

XYZ Insurance Company

Project Report

*Submitted in partial fulfillment of
the requirements for the award of the degree of*

**Master's
in
Computing**

Submitted by

Manish VS (L00150394)

Under the guidance of
Angela Sweeney



lyit

**Institiúid Teicneolaíochta
Leitir Ceanainn
Letterkenny Institute
of Technology**

Department of Computing
LETTERKENNY INSTITUTE OF TECHNOLOGY

Abstract

This report documents a project on XYZ Insurance company which describes about data flow and building a data warehouse for enhancing their analytical capabilities. The report begins by introduction to Data warehouse and reviews the technical area involved. It moves on with design requirements, Implementation. Initially the data is cleaned and loaded into staging tables then loaded into dimension -tables. Policy fact is the fact table. Finally performance evaluation is done using insert, updates and insert Update files. The business queries are then answered using tableau visualisation for enhancing analytical capabilities

Contents

Abstract	i
	Page
1 Introduction	2
1.1 Aim	2
1.2 Technical Review	2
1.2.1 Extract, Transform and Load(ETL)	3
1.2.2 Data Warehouse	3
2 Design Requirements	4
2.1 Tools Used	4
2.1.1 Talend Open Studio	4
2.1.2 MySql Workbench	5
2.1.3 Tableau	5
3 Implementation	6
3.1 Database Schema Design	6
3.1.1 Staging	6
3.1.2 Data WareHouse	7
3.2 Metadata	7
3.2.1 Excel Files	8
3.2.2 Staging and Data Warehouse tables	8
3.3 Staging Tables	9
3.3.1 Customer Detail	9
3.3.2 Location Detail	10
3.3.3 Agent Detail	11
3.3.4 Transaction Detail	12
3.4 Data warehouse Loading	13
3.4.1 Calendar Dimension	13
3.4.2 Customer dimension loading(SCD Type 2)	14
3.4.3 Location dimension loading(SCD Type 1)	16
3.4.4 Agent dimension loading(SCD Type(0,2)	16
4 Business Rule	20
4.1 Business Transaction Rule	20
4.1.1 Transaction Loading	20
4.2 Fact Table Loading	21
4.3 Tableau	23

5	Performance Evaluation	25
5.1	Customer Detail	25
5.2	Location Detail	27
5.3	Agent Detail	27
5.4	Transaction Detail	29
5.5	Business Rule Detail	30
5.6	Calendar Detail	31
5.7	Policy Detail	32
5.8	Business Report	32
6	CONCLUSION	37
	References	38

List of Figures

1.1	Business Intelligence	3
1.2	ETL Process	3
2.1	Data model	4
3.1	Staging tables schema	6
3.2	Staging tables schema	7
3.3	Staging tables schema	8
3.4	Meta data	8
3.5	customer staging job	9
3.6	Mapping to output table	9
3.7	Java expression to replace	9
3.8	tDBoutput component settings	10
3.9	Location staging job	10
3.10	Location Mapping tables	10
3.11	java expression to replace(,)	11
3.12	Agent staging job	11
3.13	Agent Mapping tables	11
3.14	Inner joining agent and location tables	12
3.15	Transaction staging job	12
3.16	Transaction Mapping tables	13
3.17	Calendar dimension loading	13
3.18	Calendar Dimension Mapping	14
3.19	SCD Type 2	15
3.20	Customer_Update dimension loading	15
3.21	Customer_IU Dimension Loading	16
3.22	Location dimension loading	16
3.24	Location_IU dim loading	16
3.23	Location SCD Type 1	17
3.25	Agent dimension loading	18
3.26	SCD Type 0,2	18
3.27	Agent_Update dimension loading	19
3.28	Agent_IU Dimension Loading	19
4.1	Transaction Business Rule	20
4.2	Transaction Business rule Mappings	21
4.3	Business rule	21
4.4	Ready indicator	21
4.5	Load Policy Table	22
4.6	policy fact Mapping	22

4.7	Inner join	22
4.8	Inner Join	23
4.9	Inner Join	23
4.10	Tableau Implementation	24
5.1	Staging table after customer insert	25
5.2	Dimension table for customer insert	25
5.3	Staging table after customer update	26
5.4	Dimension table after customer update	26
5.5	Staging table after customer IU	26
5.6	Dimension table after customer IU	26
5.7	Staging table after location insert	27
5.8	Dimension table after location insert	27
5.9	Staging table after Agent insert	27
5.10	Dimension table after Agent insert	28
5.11	Staging table after agent IU	28
5.12	Dimension table after agent IU	28
5.13	Staging table after agent U	28
5.14	Dimension table after agent U	29
5.15	Staging table for Transaction	29
5.16	dimension table for Transaction	30
5.17	Business Rule	30
5.18	Calendar dimension table	31
5.19	Policy Fact table	32
5.20	Business Query 1	32
5.21	Tableau	33
5.22	Business Query 2	34
5.23	Tableau	34
5.24	Business Query 3	35
5.25	Tableau	35
5.26	Business Query 4	36
5.27	Business Query 5	36

Chapter 1

Introduction

1.1 Aim

The aim of this project is to extract, transform and load the data of the xyz insurance company. It uses talend and mysql for data warehousing and storing the schemas. It also demonstrates about different types of dimension tables such as slowly changing dimension(SCD 0,1,2). Business Intelligence tools like tableau are used for visualization which one of the aim of this project.

1.2 Technical Review

The term business intelligence was introduced in the early 90's in business and IT industries. Later, with development of analytics, business analytics was introduced to analyse data with basic statistical models. Later in 2000 Big data and analytics was introduced to analyse and compute large complex data sets which require advanced and unique data storage, management, analysis, and visualization technologies.

Data management and warehousing was employed during this phase where "data marts" and "ETL(Extraction, Transformation and loading)" were used for analysing enterprise specific data. Scorecards and dashboards were used for data visualization. "Database query, online analytical processing (OLAP), and reporting tools based on intuitive, but simple, graphics are used to explore important data characteristics.



Figure 1.1: Business Intelligence

Later web2.0 Web intelligence, web analytics, and the user-generated content collected through Web 2.0-based social and crowd-sourcing systems have ushered in a new and exciting era of web 2.0 research in the 2000s, centered on text and web analytics for unstructured web contents.

1.2.1 Extract, Transform and Load(ETL)

ETL is extraction, Loading and Transforming. ETL tools are used to extract data from various sources and load into single or multiple systems. The data often is very large and contain lots of errors, hence they are cleaned and transformed before loading.

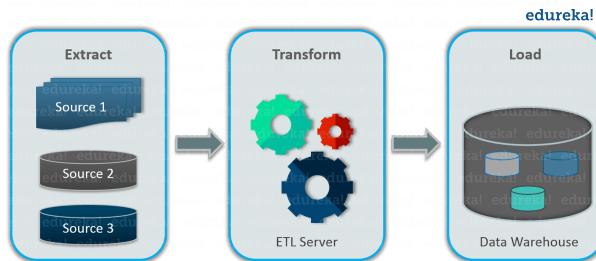


Figure 1.2: ETL Process

ETL is an important part of today's business intelligence (BI) processes and systems. It is the IT process from which data from disparate sources can be put in one place to programmatically analyze and discover business insights.

1.2.2 Data Warehouse

Data warehouse is a system where large data from different sources is collected and loaded into a single system for reporting and analysis. These analysis and reports are then further used for business decision making. It is integrated, Volatile and time variant. These systems use different design from standard operational databases.

Chapter 2

Design Requirements

The source files for this project were given. The entity relationship for different tables were given as project requirement in documents. The tables, schemas are also given.

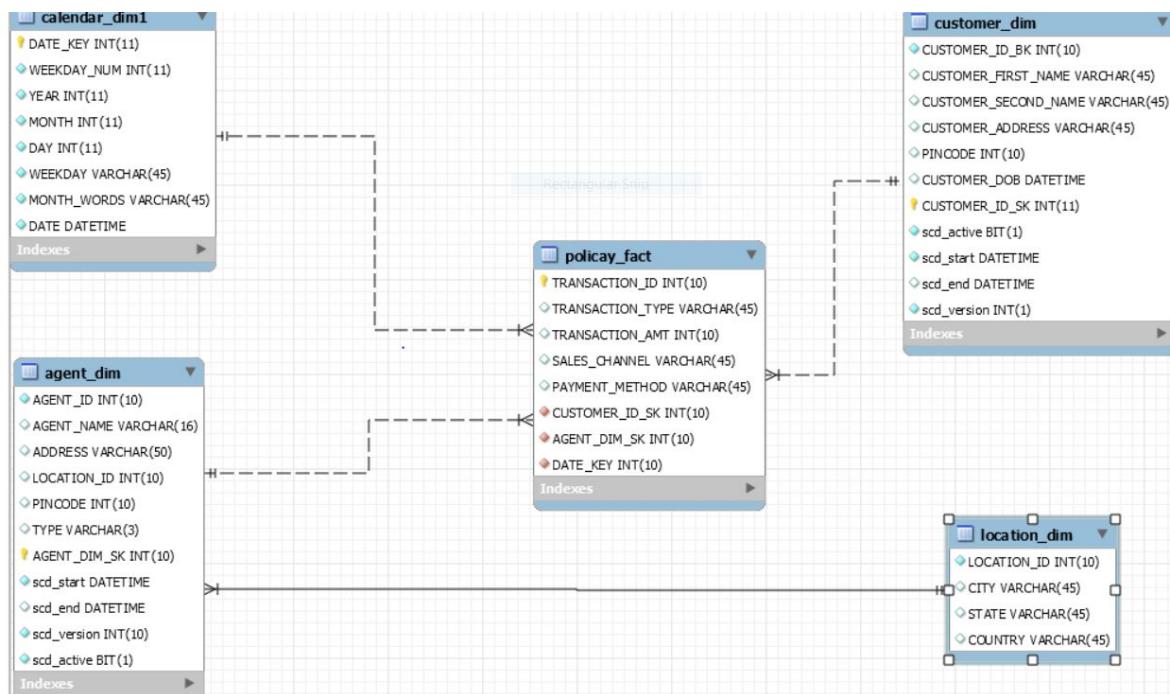


Figure 2.1: Data model

2.1 Tools Used

The tools used to complete this project are Talend open studio, Mysql and Tableau.

2.1.1 Talend Open Studio

Talend open studio is an open source data integration tool used for ETL process. It provides various software and services for data integration, data management, enterprise application integration, data quality, cloud storage and Big Data. It provides :

- 1.faster development and deployment.

- 2.Unified Platform.
- 3.Huge Community.

