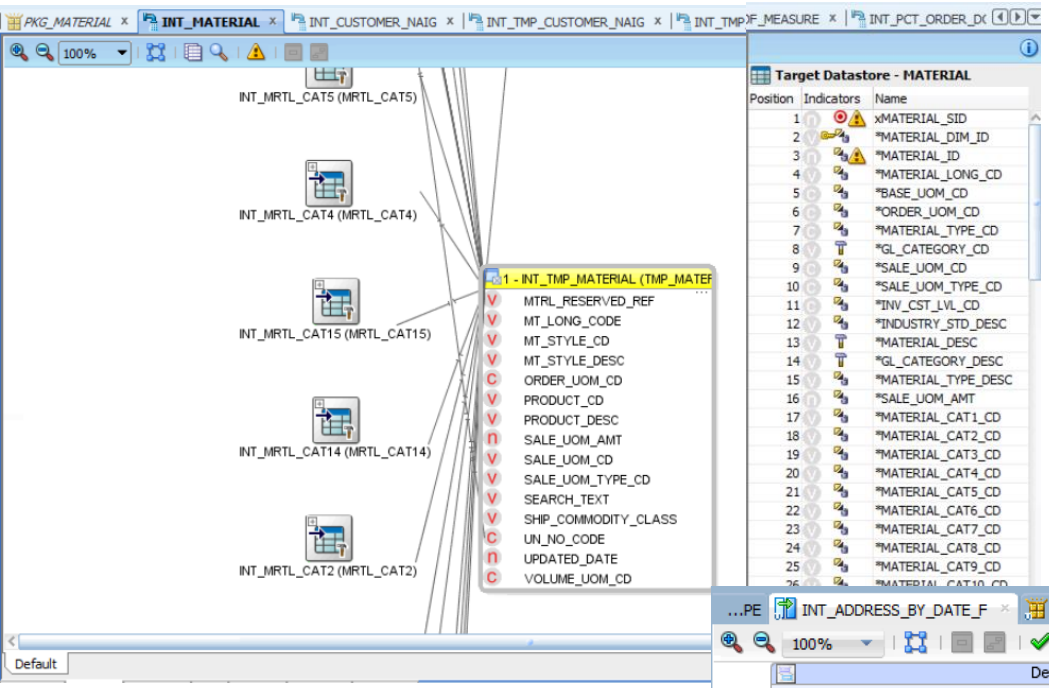
•If in the 11g the above property *is checked* and the Temporary Interface / Repository is upgraded or imported to 12c, the Reusable Mapping used is _RMS.



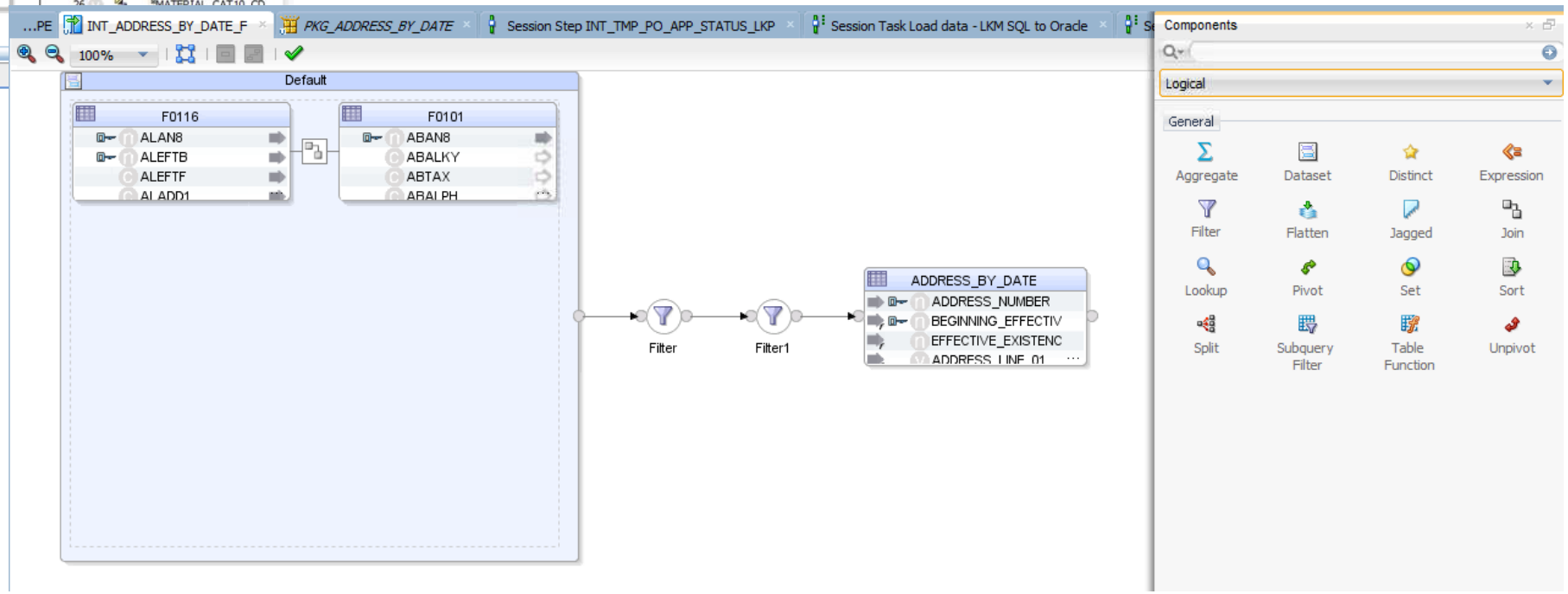
Use Temporary Interface as Derived Table (Sub-Select) - This option is available only if the source datastore is a temporary target datastore of another interface.

When this box is selected, the interface generates a sub-select corresponding to the loading of the temporary datastore. If this box is not selected, make sure to run the interface loading the temporary datastore before running the current interface.

ODI 11g User Interface

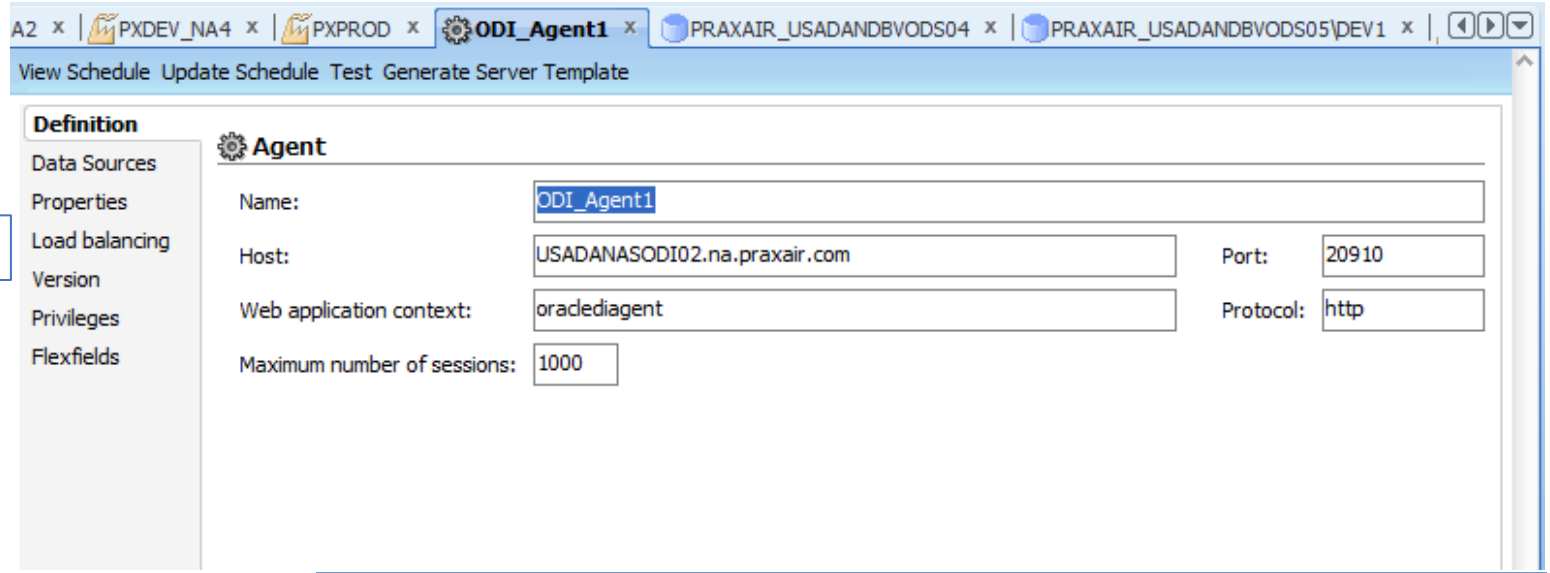


ODI 12c Declarative Flow-Based User Interface

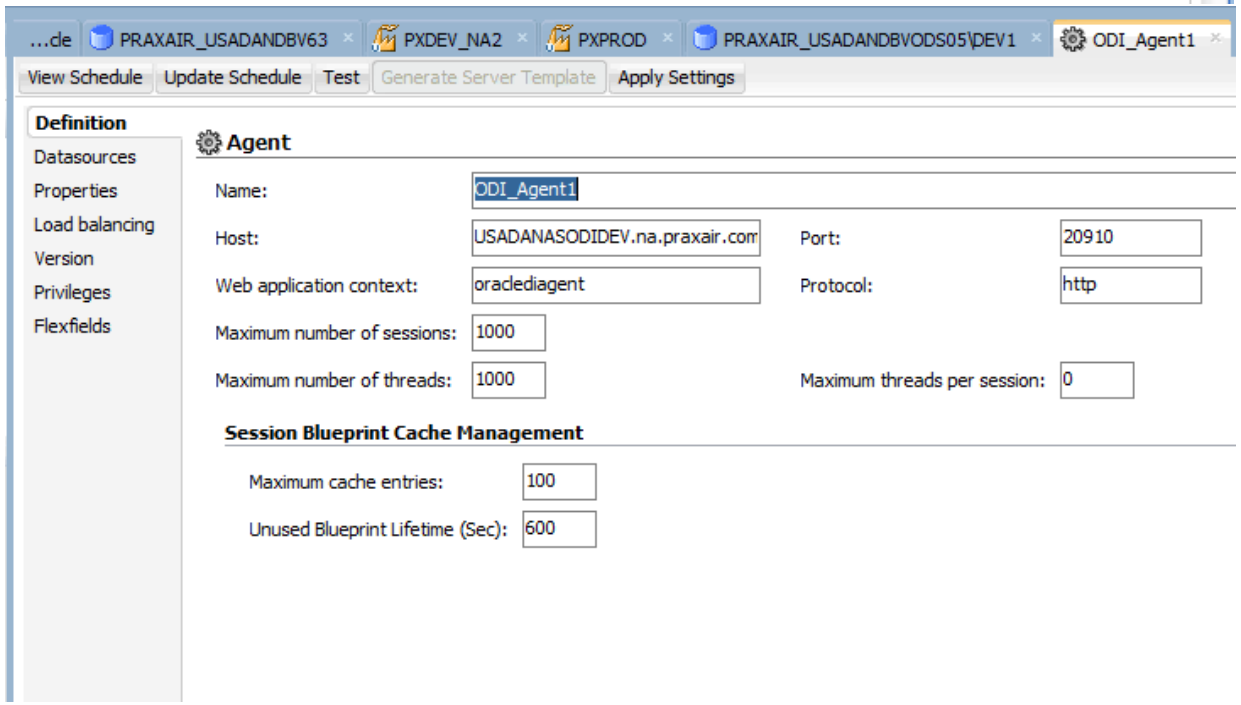


Agents Configuration

Manual configuration of odiparams.bat file in ODI 11g



UI driven configuration through configuration wizard in ODI 12c



C:\oracle\product\12.2.1.4\odi\common\bin\Configuration.cmd

Template-Style KM (11g)

Default Language | Highlight: the whole expression | Validate Syntax

Command Text
Groovy Task-local
Groovy KM-local
Validation Results

```
<%for (int i=odiRef.getDataSetMin(); i <= odiRef.getDataSetMax(); i++){%>
<%=odiRef.getDataSet(i, "Operator")%>
select  <%=odiRef.getPop("DISTINCT_ROWS")%>
        <%=odiRef.getColList(i, "", "[EXPRESSION]\t[ALIAS_SEP] [CX_COL_NAME]", ",\n\t", "", "")%>
from    <%=odiRef.getFrom(i)%>
where   (1=1)
<%=odiRef.getFilter(i)%>
<%=odiRef.getJrnFilter(i)%>
<%=odiRef.getJoin(i)%>
<%=odiRef.getGrpBy(i)%>
<%=odiRef.getHaving(i)%>
<%=odiRef.getOption("ORDER BY")%>
<%}%>
```

