Anton Slizh's

U2M10.LW.ETL Overview - Load and Transformation

Task 1

2.1. Task 01: Transformation Description

Oracle gives us the following choices for transforming data inside the database:

- Transforming Data Using SQL
- Transforming Data Using PL/SQL
- Transforming Data Using Table Functions

In the building business model, the best solutions are using the SQL and PL/SQL transformations.

The data transforming and loading processes in the preparing DWH often use the small tables and simple transformations. These actions can be successfully completed by using standard functionality of INSERT, UPDATE, MERGE statements. Also, I should note that I have used very often the 'UPSERT' functionality to INSERT new rows into the table and UPDATE existing rows. The MERGE statement is really good choice for this action.

Sometimes for realization more complex transformations the standard SQL functionality is not enough. In preparing DWH for more complex and large tables such as sales or products (with SCD2 implementation) more efficient and logically simpler is to use the PL/SQL statements. For example, a PL/SQL procedure could open multiple cursors and read data from multiple source tables, combine this data using complex business rules, and finally insert the transformed data into one or more target table. It would be difficult or impossible to express the same sequence of operations using standard SQL statements.

Task 2

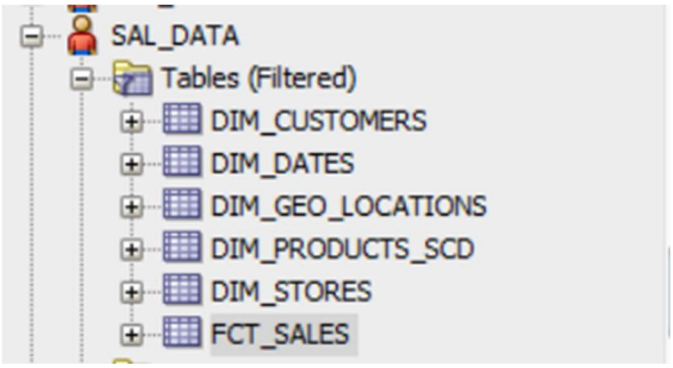
3.1. Task 02: Loading to SAL Layer Data

The Main Task is to load dimension to SAL layer

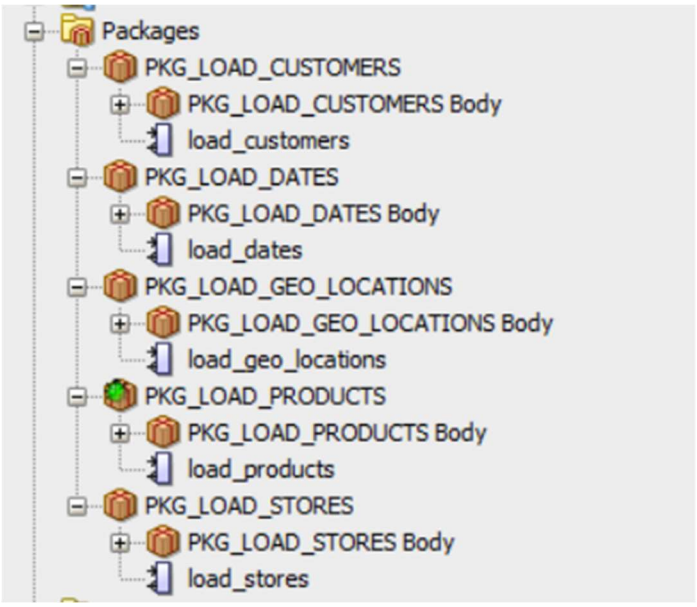
Required points:

- Create new package for Load FCT_* and DIM_* to SAL Layer
- Load Dimension
- Load SCD Dimension
- Load FCT_*

First step is to create SAL layer objects (Dimension and Fact tables) using prepared scripts.



After initializing tables we should to transform data from the DW layer into Star Layer using prepared procedures.