2.1.2 MySql Workbench

The database management is a repository used to store files. It allows users to perform DDL,DML operations such create,insert and update. For this project the database was chosen is SQL.MYSQL is open source relational database system maintained by the oracle. The reason for choosing mysql for this project is because of its simplicity. It provides functionalities like:

- 1.High Security.
- 2.On Demand Scalability.
- 3.High Performance.
- 4.Work Flow Control.

2.1.3 Tableau

The Data Visualisation in business intelligence industry tool used to know the business insights. Tableau provides faster data analysis and simpler visualisation in the form of dashboards, worksheets. For this project we have used Tableau desktop version. Tableau Desktop has a rich feature set and allows you to code and customize reports.

Chapter 3

Implementation

This section examines the working model of this project. The schemas required to work are demonstrated here.

3.1 Database Schema Design

The complete schemas for tables are created using MySQL workbench. The definitions, keys, number of columns are provided in the requirement document and created accordingly.

3.1.1 Staging

Staging area and data warehouse are defined in the different databases for better understanding of the project. For this project staging tables are created in the insurance database.

The staging tables required for this project are.

1. Customer staging table.
2. Location staging table.
3. Agent staging table.
4. Transaction staging table.

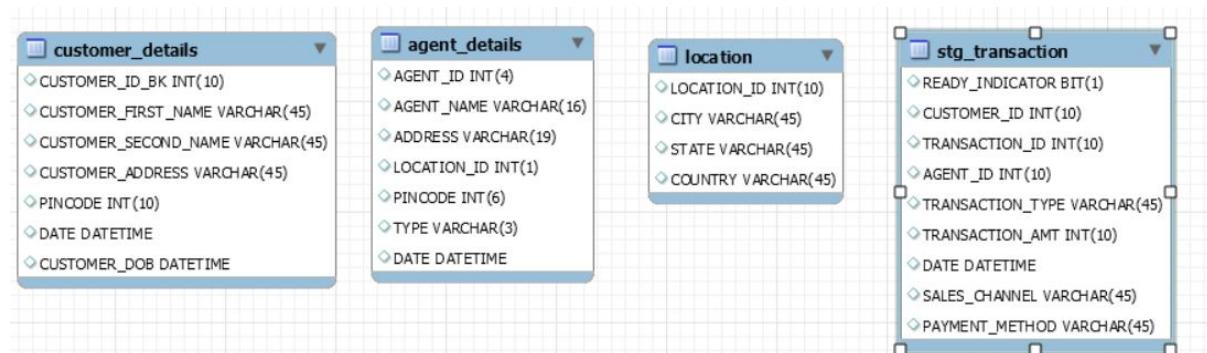


Figure 3.1: Staging tables schema

The staging area is performed to remove inconsistency in the data. There are no primary key or foreign keys defined during the staging in this project.

3.1.2 Data WareHouse

This area is used to define the dimension and fact tables for the project. Figure shows the data model of the project. After removing all the inconsistencies in the data, the data is shifted to dimension tables with appropriate foreign keys. The fact table was defined to have all surrogate keys to all tables to provide relationship between tables. The scd types are also defined here in dimension tables.

The dimension tables are:

1. Customer Dimension table.
2. Agent Dimension table.
3. Calendar Dimension table.
4. Location Dimension table.

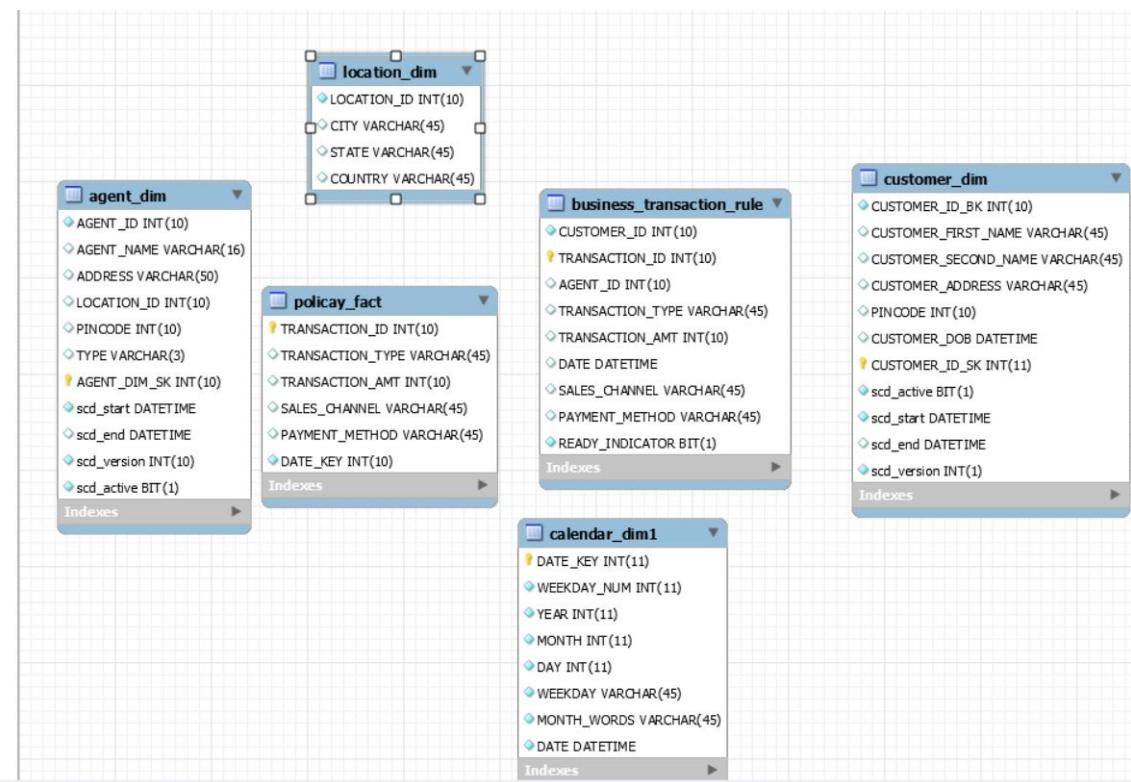


Figure 3.2: Staging tables schema

3.2 Metadata

Metadata is the data about the data. The metadata gives what, when, why and how about the data. In Talend, various sources like DB Connections, different kind of files, LDAP, Azure, Salesforce, Web Services, FTP, Hadoop Cluster and many more options are present under Talend Metadata. The main use of metadata in Talend Open Studio is that you can use these data sources in several jobs just by a simple drag and drop from the Metadata in repository panel.

3.2.1 Excel Files

Using talend excel files can be inserted into the repository for running jobs. The metadata is defined for every excel file. This includes insert, update and update/insert.

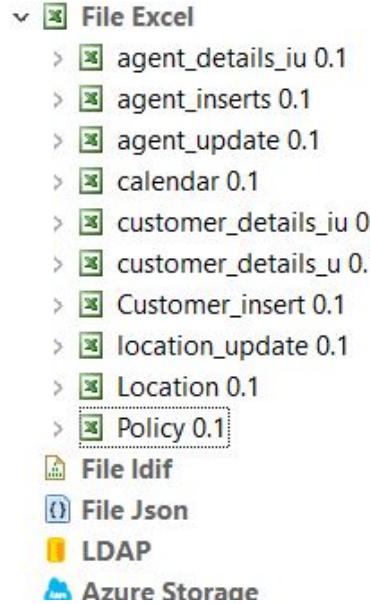


Figure 3.3: Staging tables schema

3.2.2 Staging and Data Warehouse tables

Metadata is created for every staging and dimension table, see figure 3.5.

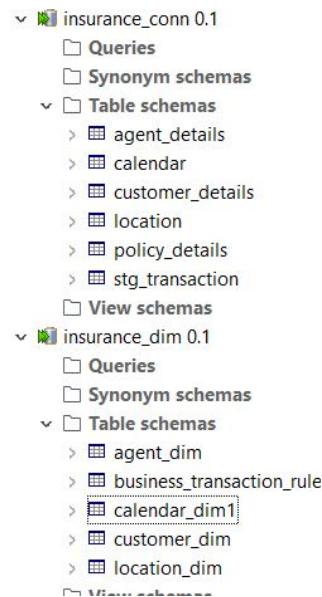


Figure 3.4: Meta data

3.3 Staging Tables

Staging is done using talend open studio. This stage is performed before loading into data warehouse. All the inconsistencies are removed in staging.

The excel files are loaded into the repository and metadata is defined for each excel file. Staging jobs are created in the talend and its respective tables are created in the MySQL database. The MySQL connection is given to the talend jobs using dbconnection in repository.

3.3.1 Customer Detail

JOB: STG CUSTOMER DETAIL LOAD The job uses tInputExcelFile to load the customer excel file and maps it using tMap component. It then writes the data to the STG Customer Detail table in MySQL. The replace() is used in the tMap to remove the unwanted characters from the file i.e cleaning of the file, refer fig.

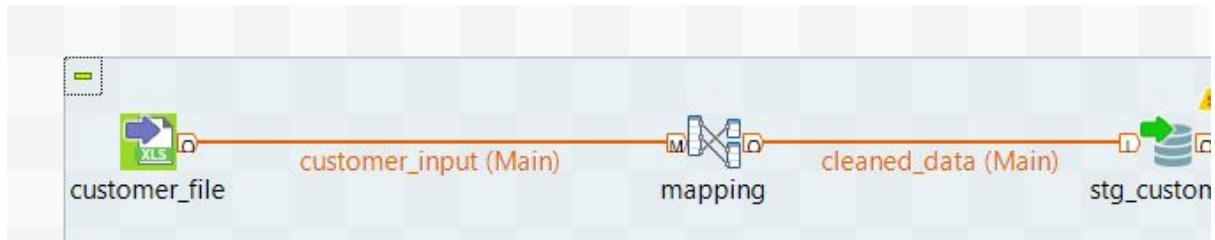


Figure 3.5: customer staging job

The image below refers to the mapping of input file to output table with java expressions to remove inconsistencies.

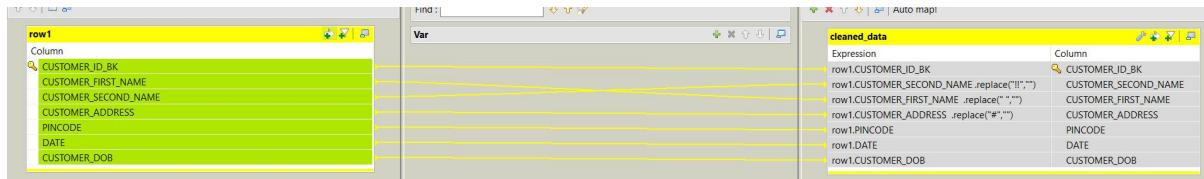


Figure 3.6: Mapping to output table

```
cust_input.CUSTOMER_SECOND_NAME.replace("!!...  
cust_input.CUSTOMER_FIRST_NAME.replace(" ", "")...  
cust_input.CUSTOMER_ADDRESS.replace("#", "")
```

Figure 3.7: Java expression to replace

The customer insert file must be loaded into stg_customer table using tdbouput component. The component should be configured, refer the figure below. The action on table must be insert. The database connection is given to load the data into insurance database.