Component-Style KM

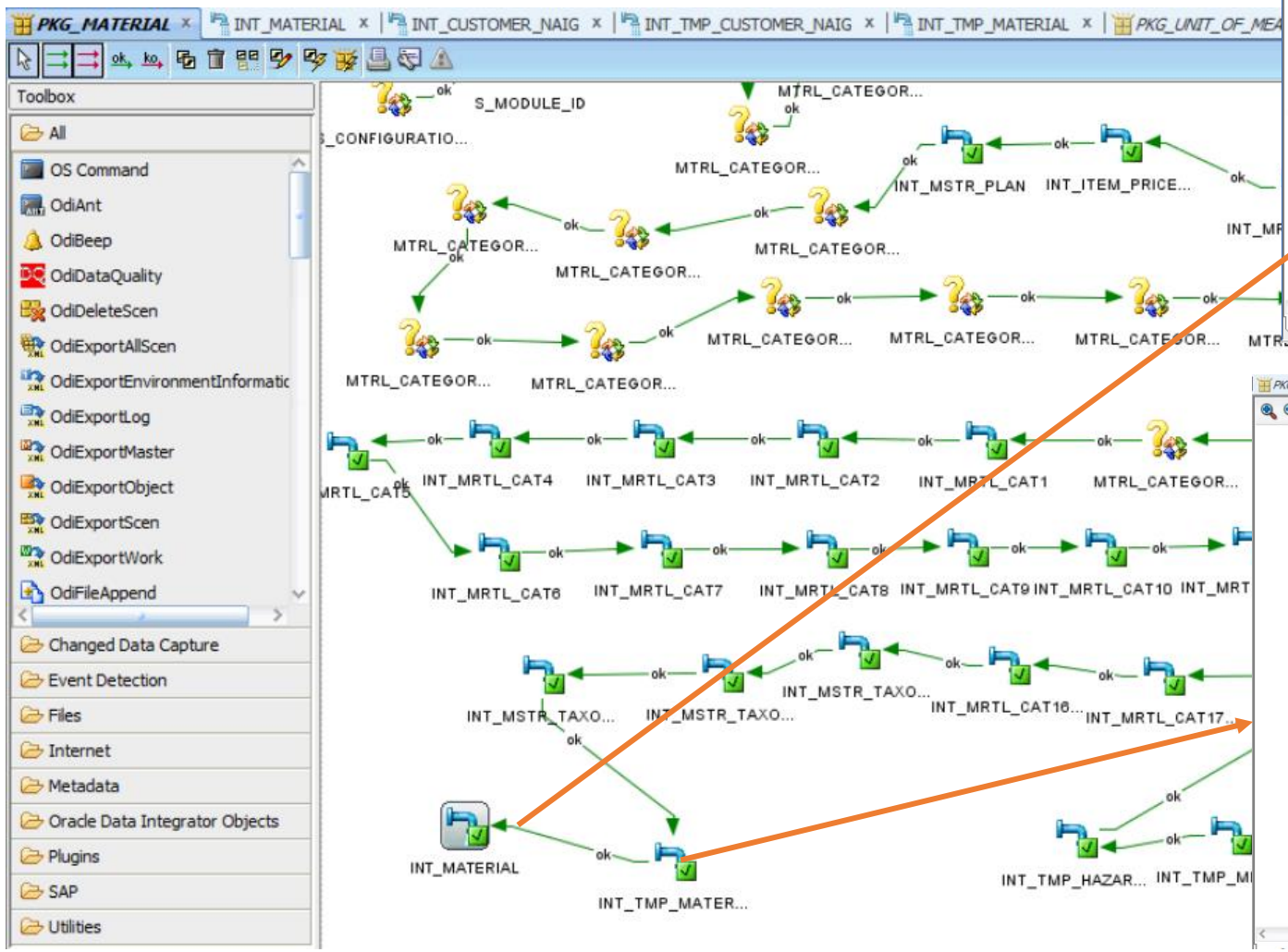
SQL | Highlight: the whole expression | Validate Syntax

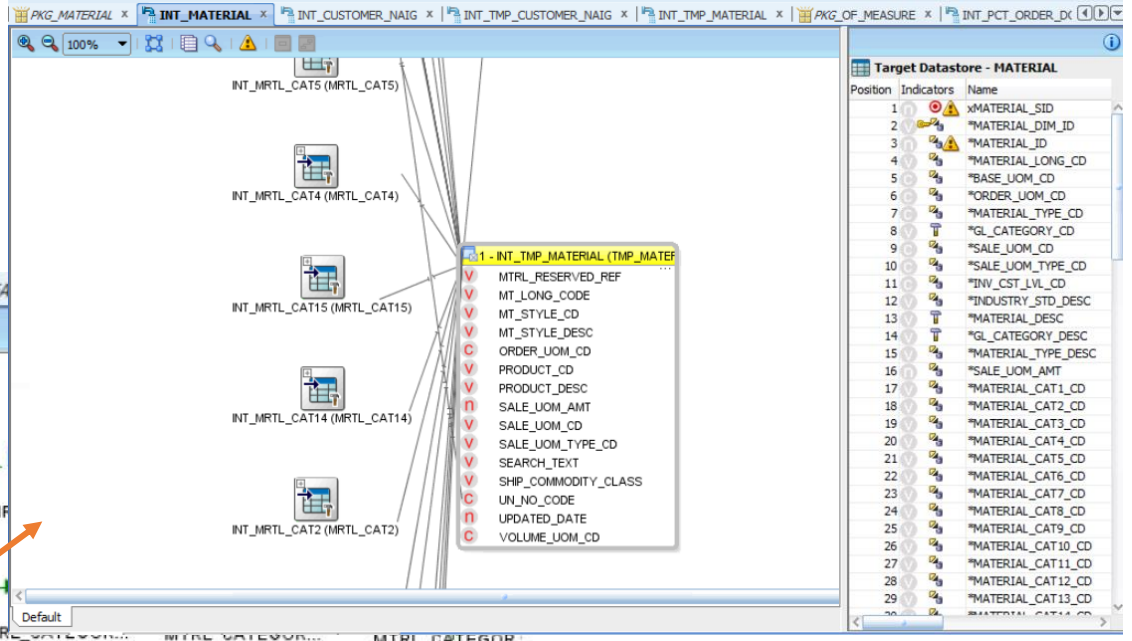
Command Text
Groovy Task-local
Groovy KM-local
Validation Results

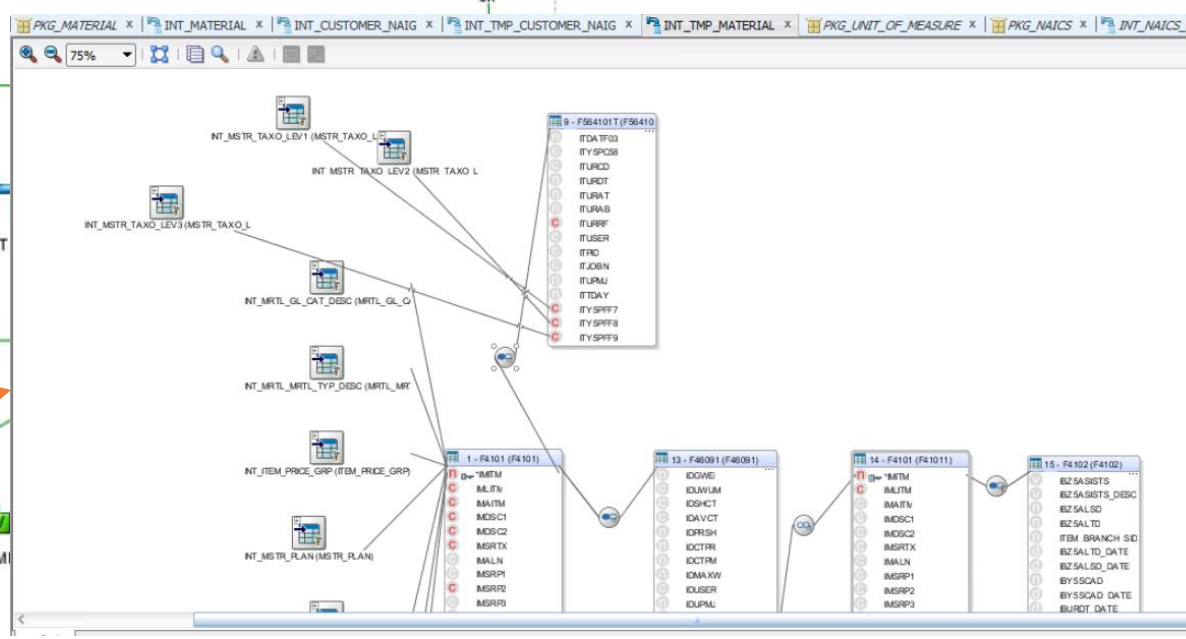
```
{#NL#}
INSERT INTO $[TARGET.getGeneratedName()] ({#NL#}
      $[TGTCOLLISTTEXT] {#NL#}
      ){#NL#}
      VALUES(  {#NL#}
      {# LIST #} <?=":"+"$[COLLIST.foreach(getName())]" ?> {#SEP#} , {#NL#}{# ENDLIST #}{#NL#}
      ){#NL#}


{#NL#}
SELECT
{# IF $[QUERY.isDistinct()] #}{#NL#}
  DISTINCT
{# ENDIF #}
{# IF ($[QUERY.getAliasList()] != 'null') #}{#NL#}
{# LIST #}  $[QUERY.getSelectList().foreach(getText())] $[QUERY.getColumnAliasSeparator()] $[QUERY.getAliasList()] {# SE
{# ELSE #}{#NL#}
{# LIST #}  $[QUERY.getSelectList().foreach(getText())] {# SEP #},{#NL#}{# ENDLIST #}
{# ENDIF #}
{#NL#}
```