Executing procedures using *load_data* script:

```
1  ALTER SESSION SET CURRENT_SCHEMA = sal_data;
2  BEGIN
3      pkg_load_dates.load_dates;
4      pkg_load_geo_locations.load_geo_locations;
5      pkg_load_products.load_products;
6      pkg_load_stores.load_stores;
7      pkg_load_customers.load_customers;
8      pkg_load_sales.load_sales;
9  END;
```

Script Output | Script Output 1 | Task completed in 1,507 seconds

```
Session altered.


PL/SQL procedure successfully completed.
```

So, let's look at the data of our final Star Schema:

DIM_CUSTOMERS

```
40  SELECT * FROM sal_data.dim_customers;
```

Script Output | Query Result | SQL | Fetched 50 rows in 0,117 seconds

	CUSTOMER_ID	FIRST_NAME	LAST_NAME	PHONE	COUNTRY	EMAIL	BIRTHDAY	INSERT_DT	UPDATE_DT
1	104970	Ethan	Lewis	+375297914925	Belarus	EthanLewis@gmail.com	31.03.81	20.08.22	20.08.22
2	109701	Joe	Campbell	+375297917249	Belarus	JoeCampbell@gmail.com	07.10.82	20.08.22	20.08.22
3	110379	Jordan	Gonzalez	+375297912894	Russian Federation	JordanGonzalez@gmail.com	21.04.83	20.08.22	20.08.22
4	104829	Harold	Martinez	+375297912502	Ukraine	HaroldMartinez@gmail.com	22.01.98	20.08.22	20.08.22
5	104781	John	Wright	+375297915635	Belarus	JohnWright@gmail.com	19.04.01	20.08.22	20.08.22
6	105204	Michelle	Davis	+375297912296	Belarus	MichelleDavis@gmail.com	06.08.99	20.08.22	20.08.22
7	106256	Jesse	Nguyen	+375297916093	Belarus	JesseNguyen@gmail.com	08.04.96	20.08.22	20.08.22
8	110305	Ashley	Lewis	+375297914980	Belarus	AshleyLewis@gmail.com	18.10.93	20.08.22	20.08.22
9	107993	Jack	Lee	+375297913888	Belarus	JackLee@gmail.com	31.05.98	20.08.22	20.08.22
10	105228	Roy	Martin	+375297913804	Belarus	RoyMartin@gmail.com	09.08.95	20.08.22	20.08.22
11	106630	Larry	Moore	+375297913493	Belarus	LarryMoore@gmail.com	22.03.97	20.08.22	20.08.22
12	107338	Kimberly	Allen	+375297915493	Belarus	KimberlyAllen@gmail.com	06.12.94	20.08.22	20.08.22
13	107447	Roy	Green	+375297916492	Kazakhstan	RoyGreen@gmail.com	19.10.01	20.08.22	20.08.22
14	109859	Rebecca	Thompson	+375297914223	Belarus	RebeccaThompson@gmail.com	01.01.89	20.08.22	20.08.22
15	107411	Ethan	White	+375297914285	Belarus	EthanWhite@gmail.com	06.10.93	20.08.22	20.08.22
16	107344	Willie	Wright	+375297915718	Belarus	WillieWright@gmail.com	24.06.88	20.08.22	20.08.22
17	109469	Bryan	Adams	+375297916607	Belarus	BryanAdams@gmail.com	01.06.99	20.08.22	20.08.22
18	105074	Mark	Wright	+375297915647	Russian Federation	MarkWright@gmail.com	27.11.88	20.08.22	20.08.22
19	105662	Ethan	Rivera	+375297917101	Belarus	EthanRivera@gmail.com	10.05.82	20.08.22	20.08.22
20	109335	Paul	Sanchez	+375297914498	Kazakhstan	PaulSanchez@gmail.com	26.02.01	20.08.22	20.08.22
21	107390	Amanda	Martinez	+375297912554	Belarus	AmandaMartinez@gmail.com	30.08.99	20.08.22	20.08.22
22	110541	Sharon	Campbell	+375297917296	Belarus	SharonCampbell@gmail.com	09.10.95	20.08.22	20.08.22
23	105001	Lawres	Carter	+375297917499	Belarus	LawresCarter@gmail.com	10.03.95	20.08.22	20.08.22