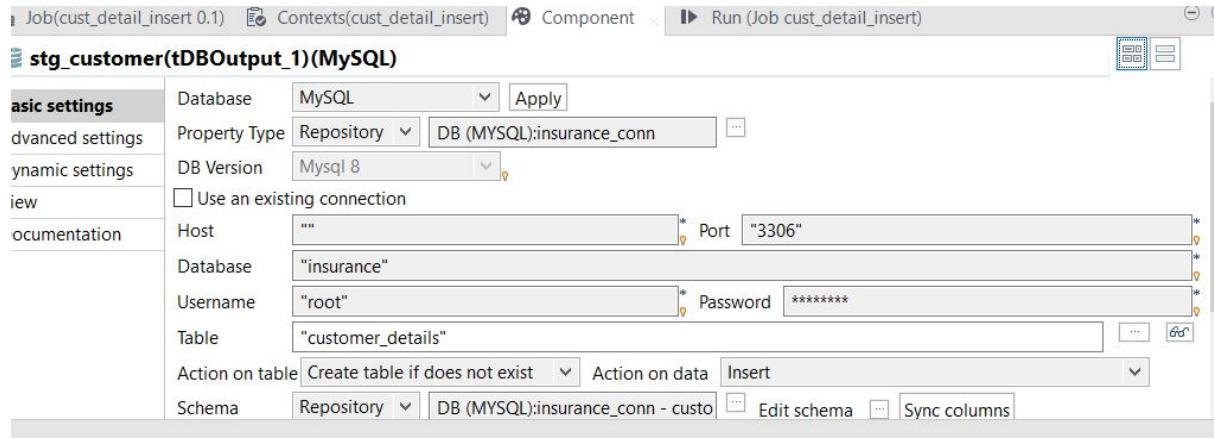


Figure 3.8: tDBOutput component settings

3.3.2 Location Detail

JOB:STG LOCATION DETAIL LOAD The job uses tInputExcelFile to load the location excel file and maps it using tMap component. It then writes the data to the STG Customer Detail table in MySQL. The replace() is used in the tMap to remove the unwanted characters from the file i.e cleaning of the file, refer fig.

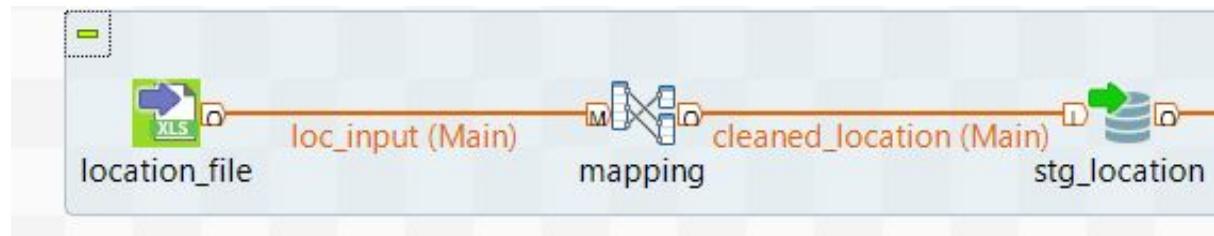


Figure 3.9: Location staging job

The image below refers to the mapping of input file to output table with java expressions to remove inconsistencies.

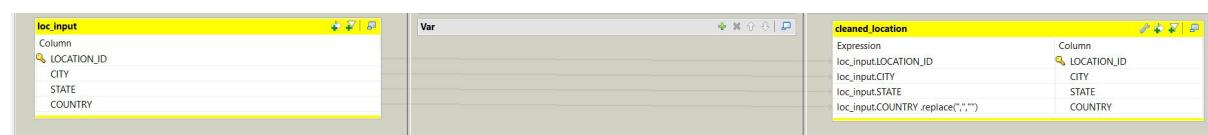


Figure 3.10: Location Mapping tables

The screenshot shows a Java code editor window. The code in the editor is:

```
loc_input.COUNTRY.replace(",","")
```

Figure 3.11: java expression to replace(,)

3.3.3 Agent Detail

JOB:STG Agent DETAIL LOAD The job uses tInputExcelFile to load the agent excel file and maps it using tMap component. It then writes the data to the STG Agent Detail table in MySQL. The file doesn't contain any errors. Though the tMap is used to maintain consistency between tables. The location id of the stg location table is made inner join with agent detail which is used for future purpose in dimensional loading.

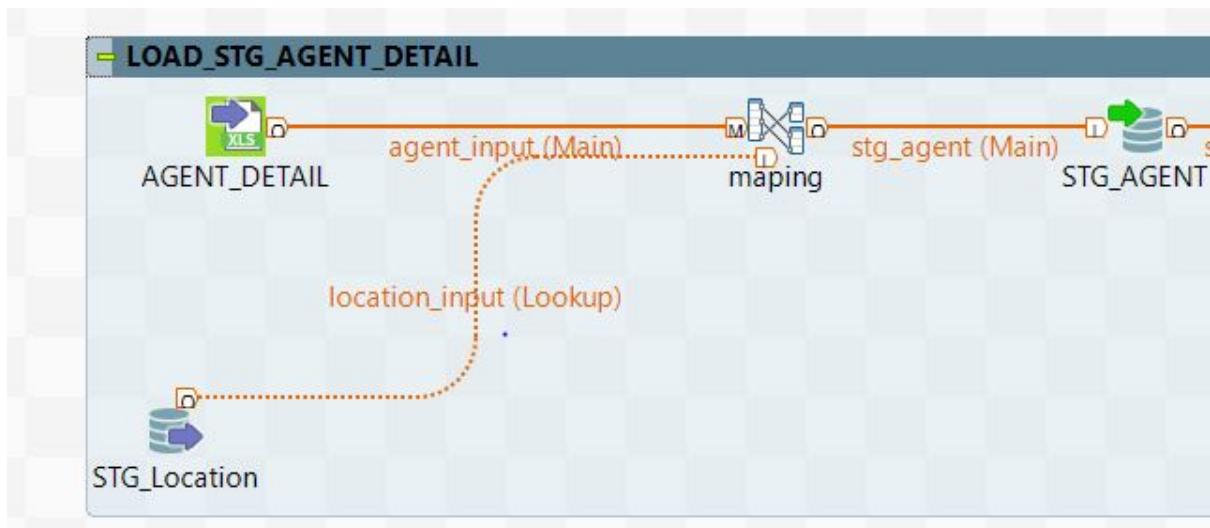


Figure 3.12: Agent staging job

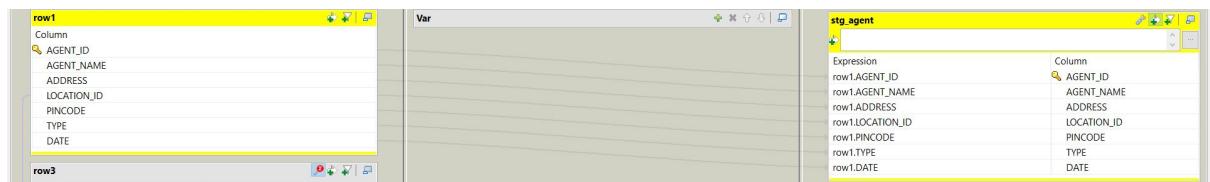


Figure 3.13: Agent Mapping tables

The agent _id of the agent table is mapped with location staging with inner join so that location_id becomes key in agent dimension which is used for future purpose where few business queries are answered regarding agent.

Figure 3.14: Inner joining agent and location tables

3.3.4 Transaction Detail

JOB:STG Transaction DETAIL LOAD The job uses tInputExcelFile to load the Transaction excel file and maps it using tMap component. It then writes the data to the STG Transaction Detail table in MySQL. The file doesn't contain any errors. Though the tMap is used to maintain consistency between tables. The STG Customer and STG Agent are linked to tMap to inner join them with STG Transaction which is used for future purpose.



Figure 3.15: Transaction staging job

The below figure refers to the mapping of input file to output table. The mapping is almost same like every job. The ready indicator is added into transaction staging table which is used for future purpose.

stg_input	stg_transaction
Column	Column
CUSTOMER_ID	READY_INDICATOR
TRANSACTION_ID	TRANSACTION_ID
AGENT_ID	AGENT_ID
TRANSACTION_TYPE	TRANSACTION_TYPE
TRANSACTION_AMT	TRANSACTION_AMT
DATE	DATE
SALES_CHANNEL	SALES_CHANNEL
PAYMENT_METHOD	PAYMENT_METHOD

Figure 3.16: Transaction Mapping tables

3.4 Data warehouse Loading

Once done with cleaning in staging tables, the data is moved into dimension and fact tables. Separate database is created to load dimension and fact tables

3.4.1 Calendar Dimension

JOB:LOAD DIM CALENDAR The job uses tInputExcelFile to load the Calendar excel file and maps it using tMap component. It then writes the data to the Calendar dimension table in MySQL. There is no need of using scd for calendar table. An extra date key is added as primary key for calendar table while loading into Calendar dimension.

