U2M8-9.LW.ETL Overview – Extraction_Transportation

Shkrabatouskaya Vera

https://github.com/VeraShkrabatouskaya/DataMola_Data-Camping-2022

Evolution of the Data Warehouse

As the data warehouse is a living IT system, sources and targets might change. Those changes must be maintained and tracked through the lifespan of the system without overwriting or deleting the old ETL process flow information. To build and keep a level of trust about the information in the warehouse, the process flow of each individual record in the warehouse can be reconstructed at any point in time in the future in an ideal case.

2. ETL Extraction – BASIC

We need to load your data warehouse regularly so that it can serve its purpose of facilitating business analysis. To do this, data from one or more operational systems needs to be extracted and copied into the data warehouse. The challenge in data warehouse environments is to integrate, rearrange and consolidate large volumes of data over many systems, thereby providing a new unified information base for business intelligence.

The process of extracting data from source systems and bringing it into the data warehouse is commonly called ETL, which stands for extraction, transformation, and loading.

The methodology and tasks of ETL have been well known for many years, and are not necessarily unique to data warehouse environments: a wide variety of proprietary applications and database systems are the IT backbone of any enterprise. Data has to be shared between applications or systems, trying to integrate them, giving at least two applications the same picture of the world. This data sharing was mostly addressed by mechanisms similar to what we now call ETL.

2.1. Task 01: Extraction Description

During extraction, the desired data is identified and extracted from many different sources, including database systems and applications. Very often, it is not possible to identify the specific subset of interest, therefore more data than necessary has to be extracted, so the identification of the relevant data will be done at a later point in time. Depending on the source system's capabilities (for example, operating system resources), some transformations may take place during this extraction process. The size of the extracted data varies from hundreds of kilobytes up to gigabytes, depending on the source system and the business situation. The same is true for the time delta between two (logically) identical extractions: the time span may vary between days/hours and minutes to near real-time.

The method for extracting data from a data warehouse is highly dependent on the original system as well as the business needs of the target data warehouse environment.

In our business model, we will use full extraction as the logical extraction method and offline extraction as the physical extraction method.

Full Extraction

The data is extracted completely from the source system. Because this extraction reflects all the data currently available on the source system, there's no need to keep track of changes to the data source since the last successful extraction. The source data will be provided as-is and no additional logical information (for example, timestamps) is necessary on the source site.

Offline Extraction

The data is not extracted directly from the source system but is staged explicitly outside the original source system. The data already has an existing structure (for example, redo logs, archive logs or transportable tablespaces) or was created by an extraction routine.

2. ETL Transportation – BASIC

After data is extracted, it has to be physically transported to the target system or to an intermediate system for further processing. Depending on the chosen way of transportation, some transformations can be done during this process, too.

2.2. Task 02: Transportation Description

Transportation is the operation of moving data from one system to another system. In a data warehouse environment, the most common requirements for transportation are in moving data from:

- A source system to a staging database or a data warehouse database
- A staging database to a data warehouse
- A data warehouse to a data mart

There are three basic choices for transporting data in warehouses:

- Transportation Using Flat Files
- Transportation Through Distributed Operations
- Transportation Using Transportable Tablespaces

For our business model, we suggest considering Transportation Using Flat Files.

Transportation Using Flat Files

The most common method for transporting data is by the transfer of flat files, using mechanisms such as FTP or other remote file system access protocols. Data is unloaded or exported from the source system into flat files and is then transported to the target platform using FTP or similar mechanisms.

Because source systems and data warehouses often use different operating systems and database systems, using flat files is often the simplest way to exchange data between heterogeneous systems with minimal transformations. However, even when transporting data between homogeneous systems, flat files are often the most efficient and most easy-to-manage mechanism for data transfer.

3. ETL Extraction – Example of Loading FCT_*

Task 02 is common for LabWork 08, 09.

3.1. Task 02: Prepare Table of Facts to DW Layer

The fact table was moved from the CL layer to the DW layer in Lab4 Unit 02.

Create a cls_t_customer table at the DW - Cleansing Level.

