

ADV04 Conditioned Bind Link

- ❖ The Objective is to build a process that will generate a JSON or a XML file with two bind links, depending on the SAL_GEN_FILE_TYPE of a new T_SALES_PERSON table
- Use of two mappings, one for JSON and the other for XML generation
- Creation of e a T_SALES_PERSON table and update of the T_CUSTOMER table to know which sales person own each customer

Table structure for T_SALES_PERSON:

- SAL_ID
- SAL_GENDER
- SAL_LASTNAME
- SAL_FIRSTNAME
- SAL_GEN_FILE_TYPE
- PK_T_SALES_PERSON

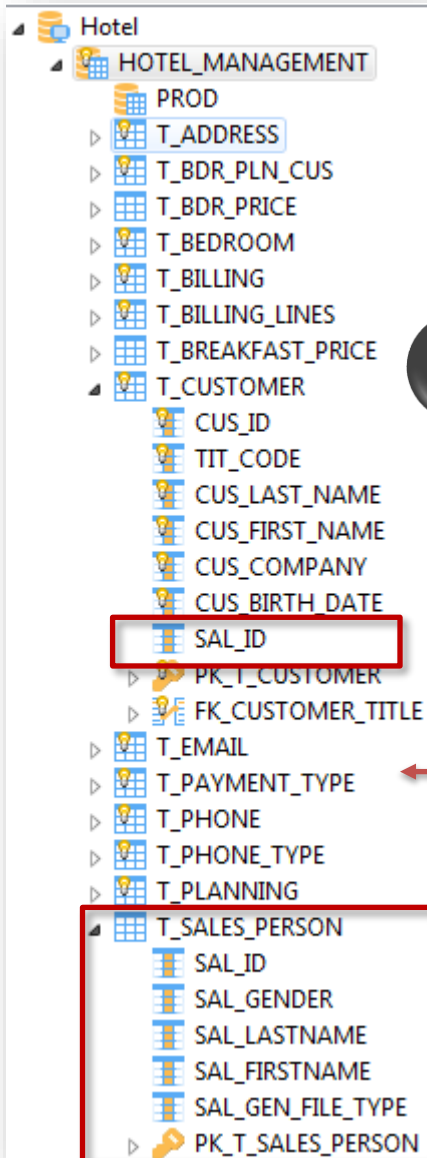
SAL_ID	SAL_GENDER	SAL_LASTNAME	SAL_FIRSTNAME	SAL_GEN_FILE_TYPE
1	M	SCHMIDT	Alexander	XML
2	M	JONES	Hales	XML
3	F	ARKET	Rosa	JSON

Table structure for T_CUSTOMER:

- CUS_ID
- TIT_CODE
- CUS_LAST_NAME
- CUS_FIRST_NAME
- CUS_COMPANY
- CUS_BIRTH_DATE
- SAL_ID
- PK_T_CUSTOMER
- FK_CUSTOMER_TITLE

CUS_ID	TIT_CODE	CUS_LAST_NAME	CUS_FIRST_NAME	CUS_COMPANY	CUS_BIRTH_DATE	SAL_ID
1	Mr	GIBBS	Jason		<null>	2
2	Mr	O'NEAL	Michael	Springfield Ltd	<null>	3
3	Mr	JIMENEZ	Tony		<null>	1
4	Mr	HAMPTON	Lionel		<null>	2
5	Mr	SILL	Joel		<null>	3
6	Mr	CARTER	Rick	Buick	<null>	1
7	Mr	BARTZ	Gary		<null>	2

- ❖ 1 - Execute the SQL file (Init Environment on Hotel.sql)
 - Reverse T_SALES_PERSON and T_CUSTOMER tables