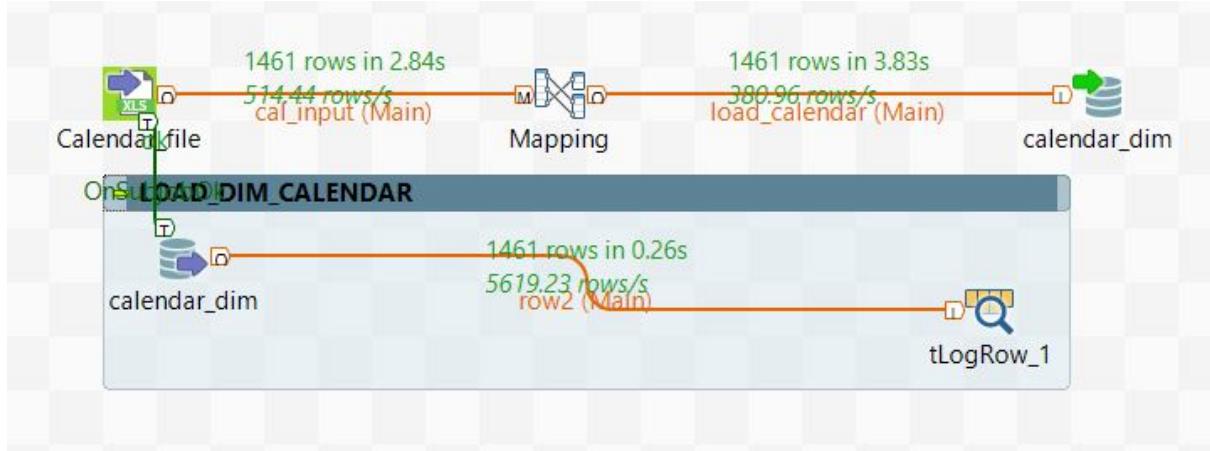


Figure 3.17: Calendar dimension loading

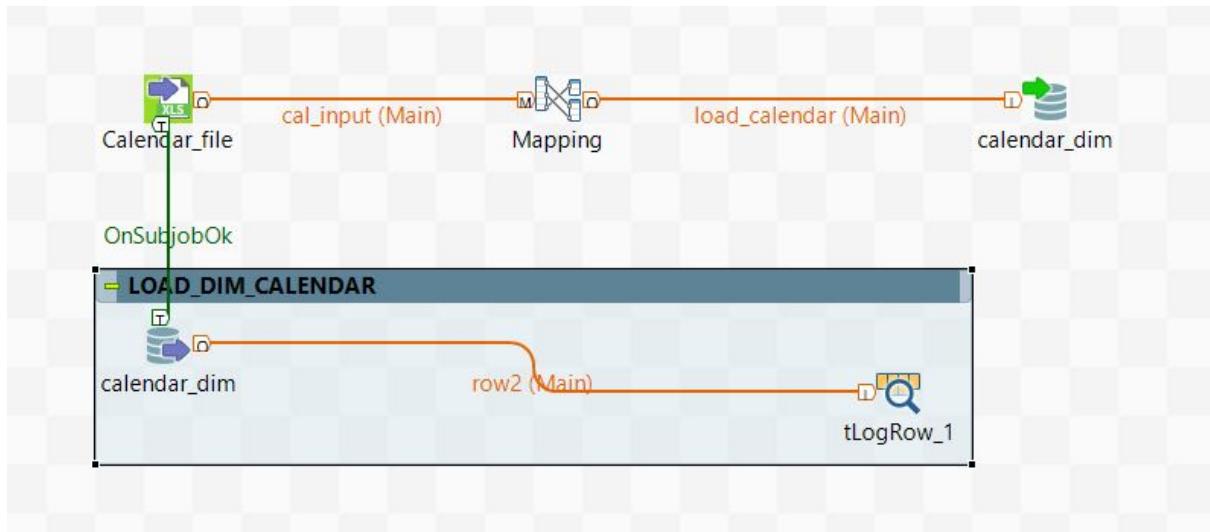


Figure 3.18: Calendar Dimension Mapping

3.4.2 Customer dimension loading(SCD Type 2)

JOB:LOAD DIM CUSTOMER DETAIL

The data in the staging table stg_customer is loaded into dimension table using tDB-SCD. The tBSCD is added to the Load stage customer detail job. Customer id is made primary key here. This is SCD Type 2(CUSTOMER FIRST NAME, CUSTOMER LAST NAME, CUSTOMER ADDRESS, CUSTOMER DOB, PINCODE).

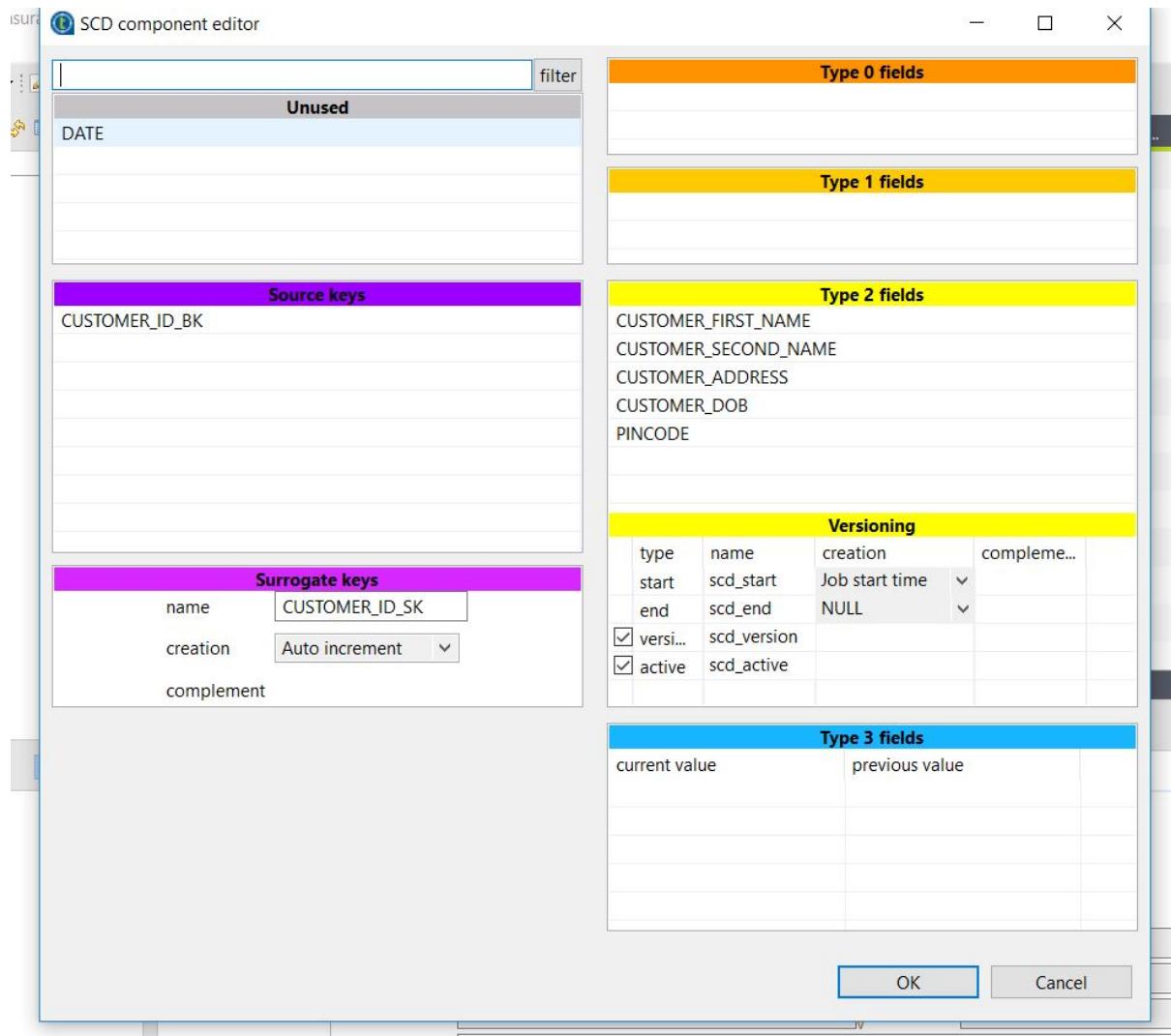


Figure 3.19: SCD Type 2

similarly same job is created for both update and insert update jobs. When update job is performed action on table must be update in component settings of tDBOutPut. if insert update file is being loaded into dimension table,action on table should be insert or update.

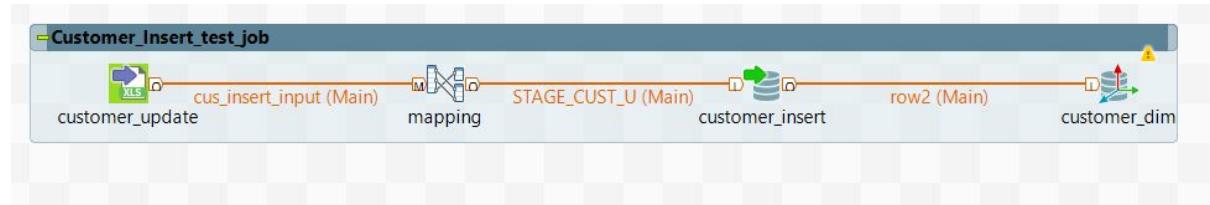


Figure 3.20: Customer_Update dimension loading

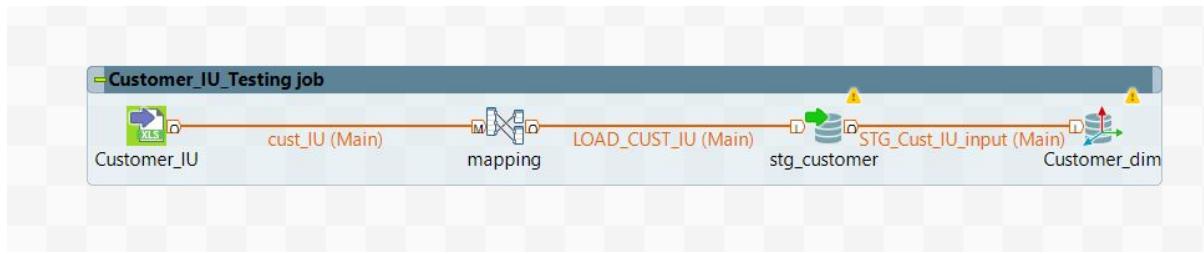


Figure 3.21: Customer_IU Dimension Loading

3.4.3 Location dimension loading(SCD Type 1)

JOB:LOAD DIM Location DETAIL The data in the staging table stg_Location is loaded into dimension table using tDBSCD. The tBSCD is added to the Load stage location detail job. This is Type 1 scd. No surrogate key and no history is maintained.

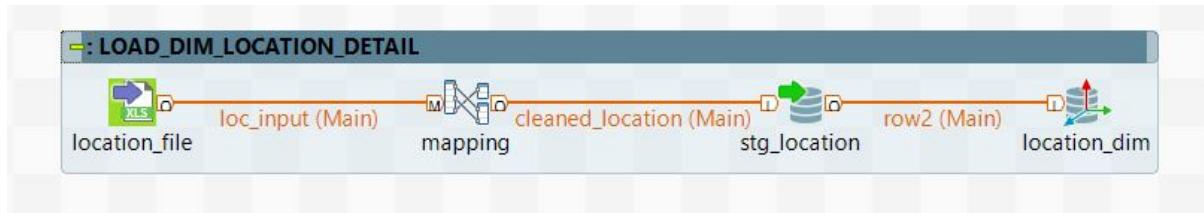


Figure 3.22: Location dimension loading

similarly same job is created for both insert and insert update jobs. When update job is performed action on table must be insert in component settings of tDBOutPut. if insert update file is being loaded into dimension table, action on table should be insert or update.

