ADV03 Not exists
In the filter of a mapping

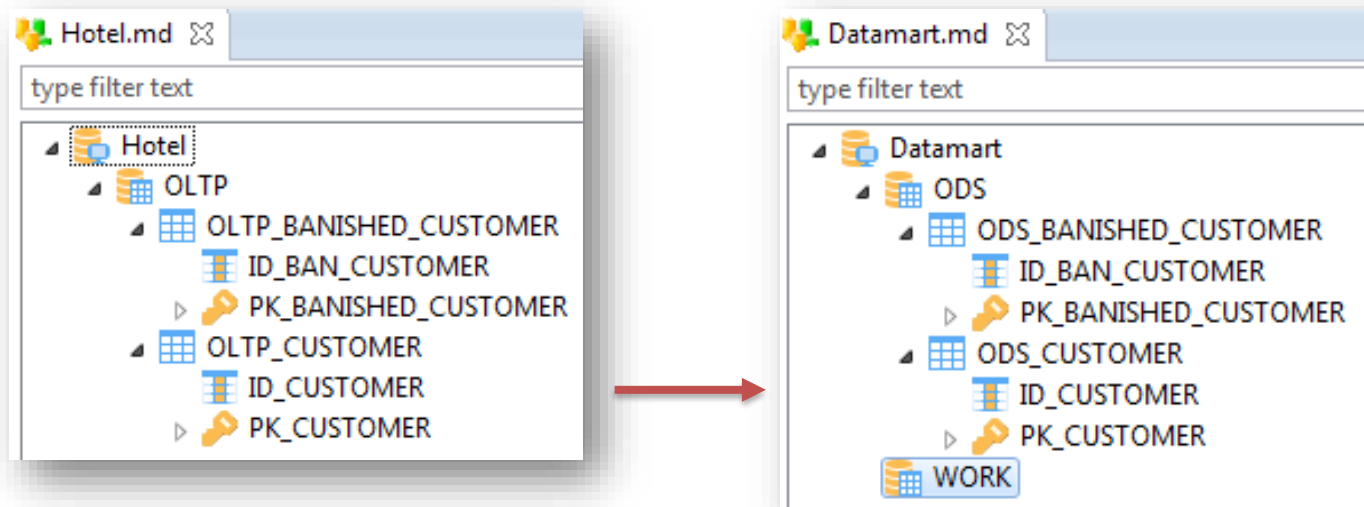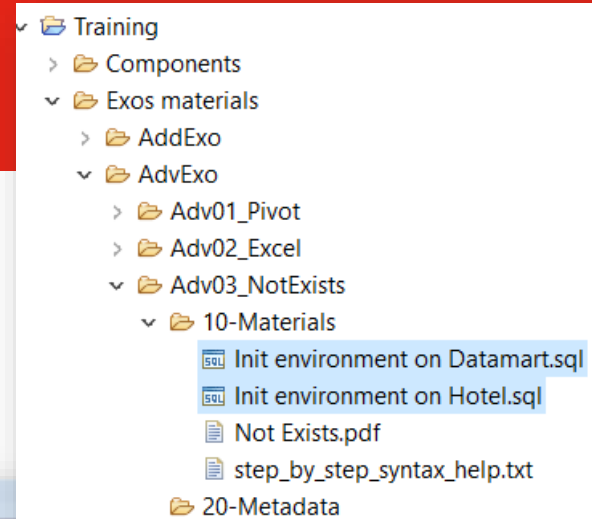A related article in Stambia.org
- https://stambia.org/doc/220-development-hints-and-tips/exists-not-exists/538-exists-not-exist-in-mappings

❖ Context

- The objective of this exercise is to integrate "customer_id" from an OLTP database (source) to an ODS one (target)

- Some customers are banished and are listed in banished tables

- Those banished customers must not be integrated in ODS database

- We are going to study the case where the banished table is in the OLTP database and the case where the banished table is in the ODS database

NEED HELP? Training/Exos materials/AdvExo/Adv03_.../step_by_step_syntax_help.txt

- **1 - Execute the 2 SQL files**
  - On the source (Hotel) & on the target (datamart)

```
Training
  Components
  Exos materials
    AddExo
    AdvExo
      Adv01_Pivot
      Adv02_Excel
      Adv03_NotExists
        10-Materials
          Init environment on Datamart.sql
          Init environment on Hotel.sql
        Not Exists.pdf
        step_by_step_syntax_help.txt
      20-Metadata
```