- cls_t_transaction

```
alter session set current_schema=DW_CL;
GRANT SELECT ON DW_CL.cls_t_transaction TO DW_DATA;

Create table cls_t_transaction (
    time_ID                     DATE,
    customer_name               VARCHAR2(50) NOT NULL,
    brand_name                  VARCHAR2(50) NOT NULL,
    customer_address            VARCHAR2(50) NOT NULL,
    customer_city               VARCHAR2(30) NOT NULL,
    customer_country            VARCHAR2(30) NOT NULL,
    customer_email              VARCHAR2(50) NOT NULL,
    customer_office_phone       VARCHAR2(30) NOT NULL,
    customer_mobile_phone       VARCHAR2(30) NOT NULL,
    product_name                VARCHAR2(50) NOT NULL,
    agency_name                 VARCHAR2(50) NOT NULL,
    agency_city                 VARCHAR2(30) NOT NULL,
    agency_country              VARCHAR2(30) NOT NULL,
    agency_address              VARCHAR2(50) NOT NULL,
    agency_postcode             VARCHAR2(6) NOT NULL,
    agency_email                VARCHAR2(30) NOT NULL,
    agency_office_phone         VARCHAR2(30) NOT NULL,
    agency_mobile_phone         VARCHAR2(30) NOT NULL,
    agency_Fee_percent          DECIMAL (10,2),
    agency_VAT_percent          DECIMAL (10,2),
```

Script Output

Task completed in 4.284 seconds

```
Session altered.

Table CLS_T_TRANSACTION created.
```

Let's create packages to get data from Storage level SA_* in DW - Cleanup level for the table cls_t_transaction.

```
alter session set current_schema = DW_CL;

CREATE OR REPLACE PACKAGE pkg_etl_cls_tran
AS
    PROCEDURE load_cls_transaction;
END pkg_etl_cls_transaction;
/
```

```
Session altered.

Package PKG_ETL_CLS_TRANSACTION compiled
```

Grant permissions to user DW_CL in tablespace ts_dw_cl to use data from table SA_TRANSACTION_FACT_DATA in tablespace ts_sa_promotions_data_01.

```
alter session set current_schema=sa_promotions;
GRANT SELECT ON SA_TRANSACTION_FACT_DATA TO DW_CL;
```

```
Session altered.

Grant succeeded.
```

```
alter session set current_schema = DW_CL;

CREATE OR REPLACE PACKAGE body pkg_etl_cls_transaction
AS
  PROCEDURE load_cls_transaction
   AS
      CURSOR cursor_cls_transaction
      IS
         SELECT DISTINCT time_ID
                        ,customer_name
                        ,brand_name
                        ,customer_address
                        ,customer_city
                        ,customer_country
                        ,customer_email
                        ,customer_office_phone
                        ,customer_mobile_phone
```

```
Session altered.

Grant succeeded.

Session altered.

Package Body PKG_ETL_CLS_TRANSACTION compiled
```

```
alter session set current_schema = DW_CL;
alter user DW_CL QUOTA UNLIMITED ON ts_dw_cl;

EXEC pkg_etl_cls_transaction.load_cls_transaction;
```

Script Output | Task completed in 21.231 seconds

```
Session altered.

User DW_CL altered.

PL/SQL procedure successfully completed.
```

```
SELECT * FROM cls_t_transaction
order by 1;
```