DIM_DATES

```
40 SELECT * FROM sal_data.dim_dates ORDER BY date_id;
```

Script Output × Query Result ×

SQL | Fetched 50 rows in 0,029 seconds

	DATE_ID	DAY_NAME	DAY_NUMBER_IN_WEEK	DAY_NUMBER_IN_MONTH	DAY_NUMBER_IN_YEAR	CALENDAR_WEEK_NUMBER	WEEK_ENDING_DATE	CALENDAR_MONTH_N
1	01.01.21	Пятница	5	01	001	1	03.01.21	01
2	02.01.21	Суббота	6	02	002	1	03.01.21	01
3	03.01.21	Воскресенье	7	03	003	1	03.01.21	01
4	04.01.21	Понедельник	1	04	004	1	10.01.21	01
5	05.01.21	Вторник	2	05	005	1	10.01.21	01
6	06.01.21	Среда	3	06	006	1	10.01.21	01
7	07.01.21	Четверг	4	07	007	1	10.01.21	01
8	08.01.21	Пятница	5	08	008	2	10.01.21	01
9	09.01.21	Суббота	6	09	009	2	10.01.21	01
10	10.01.21	Воскресенье	7	10	010	2	10.01.21	01
11	11.01.21	Понедельник	1	11	011	2	17.01.21	01
12	12.01.21	Вторник	2	12	012	2	17.01.21	01
13	13.01.21	Среда	3	13	013	2	17.01.21	01
14	14.01.21	Четверг	4	14	014	2	17.01.21	01
15	15.01.21	Пятница	5	15	015	3	17.01.21	01
16	16.01.21	Суббота	6	16	016	3	17.01.21	01
17	17.01.21	Воскресенье	7	17	017	3	17.01.21	01
18	18.01.21	Понедельник	1	18	018	3	24.01.21	01
19	19.01.21	Вторник	2	19	019	3	24.01.21	01
20	20.01.21	Среда	3	20	020	3	24.01.21	01
21	21.01.21	Четверг	4	21	021	3	24.01.21	01
22	22.01.21	Пятница	5	22	022	4	24.01.21	01
23	23.01.21	Суббота	6	23	023	4	24.01.21	01
24	24.01.21	Воскресенье	7	24	024	4	24.01.21	01
25	25.01.21	Понедельник	1	25	025	4	31.01.21	01

DIM_GEO_LOCATIONS

```
40 SELECT * FROM sal_data.dim_geo_locations;
```

Script Output × Query Result ×

SQL | Fetched 50 rows in 0,06 seconds

	GEO_ID	GROUP_ID	GROUP_DESC	SUB_GROUP_ID	SUB_GROUP_DESC	SYSTEM_CODE	SYSTEM_DESC	REGION_DESC
1	242	2	Economic groupings	203	Small island developing states	WORLD	The UN World structure	South America
2	243	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Western Asia
3	244	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Caribbean
4	245	2	Economic groupings	205	Developed countries	WORLD	The UN World structure	Northern Europe
5	246	2	Economic groupings	205	Developed countries	WORLD	The UN World structure	Northern America
6	247	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Caribbean
7	248	2	Economic groupings	201	Least developed countries	WORLD	The UN World structure	Eastern Africa
8	249	2	Economic groupings	201	Least developed countries	WORLD	The UN World structure	Middle Africa
9	250	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Central Asia
10	251	(null)	(null)	(null)	(null)	WORLD	The UN World structure	South-Eastern Asia
11	252	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Central America
12	253	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Northern Europe
13	254	3	Unions groupings	303	European Union	WORLD	The UN World structure	Eastern Europe
14	255	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Western Asia
15	256	2	Economic groupings	201	Least developed countries	WORLD	The UN World structure	Eastern Africa
16	257	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Eastern Asia
17	258	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Western Asia
18	259	2	Economic groupings	202	Landlocked developing countries	WORLD	The UN World structure	Eastern Asia
19	260	2	Economic groupings	203	Small island developing states	WORLD	The UN World structure	Caribbean
20	261	2	Economic groupings	203	Small island developing states	WORLD	The UN World structure	Caribbean
21	262	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Eastern Asia
22	263	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Eastern Africa
23	264	(null)	(null)	(null)	(null)	WORLD	The UN World structure	Western Asia
24	265	2	Economic groupings	201	Least developed countries	WORLD	The UN World structure	Eastern Asia
25	266	2	Economic groupings	202	Landlocked developing countries	WORLD	The UN World structure	Southern Africa