A ODI 11c Package upgraded to ODI 12c – PKG_MATERIAL

11g Package : PKG_MATERIAL

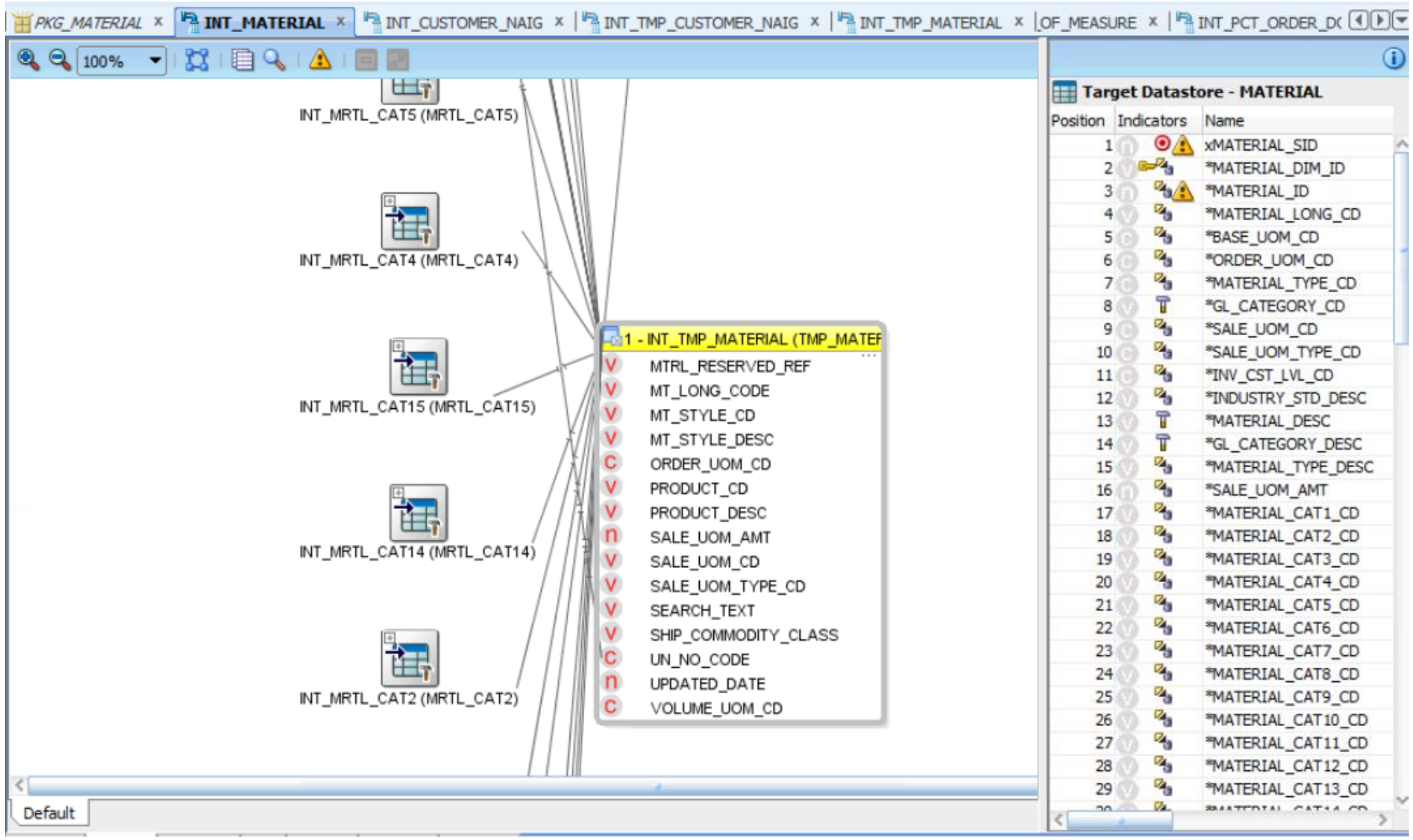






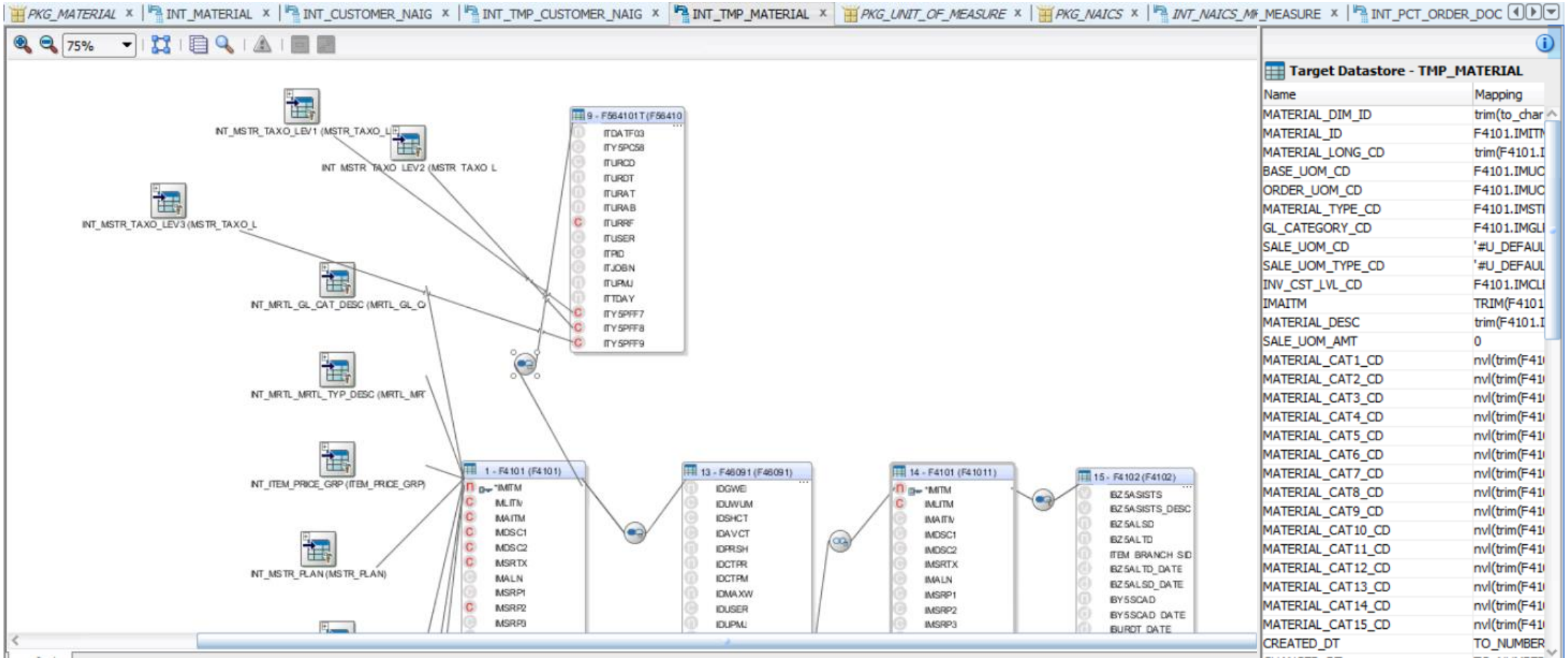
11g Interface : INT_MATERIAL

Main Interface

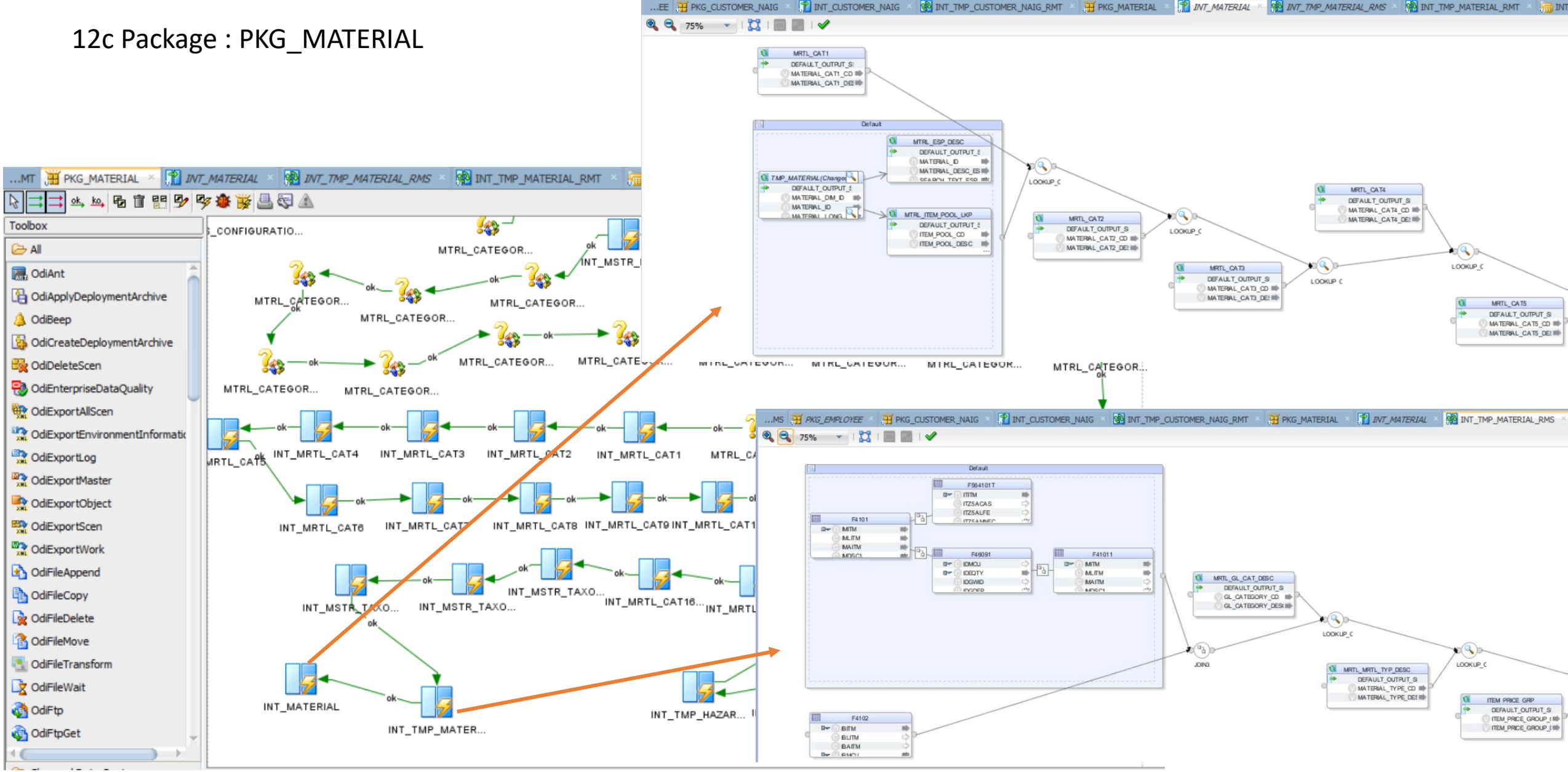


11g Temp Interface : INT_TMP_MATERIAL

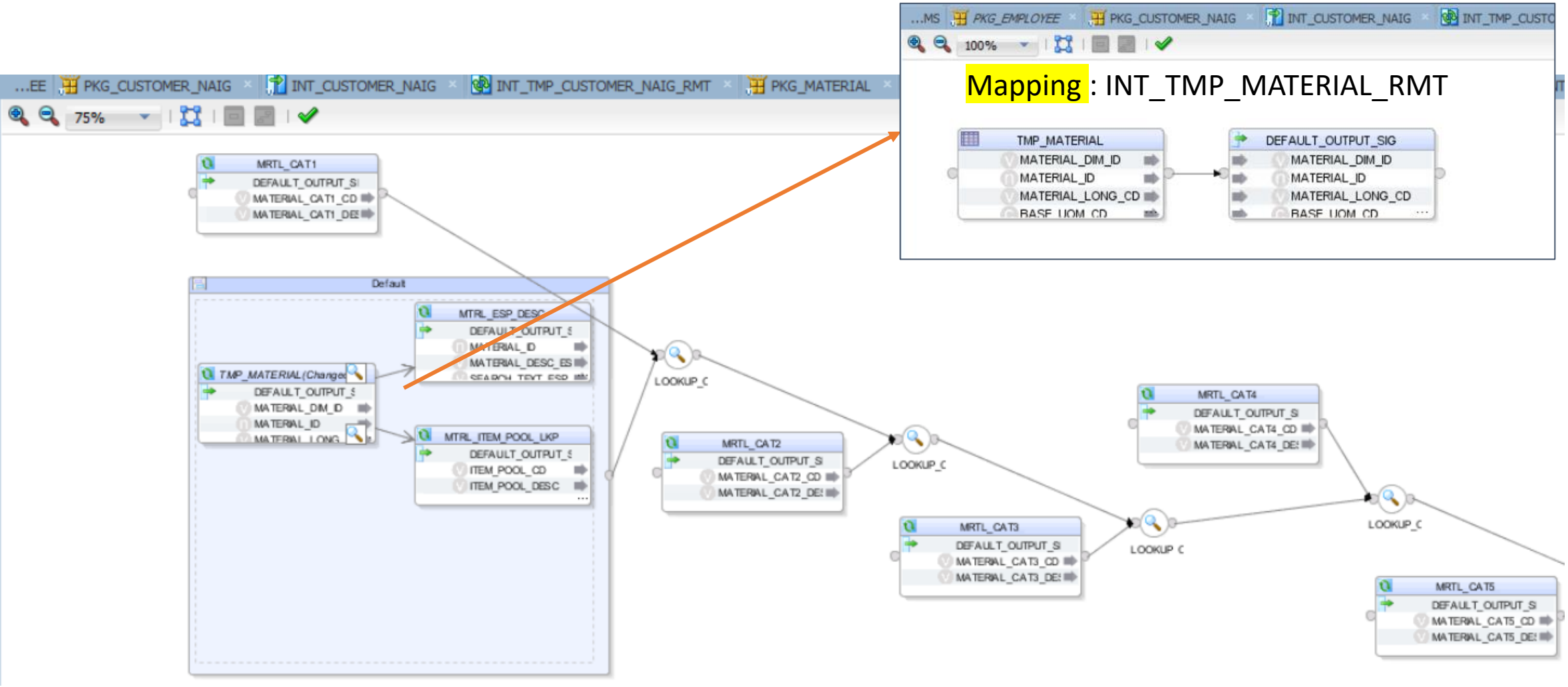
Temp Interface



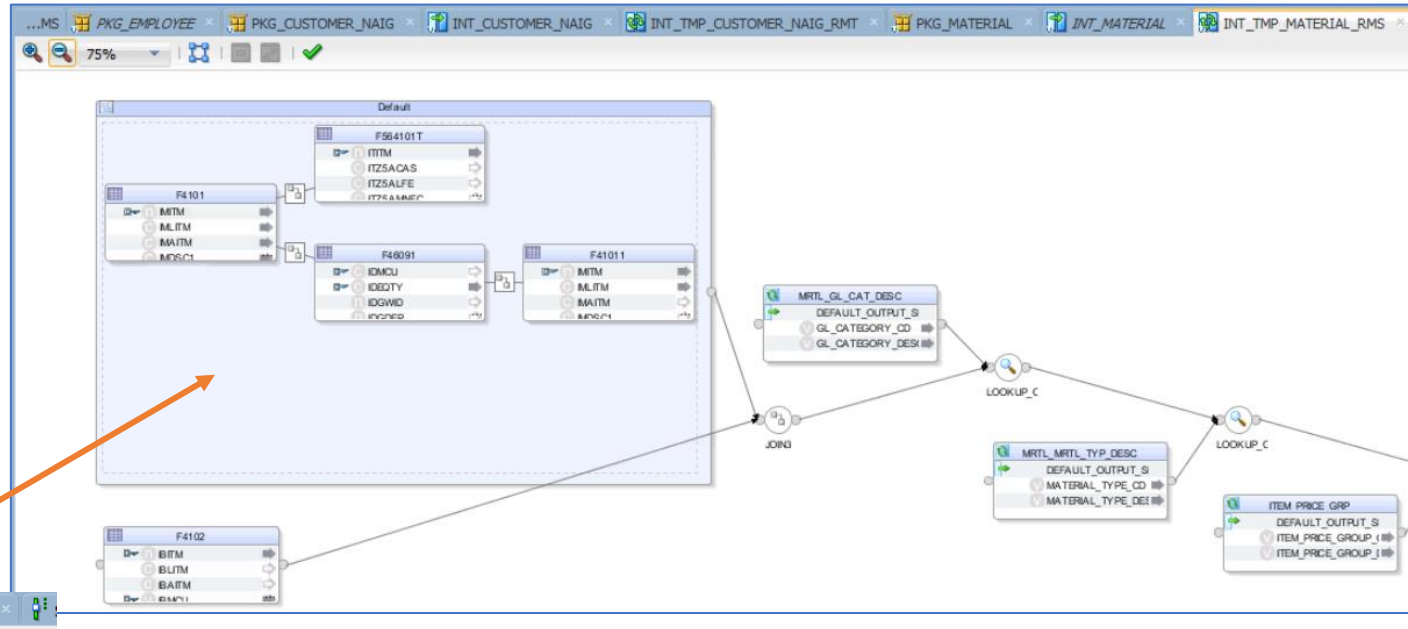
12c Package : PKG_MATERIAL



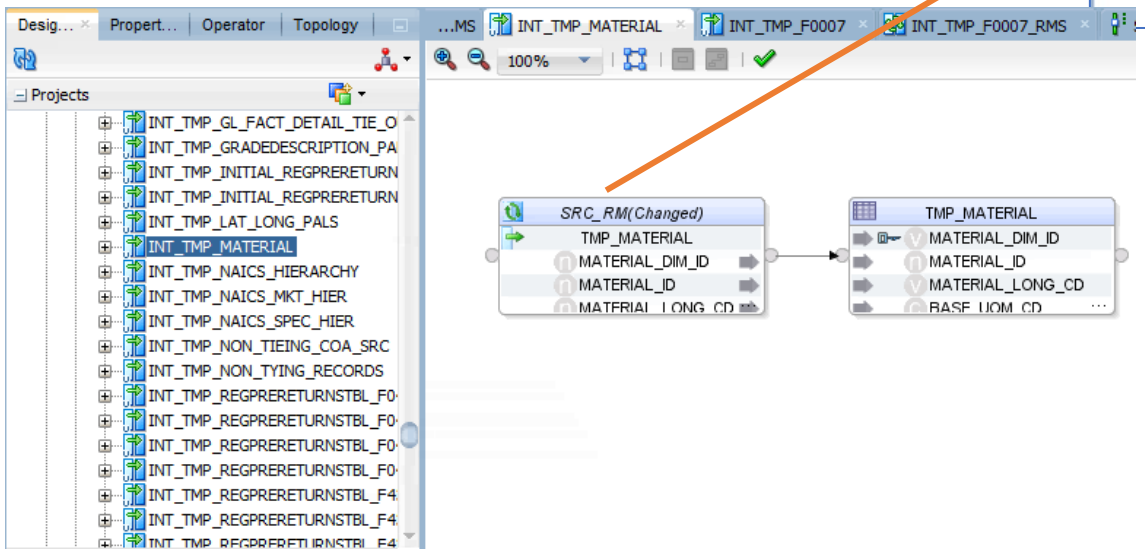
Main Mapping : INT_MATERIAL



INT_TMP_MATERIAL_RMS (reusable mapping source)