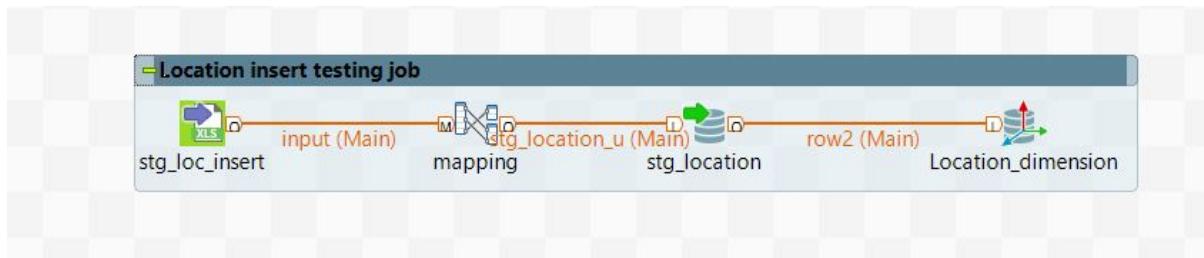


Figure 3.24: Location_IU dim loading

3.4.4 Agent dimension loading(SCD Type(0,2))

JOB:LOAD DIM Agent DETAIL The data in the staging table stg_agent is loaded into dimension table using tDBSCD. The tBSCD is added to the Load stage agent detail job. This is Type 2 and o scd. Location id is type O where as TYPE 2 (ADDRESS, AGENT NAME, PIN CODE, TYPE). A Agent DIM SK surrogate key is generated.

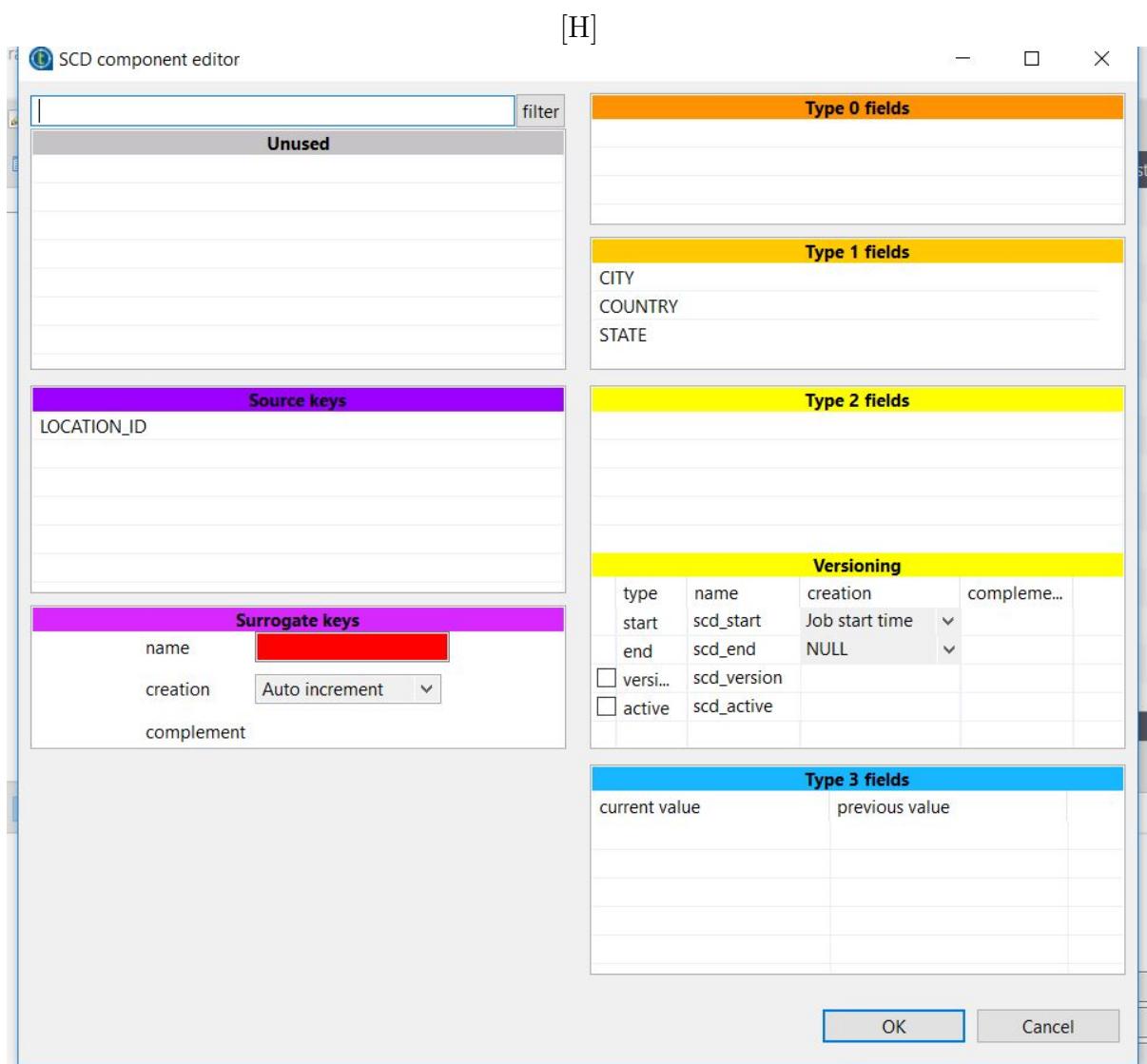


Figure 3.23: Location SCD Type 1

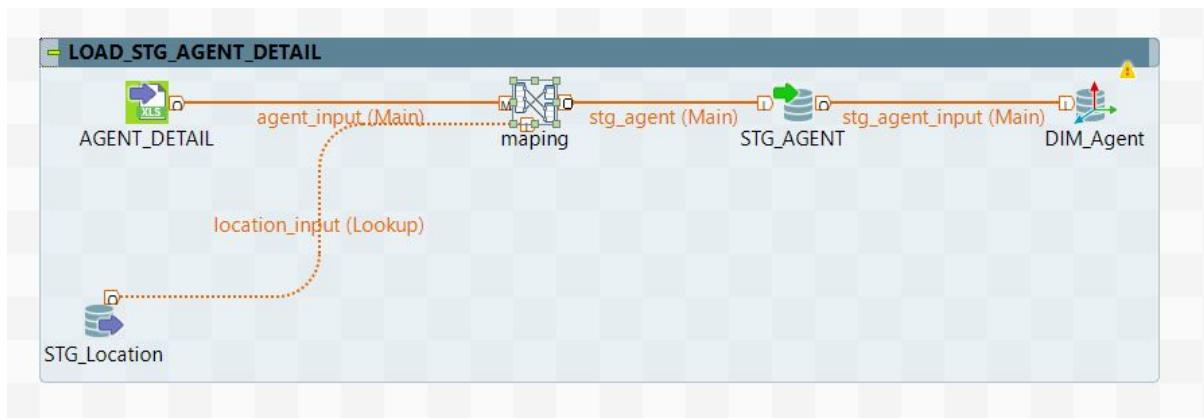


Figure 3.25: Agent dimension loading

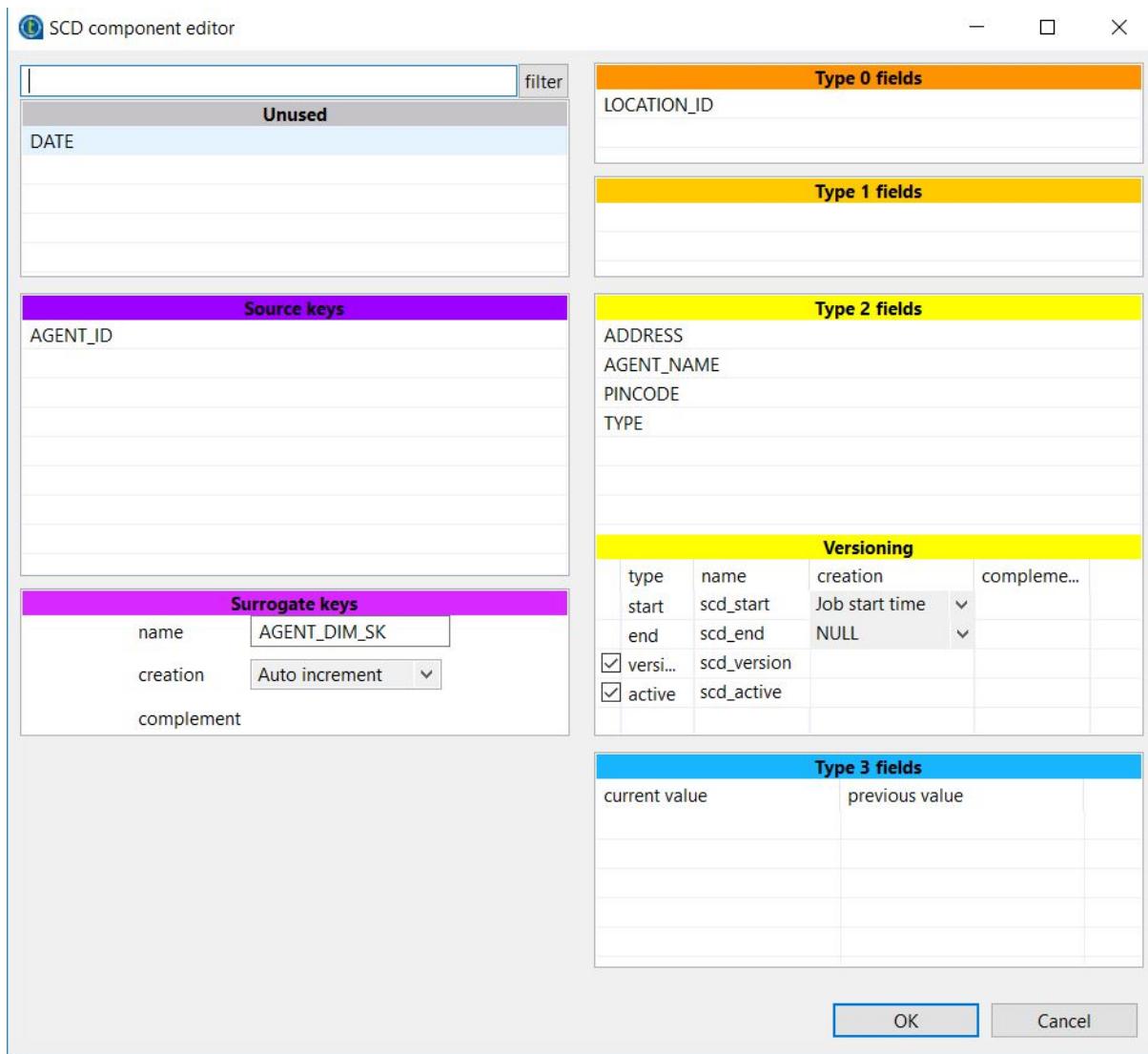


Figure 3.26: SCD Type 0,2

similarly same job is created for both update and insert update jobs. When update job

is performed action on table must be update in component settings of tDBOutPut.if insert update file is being loaded into dimension table,action on table should be insert or update.