Script Output | Query Result | SQL | Fetched 50 rows in 2.577 seconds

	TIME_ID	CUSTOMER_NAME	BRAND_NAME	CUSTOMER_ADDRESS	CUSTOMER_CITY	CUSTOMER_COUNTRY	CUSTOMER_EMAIL
1	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
2	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
3	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
4	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
5	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
6	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
7	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
8	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
9	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
10	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
11	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
12	01-JAN-22	Visa International Service Association	VISA	900 Metro Center Blvd.	Foster City	United States	https://usa.visa.com/contact-us.html
13	01-JAN-22	Visa International Service Association	VISA	900 Metro Center Blvd.	Foster City	United States	https://usa.visa.com/contact-us.html
14	01-JAN-22	Visa International Service Association	VISA	900 Metro Center Blvd.	Foster City	United States	https://usa.visa.com/contact-us.html
15	01-JAN-22	Visa International Service Association	VISA	900 Metro Center Blvd.	Foster City	United States	https://usa.visa.com/contact-us.html
16	01-JAN-22	Visa International Service Association	VISA	900 Metro Center Blvd.	Foster City	United States	https://usa.visa.com/contact-us.html
17	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
18	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
19	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
20	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
21	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
22	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS
23	01-JAN-22	Samsung Electronics	SAMSUNG	85 Challenger Rd. Ridgefield Park	New Jersey	United States	www.facebook.com/SamsungUS

Let's create the DW_DATA.FCT_business tables and sequences at the DW layer for the DW_DATA user in the ts_dw_data_01 tablespace.

- DW_DATA.FCT_business

```
--alter session set current_schema = DW_DATA;
--drop table FCT_business CASCADE CONSTRAINTS;

CREATE TABLE FCT_business(
    Business_Fact_ID NUMBER(10),
    TIME_ID                          DATE,
    customer_ID                      NUMBER(10),
    employee_ID                      NUMBER(10),
    agency_ID                        NUMBER(10),
    gen_period_ID                    NUMBER(10),
    product_ID                       NUMBER(10),
    promotion_ID                     NUMBER(10),
    gross_profit_dollar_amount       DECIMAL (30,2),
    net_profit_dollar_amount         DECIMAL (30,2),
    gross_revenue_dollar_amount      DECIMAL (30,2),
    net_revenue_dollar_amount        DECIMAL (30,2),
```

Script Output | Query Result | Task completed in 5.194 seconds

```
Session altered.

Table FCT_BUSINESS created.

Table FCT_BUSINESS altered.

Table FCT_BUSINESS altered.

Table FCT_BUSINESS altered.
```

```
select * from FCT_business order by 1;
```

Script Output | Query Result

SQL | All Rows Fetched: 0 in 1.353 seconds

BUSINESS...	TIME_ID	CUSTOME...	EMPLOYE...	AGENCY_ID	GEN_PERI...	PRODUCT...	PROMOTI...	GROSS_P...	NET_PRO...	GROSS_R...	NET_REV...	GROSS_C...	NET_COS...	GROSS_S...

```
CREATE SEQUENCE DW_DATA.SQ_FCT_business
    START WITH        1
    INCREMENT BY      1
    NOCACHE
    NOCYCLE;
```

Script Output | Task completed in 0.041 seconds

```
Sequence DW_DATA.SQ_FCT_BUSINESS created.
```

Let's create packages to move all the data from the Cleansing Level to the DW Level, with the natural keys converted to primary keys.

- pkg_etl_dw_transaction

...sql | pkg_etl_cls_transaction-impl.sql | cls_t_transaction.sql | pkg_etl_dw_FCT_business-def.sql

SQL Worksheet | History

Worksheet | Query Builder

```
alter session set current_schema = DW_DATA;

CREATE OR REPLACE PACKAGE pkg_etl_dw_transaction
AS
    PROCEDURE load_dw_transaction;
END pkg_etl_dw_transaction;
/
```

Script Output | Task completed in 0.055 seconds

```
Session altered.

Package PKG_ETL_DW_TRANSACTION compiled
```

...sql | pkg_etl_cls_transaction-impl.sql | cls_t_transaction.sql | pkg_etl_dw_FCT_business-def.sql | pkg_etl_dw_FCT_business-impl.sql

SQL Worksheet | History

Worksheet | Query Builder

```
alter session set current_schema = DW_CL;
GRANT SELECT ON DW_CL.cls_t_transaction TO DW_DATA;

alter session set current_schema = DW_DATA;
```