DIM_STORES

```
40  SELECT * FROM sal_data.dim_stores;
```

Script Output | Query Result | SQL | All Rows Fetched: 16 in 0,024 seconds

	STORE_ID	ADDRESS	COUNTRY	REGION	CITY	PHONE	INSERT_DT	UPDATE_DT
1	397	Gomel, Kalinouskaga, 6	Belarus	Gomel	Gomel	+375291110892	20.08.22	20.08.22
2	398	Minks, Byady, 3	Belarus	Minsk	Minsk	+375291122892	20.08.22	20.08.22
3	399	Astana, Kalinouskaga, 90	Kazakhstan	Astana	Astana	+99677131211	20.08.22	20.08.22
4	400	Novogrudok, Kalinouskaga, 1	Belarus	Grodno	Novogrudok	+375788822892	20.08.22	20.08.22
5	401	Mogilev, Kalinouskaga, 17	Belarus	Mogilev	Mogilev	+375290121234	20.08.22	20.08.22
6	402	Minks, Kalinouskaga, 33	Belarus	Minsk	Minsk	+375291122892	20.08.22	20.08.22
7	403	Grodno, Savetskaya, 2	Belarus	Grodno	Grodno	+375291122552	20.08.22	20.08.22
8	404	Grodno, Kirova, 13	Belarus	Grodno	Grodno	+375294322892	20.08.22	20.08.22
9	405	Brest, Kalinouskaga, 47	Belarus	Brest	Brest	+375290120092	20.08.22	20.08.22
10	406	Kyiv, Kalinouskaga, 11	Ukraine	Kyiv	Kyiv	+380291144894	20.08.22	20.08.22
11	407	Minks, Zaporozhskaya, 73	Belarus	Minsk	Minsk	+375291522852	20.08.22	20.08.22
12	408	Grodno, Kalinouskaga, 7	Belarus	Grodno	Grodno	+375290122892	20.08.22	20.08.22
13	409	Moscow, Kalinouskaga, 12	Russian Federation	Moscow	Moscow	+71232131232	20.08.22	20.08.22
14	410	Saint Petersburg, Kalinouskaga, 90	Russian Federation	Leningrad	Saint Petersburg	+71277131232	20.08.22	20.08.22
15	411	Vitebsk, Kalinouskaga, 9	Belarus	Vitebsk	Vitebsk	+375291111892	20.08.22	20.08.22
16	412	Slonim, Skaryny, 18	Belarus	Grodno	Slonim	+375290292892	20.08.22	20.08.22

DIM_PRODUCTS_SCD

```
40  SELECT product_id, product_src_id, price,description, type, brand, start_dt, end_dt,volume, producer_country,  shelf_width, shelf_hight, shelf_depth,
41          package, package_color, package_reusable, taste, alcohol
42  FROM sal_data.dim_products_scd;
```