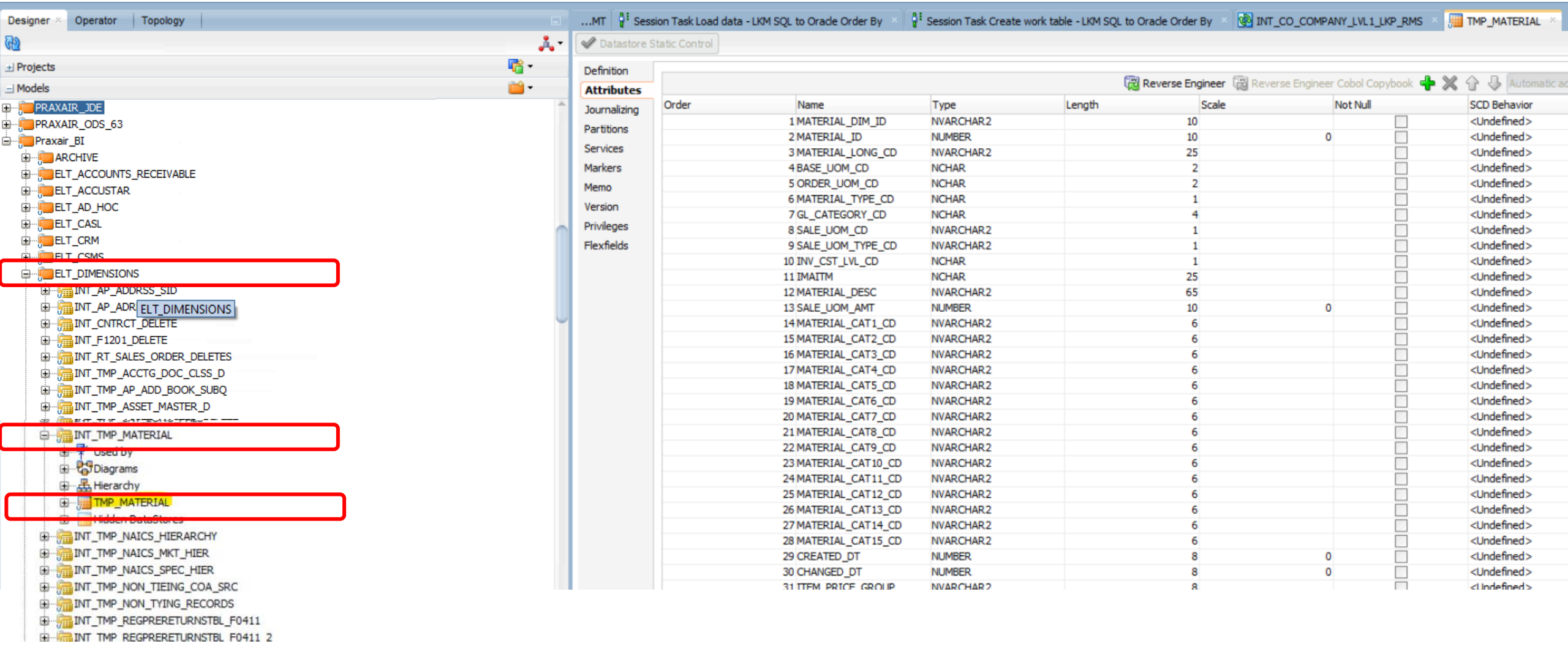


INT_TMP_MATERIAL



Model : INT_TMP_MATERIAL

New Model folder Praxair_BI is created.
New Model got created for Reusable Mapping.



Order	Name	Type	Length	Scale	Not Null	SCD Behavior
1	MATERIAL_DIM_ID	NVARCHAR2	10			<Undefined>
2	MATERIAL_ID	NUMBER	10	0		<Undefined>
3	MATERIAL_LONG_CD	NVARCHAR2	25			<Undefined>
4	BASE_UOM_CD	NCHAR	2			<Undefined>
5	ORDER_UOM_CD	NCHAR	2			<Undefined>
6	MATERIAL_TYPE_CD	NCHAR	1			<Undefined>
7	GL_CATEGORY_CD	NCHAR	4			<Undefined>
8	SALE_UOM_CD	NVARCHAR2	1			<Undefined>
9	SALE_UOM_TYPE_CD	NVARCHAR2	1			<Undefined>
10	INV_CST_LVL_CD	NCHAR	1			<Undefined>
11	IMAITM	NCHAR	25			<Undefined>
12	MATERIAL_DESC	NVARCHAR2	65			<Undefined>
13	SALE_UOM_AMT	NUMBER	10	0		<Undefined>
14	MATERIAL_CAT1_CD	NVARCHAR2	6			<Undefined>
15	MATERIAL_CAT2_CD	NVARCHAR2	6			<Undefined>
16	MATERIAL_CAT3_CD	NVARCHAR2	6			<Undefined>
17	MATERIAL_CAT4_CD	NVARCHAR2	6			<Undefined>
18	MATERIAL_CAT5_CD	NVARCHAR2	6			<Undefined>
19	MATERIAL_CAT6_CD	NVARCHAR2	6			<Undefined>
20	MATERIAL_CAT7_CD	NVARCHAR2	6			<Undefined>
21	MATERIAL_CAT8_CD	NVARCHAR2	6			<Undefined>
22	MATERIAL_CAT9_CD	NVARCHAR2	6			<Undefined>
23	MATERIAL_CAT10_CD	NVARCHAR2	6			<Undefined>
24	MATERIAL_CAT11_CD	NVARCHAR2	6			<Undefined>
25	MATERIAL_CAT12_CD	NVARCHAR2	6			<Undefined>
26	MATERIAL_CAT13_CD	NVARCHAR2	6			<Undefined>
27	MATERIAL_CAT14_CD	NVARCHAR2	6			<Undefined>
28	MATERIAL_CAT15_CD	NVARCHAR2	6			<Undefined>
29	CREATED_DT	NUMBER	8	0		<Undefined>
30	CHANGED_DT	NUMBER	8	0		<Undefined>
31	ITEM_PRICE_GROUP	NVARCHAR2	8			<Undefined>

ODI Operator 11g vs 12c

ODI 11g

- 69 - INT_TMP_MATERIAL - Jan 13, 2022 12:53:46 AM
  - 566 - Loading - SrcSet0 - Drop work table
  - 567 - Loading - SrcSet0 - Create work table
  - 568 - Loading - SrcSet0 - Load data
  - 569 - Loading - SrcSet0 - Analyze work table
  - 571 - Integration - INT_TMP_MATERIAL - Create target table
  - 572 - Integration - INT_TMP_MATERIAL - Drop flow table
  - 573 - Integration - INT_TMP_MATERIAL - Create flow table I$
  - 574 - Integration - INT_TMP_MATERIAL - Truncate target table
  - 575 - Integration - INT_TMP_MATERIAL - Insert flow into I$ table
  - 576 - Integration - INT_TMP_MATERIAL - Create Index on flow table
  - 577 - Integration - INT_TMP_MATERIAL - Analyze integration table
  - 578 - Integration - INT_TMP_MATERIAL - Flag rows for update
  - 579 - Integration - INT_TMP_MATERIAL - Flag useless rows
  - 580 - Integration - INT_TMP_MATERIAL - Insert new rows
  - 581 - Integration - INT_TMP_MATERIAL - Commit transaction
  - 582 - Integration - INT_TMP_MATERIAL - Drop flow table
  - 1000570 - Loading - SrcSet0 - Drop work table

- 70 - INT_MATERIAL - Jan 13, 2022 12:56:11 AM
  - 583 - Integration - INT_MATERIAL - Drop flow table
  - 584 - Integration - INT_MATERIAL - Create flow table I$
  - 585 - Integration - INT_MATERIAL - Insert flow into I$ table
  - 586 - Integration - INT_MATERIAL - Create Index on flow table
  - 587 - Integration - INT_MATERIAL - Analyze integration table
  - 588 - Control - MATERIAL - drop check table
  - 589 - Control - MATERIAL - drop check view
  - 590 - Control - MATERIAL - drop check EV
  - 591 - Control - MATERIAL - create check table
  - 592 - Control - MATERIAL - delete previous check sum
  - 593 - Control - MATERIAL - drop error table
  - 594 - Control - MATERIAL - create error table
  - 595 - Control - MATERIAL - delete previous errors
  - 596 - Control - MATERIAL - Create index on PK
  - 644 - Control - MATERIAL - insert Not Null errors
  - 645 - Control - MATERIAL - delete errors from controlled table
  - 646 - Control - MATERIAL - insert check sum into check table
  - 647 - Integration - INT_MATERIAL - Flag rows for update
  - 648 - Integration - INT_MATERIAL - Flag useless rows
  - 649 - Integration - INT_MATERIAL - Update existing rows
  - 650 - Integration - INT_MATERIAL - Insert new rows
  - 651 - Integration - INT_MATERIAL - Commit transaction
  - 652 - Integration - INT_MATERIAL - Analyze target table

ODI 12c

- 69 - INT_TMP_MATERIAL - Jan 12, 2022 9:45:26 AM
  - 10 - SERIAL - MAP_MAIN
    - 20 - SERIAL - EU - PXDEV_SOURCE_UNIT
      - 30 - Drop work table - LKM SQL to Oracle
      - 40 - Create work table - LKM SQL to Oracle
      - 50 - Load data - LKM SQL to Oracle
      - 60 - Analyze work table - LKM SQL to Oracle
    - 70 - SERIAL - EU - PXDEV_STAGE_UNIT
      - 80 - Create target table - IKM Oracle Incremental Update
      - 90 - Drop flow table - IKM Oracle Incremental Update
      - 100 - Create flow table I$ - IKM Oracle Incremental Update
      - 110 - Truncate target table - IKM Oracle Incremental Update
      - 120 - Insert flow into I$ table - IKM Oracle Incremental Update
      - 130 - Create Index on flow table - IKM Oracle Incremental Update
      - 140 - Analyze integration table - IKM Oracle Incremental Update
      - 150 - Flag rows for update - IKM Oracle Incremental Update
      - 160 - Flag useless rows - IKM Oracle Incremental Update
      - 170 - Disable constraints - IKM Oracle Incremental Update
      - 180 - Disable indexes - IKM Oracle Incremental Update
      - 190 - Insert new rows - IKM Oracle Incremental Update
      - 200 - Enable indexes - IKM Oracle Incremental Update
      - 210 - Enable constraints - IKM Oracle Incremental Update
      - 220 - Commit transaction - IKM Oracle Incremental Update
    - 230 - SERIAL - MAP_CLEANUP
      - 240 - SERIAL - EU - PXDEV_SOURCE_UNIT
        - 250 - Drop work table - LKM SQL to Oracle
      - 260 - SERIAL - EU - PXDEV_STAGE_UNIT
        - 270 - Drop flow table - IKM Oracle Incremental Update
- 70 - INT_MATERIAL - Jan 12, 2022 9:47:59 AM
  - 10 - SERIAL - MAP_MAIN
    - 20 - SERIAL - EU - PXDEV_TARGET_UNIT
      - 30 - Drop flow table - IKM Oracle Incremental Update
      - 40 - Create flow table I$ - IKM Oracle Incremental Update
      - 50 - Analyze target table - IKM Oracle Incremental Update
      - 60 - Insert flow into I$ table - IKM Oracle Incremental Update
      - 70 - Create Index on flow table - IKM Oracle Incremental Update
      - 80 - Analyze integration table - IKM Oracle Incremental Update
      - 90 - drop check table - CKM Oracle
      - 100 - drop check view - CKM Oracle
      - 110 - drop check EV - CKM Oracle
      - 120 - create check table - CKM Oracle
      - 130 - delete previous check sum - CKM Oracle
      - 140 - drop error table - CKM Oracle
      - 150 - create error table - CKM Oracle
      - 160 - delete previous errors - CKM Oracle
      - 170 - Create index on PK - CKM Oracle
      - 180 - insert PK errors - CKM Oracle
      - 190 - insert Not Null errors - CKM Oracle
      - 650 - insert Not Null errors - CKM Oracle
      - 660 - delete errors from controlled table - CKM Oracle
      - 670 - insert check sum into check table - CKM Oracle
      - 680 - Flag rows for update - IKM Oracle Incremental Update
      - 690 - Flag useless rows - IKM Oracle Incremental Update
      - 700 - Disable constraints - IKM Oracle Incremental Update
      - 710 - Disable indexes - IKM Oracle Incremental Update
      - 720 - Update existing rows - IKM Oracle Incremental Update
      - 730 - Insert new rows - IKM Oracle Incremental Update
      - 740 - Enable indexes - IKM Oracle Incremental Update
      - 750 - Enable constraints - IKM Oracle Incremental Update
      - 760 - Commit transaction - IKM Oracle Incremental Update
      - 770 - Analyze target table - IKM Oracle Incremental Update