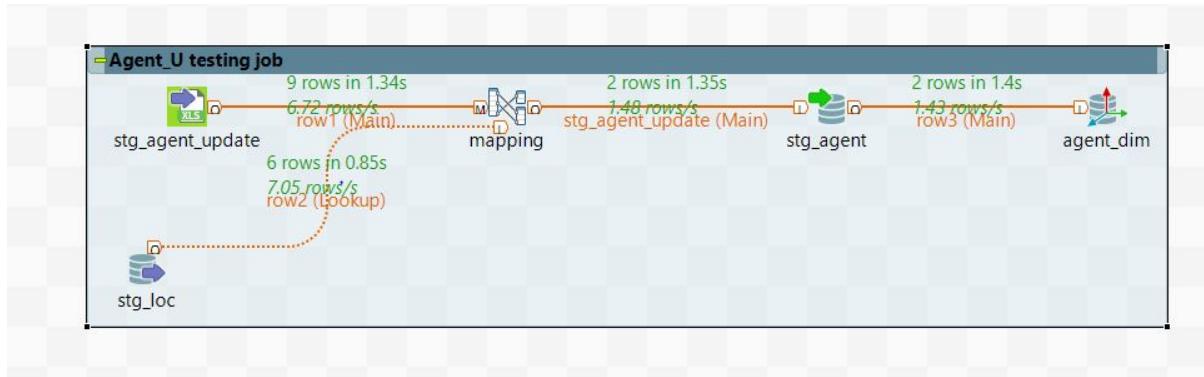


Figure 3.27: Agent_Update dimension loading

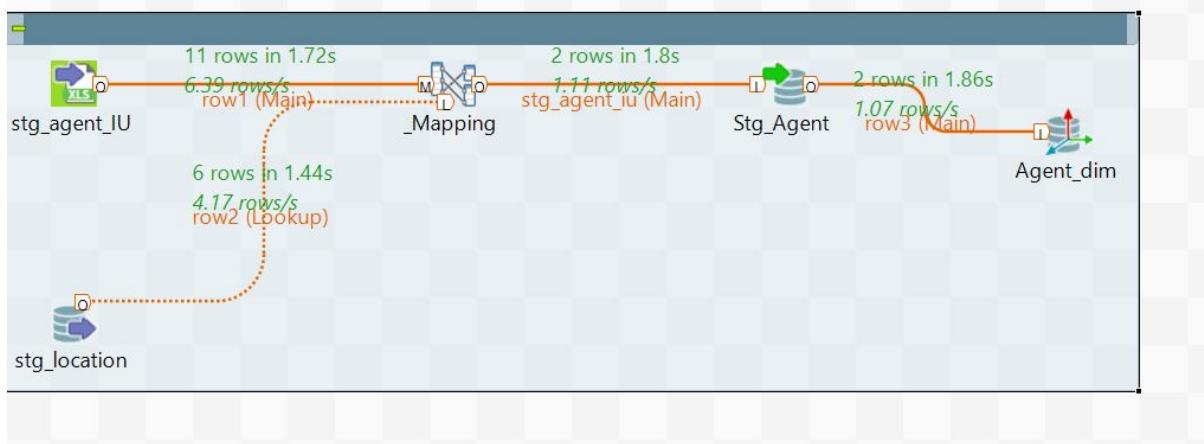


Figure 3.28: Agent_IU Dimension Loading

Chapter 4

Business Rule

The xyz company wants to know the performance of each agent. A Business rule is implemented to analyse the performance of each agent.

4.1 Business Transaction Rule

4.1.1 Transaction Loading

JOB:STG TRANSACTION BUSINESS RULE

The STG transaction table is loaded into dimension table using business rule.e, XYZ has given the requirement stating that only those customer id's and agent id's which are present in the customer dim and agent dim respectively must be populated into the fact table. If customer id coming from the transaction file is not present in customer dim then do not populate this id. Similarly, if the agent id coming from the transaction file is not present in AGENT dim, then it is not populated.

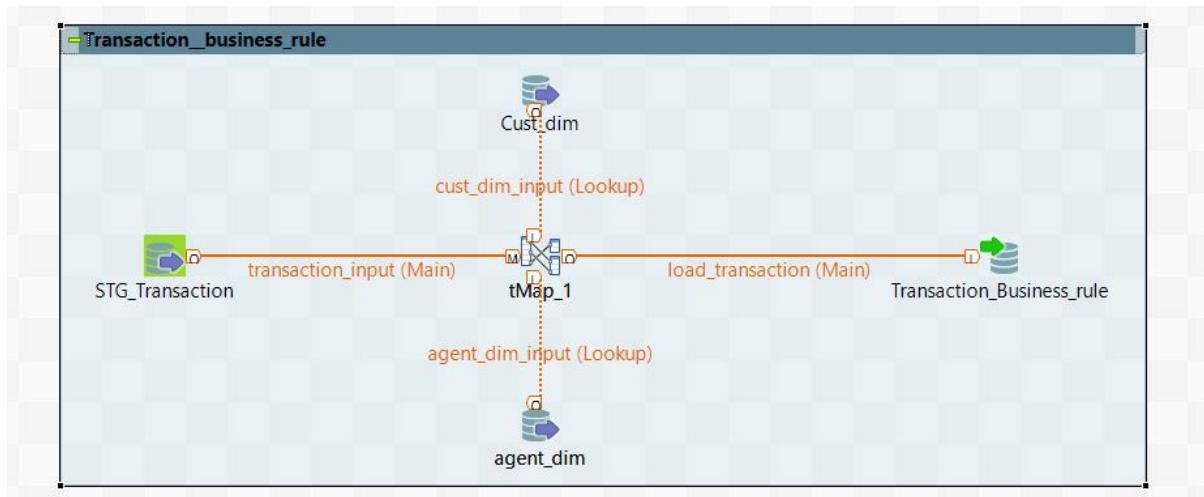


Figure 4.1: Transaction Business Rule

The STG Transaction job is added with tboutput component to load it into transaction dimension table. The inputs are from customer dim and agent dim Left Outer join with

STG Transaction table so that customer id from customer dimension table and agent id from agent dimension table is mapped with business logic(Look up on customer dim on customer id, if the record is not present then the record is rejected If the record is present in customer dim, lookup agent dim on agent id. If present update the Ready Indicator to “true” else the record is rejected.

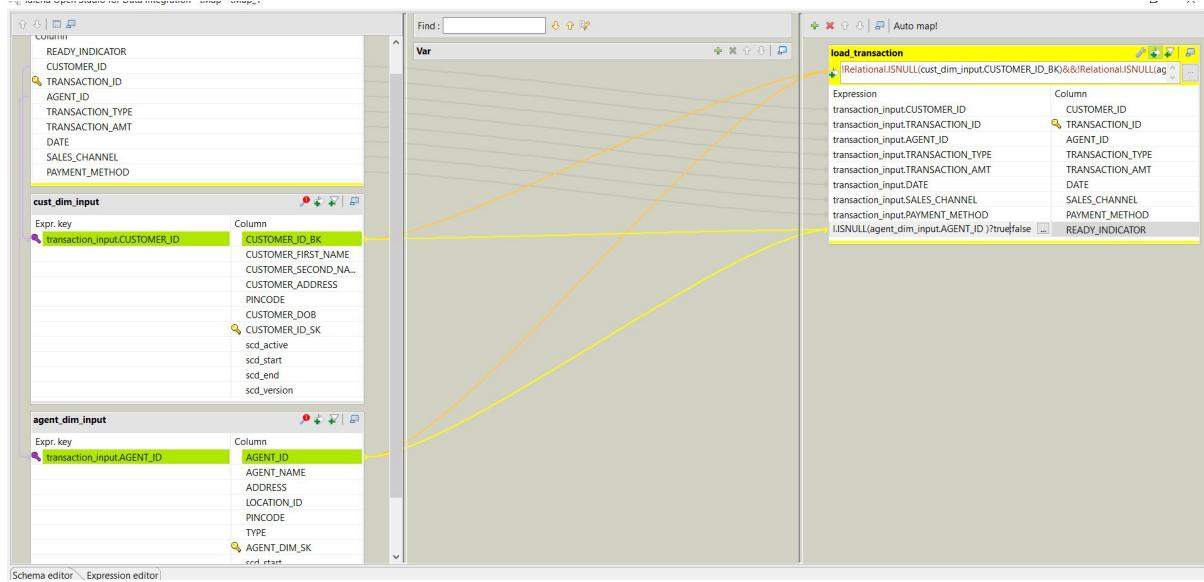


Figure 4.2: Transaction Business rule Mappings

Business Rule Logic: Look up on customer dim on customer id, if the record is not present then the record is rejected.If the record is present in customer_dim, lookup agent_dim on agent id. If present update the Ready Indicator to “true” else the record is rejected.

[!Relational.ISNULL\(cust_dim_input.CUSTOMER_ID_BK\)&&!Relational.ISNULL\(agent_dim_input.AGENT_ID \)](#)

Figure 4.3: Business rule

Ready_Indicator:



Figure 4.4: Ready indicator

4.2 Fact Table Loading

The policy fact table of XYZ insurance company is loaded into policy_fact table,see figure.