Script Output | Query Result | SQL | Fetched 50 rows in 0,029 seconds

	PRODUCT_ID	PRODUCT_SRC_ID	PRICE	DESCRIPTION	TYPE	BRAND	START_DT	END_DT	VOLUME	PRODUCER_COUNTRY	SHELF_WIDTH	SHELF_HIGHT	SHELF_DEPTH	PACKAGE
1	2782066	89234312932	2,5	Lidski Classic 2 liter	Classic	Lidski	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
2	2782067	89234112313	1,8	Lidski Classic 1.5 liter	Classic	Lidski	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
3	2782068	89233222932	1,5	Lidski Classic 1 liter	Classic	Lidski	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
4	2782069	29525311123	1,6	Lidski Classic 0.5 liter	Classic	Lidski	20.08.22	31.12.99	500	Belarus	5	15	5	Can
5	2782070	8123324129	2,8	Lidski Dark 2 liter	Dark	Lidski	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
6	2782071	8941241231	2	Lidski Dark 1.5 liter	Dark	Lidski	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
7	2782072	8923312315	1,55	Lidski Dark 1 liter	Dark	Lidski	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
8	2782073	2935123123	1,7	Lidski Dark 0.5 liter	Dark	Lidski	20.08.22	31.12.99	500	Belarus	5	15	5	Can
9	2782074	8123333333	2,8	Lidski Light 2 liter	Light	Lidski	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
10	2782075	8941242131	2	Lidski Light 1.5 liter	Light	Lidski	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
11	2782076	8932512388	1,55	Lidski Light 1 liter	Light	Lidski	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
12	2782077	2555223123	1,7	Lidski Light 0.5 liter	Light	Lidski	20.08.22	31.12.99	500	Belarus	5	15	5	Can
13	2782078	8233231233932	3	Hatni Classic 2 liter	Classic	Hatni	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
14	2782079	8123125127313	2,1	Hatni Classic 1.5 liter	Classic	Hatni	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
15	2782080	8928232932	1,77	Hatni Classic 1 liter	Classic	Hatni	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
16	2782081	23215311123	1,8	Hatni Classic 0.5 liter	Classic	Hatni	20.08.22	31.12.99	500	Belarus	5	15	5	Can
17	2782082	821233124129	3	Hatni Dark 2 liter	Dark	Hatni	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
18	2782083	89412123231	2,1	Hatni Dark 1.5 liter	Dark	Hatni	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
19	2782084	8928452315	1,77	Hatni Dark 1 liter	Dark	Hatni	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
20	2782085	2935235123	1,8	Hatni Dark 0.5 liter	Dark	Hatni	20.08.22	31.12.99	500	Belarus	5	15	5	Can
21	2782086	8111543333	2,9	Hatni Light 2 liter	Light	Hatni	20.08.22	31.12.99	2000	Belarus	10	30	10	Plastic
22	2782087	891236782131	2	Hatni Light 1.5 liter	Light	Hatni	20.08.22	31.12.99	1500	Belarus	10	25	10	Plastic
23	2782088	823256672388	1,8	Hatni Light 1 liter	Light	Hatni	20.08.22	31.12.99	1000	Belarus	8	20	8	Plastic
24	2782089	212654312643	1,89	Hatni Light 0.5 liter	Light	Hatni	20.08.22	31.12.99	500	Belarus	5	15	5	Can

FCT_SALES

```
78  SELECT * FROM sal_data.fct_sales;
```

Script Output × | Query Result ×

SQL | Fetched 50 rows in 0,069 seconds

	SALE_ID	DATE_ID	PRODUCT_ID	CUSTOMER_ID	STORE_ID	GEO_ID	AMOUNT	POS_TRANSACTION	INSERT_DT	SUM
1	65116278	29.11.21	2782085	107091	410	473	2	20211129000115	20.08.22	3,6
2	65116279	11.02.22	2782139	107091	399	476	3	20220211000116	20.08.22	5,4
3	65116280	23.08.21	2782130	106489	402	412	1	20210823000201	20.08.22	2,5
4	65116281	17.07.22	2782129	110641	402	412	1	20220717000301	20.08.22	1,6
5	65116282	07.03.21	2782096	110547	404	412	2	20210307000405	20.08.22	3,54
6	65116283	20.12.21	2782126	110547	405	412	2	20211220000409	20.08.22	5
7	65116284	13.10.21	2782124	110547	406	453	2	20211013000413	20.08.22	3
8	65116285	27.07.22	2782103	104332	402	412	3	20220727000501	20.08.22	9
9	65116286	15.02.22	2782133	108251	407	412	2	20220215000603	20.08.22	3,2
10	65116287	30.04.21	2782067	108251	406	453	1	20210430000613	20.08.22	1,8
11	65116288	29.05.22	2782095	108251	399	476	1	20220529000616	20.08.22	2,1
12	65116289	04.12.21	2782143	107279	410	473	2	20211204000715	20.08.22	3,6
13	65116290	29.10.21	2782109	109922	407	412	1	20211029000803	20.08.22	2,9
14	65116291	03.04.22	2782107	110642	412	412	3	20220403000907	20.08.22	10,5
15	65116292	11.06.21	2782083	110642	409	473	1	20210611000914	20.08.22	2,1
16	65116293	05.06.21	2782144	105675	397	412	3	20210605001012	20.08.22	4,5
17	65116294	26.06.21	2782122	105675	406	453	3	20210626001013	20.08.22	5,7
18	65116295	06.05.21	2782128	105675	410	473	3	20210506001015	20.08.22	4,5
19	65116296	18.06.21	2782104	105675	399	476	3	20210618001016	20.08.22	6,6
20	65116297	13.12.21	2782140	106066	407	412	3	20211213001103	20.08.22	4,5
21	65116298	11.04.21	2782110	106066	412	412	2	20210411001107	20.08.22	10
22	65116299	30.04.22	2782104	110643	398	412	1	20220430001202	20.08.22	2,2
23	65116300	15.08.21	2782104	110643	404	412	1	20210815001205	20.08.22	2,2
24	65116301	05.07.22	2782129	110643	404	412	3	20220705001205	20.08.22	4,8
25	65116302	17.02.21	2782118	110643	406	453	1	20210217001213	20.08.22	1,9