Steps are Categorized based on stages

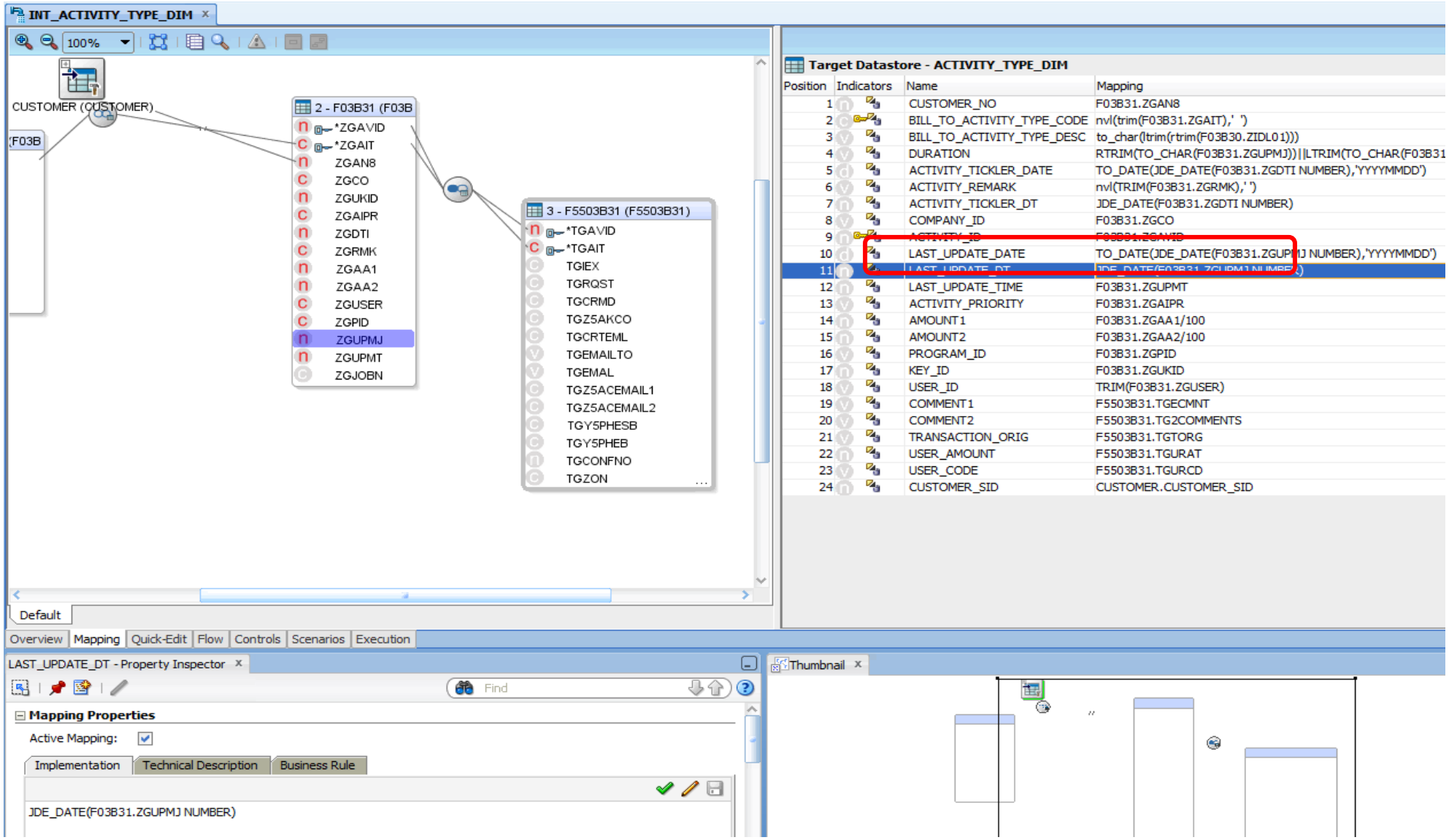
ODI 11g to ODI 12c Migration Issues

- Context Issue
- Refresh Variable Issue
- File Missing issue
- column length issue
- Batch script not in tidal scheduler
- Column missing on target issue
- Data issue – Too much of data or too less data (Extract from Date and to Date settings)

- jde_date issue
- Reusable mapping issue - (set, distinct, aggregate components)
- Variable calling issue

JDE_DATE Issue

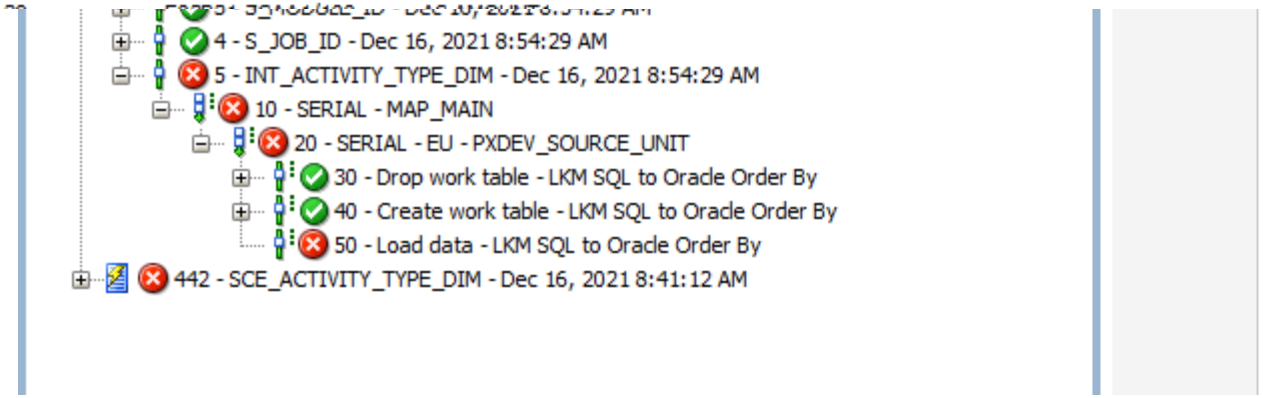
11g Code



JDE_DATE Issue

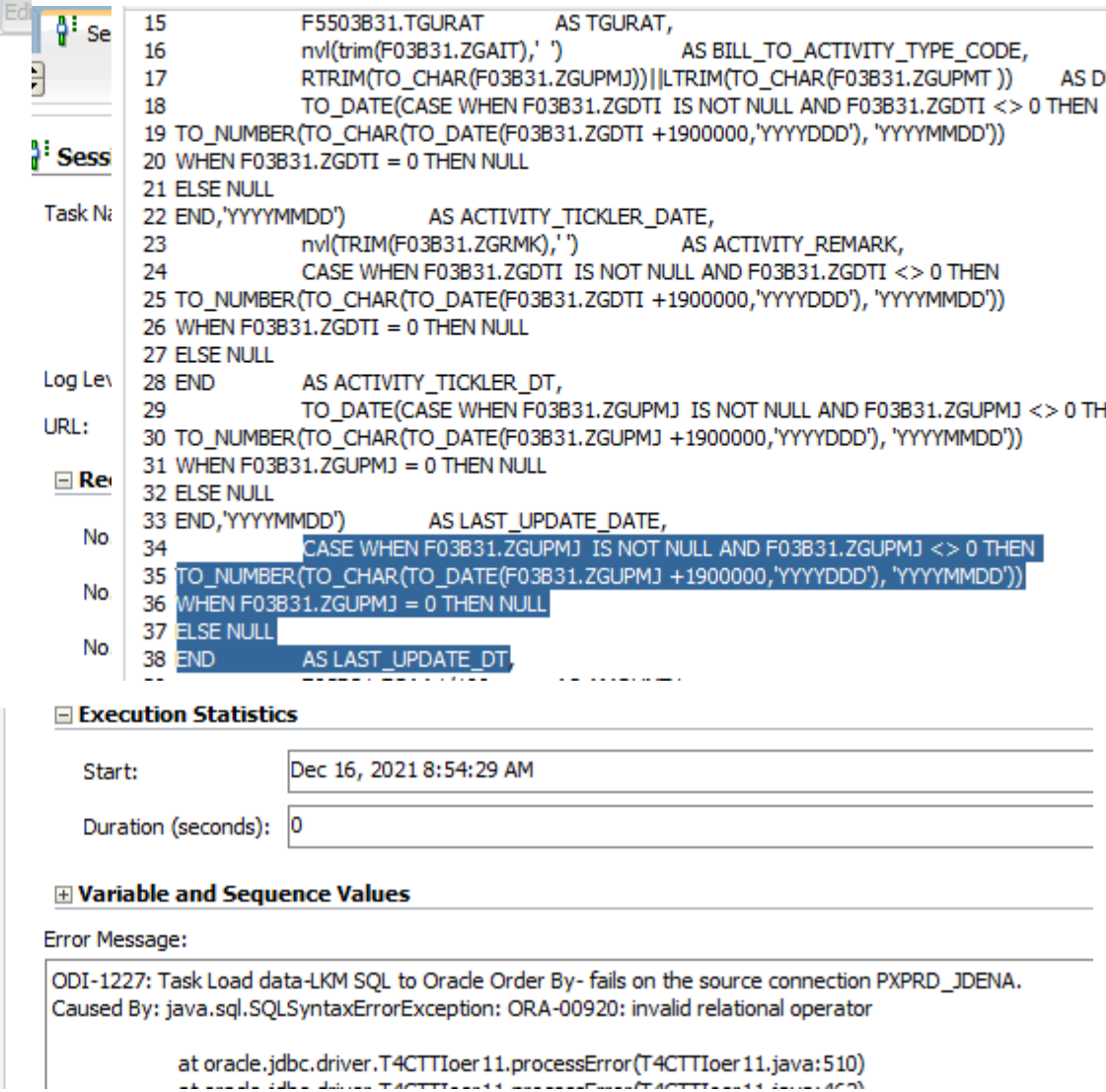
12c Code Before

```
15              F5503B31.TGURAT         AS TGURAT,
16              nvl(trim(F03B31.ZGAIT),' ')             AS BILL_TO_ACTIVITY_TYPE_CODE,
17              RTRIM(TO_CHAR(F03B31.ZGUPMJ))||LTRIM(TO_CHAR(F03B31.ZGUPMT ))       AS DURATION,
18              TO_DATE(CASE WHEN F03B31.ZGDTI NUMBER  IS NOT NULL AND F03B31.ZGDTI NUMBER <> 0 THEN
19 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGDTI NUMBER +1900000,'YYYYDDD'), 'YYYYMMDD'))
20 WHEN F03B31.ZGDTI NUMBER = 0 THEN NULL
21 ELSE NULL
22 END,'YYYYMMDD')          AS ACTIVITY_TICKLER_DATE,
23              nvl(TRIM(F03B31.ZGRMK),' ')             AS ACTIVITY_REMARK,
24              CASE WHEN F03B31.ZGDTI NUMBER  IS NOT NULL AND F03B31.ZGDTI NUMBER <> 0 THEN
25 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGDTI NUMBER +1900000,'YYYYDDD'), 'YYYYMMDD'))
26 WHEN F03B31.ZGDTI NUMBER = 0 THEN NULL
27 ELSE NULL
28 END           AS ACTIVITY_TICKLER_DT,
29              TO_DATE(CASE WHEN F03B31.ZGUPMJ NUMBER  IS NOT NULL AND F03B31.ZGUPMJ NUMBER <> 0 THEN
30 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGUPMJ NUMBER +1900000,'YYYYDDD'), 'YYYYMMDD'))
31 WHEN F03B31.ZGUPMJ NUMBER = 0 THEN NULL
32 ELSE NULL
33 END,'YYYYMMDD')          AS LAST_UPDATE_DATE,
34              CASE WHEN F03B31.ZGUPMJ NUMBER  IS NOT NULL AND F03B31.ZGUPMJ NUMBER <> 0 THEN
35 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGUPMJ NUMBER +1900000,'YYYYDDD'), 'YYYYMMDD'))
36 WHEN F03B31.ZGUPMJ NUMBER = 0 THEN NULL
37 ELSE NULL
38 END           AS LAST_UPDATE_DT,
```