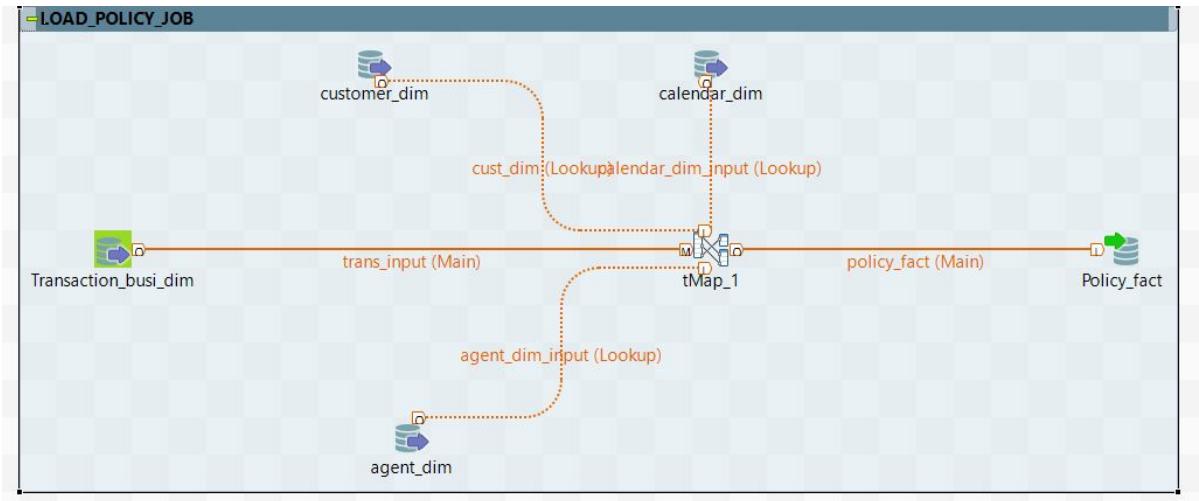


Figure 4.5: Load Policy Table

The STG_Transaction is loaded into Policy Fact dimension using tMap, the customer_dimension,agent_calendar_dimension are mapped with stg transaction with inner join to fact table.



Figure 4.6: policy fact Mapping

The customer_id from customer_dimension table, date_key from calendar table is inner joined with the transaction_id of transaction dimension table to make them foreign keys in policy fact.

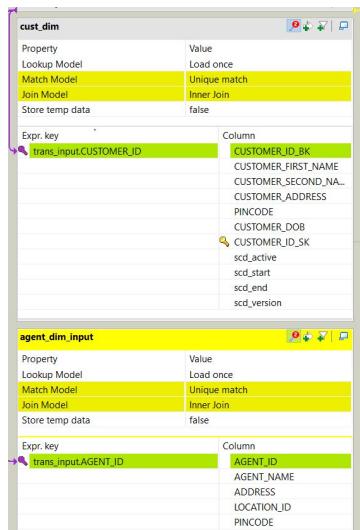


Figure 4.7: Inner join

The transaction _id from transaction table is mapped with agent_id from agent dimension table. The two tables are inner joined.

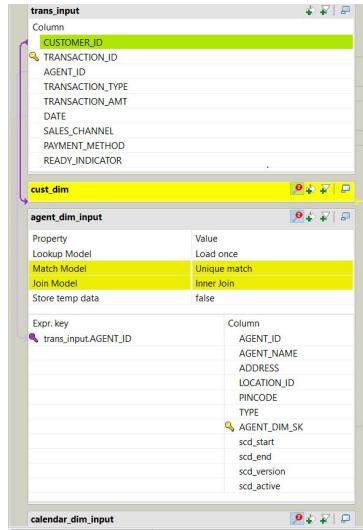


Figure 4.8: Inner Join

The transaction _id from transaction table is mapped with date_key from calendar dimension table. The two tables are inner joined.

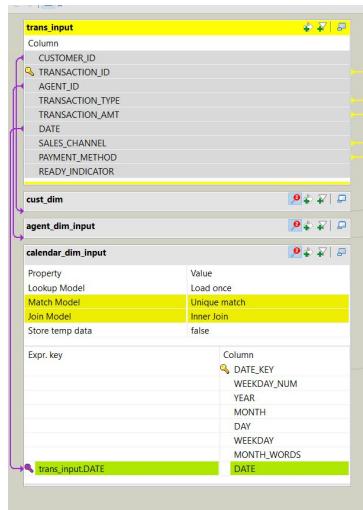


Figure 4.9: Inner Join

4.3 Tableau

After the completion of data warehouse, instance of tableau is created. The XYZ insurance company uses this to draw business insights.

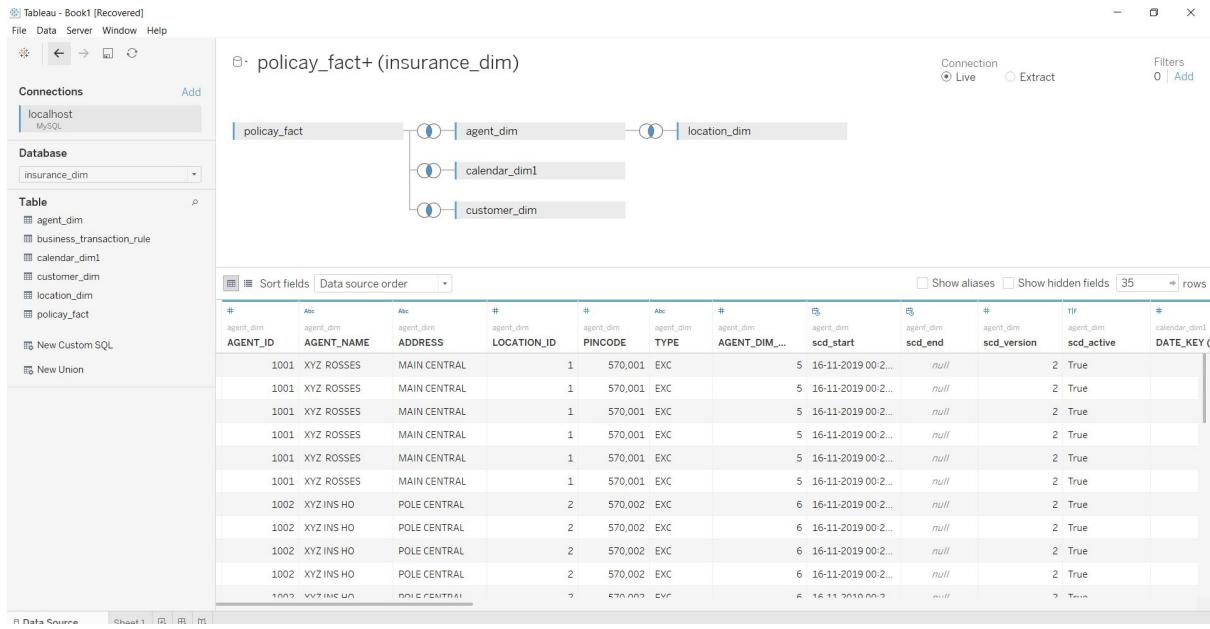


Figure 4.10: Tableau Implementation

Chapter 5

Performance Evaluation

The requirement document is supplied with test cases i.e inserts, updates and insert updates to be loaded into dimension tables using slowly changing dimensions. It is checked the functionality of the jobs. Later the business queries are answered using SQL queries and visualised using tableau which is another option for testing the performance of the jobs.

5.1 Customer Detail

CUSTOMER_ID_BK	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME	CUSTOMER_ADDRESS	PINCODE	DATE	CUSTOMER_DOB
12345 FIRNAME	A		123,1ST MAIN, 2ND CROSS	520083	01/01/2010 0:00	04/04/1977 0:00
678910 FNAME	B		342,2ND MAIN, 15TH CROSS	520084	02/01/2011 0:00	01/01/1971 0:00
11121314 FIRSAME	C		32, 1ST MAIN, 1ST CROSS	520085	03/01/2010 0:00	01/01/1979 0:00
15161718 FAME	D		151, 15TH MAIN, 39TH CROSS	520086	04/01/2013 0:00	01/01/1978 0:00
19202122 FIRSTAME	E		155, 2ND CROSS,11TH AVENUE	520087	04/05/2012 0:00	01/01/1984 0:00

Figure 5.1: Staging table after customer insert

CUSTOMER_ID_BK	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME	CUSTOMER_ADDRESS	PINCODE	CUSTOMER_DOB	CUSTOMER_ID_SCD	scd_start	scd_end	scd_version	scd_active
12345 FIRNAME	A		123,1ST MAIN, 2ND CROSS	520083	04/04/1977 0:00	1	17/11/2019 23:38		1	1
678910 FNAME	B		342,2ND MAIN, 15TH CROSS	520084	01/01/1971 0:00	2	17/11/2019 23:38		1	1
11121314 FIRSAME	C		32, 1ST MAIN, 1ST CROSS	520085	01/01/1979 0:00	3	17/11/2019 23:38		1	1
15161718 FAME	D		151, 15TH MAIN, 39TH CROSS	520086	01/01/1978 0:00	4	17/11/2019 23:38		1	1
19202122 FIRSTAME	E		155, 2ND CROSS,11TH AVENUE	520087	01/01/1984 0:00	5	17/11/2019 23:38		1	1

Figure 5.2: Dimension table for customer insert

CUSTOMER_ID_BK	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME	CUSTOMER_ADDRESS	PINCODE	DATE	CUSTOMER_DOB
12345 FIRNAMES	AB	123,1ST MAIN, 2ND CROSS	520099	02/02/2012 0:00	04/04/1977 0:00	
678910 FNAME	B	342,2ND MAIN, 15TH CROSS	520084	02/01/2011 0:00	01/01/1971 0:00	
11121314 FIRSAME	C	32, 1ST MAIN, 1ST CROSS	520085	03/01/2010 0:00	01/01/1979 0:00	
15161718 FAME	D	151, 15TH MAIN, 39TH CROSS	520086	04/01/2013 0:00	01/01/1978 0:00	
19202122 FIRSTAME	EC	155, 2ND CROSS,11TH AVENUE	520087	05/05/2012 0:00	01/01/1984 0:00	

Figure 5.3: Staging table after customer update

A	B	C	D	E	F	G	H	I	J	K
CUSTOMER_ID_BK	CUSTOMER_FIRST_NAM	CUSTOMER_LAST_NAM	CUSTOMER_ADDRESS	PINCODE	CUSTOMER_DOB	CUSTOMER_ID_SK	scd_start	scd_end	scd_ver	scd_acti
12345 FIRNAME	A	123,1ST MAIN, 2ND CROSS	520083	04/04/1977 0:00	1 17/11/2019 23:38 17/11/2019 23:43	1	0			
678910 FNAME	B	342,2ND MAIN, 15TH CROSS	520084	01/01/1971 0:00	2 17/11/2019 23:38	1	1			
11121314 FIRSAME	C	32, 1ST MAIN, 1ST CROSS	520085	01/01/1979 0:00	3 17/11/2019 23:38	1	1			
15161718 FAME	D	151, 15TH MAIN, 39TH CROSS	520086	01/01/1978 0:00	4 17/11/2019 23:38	1	1			
19202122 FIRSTAME	E	155, 2ND CROSS,11TH AVENU	520087	01/01/1984 0:00	5 17/11/2019 23:38 17/11/2019 23:43	1	0			
12345 FIRNAMES	AB	123,1ST MAIN, 2ND CROSS	520099	04/04/1977 0:00	6 17/11/2019 23:43	2	1			
19202122 FIRSTAME	EC	155, 2ND CROSS,11TH AVENU	520087	01/01/1984 0:00	7 17/11/2019 23:43	2	1			