The DIM_PRODUCTS_SCD table is SCD2 type dimension. So, for each row we have the start date and end date which represent the time period of being actual. The surrogate key for each product is *product_id* and the natural key is *product_src_id.*

Let's try to update any product and show the SCD2 mechanism in use. For example, choose the first product in the screen – Lidski Classic 2 liter (good choice by the way).

Product_id – 278066, Product_src_id – 89234312932.

Manually add to the DW layer new row with the same *product_src_id* attribute value, but with the updated *producer_country* attribute value and the tomorrow *insert_dt.*

```
45  INSERT INTO dw_data.dw_products
46  VALUES (
47          dw_data.seq_products.NEXTVAL,
48          '89234312932',
49          2.5,
50          'Lidski Classic 2 liter',
51          'Classic',
52          'Lidski',
53          'Germany',
54          2000,
55          10,
56          30,
57          10,
58          'Plastic',
59          'Brown',
60          'Reusable',
61          'Classic',
62          1.2,
63          SYSDATE + 1
64          );
```

Script Output | Query Result | Task completed in 0,046 seconds

```
1 row inserted.
```

Now let's reload products dimension data.

```
64  BEGIN
65      sal_data.pkg_load_products.load_products;
66  END;
```

Script Output | Query Result | Task completed in 0,039 seconds

```
PL/SQL procedure successfully completed.
```

Look at our updated product at the dimension:

```
69  SELECT product_id, product_src_id, price,description, type, brand, start_dt, end_dt,volume, producer_country,  shelf_width, shelf_hight, shelf_depth,
70         package, package_color, package_reusable, taste, alcohol
71  FROM sal_data.dim_products_scd
72  WHERE product_src_id = '89234312932';
```

Script Output | Query Result | SQL | All Rows Fetched: 2 in 0,023 seconds

	PRODUCT_ID	PRODUCT_SRC_ID	PRICE	DESCRIPTION	TYPE	BRAND	START_DT	END_DT	VOLUME	PRODUCER_COUNTRY	SHELF_WIDTH	SHELF_HIGHT	SHELF_DEPTH	PACKAGE	PACKAGE_C
1	2782066	89234312932	2,5	Lidski Classic 2 liter	Classic	Lidski	20.08.22	21.08.22	2000	Belarus	10	30	10	Plastic	Brown
2	2782146	89234312932	2,5	Lidski Classic 2 liter	Classic	Lidski	21.08.22	31.12.99	2000	Germany	10	30	10	Plastic	Brown

We can see that new row became actual for this product (*product_src_id*). So, this way we can store the history of changes in our product dimension.

Also let's create some Data Marts on our Star Scheme. I have used the VIEWs which select all necessary data from prepared Star Scheme.

DM_ALL_CUSTOMERS

```sql
CREATE OR REPLACE VIEW sal_data.all_customers
AS
    SELECT c.customer_id,
           first_name,
           last_name,
           phone,
           country,
           email,
           birthday,
           amount,
           sum
    FROM sal_data.dim_customers c
    JOIN (SELECT customer_id,
                 count(amount) AS amount,
                 sum(sum) AS sum
          FROM sal_data.fct_sales
          GROUP BY customer_id) s
    ON s.customer_id = c.customer_id
    ORDER BY sum DESC;
```

```sql
80  SELECT * FROM sal_data.all_customers;
```