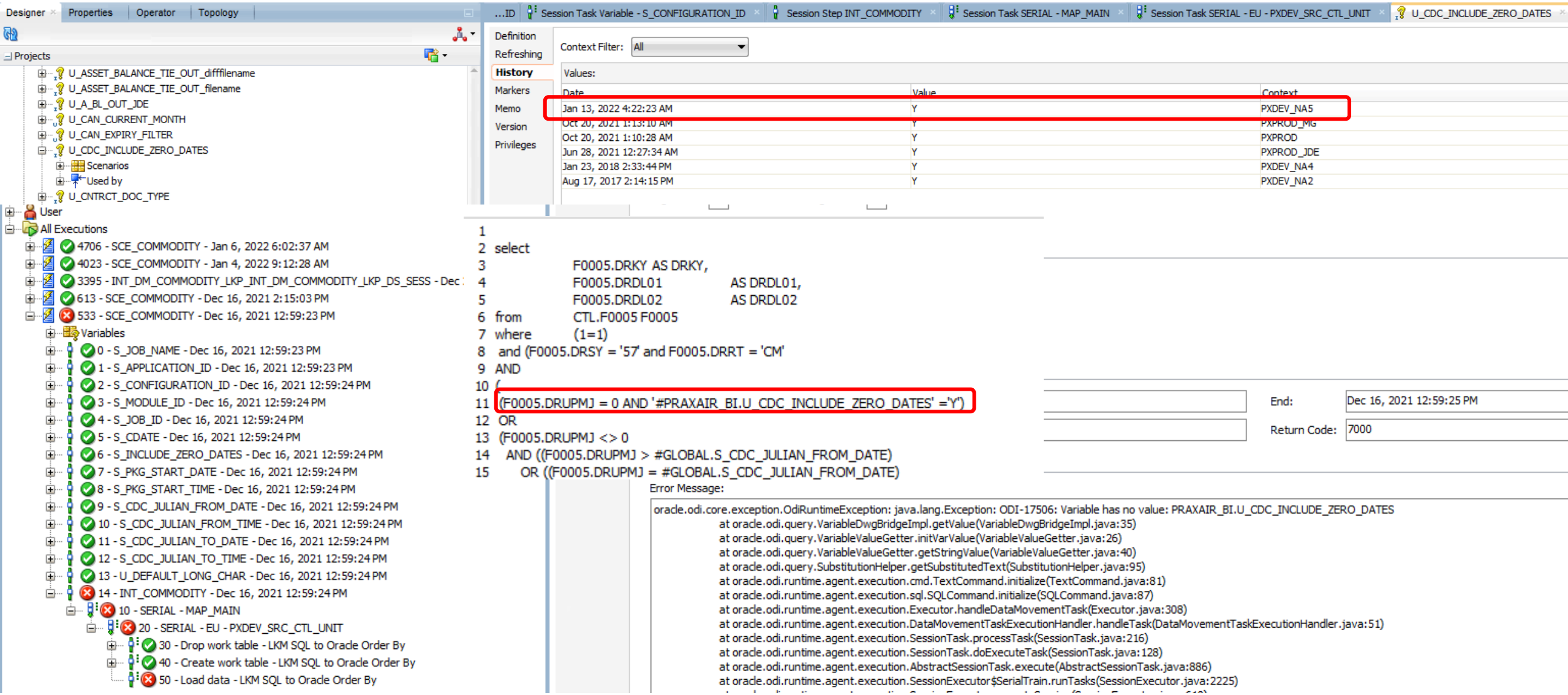


12c Code After

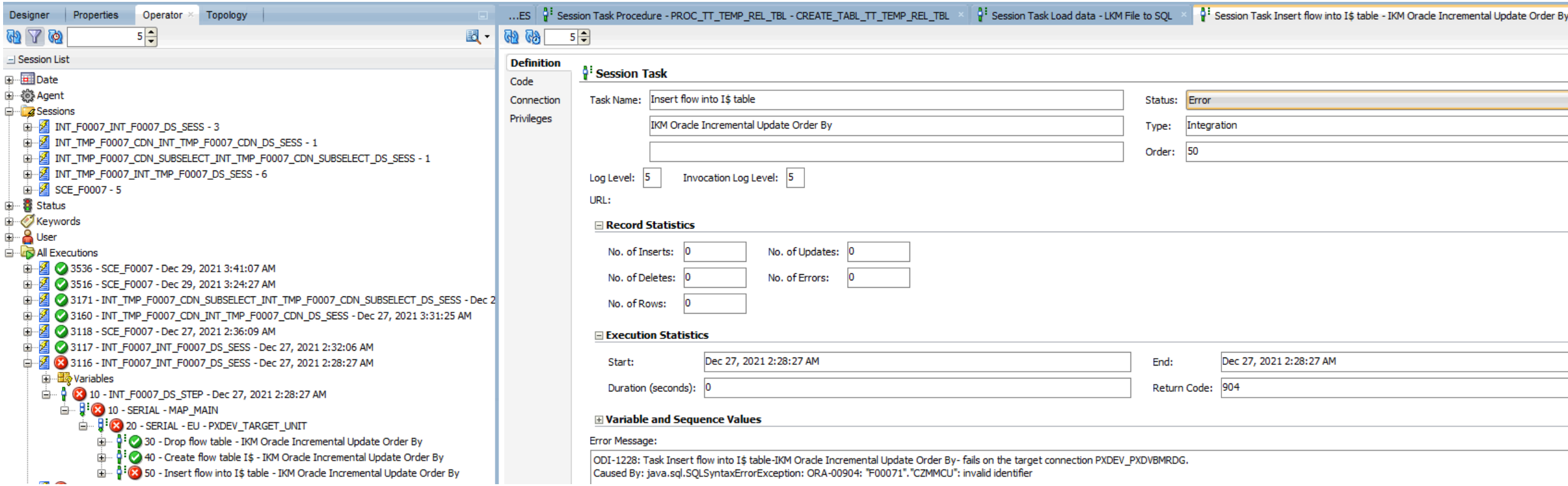
```
15              F5503B31.TGURAT         AS TGURAT,
16              nvl(trim(F03B31.ZGAIT),' ')             AS BILL_TO_ACTIVITY_TYPE_CODE,
17              RTRIM(TO_CHAR(F03B31.ZGUPMJ))||LTRIM(TO_CHAR(F03B31.ZGUPMT ))       AS DURAT
18              TO_DATE(CASE WHEN F03B31.ZGDTI  IS NOT NULL AND F03B31.ZGDTI <> 0 THEN
19 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGDTI +1900000,'YYYYDDD'), 'YYYYMMDD'))
20 WHEN F03B31.ZGDTI = 0 THEN NULL
21 ELSE NULL
22 END,'YYYYMMDD')          AS ACTIVITY_TICKLER_DATE,
23              nvl(TRIM(F03B31.ZGRMK),' ')             AS ACTIVITY_REMARK,
24              CASE WHEN F03B31.ZGDTI  IS NOT NULL AND F03B31.ZGDTI <> 0 THEN
25 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGDTI +1900000,'YYYYDDD'), 'YYYYMMDD'))
26 WHEN F03B31.ZGDTI = 0 THEN NULL
27 ELSE NULL
28 END           AS ACTIVITY_TICKLER_DT,
29              TO_DATE(CASE WHEN F03B31.ZGUPMJ  IS NOT NULL AND F03B31.ZGUPMJ <> 0 THEN
30 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGUPMJ +1900000,'YYYYDDD'), 'YYYYMMDD'))
31 WHEN F03B31.ZGUPMJ = 0 THEN NULL
32 ELSE NULL
33 END,'YYYYMMDD')          AS LAST_UPDATE_DATE,
34              CASE WHEN F03B31.ZGUPMJ  IS NOT NULL AND F03B31.ZGUPMJ <> 0 THEN
35 TO_NUMBER(TO_CHAR(TO_DATE(F03B31.ZGUPMJ +1900000,'YYYYDDD'), 'YYYYMMDD'))
36 WHEN F03B31.ZGUPMJ = 0 THEN NULL
37 ELSE NULL
38 END           AS LAST_UPDATE_DT,
```



Refresh Variable Issue



reusable mapping issue

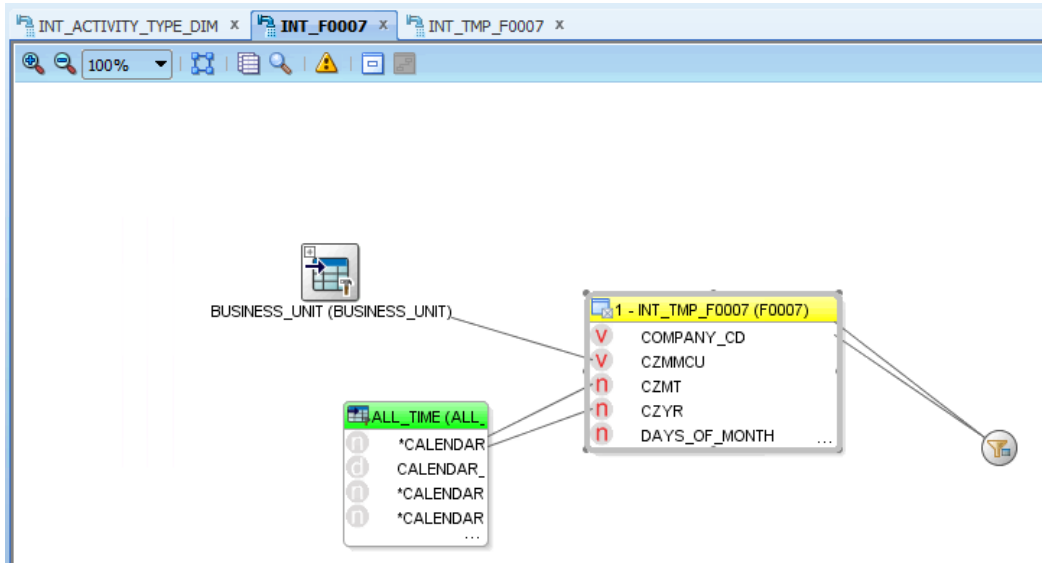


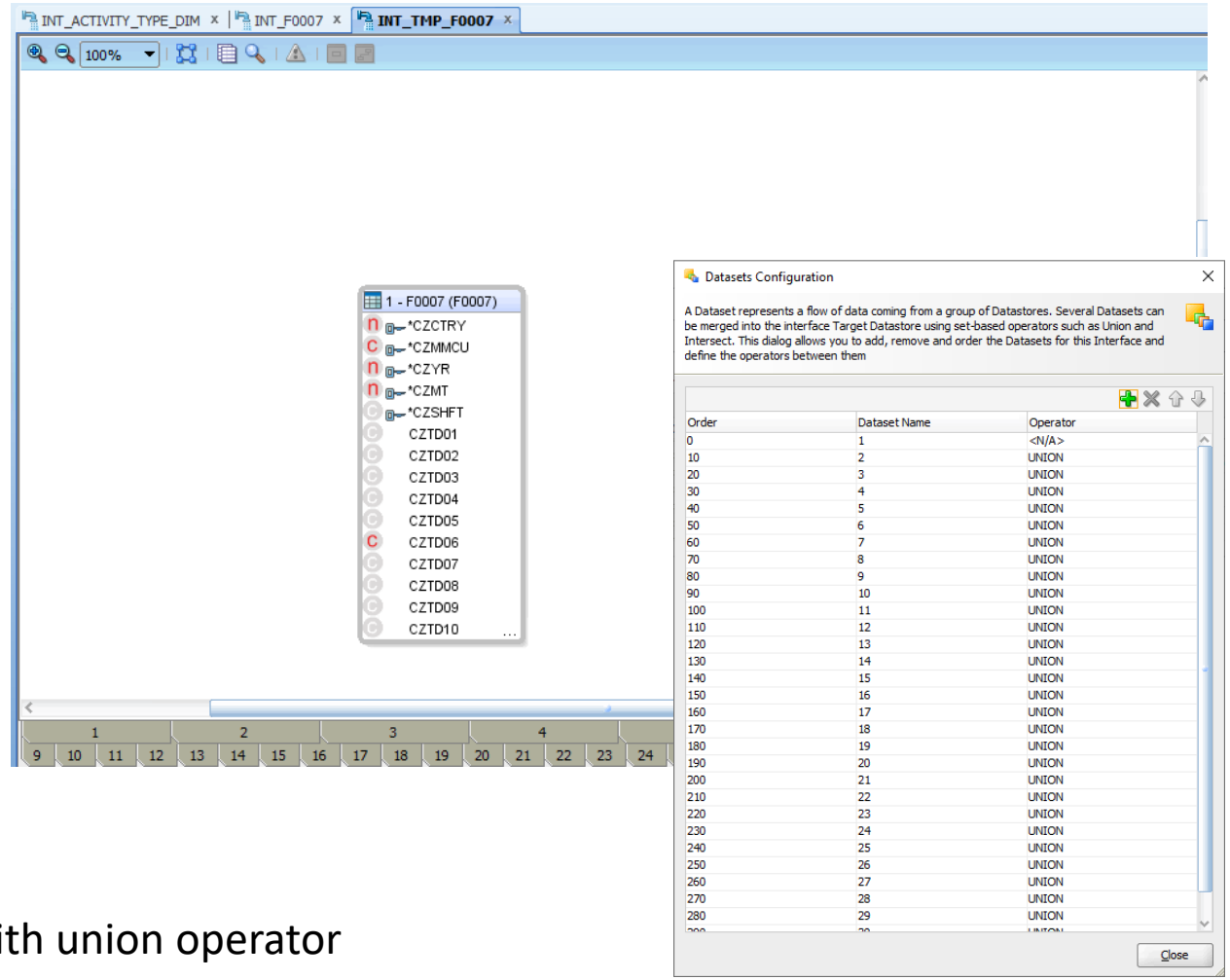
Error Message:

ODI-1228: Task Insert flow into I$ table-IKM Oracle Incremental Update Order By- fails on the target connection PXDEV_PXDVBMRDG.
Caused By: java.sql.SQLSyntaxErrorException: ORA-00904: "F00071"."CZMMCU": invalid identifier