Script Output | Task completed in 0.1 seconds

```
Session altered.

Grant succeeded.

Session altered.
```

```sql
alter session set current_schema = DW_DATA;

CREATE OR REPLACE PACKAGE body pkg_etl_dw_transaction
AS
  PROCEDURE load_dw_transaction
   AS
     BEGIN
      DECLARE
          TYPE CURSOR_VARCHAR IS TABLE OF varchar2(50);
          TYPE CURSOR_DECIMAL IS TABLE OF DECIMAL(30,2);
          TYPE CURSOR_DATE IS TABLE OF DATE;
          TYPE CURSOR_NUMBER IS TABLE OF number(10);

          TYPE BIG_CURSOR IS REF CURSOR;

          ALL_INF BIG_CURSOR;

          TIME_ID CURSOR_DATE;
          customer_ID CURSOR_NUMBER;
          employee_ID CURSOR_NUMBER;
          agency_ID CURSOR_NUMBER;
          gen_period_ID CURSOR_NUMBER;
          product_ID CURSOR_NUMBER;
          promotion_ID CURSOR_NUMBER;
          Business_Fact_ID CURSOR_NUMBER;
          gross_profit_dollar_amount CURSOR_DECIMAL;
          net_profit_dollar_amount CURSOR_DECIMAL;
```

Script Output — Task completed in 0.308 seconds

```
Package Body PKG_ETL_DW_TRANSACTION compiled
```

```sql
alter session set current_schema = DW_DATA;
alter user DW_DATA QUOTA UNLIMITED ON TS_DW_DATA_01;

EXEC pkg_etl_dw_transaction.load_dw_transaction;
```

Script Output — Task completed in 129.439 seconds

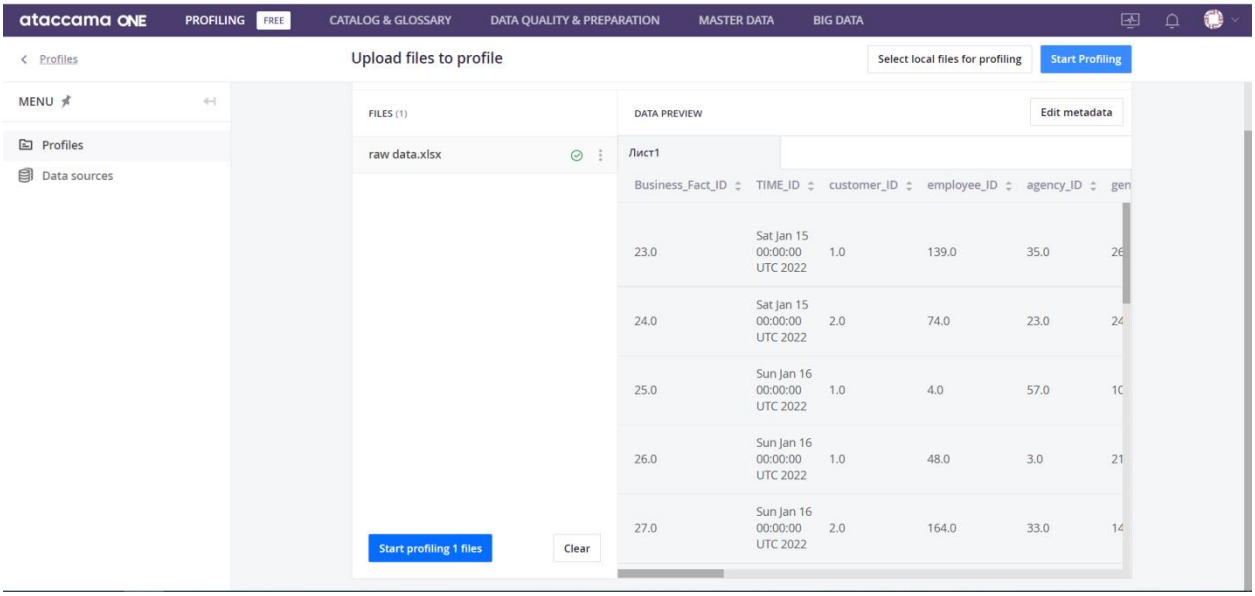
```
Session altered.

User DW_DATA altered.

PL/SQL procedure successfully completed.
```