Hotel(6).sql

Hotel/sa   ☑ Limit Rows: 100

```sql
CREATE SCHEMA OLTP;
Execute current SQL. Current SQL is the selected text or the complete file content if nothing is selected.
CREATE TABLE OLTP.OLTP_BANISHED_CUSTOMER
(
    ID_BAN_CUSTOMER INTEGER NOT NULL
    , CONSTRAINT PK_BANISHED_CUSTOMER PRIMARY KEY (ID_BAN_CUSTOMER)
);
CREATE TABLE OLTP.OLTP_CUSTOMER
(
    ID_CUSTOMER INTEGER NOT NULL
    , CONSTRAINT PK_CUSTOMER PRIMARY KEY (ID_CUSTOMER)
);
INSERT INTO OLTP.OLTP_CUSTOMER
SELECT GEN_ID
FROM   UNNEST(SEQUENCE_ARRAY(1, 1000, 1)) AS GENERATE_SERIES(GEN_ID);
INSERT INTO OLTP.OLTP_BANISHED_CUSTOMER
SELECT GEN_ID
FROM   UNNEST(SEQUENCE_ARRAY(50, 199, 1)) AS GENERATE_SERIES(GEN_ID);
```
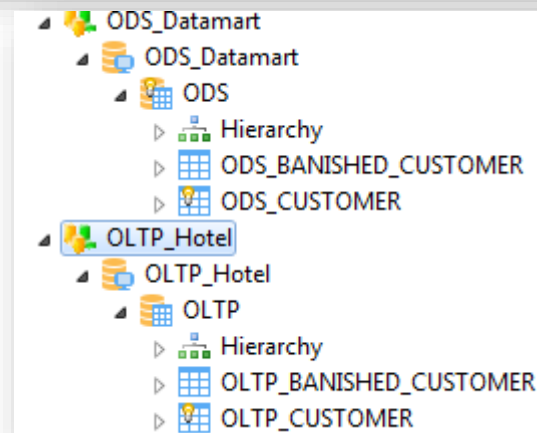
1
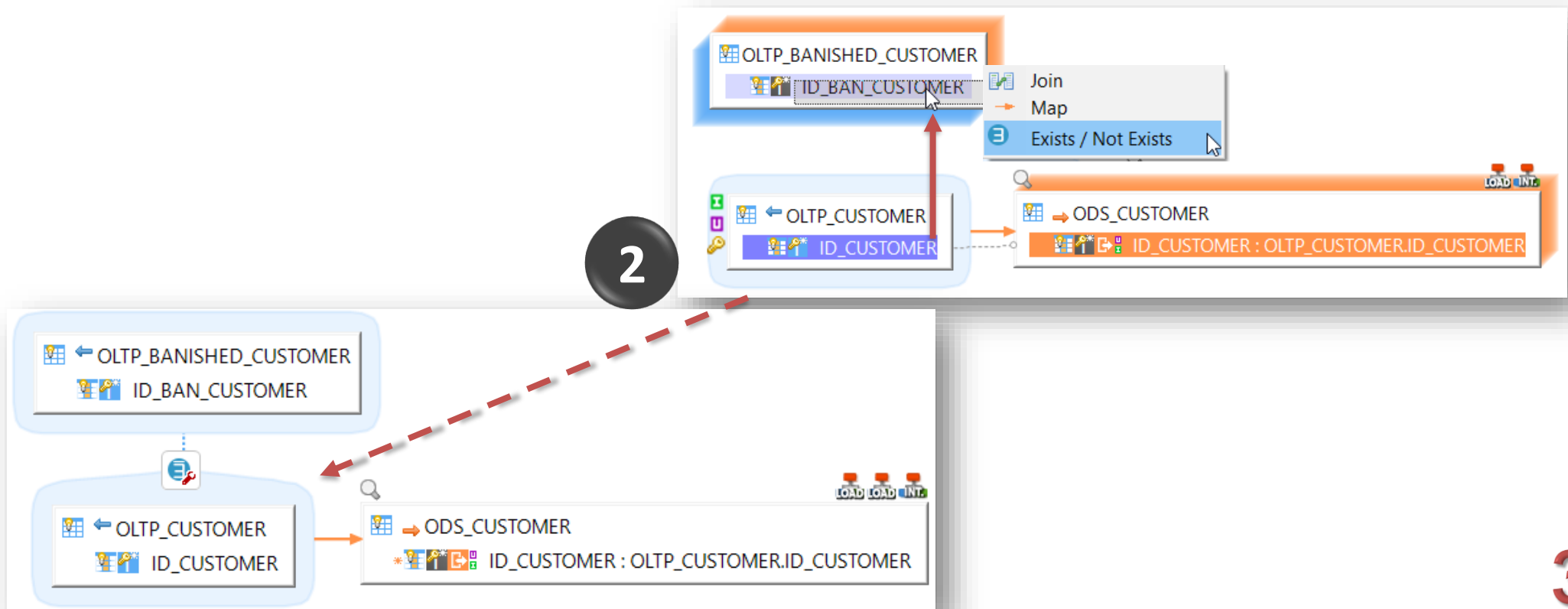
ODS(8).sql