reusable mapping issue

ODI 11g Interface for INT_TMP_F0007



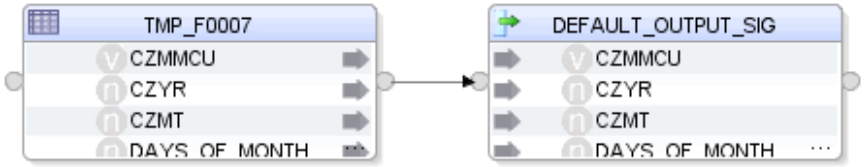


F0007 Interface with Multiple data sets and target data with union operator

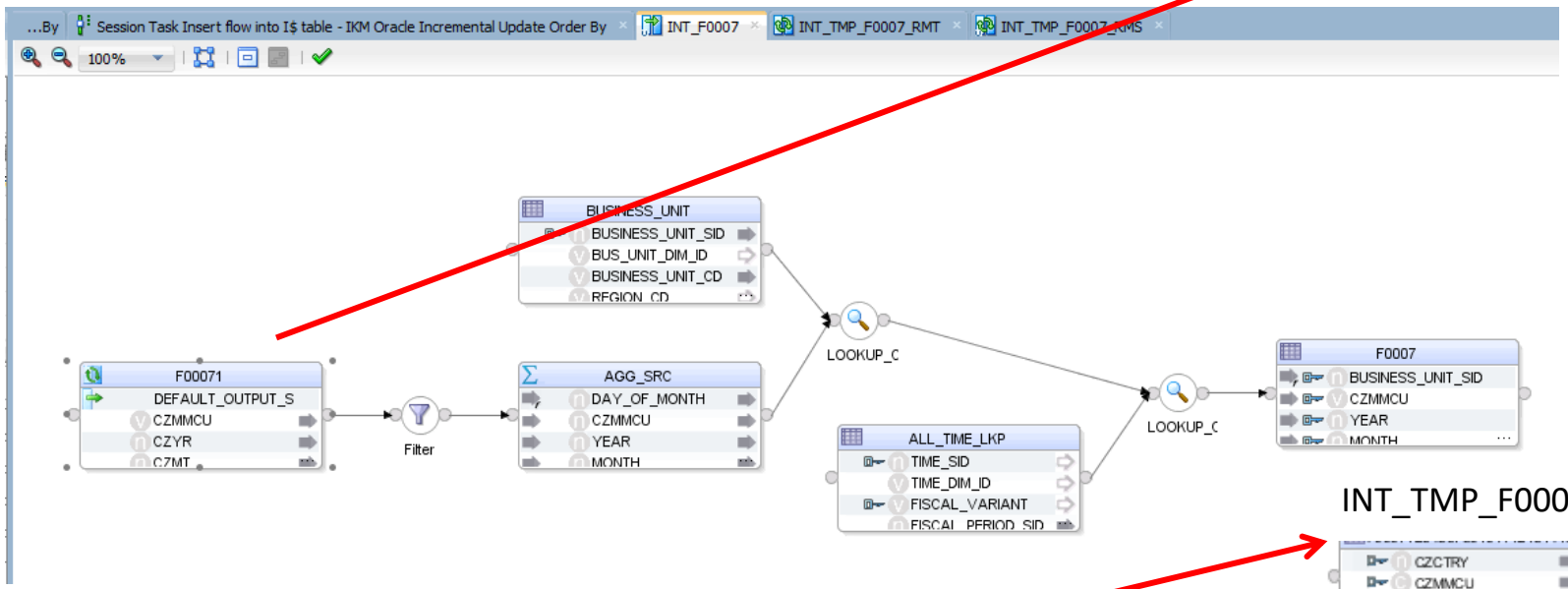
reusable mapping issue

ODI 12c Mapping

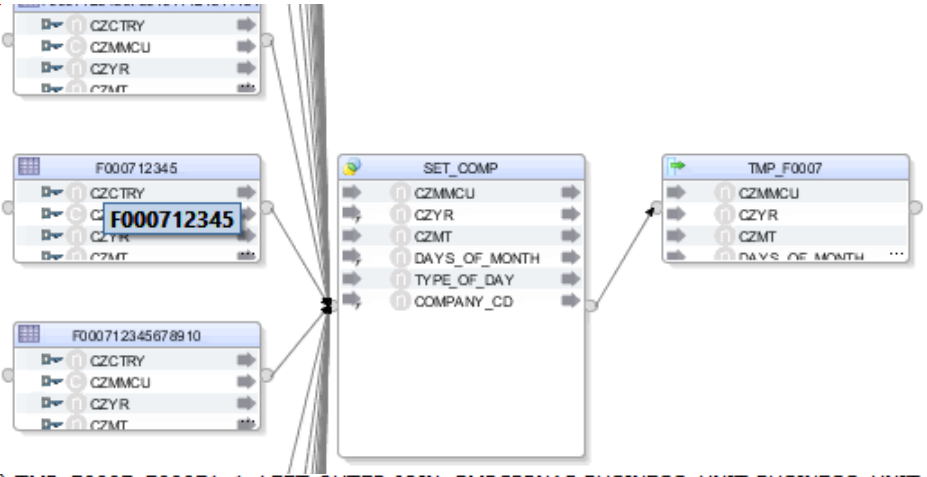
INT_TMP_F0007_RMT



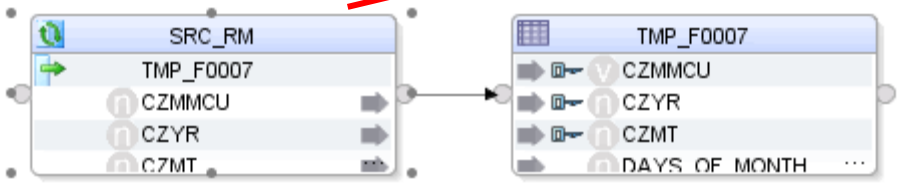
INT_F0007



INT_TMP_F0007_RMS



INT_TMP_F0007

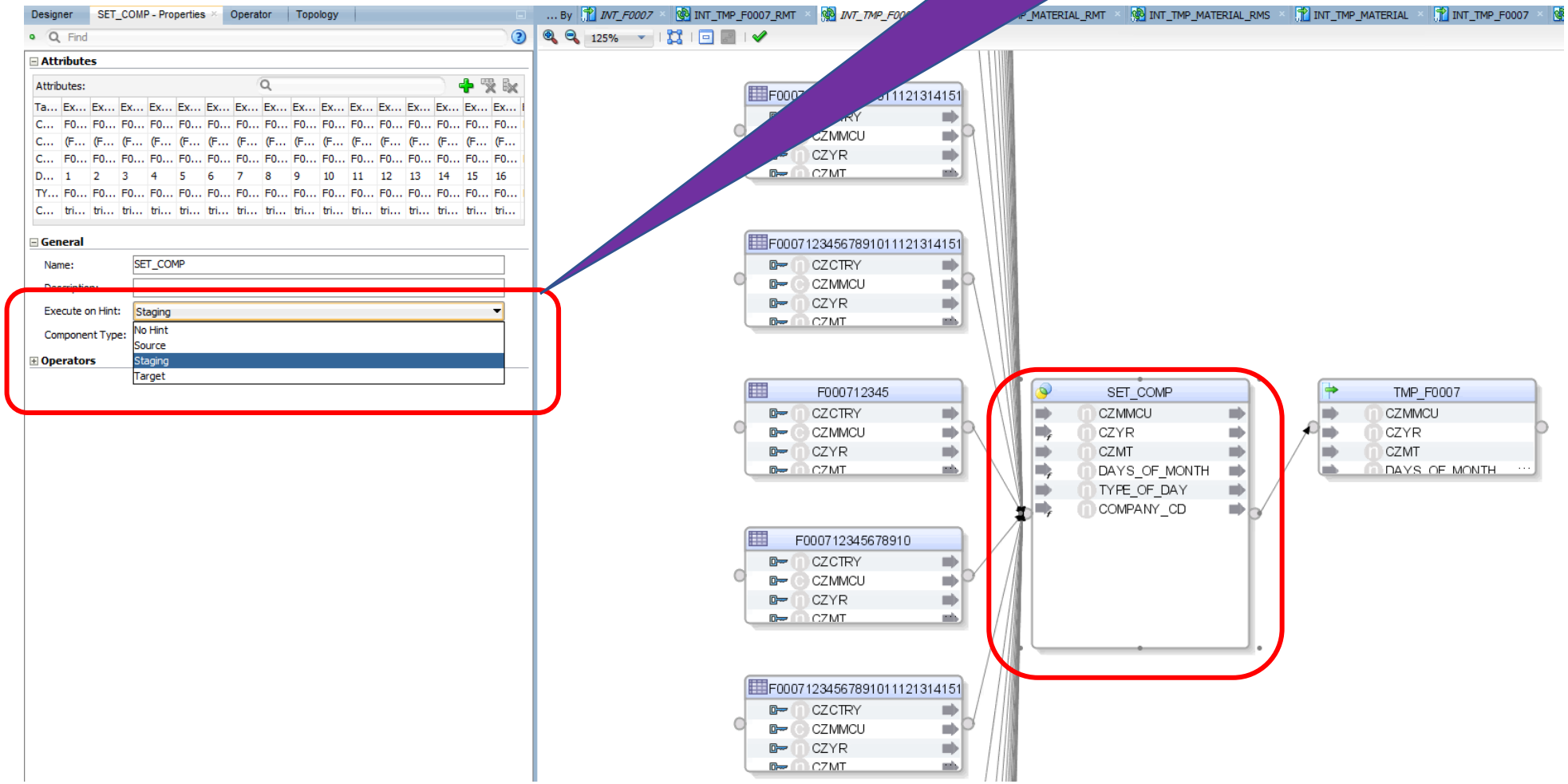



```
) TMP_F0007_F00071_1  LEFT OUTER JOIN  BMRCRPNA5.BUSINESS_UNIT BUSINESS_UNIT
  ON  TRIM(F00071.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)
   ) LEFT OUTER JOIN  BMRCRPNA5.ALL_TIME ALL_TIME_LKP
  ON  F00071.CZYR=ALL_TIME_LKP.CALENDAR_YEAR
AND F00071.CZMT=ALL_TIME_LKP.CALENDAR_MONTH
AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
```

```
52  ) TMP_F0007_F00071_1  LEFT OUTER JOIN  BMRCRPNA5.BUSINESS_UNIT BUSINESS_UNIT
53    ON  TRIM(TMP_F0007_F00071_1.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)
54     ) LEFT OUTER JOIN  BMRCRPNA5.ALL_TIME ALL_TIME_LKP
55    ON  TMP_F0007_F00071_1.YEAR=ALL_TIME_LKP.CALENDAR_YEAR
56  AND TMP_F0007_F00071_1.MONTH=ALL_TIME_LKP.CALENDAR_MONTH
57    AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
```

reusable mapping issue – Resolution

Change Execute on Hint from **Source** to **Staging**



reusable mapping issue

Comparison of ODI 11g and ODI 12c Code

```
/* DETECTION_STRATEGY = NOT_EXISTS */
insert /*+ append */ into BMRODISTGNA.I$_F0007
(
   BUSINESS_UNIT_SID,
   CZMMCU,
   YEAR,
   MONTH,
   DAY_OF_MONTH,
   TYPE_OF_DAY,
   FISCAL_PERIOD_SID,
   COMPANY_CD,
   IND_UPDATE
)
select
BUSINESS_UNIT_SID,
   CZMMCU,
   YEAR,
   MONTH,
   DAY_OF_MONTH,
   TYPE_OF_DAY,
   FISCAL_PERIOD_SID,
   COMPANY_CD,
   IND_UPDATE
 from (
select
   NVL(BUSINESS_UNIT.BUSINESS_UNIT_SID,0) BUSINESS_UNIT_SID,
   F0007.CZMMCU CZMMCU,
   F0007.CZYR YEAR,
   F0007.CZMT MONTH,
   COUNT(F0007.DAYS_OF_MONTH) DAY_OF_MONTH,
   F0007.TYPE_OF_DAY TYPE_OF_DAY,
   ALL_TIME_LKP.FISCAL_PERIOD_SID FISCAL_PERIOD_SID,
   NVL(F0007.COMPANY_CD,' ') COMPANY_CD,

   'I' IND_UPDATE

from (BMRODISTGNA.TMP_F0007 F0007 LEFT OUTER JOIN BMRCRPNA6.BUSINESS_UNIT BUSINESS_UNIT ON TRIM(F0007.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)) LEFT OU

AND F0007.CZMT=ALL_TIME_LKP.CALENDAR_MONTH
AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
where   (1=1)

And (F0007.TYPE_OF_DAY='W'
AND
(length(trim(F0007.COMPANY_CD)) = 5 or trim(F0007.COMPANY_CD) is null))

Group By NVL(BUSINESS_UNIT.BUSINESS_UNIT_SID,0),
 F0007.CZMMCU,
 F0007.CZYR,
 F0007.CZMT,
 F0007.TYPE_OF_DAY,
 ALL_TIME_LKP.FISCAL_PERIOD_SID,
 NVL(F0007.COMPANY_CD, ' ')
) S
where NOT EXISTS
   ( select 1 from BMRCRPNA6.F0007 T
   where   T.BUSINESS_UNIT_SID = S.BUSINESS_UNIT_SID
   and   T.CZMMCU = S.CZMMCU
   and   T.YEAR   = S.YEAR
   and   T.MONTH   = S.MONTH
   and   T.COMPANY_CD   = S.COMPANY_CD
      and ((T.DAY_OF_MONTH = S.DAY_OF_MONTH) or (T.DAY_OF_MONTH IS NULL and S.DAY_OF_MONTH IS NULL)) and
      ((T.TYPE_OF_DAY = S.TYPE_OF_DAY) or (T.TYPE_OF_DAY IS NULL and S.TYPE_OF_DAY IS NULL)) and
      ((T.FISCAL_PERIOD_SID = S.FISCAL_PERIOD_SID) or (T.FISCAL_PERIOD_SID IS NULL and S.FISCAL_PERIOD_SID IS NULL))
      )
```

```
1  /* DETECTION_STRATEGY = NOT_EXISTS */
2  insert /*+ append */ into BMRODISTGNA.I$_F0007
3  (
4     BUSINESS_UNIT_SID,
5     CZMMCU,
6     YEAR,
7     MONTH,
8     DAY_OF_MONTH,
9     TYPE_OF_DAY,
10    FISCAL_PERIOD_SID,
11    COMPANY_CD,
12    IND_UPDATE
13 )
14 select
15 BUSINESS_UNIT_SID,
16    CZMMCU,
17    YEAR,
18    MONTH,
19    DAY_OF_MONTH,
20    TYPE_OF_DAY,
21    FISCAL_PERIOD_SID,
22    COMPANY_CD,
23    IND_UPDATE
24  from (
25 select
26    NVL(BUSINESS_UNIT.BUSINESS_UNIT_SID,0) BUSINESS_UNIT_SID,
27    TMP_F0007_F00071_1.CZMMCU CZMMCU,
28    TMP_F0007_F00071_1.YEAR YEAR,
29    TMP_F0007_F00071_1.MONTH MONTH,
30    TMP_F0007_F00071_1.DAY_OF_MONTH DAY_OF_MONTH,
31    TMP_F0007_F00071_1.TYPE_OF_DAY TYPE_OF_DAY,
32    ALL_TIME_LKP.FISCAL_PERIOD_SID FISCAL_PERIOD_SID,
33    TMP_F0007_F00071_1.COMPANY_CD COMPANY_CD,
34
35    'I' IND_UPDATE
36
37 from (
38 SELECT
39    COUNT(TMP_F0007_F00071.DAYS_OF_MONTH) AS DAY_OF_MONTH ,
40    TMP_F0007_F00071.CZMMCU AS CZMMCU ,
41    TMP_F0007_F00071.CZYR AS YEAR ,
42    TMP_F0007_F00071.CZMT AS MONTH ,
43    TMP_F0007_F00071.TYPE_OF_DAY AS TYPE_OF_DAY ,
44    NVL(TMP_F0007_F00071.COMPANY_CD, ' ') AS COMPANY_CD
45 FROM
46    BMRODISTGNA.TMP_F0007 TMP_F0007_F00071
47 WHERE
48    (TMP_F0007_F00071.TYPE_OF_DAY='W'
49 AND
50 (length(trim(TMP_F0007_F00071.COMPANY_CD)) = 5 or trim(TMP_F0007_F00071.COMPANY_CD) is null)
51 )
52 GROUP BY
53    TMP_F0007_F00071.CZMMCU, TMP_F0007_F00071.CZYR, TMP_F0007_F00071.CZMT, TMP_F0007_F00071.TYPE_OF_DAY, NVL(TMP_F0007_F00071.COMPANY_CD, ' ')
54    ) TMP_F0007_F00071_1  LEFT OUTER JOIN  BMRCRPNA5.BUSINESS_UNIT BUSINESS_UNIT
55      ON  TRIM(F00071.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)
56       )  LEFT OUTER JOIN  BMRCRPNA5.ALL_TIME ALL_TIME_LKP
57      ON  F00071.CZYR=ALL_TIME_LKP.CALENDAR_YEAR
58 AND F00071.CZMT=ALL_TIME_LKP.CALENDAR_MONTH
59 AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
60 where   (1=1)
61
62
63
64
65
66
67
68
69
70
71
72
73 ) S
74 where NOT EXISTS
75    ( select 1 from BMRCRPNA5.F0007 T
76    where   T.BUSINESS_UNIT_SID = S.BUSINESS_UNIT_SID
77    and   T.CZMMCU = S.CZMMCU
78    and   T.YEAR   = S.YEAR
79    and   T.MONTH   = S.MONTH
80    and   T.COMPANY_CD   = S.COMPANY_CD
81       and ((T.DAY_OF_MONTH = S.DAY_OF_MONTH) or (T.DAY_OF_MONTH IS NULL and S.DAY_OF_MONTH IS NULL)) and
82       ((T.TYPE_OF_DAY = S.TYPE_OF_DAY) or (T.TYPE_OF_DAY IS NULL and S.TYPE_OF_DAY IS NULL)) and
83       ((T.FISCAL_PERIOD_SID = S.FISCAL_PERIOD_SID) or (T.FISCAL_PERIOD_SID IS NULL and S.FISCAL_PERIOD_SID IS NULL))
84       )
85
```

reusable mapping issue

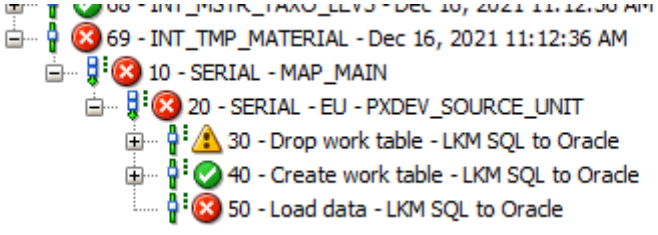
Comparison of ODI 12c Failed and ODI 12c Fixed Code