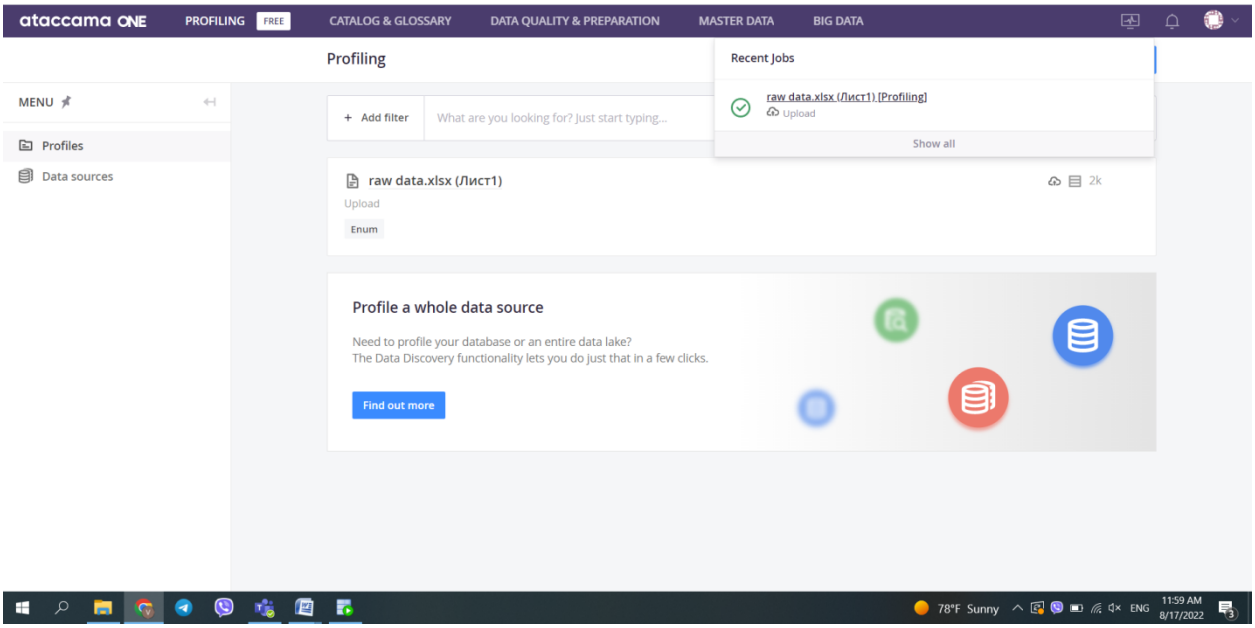
```sql
SELECT * from DW_DATA.FCT_business ORDER BY 1;
```

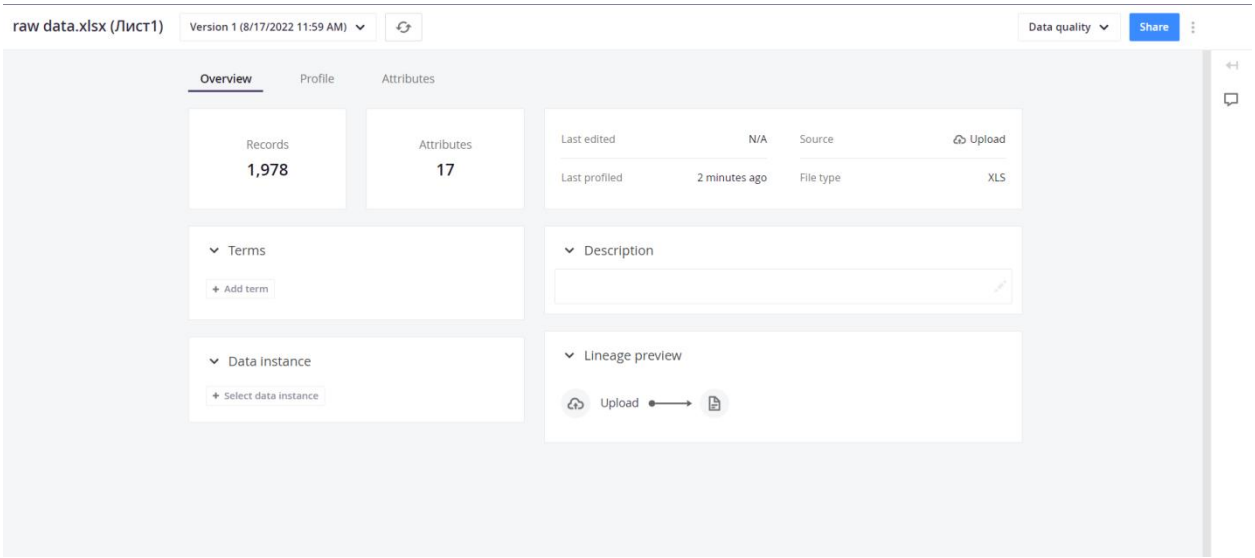
Query Result — Fetched 50 rows in 0.813 seconds