Script Output | Query Result

SQL | Fetched 50 rows in 0,089 seconds

	CUSTOMER_ID	FIRST_NAME	LAST_NAME	PHONE	COUNTRY	EMAIL	BIRTHDAY	AMOUNT	SUM
1	109293	Donna	Smith	+375297911399	Belarus	DonnaSmith@gmail.com	19.03.97	71	345,5
2	109130	Lisa	Gonzalez	+375297912927	Belarus	LisaGonzalez@gmail.com	13.05.82	74	325,25
3	108651	Gabriel	White	+375297914307	Russian Federation	GabrielWhite@gmail.com	11.02.91	66	323,07
4	104956	Margaret	White	+375297914338	Belarus	MargaretWhite@gmail.com	28.06.91	70	321,76
5	105153	Adam	Ramirez	+375297914790	Belarus	AdamRamirez@gmail.com	30.08.96	68	315,31
6	105989	Stephanie	Lopez	+375297912814	Belarus	StephanieLopez@gmail.com	04.03.86	68	313,1
7	105209	Wayne	Martinez	+375297912521	Kazakhstan	WayneMartinez@gmail.com	18.08.89	62	313,09
8	106958	Jeremy	Allen	+375297915440	Belarus	JeremyAllen@gmail.com	20.02.86	65	312,4
9	107745	Alexan	Baker	+375297916828	Belarus	AlexanBaker@gmail.com	24.02.92	62	308,54
10	109359	Roy	Torres	+375297915980	Belarus	RoyTorres@gmail.com	05.12.85	61	308,36
11	109001	Jonat	Baker	+375297916819	Russian Federation	JonatBaker@gmail.com	28.04.92	70	308,14
12	106832	Benja	Moore	+375297913497	Kazakhstan	BenjaMoore@gmail.com	23.01.91	68	307,36
13	104935	Deborah	Lopez	+375297912813	Belarus	DeborahLopez@gmail.com	16.10.90	67	307,03
14	108818	Mark	Miller	+375297912063	Belarus	MarkMiller@gmail.com	17.11.95	63	304,52
15	105027	Frank	Wilson	+375297912989	Russian Federation	FrankWilson@gmail.com	23.03.96	64	303,94
16	108213	Emily	Young	+375297915366	Belarus	EmilyYoung@gmail.com	22.10.95	70	303,2
17	107846	Gerald	Hall	+375297916981	Belarus	GeraldHall@gmail.com	19.07.88	61	302,44
18	106823	Aaron	Wilson	+375297912996	Belarus	AaronWilson@gmail.com	25.06.90	68	302,11
19	110415	Jonat	Scott	+375297915795	Russian Federation	JonatScott@gmail.com	06.05.86	62	302,03
20	110189	Timothy	Harris	+375297914377	Belarus	TimothyHarris@gmail.com	10.10.95	65	302,03
21	105465	Joe	Hill	+375297916225	Belarus	JoeHill@gmail.com	28.08.86	65	301,76
22	109163	Karen	Allen	+375297915486	Ukraine	KarenAllen@gmail.com	21.04.89	63	300,98
23	106977	Alan	Green	+375297916487	Belarus	AlanGreen@gmail.com	13.08.96	64	300,02
24	110576	Dennis	Taylor	+375297913377	Belarus	DennisTaylor@gmail.com	26.03.84	66	297,48
25	105505	Ashley	Brown	+375297911780	Belarus	AshleyBrown@gmail.com	17.08.90	65	296,56
26	109033	Gabriel	Rodriguez	+375297912387	Belarus	GabrielRodriguez@gmail.com	06.04.83	66	296,16

DM_ALL_BRANDS

```
CREATE OR REPLACE VIEW sal_data.all_brands
AS
    SELECT  s.date_id,
            p.brand,
            COUNT(s.amount) AS amount,
            SUM(s.sum) AS sum
    FROM sal_data.fct_sales s
    LEFT JOIN sal_data.dim_products_scd p
    ON s.product_id = p.product_id
    GROUP BY s.date_id, p.brand
    ORDER BY date_id, sum DESC;
```

```
80  SELECT * FROM sal_data.all_brands;
```

Script Output | Query Result | SQL | Fetched 50 rows in 0,146 seconds

	DATE_ID	BRAND	AMOUNT	SUM
1	01.01.21	Hatni	150	624,87
2	01.01.21	Lidski	152	543,55
3	01.01.21	Zaporozhski	73	472,5
4	01.01.21	Alivarski	84	360,92
5	01.01.21	Ruski	62	205,5
6	02.01.21	Lidski	174	687,45
7	02.01.21	Hatni	126	510,1
8	02.01.21	Zaporozhski	78	494,3
9	02.01.21	Alivarski	74	293,46
10	02.01.21	Ruski	68	213,8
11	03.01.21	Lidski	178	693,75
12	03.01.21	Zaporozhski	89	534,3
13	03.01.21	Hatni	119	492,13
14	03.01.21	Alivarski	80	383,24
15	03.01.21	Ruski	89	278,5