1

Hotel/sa ☒ Limit Rows: 100

```

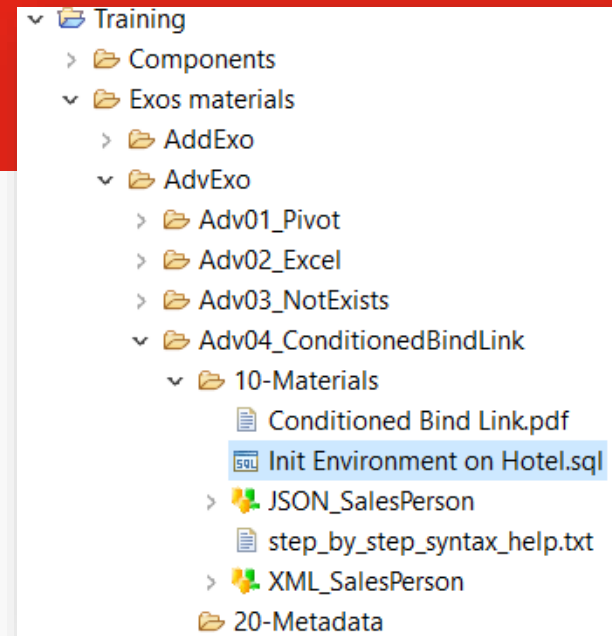
CREATE TABLE HOTEL_MANAGEMENT.T_SALES_PERSON
(
    SAL_ID INTEGER NOT NULL,
    SAL_GENDER VARCHAR(8) ,
    SAL_LASTNAME VARCHAR(32) NOT NULL,
    SAL_FIRSTNAME VARCHAR(25),
    SAL_GEN_FILE_TYPE VARCHAR(4),
    CONSTRAINT PK_T_SALES_PERSON PRIMARY KEY (SAL_ID));

INSERT INTO HOTEL_MANAGEMENT.T_SALES_PERSON VALUES (1,'M','SCHMIDT','Alexander','XML');
INSERT INTO HOTEL_MANAGEMENT.T_SALES_PERSON VALUES (2,'M','JONES','Hales','XML');
INSERT INTO HOTEL_MANAGEMENT.T_SALES_PERSON VALUES (3,'F','ARKET','Rosa','JSON');

ALTER TABLE HOTEL_MANAGEMENT.T_CUSTOMER ADD COLUMN SAL_ID INTEGER;

UPDATE HOTEL_MANAGEMENT.T_CUSTOMER SET SAL_ID=MOD(CUS_ID,3)+1;
    
```

Execute current SQL. Current SQL is the selected text or the complete file content if nothing is selected.



- ❖ 2 - Use the two XML and JSON files in the two mappings specified in next slides

XML_SalesPerson.md

type filter text

- salesPerson.xsd
 - salesPerson
 - Sequence
 - salesPerson
 - salesGender
 - salesId
 - salesName
 - Sequence
 - customer
 - customerId
 - customerName
 - customerTitleCode