	BUSINESS_FACT_ID	TIME_ID	CUSTOMER_ID	EMPLOYEE_ID	AGENCY_ID	GEN_PERIOD_ID	PRODUCT_ID	PROMOTION_ID	GROSS_PROFIT_DOLLAR_AMOUNT	NET_PROFIT_DOLLAR_AMOUNT	GROSS_RI
1	1	02-JAN-22	1	111	3	265625	15	267665	-113.55	-95.42	
2	2	01-JAN-22	1	185	23	259708	13	81489	1185.07	1039.53	
3	3	05-JAN-22	2	150	58	25958	14	168631	175.58	146.32	
4	4	06-JAN-22	1	86	1	191010	7	82335	-74.24	-65.7	
5	5	04-JAN-22	1	138	56	295257	10	245987	1798.64	1591.71	
6	6	07-JAN-22	1	112	10	156322	12	82781	24.92	20.94	
7	7	08-JAN-22	1	168	1	1313	10	28697	861.37	762.27	
8	8	06-JAN-22	2	78	12	156272	8	313	-157.24	-136.73	
9	9	08-JAN-22	2	189	8	139650	14	169246	420.15	350.13	
10	10	11-JAN-22	1	142	2	1714	1	169868	267.08	224.44	
11	11	11-JAN-22	2	63	5	132058	14	247753	881.9	741.09	
12	12	12-JAN-22	2	19	10	267916	8	203894	1915.4	1609.58	
13	13	13-JAN-22	1	10	42	236563	7	322953	3437.03	2864.19	
14	14	16-JAN-22	1	4	57	109542	6	1267	3277.53	2664.66	
15	15	16-JAN-22	1	48	3	214382	12	258232	1733.96	1457.11	
16	16	17-JAN-22	1	119	38	182699	2	41551	-222.98	-182.77	
17	17	19-JAN-22	1	21	56	182903	13	227482	353.39	312.74	
18	18	20-JAN-22	1	83	2	287401	2	128186	-320.04	-268.94	
19	19	22-JAN-22	1	185	38	38173	4	21649	49.91	40.91	
20	20	22-JAN-22	1	117	53	168926	4	21677	1654.62	1575.83	
21	21	22-JAN-22	2	26	22	183832	14	259579	1181.25	984.37	
22	22	21-JAN-22	1	76	22	318990	2	205721	160.07	133.39	

Let's look at sampling data from table DW_DATA.FCT_business using https://app.ataccama.com/.







raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

Unique: 0 (0%)Distinct: 186 (100%)Null: 0 (0%)

# Business_Fact_ID	TIME_ID	# customer_ID (Enum)	# employee_ID	# agency_ID	# gen_period_
All values are unique (Unique)	Unique Distinct Null	Enum (Only 2 distinct values)	Count: Unique Distinct Null	Count: Unique Distinct Null	All values a (Uniq)
23	2022-01-15 00:00:00	1	139	35	268155
24	2022-01-15 00:00:00	2	74	23	245569
25	2022-01-16 00:00:00	1	4	57	109542
26	2022-01-16 00:00:00	1	48	3	214382
27	2022-01-16 00:00:00	2	164	33	141006