DM_STORE_SALES

```
CREATE OR REPLACE VIEW sal_data.stores_sales
AS
    SELECT date_id,
           country,
           address,
           COUNT(amount) AS amount,
           SUM(sum) AS sum
    FROM sal_data.fct_sales sl
    LEFT JOIN sal_data.dim_stores st
    ON sl.store_id = st.store_id
    GROUP BY date_id, country, address
    ORDER BY date_id, country, sum DESC;
```

```
80  SELECT * FROM sal_data.stores_sales;
```

Script Output | Query Result

SQL | Fetched 50 rows in 0,113 seconds

	DATE_ID	COUNTRY	ADDRESS	AMOUNT	SUM
1	01.01.21	Belarus	Novogrudok, Kalinouskaga, 1	44	190,81
2	01.01.21	Belarus	Slonim, Skaryny, 18	42	187,76
3	01.01.21	Belarus	Mogilev, Kalinouskaga, 17	44	181,59
4	01.01.21	Belarus	Gomel, Kalinouskaga, 6	35	177,21
5	01.01.21	Belarus	Minks, Byady, 3	37	160,31
6	01.01.21	Belarus	Minks, Kalinouskaga, 33	37	143,51
7	01.01.21	Belarus	Minks, Zaporozhskaya, 73	31	128,38
8	01.01.21	Belarus	Grodno, Kalinouskaga, 7	29	123,9
9	01.01.21	Belarus	Grodno, Kirova, 13	31	109,47
10	01.01.21	Belarus	Grodno, Savetskaya, 2	31	108,39
11	01.01.21	Belarus	Brest, Kalinouskaga, 47	25	98,05
12	01.01.21	Belarus	Vitebsk, Kalinouskaga, 9	21	97,12
13	01.01.21	Kazakhstan	Astana, Kalinouskaga, 90	19	103,57
14	01.01.21	Russian Federation	Saint Petersburg, Kalinouskaga, 90	30	147,57
15	01.01.21	Russian Federation	Moscow, Kalinouskaga, 12	31	124,68
16	01.01.21	Ukraine	Kyiv, Kalinouskaga, 11	34	125,02
17	02.01.21	Belarus	Grodno, Savetskaya, 2	39	162,33
18	02.01.21	Belarus	Grodno, Kalinouskaga, 7	35	156,9
19	02.01.21	Belarus	Grodno, Kirova, 13	38	156,52
20	02.01.21	Belarus	Novogrudok, Kalinouskaga, 1	36	145,48
21	02.01.21	Belarus	Brest, Kalinouskaga, 47	32	143,89
22	02.01.21	Belarus	Minks, Zaporozhskaya, 73	33	140,68

And at the end let's create the script which join together all load procedures to execute whole ETL process.

```
1  BEGIN
2      -- Cleansing Layer
3      dw_cl.pkg_load_products.load_products;
4      dw_cl.pkg_load_stores.load_stores;
5      dw_cl.pkg_load_customers.load_customers;
6      dw_cl.pkg_load_sales.load_sales;
7
8      -- DW Layer
9      dw_data.pkg_load_dates.load_dates;
10     dw_data.pkg_load_geo_locations.load_geo_locations;
11     dw_data.pkg_load_products.load_products;
12     dw_data.pkg_load_stores.load_stores;
13     dw_data.pkg_load_customers.load_customers;
14     dw_data.pkg_load_sales.load_sales;
15
16     -- SAL Layer
17     sal_data.pkg_load_dates.load_dates;
18     sal_data.pkg_load_geo_locations.load_geo_locations;
19     sal_data.pkg_load_products.load_products;
20     sal_data.pkg_load_stores.load_stores;
21     sal_data.pkg_load_customers.load_customers;
22     sal_data.pkg_load_sales.load_sales;
23 END;
```

Script Output

Task completed in 6,349 seconds

```
PL/SQL procedure successfully completed.
```