sales=http://stambia.org/samples/salesPerson
xs=http://www.w3.org/2001/XMLSchema

JSON_SalesPerson.md

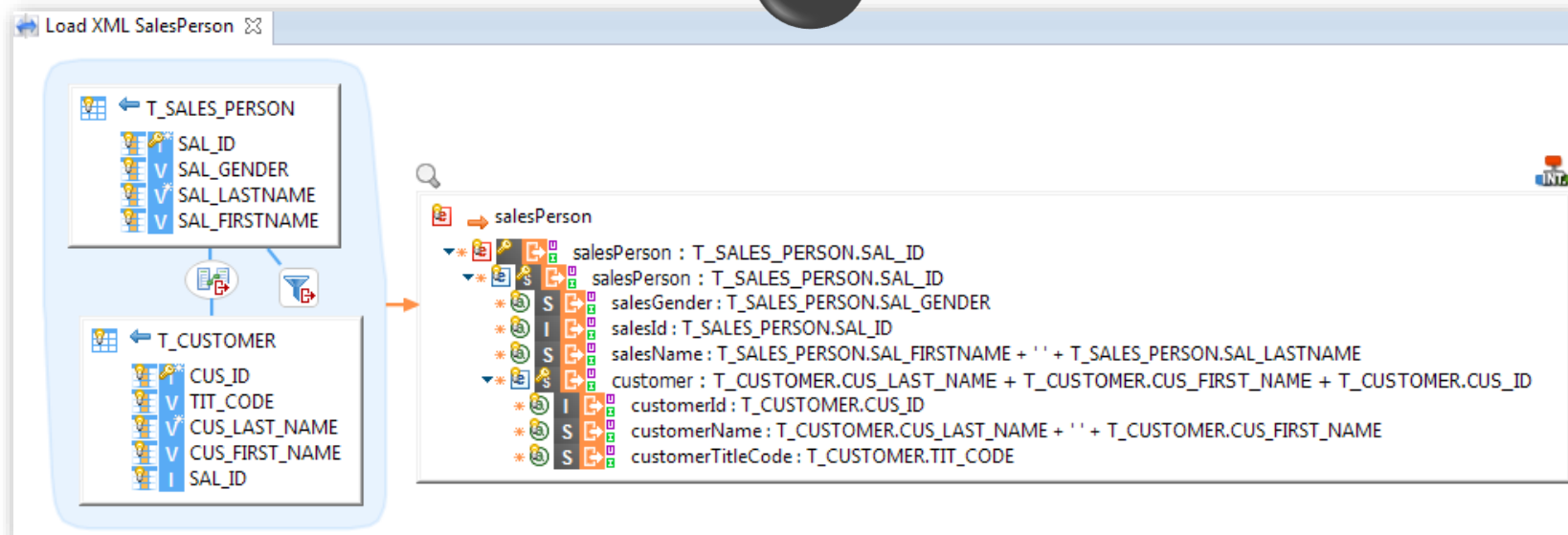
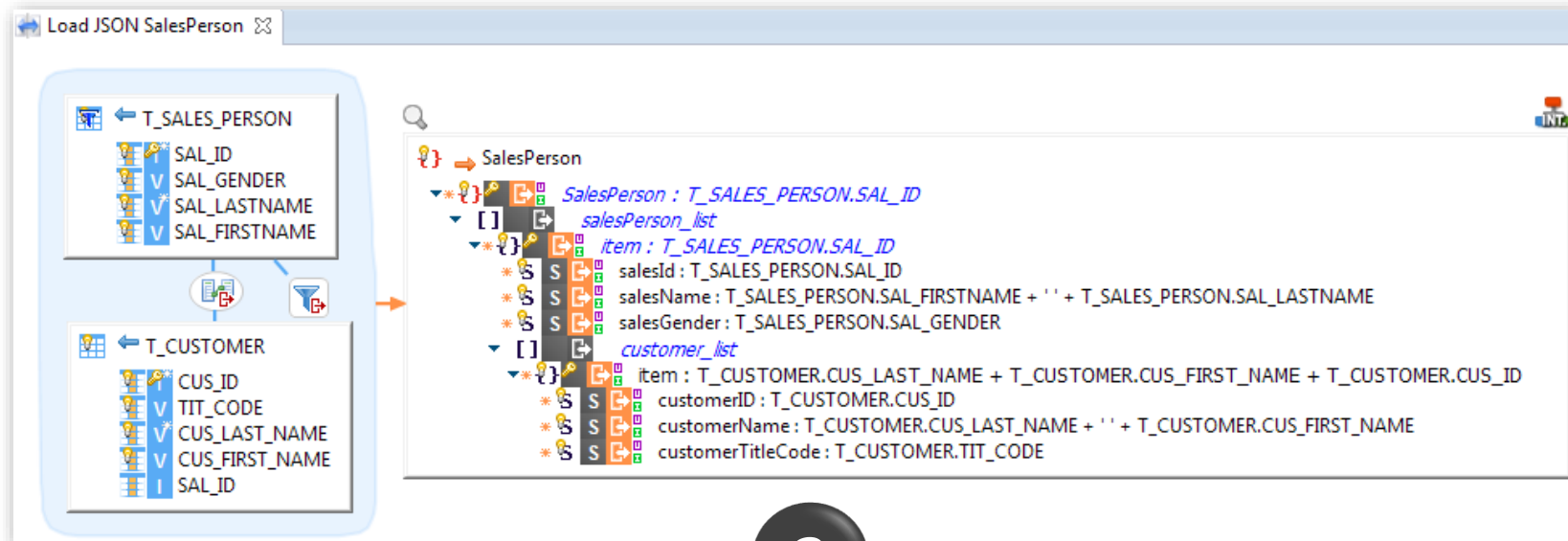
type filter text

- Schema
 - SalesPerson
 - salesPerson_list
 - item
 - salesId
 - salesName
 - salesGender
 - customer_list
 - item
 - customerID
 - customerName
 - customerTitleCode