Datamart/sa   ☑ Limit Rows: 100

```sql
CREATE SCHEMA ODS;
Execute current SQL. Current SQL is the selected text or the complete file content if nothing is selected.
CREATE SCHEMA WORK;
CREATE TABLE ODS.ODS_BANISHED_CUSTOMER
(
    ID_BAN_CUSTOMER INTEGER NOT NULL
    , CONSTRAINT PK_BANISHED_CUSTOMER PRIMARY KEY (ID_BAN_CUSTOMER)
);
CREATE TABLE ODS.ODS_CUSTOMER
(
    ID_CUSTOMER INTEGER NOT NULL
    , CONSTRAINT PK_CUSTOMER PRIMARY KEY (ID_CUSTOMER)
);

INSERT INTO ODS.ODS_BANISHED_CUSTOMER
SELECT GEN_ID
FROM   UNNEST(SEQUENCE_ARRAY(50, 99, 1)) AS GENERATE_SERIES(GEN_ID);
```
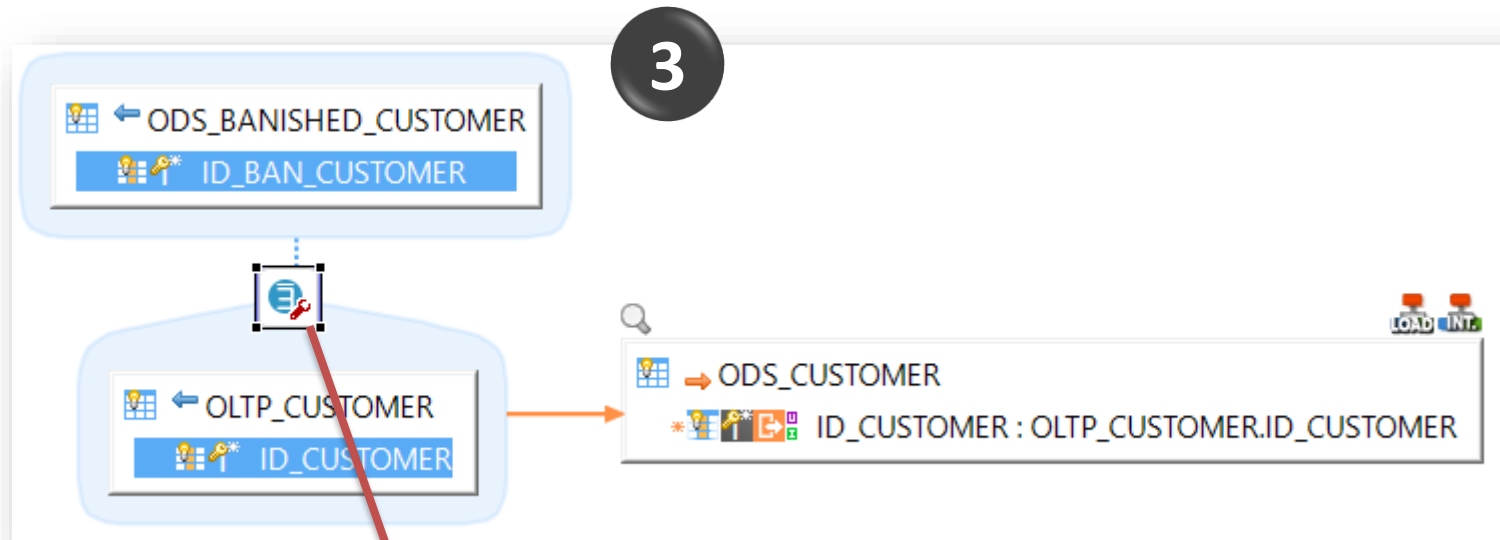
- **Reverse the schema and the tables**

```
ODS_Datamart
  ODS_Datamart
    ODS
      Hierarchy
      ODS_BANISHED_CUSTOMER
      ODS_CUSTOMER
OLTP_Hotel
  OLTP_Hotel
    OLTP
      Hierarchy
      OLTP_BANISHED_CUSTOMER
      OLTP_CUSTOMER
```

2

❖ 2 - Create a mapping in which we will use the source banished table

■ Drag & drop source and target customer tables

■ Add banished source customer table

■ Drag & drop ID_CUSTOMER from OLTP_CUSTOMER table on ID_CUSTOMER of the banished table and choose Exists/Not exists

3

❖ 3 – Do the same on the target tables :

Results

❖ First mapping

- ▪ On source

  - • OLTP_CUSTOMER   ID = [1:1000]
  - • OLTP_BANISHED_CUSTOMER   ID = [50:199]

- ▪ On target

  - • ODS_CUSTOMER   ID = [1:49-200:1000]

❖ Second mapping

- ▪ On source

  - • OLTP CUSTOMER   ID = [1:1000]

- ▪ On target

  - • ODS_BANISHED_CUSTOMER   ID = [50:99]
  - • ODS_CUSTOMER   ID = [1:49-100:1000]