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

Unique: 0 (0%)Distinct: 200 (100%)Null: 0 (0%)

# Business_Fact_ID	TIME_ID	# customer_ID (Enum)	# employee_ID	# agency_ID	# gen_period_
All values are unique (Unique)	Count: Unique Distinct Null	Enum (Only 2 distinct values)	Unique Distinct Null	Count: Unique Distinct Null	All values a (Uniq)
23	2022-01-15 00:00:00	1	139	35	268155
24	2022-01-15 00:00:00	2	74	23	245569
25	2022-01-16 00:00:00	1	4	57	109542
26	2022-01-16 00:00:00	1	48	3	214382
27	2022-01-16 00:00:00	2	164	33	141006

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

Unique: 0 (0%)Distinct: 60 (100%)Null: 0 (0%)

# Business_Fact_ID	TIME_ID	# customer_ID (Enum)	# employee_ID	# agency_ID	# gen_period_
All values are unique (Unique)	Count: Unique Distinct Null	Enum (Only 2 distinct values)	Count: Unique Distinct Null	Unique Distinct Null	All values a (Uniq)
23	2022-01-15 00:00:00	1	139	35	268155
24	2022-01-15 00:00:00	2	74	23	245569
25	2022-01-16 00:00:00	1	4	57	109542
26	2022-01-16 00:00:00	1	48	3	214382
27	2022-01-16 00:00:00	2	164	33	141006

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

# gen_period_ID	# product_ID Enum	# promotion_ID	# gross_profit_dollar_am...	# net_profit_dollar_amo...	# gross_reven
All values are unique Unique	Enum Only 16 distinct values	All values are unique Unique	Duplicates 12 duplicate values	Duplicates 9 duplicate values	Duplic 9 duplicate
268155	2	226766	6430.28	5314.28	6804.17
245569	9	301536	1057.46	927.6	1617.2
109542	6	1267	3277.53	2664.66	3821.19
214382	12	258232	1733.96	1457.11	2231.38
141006	9	301963	-385.14	-337.85	40.08

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

# gross_revenue_dollar_...	# net_revenue_dollar_a...	# gross_cost_dollar_amo...	# net_cost_dollar_amount	# gross_salary_employee...	# net_salary_
Duplicates 9 duplicate values	Duplicates 5 duplicate values	Unique: 1191 (43%)Distinct: 1548 (56%)Null: 0 (0%) Unique Distinct Null	Unique Distinct Null	Count Unique Distinct Null	Count Unique Disti
6804.17	5623.28	373.89	309	373.89	309
1617.2	1418.6	559.74	491	559.74	491
3821.19	3106.66	543.66	442	543.66	442
2231.38	1875.11	497.42	418	497.42	418
40.08	35.15	425.22	373	425.22	373

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

ue_dollar_a...	# gross_cost_dollar_amo...	# net_cost_dollar_amount	# gross_salary_employee...	# net_salary_employee_...	# gross_profit_margin_p...
licates ate values	Count Unique Distinct Null	Count Unique Distinct Null	Count Unique Distinct Null	Count Unique Distinct Null	Count Unique Distinct Null
	373.89	309	373.89	309	94.5
	559.74	491	559.74	491	65.39
	543.66	442	543.66	442	85.77
	497.42	418	497.42	418	77.71
	425.22	373	425.22	373	-961.05

raw data.xlsx (Лист1) Version 1 (8/17/2022 11:59 AM) Data quality Share

Overview Profile Attributes

Data Attributes 17 Filter attributes

Name	Term
Business_Fact_ID	
TIME_ID	
customer_ID	Enum
employee_ID	
agency_ID	
gen_period_ID	
product_ID	Enum
promotion_ID	
gross_profit_dollar_amount	
net_profit_dollar_amount	

As we can notice, there are no null values in the columns of our FACT TABLE. The CL layer checks the data for duplicates and null values and then distributes them to the DW layer as unique records with an ID.