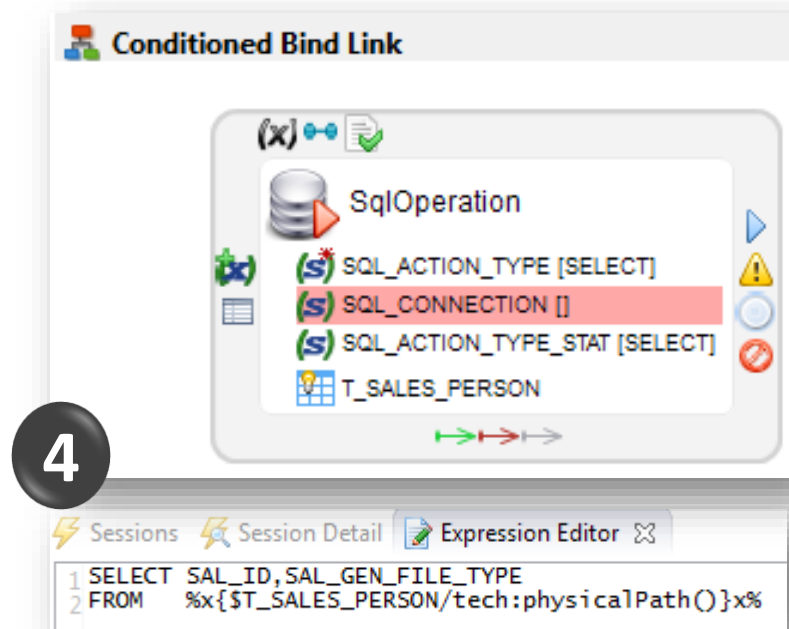
2

- Training
 - Components
 - Exos materials
 - AddExo
 - AdvExo
 - Adv01_Pivot
 - Adv02_Excel
 - Adv03_NotExists
 - Adv04_ConditionedBindLink
 - 10-Materials
 - Conditioned Bind Link.pdf
 - Init Environment on Hotel.sql
 - JSON_SalesPerson
 - step_by_step_syntax_help.txt
 - XML_SalesPerson
 - 20-Metadata

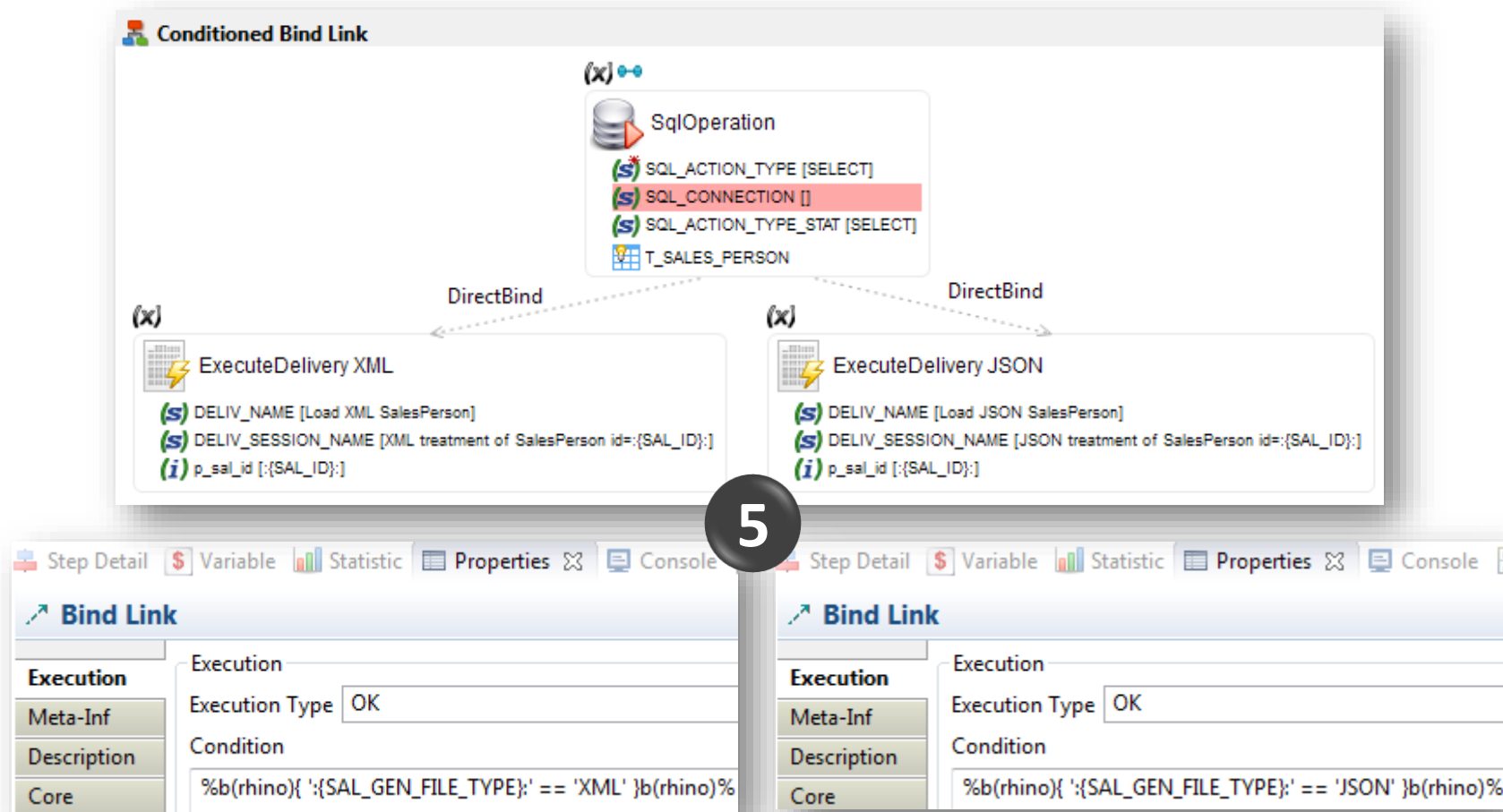
❖ 3 - Create the two mappings ad below with a filter on SAL_ID