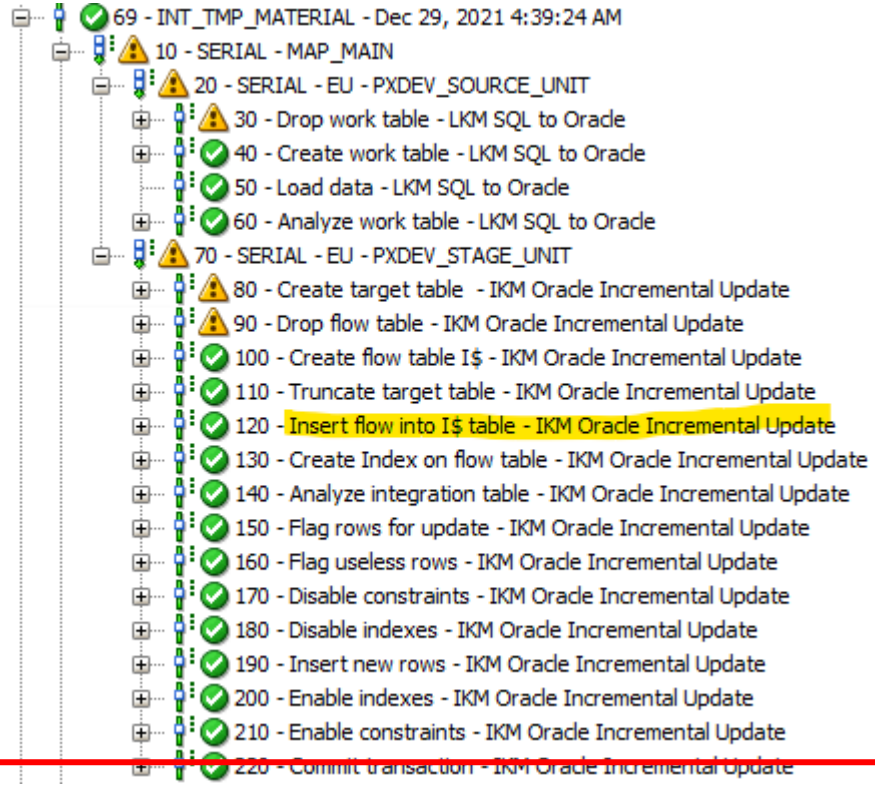
```
/* DETECTION_STRATEGY = NOT_EXISTS */
insert /*+ append */ into BMRODISTGNA.I$_F0007
(
   BUSINESS_UNIT_SID,
   CZMMCU,
   YEAR,
   MONTH,
   DAY_OF_MONTH,
   TYPE_OF_DAY,
   FISCAL_PERIOD_SID,
   COMPANY_CD,
   IND_UPDATE
)
select
BUSINESS_UNIT_SID,
   CZMMCU,
   YEAR,
   MONTH,
   DAY_OF_MONTH,
   TYPE_OF_DAY,
   FISCAL_PERIOD_SID,
   COMPANY_CD,
   IND_UPDATE
 from (
select
   NVL(BUSINESS_UNIT.BUSINESS_UNIT_SID,0) BUSINESS_UNIT_SID,
   TMP_F0007_F00071_1.CZMMCU CZMMCU,
   TMP_F0007_F00071_1.YEAR YEAR,
   TMP_F0007_F00071_1.MONTH MONTH,
   TMP_F0007_F00071_1.DAY_OF_MONTH DAY_OF_MONTH,
   TMP_F0007_F00071_1.TYPE_OF_DAY TYPE_OF_DAY,
   ALL_TIME_LKP.FISCAL_PERIOD_SID FISCAL_PERIOD_SID,
   TMP_F0007_F00071_1.COMPANY_CD COMPANY_CD,
   'I' IND_UPDATE
from ((
SELECT
  COUNT(TMP_F0007_F00071.DAYS_OF_MONTH) AS DAY_OF_MONTH ,
  TMP_F0007_F00071.CZMMCU AS CZMMCU ,
  TMP_F0007_F00071.CZYR AS YEAR ,
  TMP_F0007_F00071.CZMT AS MONTH ,
  TMP_F0007_F00071.TYPE_OF_DAY AS TYPE_OF_DAY ,
  NVL(TMP_F0007_F00071.COMPANY_CD, ' ') AS COMPANY_CD
FROM
  BMRODISTGNA.TMP_F0007 TMP_F0007_F00071
WHERE
  (TMP_F0007_F00071.TYPE_OF_DAY='W'
AND
(length(trim(TMP_F0007_F00071.COMPANY_CD)) = 5 or trim(TMP_F0007_F00071.COMPANY_CD) is null)
)
GROUP BY
  TMP_F0007_F00071.CZMMCU, TMP_F0007_F00071.CZYR, TMP_F0007_F00071.CZMT, TMP_F0007_F00071.TYPE_OF_DAY, NVL(TMP_F0007_F00071.COMPANY_CD, ' ')
  ) TMP_F0007_F00071_1  LEFT OUTER JOIN  BMRCRPNA5.BUSINESS_UNIT BUSINESS_UNIT
    ON  TRIM(F00071.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)
     ) LEFT OUTER JOIN  BMRCRPNA5.ALL_TIME ALL_TIME_LKP
    ON  F00071.CZYR=ALL_TIME_LKP.CALENDAR_YEAR
AND F00071.CZMT=ALL_TIME_LKP.CALENDAR_MONTH
AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
where   (1=1)
) F
where NOT EXISTS
   ( select 1 from BMRCRPNA5.F0007 T
   where   T.BUSINESS_UNIT_SID = S.BUSINESS_UNIT_SID
   and   T.CZMMCU = S.CZMMCU
   and   T.YEAR   = S.YEAR
   and   T.MONTH   = S.MONTH
   and   T.COMPANY_CD   = S.COMPANY_CD
       and ((T.DAY_OF_MONTH = S.DAY_OF_MONTH) or (T.DAY_OF_MONTH IS NULL and S.DAY_OF_MONTH IS NULL)) and
      ((T.TYPE_OF_DAY = S.TYPE_OF_DAY) or (T.TYPE_OF_DAY IS NULL and S.TYPE_OF_DAY IS NULL)) and
      ((T.FISCAL_PERIOD_SID = S.FISCAL_PERIOD_SID) or (T.FISCAL_PERIOD_SID IS NULL and S.FISCAL_PERIOD_SID IS NULL))
      )
```

```
1  /* DETECTION_STRATEGY = NOT_EXISTS */
2  insert /*+ append */ into BMRODISTGNA.I$_F0007
3  (
4     BUSINESS_UNIT_SID,
5     CZMMCU,
6     YEAR,
7     MONTH,
8     DAY_OF_MONTH,
9     TYPE_OF_DAY,
10    FISCAL_PERIOD_SID,
11    COMPANY_CD,
12    IND_UPDATE
13 )
14 select
15 BUSINESS_UNIT_SID,
16    CZMMCU,
17    YEAR,
18    MONTH,
19    DAY_OF_MONTH,
20    TYPE_OF_DAY,
21    FISCAL_PERIOD_SID,
22    COMPANY_CD,
23    IND_UPDATE
24  from (
25 select
26    NVL(BUSINESS_UNIT.BUSINESS_UNIT_SID,0) BUSINESS_UNIT_SID,
27    TMP_F0007_F00071_1.CZMMCU CZMMCU,
28    TMP_F0007_F00071_1.YEAR YEAR,
29    TMP_F0007_F00071_1.MONTH MONTH,
30    TMP_F0007_F00071_1.DAY_OF_MONTH DAY_OF_MONTH,
31    TMP_F0007_F00071_1.TYPE_OF_DAY TYPE_OF_DAY,
32    ALL_TIME_LKP.FISCAL_PERIOD_SID FISCAL_PERIOD_SID,
33    TMP_F0007_F00071_1.COMPANY_CD COMPANY_CD,
34    'I' IND_UPDATE
35 from ((
36 SELECT
37   COUNT(TMP_F0007_F00071.DAYS_OF_MONTH) AS DAY_OF_MONTH ,
38   TMP_F0007_F00071.CZMMCU AS CZMMCU ,
39   TMP_F0007_F00071.CZYR AS YEAR ,
40   TMP_F0007_F00071.CZMT AS MONTH ,
41   TMP_F0007_F00071.TYPE_OF_DAY AS TYPE_OF_DAY ,
42   NVL(TMP_F0007_F00071.COMPANY_CD, ' ') AS COMPANY_CD
43 FROM
44   BMRODISTGNA.TMP_F0007 TMP_F0007_F00071
45 WHERE
46   (TMP_F0007_F00071.TYPE_OF_DAY='W'
47 AND
48 (length(trim(TMP_F0007_F00071.COMPANY_CD)) = 5 or trim(TMP_F0007_F00071.COMPANY_CD) is null)
49 )
50 GROUP BY
51   TMP_F0007_F00071.CZMMCU, TMP_F0007_F00071.CZYR, TMP_F0007_F00071.CZMT, TMP_F0007_F00071.TYPE_OF_DAY, NVL(TMP_F0007_F00071.COMPANY_CD, ' ')
52   ) TMP_F0007_F00071_1  LEFT OUTER JOIN  BMRCRPNA5.BUSINESS_UNIT BUSINESS_UNIT
53     ON  TRIM(TMP_F0007_F00071_1.CZMMCU)=TRIM(BUSINESS_UNIT.BUSINESS_UNIT_CD)
54      ) LEFT OUTER JOIN  BMRCRPNA5.ALL_TIME ALL_TIME_LKP
55     ON  TMP_F0007_F00071_1.YEAR=ALL_TIME_LKP.CALENDAR_YEAR
56 AND TMP_F0007_F00071_1.MONTH=ALL_TIME_LKP.CALENDAR_MONTH
57 AND ALL_TIME_LKP.FISCAL_PERIOD_SID<>-1
58 where   (1=1)
59 ) F
60 where NOT EXISTS
61    ( select 1 from BMRCRPNA5.F0007 T
62    where   T.BUSINESS_UNIT_SID = S.BUSINESS_UNIT_SID
63    and   T.CZMMCU = S.CZMMCU
64    and   T.YEAR   = S.YEAR
65    and   T.MONTH   = S.MONTH
66    and   T.COMPANY_CD   = S.COMPANY_CD
67        and ((T.DAY_OF_MONTH = S.DAY_OF_MONTH) or (T.DAY_OF_MONTH IS NULL and S.DAY_OF_MONTH IS NULL)) and
68       ((T.TYPE_OF_DAY = S.TYPE_OF_DAY) or (T.TYPE_OF_DAY IS NULL and S.TYPE_OF_DAY IS NULL)) and
69       ((T.FISCAL_PERIOD_SID = S.FISCAL_PERIOD_SID) or (T.FISCAL_PERIOD_SID IS NULL and S.FISCAL_PERIOD_SID IS NULL))
70       )
71
```

Variable Calling issue

PKG_MATERIAL : ODI12c Failed vs Fixed Code






```
nvl(trim(#F4101.MTRL_CATEGORY1_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT1_CD,
nvl(trim(#F4101.MTRL_CATEGORY2_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT2_CD,
nvl(trim(#F4101.MTRL_CATEGORY3_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT3_CD,
nvl(trim(#F4101.MTRL_CATEGORY4_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT4_CD,
nvl(trim(#F4101.MTRL_CATEGORY5_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT5_CD,
nvl(trim(#F4101.MTRL_CATEGORY6_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT6_CD,
nvl(trim(#F4101.MTRL_CATEGORY7_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT7_CD,
nvl(trim(#F4101.MTRL_CATEGORY8_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT8_CD,
nvl(trim(#F4101.MTRL_CATEGORY9_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR')  AS MATERIAL_CAT9_CD,
nvl(trim(#F4101.MTRL_CATEGORY10_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT10_CD,
nvl(trim(#F4101.MTRL_CATEGORY11_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT11_CD,
nvl(trim(#F4101.MTRL_CATEGORY12_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT12_CD,
nvl(trim(#F4101.MTRL_CATEGORY13_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT13_CD,
nvl(trim(#F4101.MTRL_CATEGORY14_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT14_CD,
nvl(trim(#F4101.MTRL_CATEGORY15_CD),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT15_CD,
```

```
119  nvl(trim(DEFAULT_A.IMPRP1),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT1_CD ,
120  nvl(trim(DEFAULT_A.IMPRP2),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT2_CD ,
121  nvl(trim(DEFAULT_A.IMPRP3),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT3_CD ,
122  nvl(trim(DEFAULT_A.IMPRP4),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT4_CD ,
123  nvl(trim(DEFAULT_A.IMPRP5),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT5_CD ,
124  nvl(trim(DEFAULT_A.IMPRP6),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT6_CD ,
125  nvl(trim(DEFAULT_A.IMPRP7),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT7_CD ,
126  nvl(trim(DEFAULT_A.IMSRP1),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT8_CD ,
127  nvl(trim(DEFAULT_A.IMSRP2),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT9_CD ,
128  nvl(trim(DEFAULT_A.IMSRP3),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT10_CD ,
129  nvl(trim(DEFAULT_A.IMSRP4),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT11_CD ,
130  nvl(trim(DEFAULT_A.IMSRP5),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT12_CD ,
131  nvl(trim(DEFAULT_A.IMSRP6),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT13_CD ,
132  nvl(trim(DEFAULT_A.IMSRP7),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT14_CD ,
133  nvl(trim(DEFAULT_A.IMSRP8),'#PRAXAIR_BI.U_DEFAULT_SHORT_CHAR') AS MATERIAL_CAT15_CD
```

Thank you
Q & A