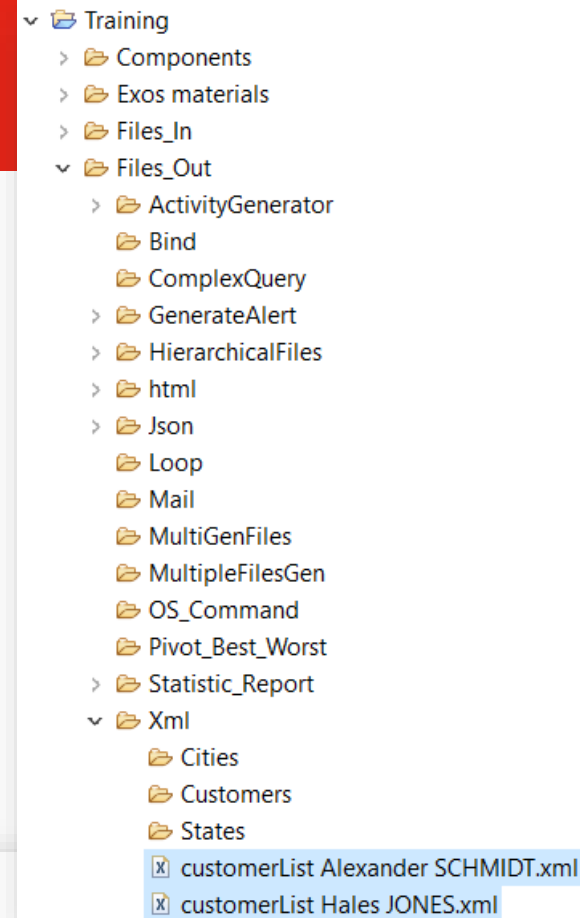


- ❖ 4 - Create a process and add a SqlOperation with the below SQL code
 - Drag & drop the T_SALES_PERSON table in the SqlOperation
 - Set the SQL_ACTION_TYPE property to SELECT



- ❖ 5 - Add two executeDelivery actions in the process
 - Add two bind links with condition on each
 - Define the properties of each executeDelivery action as below
 - Add p_sal_id parameter on each executeDelivery





6

Sessions Session Detail Expression Editor

Name: Count: 200 ☒ My sessions ☐ Current

Start Time	Name	Status
2015/12/08 17:57:21.222	Conditioned Bind Link	EXECUTE
2015/12/08 17:57:22.	XML treatment of SalesPerson id=2	EXECUTE
2015/12/08 17:57:21.	JSON treatment of SalesPerson id=3	EXECUTED
2015/12/08 17:57:21.	XML treatment of SalesPerson id=1	EXECUTED