Figure 5.4: Dimension table after customer update

A	B	C	D	E	F	G
CUSTOMER_ID_BK	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME	CUSTOMER_ADDRESS	PINCODE	DATE	CUSTOMER_DOB
12345 FIRNAMES	AB	123,1ST MAIN, 2ND CROSS	520099	02/02/2012 0:00	04/04/1977 0:00	
678910 FNAME	B	342,2ND MAIN, 15TH CROSS	520084	02/01/2011 0:00	01/01/1971 0:00	
11121314 FIRSAME	C	32, 1ST MAIN, 1ST CROSS	520085	03/01/2010 0:00	01/01/1979 0:00	
15161718 FAME	D	151, 15TH MAIN, 39TH CROSS	520086	04/01/2013 0:00	01/01/1978 0:00	
19202122 FIRSTAME	E	155, 2ND CROSS,11TH AVEN	520087	05/05/2012 0:00	01/01/1984 0:00	
23242526 FNAMES	AM	155, 2ND CROSS,11TH AVENUE	520087	05/05/2012 0:00	01/01/1971 0:00	

Figure 5.5: Staging table after customer IU

CUSTOMER_ID_BI	CUSTOMER_FIRST_NAM	CUSTOMER_LAST_NAM	CUSTOMER_ADDRESS	PINCODE	CUSTOMER_DOB	CUSTOMER_ID_SK	scd_start	scd_end	scd_ver	scd_acti
12345 FIRNAME	A	123,1ST MAIN, 2ND CROSS	520083	04/04/1977 0:00	1 17/11/2019 23:38 17/11/2019 23:43	1	0			
678910 FNAME	B	342,2ND MAIN, 15TH CROSS	520084	01/01/1971 0:00	2 17/11/2019 23:38	1	1			
11121314 FIRSAME	C	32, 1ST MAIN, 1ST CROSS	520085	01/01/1979 0:00	3 17/11/2019 23:38	1	1			
15161718 FAME	D	151, 15TH MAIN, 39TH CROS	520086	01/01/1978 0:00	4 17/11/2019 23:38	1	1			
19202122 FIRSTAME	E	155, 2ND CROSS,11TH AVEN	520087	01/01/1984 0:00	5 17/11/2019 23:38 17/11/2019 23:43	1	0			
12345 FIRNAMES	AB	123,1ST MAIN, 2ND CROSS	520099	04/04/1977 0:00	6 17/11/2019 23:43	2	1			
19202122 FIRSTAME	EC	155, 2ND CROSS,11TH AVEN	520087	01/01/1984 0:00	7 17/11/2019 23:43	2	1			
23242526 FNAMES	AD	123,1ST MAIN, 2ND CROSS	520099	01/01/1971 0:00	8 17/11/2019 23:48 17/11/2019 23:48	1	0			
23242526 FNAMES	AM	155, 2ND CROSS,11TH AVEN	520087	01/01/1971 0:00	9 17/11/2019 23:48	2	1			

Figure 5.6: Dimension table after customer IU

5.2 Location Detail

LOCATION_ID	CITY	STATE	COUNTRY
1	GLENIES	DONEGAL	IRELAND
2	CLIFFONY	SLIGO	IRELAND
3	LISTOWEL	KERRY	IRELAND
4	CLONES	MONAGHAN	IRELAND
5	LONDON	NA	UNITED KINGDOM

Figure 5.7: Staging table after location insert

LOCATION_ID	CITY	STATE	COUNTRY
1	GLENIES	DONEGAL	IRELAND
2	CLIFFONY	SLIGO	IRELAND
3	LISTOWEL	KERRY	IRELAND
4	CLONES	MONAGHAN	IRELAND
5	LONDON	NA	UNITED KINGDOM
6	TEST	TEST	TEST

Figure 5.8: Dimension table after location insert

5.3 Agent Detail

AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	DATE
1001	XYZ ROSES	MAIN CENTRAL	1	570001	EXC	01/01/1999 0:00
1002	XYZ INS HO	POLE CENTRAL	2	570002	EXC	02/02/1999 0:00
1003	XYZ LISTOWEL	MAIN STREET CENTRAL	3	570003	EXC	02/02/1999 0:00
1004	HEGARTY HO	POLE STREET CENTRAL	4	570004	EXC	01/01/1999 0:00
1005	LONDON HO	LONDON CENTRAL	5	570005	EXC	10/10/1999 0:00
1006	XYZ AREA 1	#92, TALL TOWERS	1	570006	IND	01/01/1999 0:00
1007	XYZ AREA 1	#8, BANK ARCADE	2	570007	IND	02/02/1999 0:00
1008	HEGARTY AREA 1	#51, 22 BUILDING	3	570008	IND	02/02/1999 0:00
1009	XYZ AGENT AREA 1	#42, MAINSTREET	4	570009	IND	01/01/1999 0:00

Figure 5.9: Staging table after Agent insert

AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	AGENT_ID_SK	scd_start	scd_end	scd_version	scd_active
1001 XYZ ROSSES	MAIN CENTRAL		1	570001	EXC	1	18/11/2019 0:28		1	1
1002 XYZ INS HO	POLE CENTRAL		2	570002	EXC	2	18/11/2019 0:28		1	1
1003 XYZ LISTOWEL	MAIN STREET CENTRAL		3	570003	EXC	3	18/11/2019 0:28		1	1
1004 HEGARTY HO	POLE STREET CENTRAL		4	570004	EXC	4	18/11/2019 0:28		1	1
1005 LONDON HO	LONDON CENTRAL		5	570005	EXC	5	18/11/2019 0:28		1	1
1006 XYZ AREA 1	#92, TALL TOWERS		1	570006	IND	6	18/11/2019 0:28		1	1
1007 XYZ AREA 1	#8, BANK ARCADE		2	570007	IND	7	18/11/2019 0:28		1	1
1008 HEGARTY AREA 1	#51, 22 BUILDING		3	570008	IND	8	18/11/2019 0:28		1	1
1009 XYZ AGENT AREA 1	#42, MAINSTREET		4	570009	IND	9	18/11/2019 0:28		1	1

Figure 5.10: Dimension table after Agent insert

AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	DATE
1001 XYZ ROSSES	MAIN CENTRAL		1	570001	EXC	01/01/1999 0:00
1002 XYZ INS HO	POLE CENTRAL		2	570002	EXC	02/02/1999 0:00
1003 XYZ LISTOWEL	MAIN STREET CENTRAL		3	570003	EXC	02/02/1999 0:00
1004 HEGARTY HO	POLE STREET CENTRAL		4	570004	EXC	01/01/1999 0:00
1005 LONDON HO	LONDON CENTRAL		5	570005	EXC	10/10/1999 0:00
1006 XYZ AREA 1	#92, TALL TOWERS		1	570006	IND	01/01/1999 0:00
1007 XYZ AREA 1	#8, BANK ARCADE		2	570007	IND	02/02/1999 0:00
1008 HEGARTY AREA 1	#51, 22 BUILDING		3	570008	IND	02/02/1999 0:00
1009 XYZ AGENT AREA 1	#42, MAINSTREET		4	570009	IND	01/01/1999 0:00
2003 LONDON AREA 1	#6, BIG BEN BUILDING,CENTRAL		5	570009	IND	12/12/1999 0:00

Figure 5.11: Staging table after agent IU

AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	AGENT_ID_SK	scd_start	scd_end	scd_version	scd_active
1001 XYZ ROSSES	MAIN CENTRAL		1	570001	EXC	1	18/11/2019 0:28		1	1
1002 XYZ INS HO	POLE CENTRAL		2	570002	EXC	2	18/11/2019 0:28		1	1
1003 XYZ LISTOWEL	MAIN STREET CENTRAL		3	570003	EXC	3	18/11/2019 0:28		1	1
1004 HEGARTY HO	POLE STREET CENTRAL		4	570004	EXC	4	18/11/2019 0:28		1	1
1005 LONDON HO	LONDON CENTRAL		5	570005	EXC	5	18/11/2019 0:28		1	1
1006 XYZ AREA 1	#92, TALL TOWERS		1	570006	IND	6	18/11/2019 0:28		1	1
1007 XYZ AREA 1	#8, BANK ARCADE		2	570007	IND	7	18/11/2019 0:28		1	1
1008 HEGARTY AREA 1	#51, 22 BUILDING		3	570008	IND	8	18/11/2019 0:28		1	1
1009 XYZ AGENT AREA 1	#42, MAINSTREET		4	570009	IND	9	18/11/2019 0:28		1	1
2003 LONDON AREA 1	#6, BIG BEN BUILDING		5	570009	IND	10	18/11/2019 0:34	18/11/2019 0:34	1	0
2003 LONDON AREA 1	#6, BIG BEN BUILDING,CENTRAL		5	570009	IND	11	18/11/2019 0:34		2	1

Figure 5.12: Dimension table after agent IU

A	B	C	D	E	F	G
AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	DATE
1001 XYZ ROSSES	MAIN STREET		1	570001	EXE	02/12/1999 0:00
1002 XYZ INS HO	POLE STREET		2	570002	EXE	02/05/1999 0:00
1003 XYZ LISTOWEL	MAIN STREET CENTRAL		3	570003	EXC	02/02/1999 0:00
1004 HEGARTY HO	POLE STREET CENTRAL		4	570004	EXC	01/01/1999 0:00
1005 LONDON HO	LONDON CENTRAL		5	570005	EXC	10/10/1999 0:00
1006 XYZ AREA 1	#92, TALL TOWERS		1	570006	IND	01/01/1999 0:00
1007 XYZ AREA 1	#8, BANK ARCADE		2	570007	IND	02/02/1999 0:00
1008 HEGARTY AREA 1	#51, 22 BUILDING		3	570008	IND	02/02/1999 0:00
1009 XYZ AGENT AREA 1	#42, MAINSTREET		4	570009	IND	01/01/1999 0:00
2003 LONDON AREA 1	#6, BIG BEN BUILDING,CENTRAL		5	570009	IND	12/12/1999 0:00

Figure 5.13: Staging table after agent U

|

A	B	C	D	E	F	G	H	I	J	K
AGENT_ID	AGENT_NAME	ADDRESS	LOCATION_ID	PINCODE	TYPE	AGENT_ID_SK	scd_start	scd_end	scd_version	scd_active
1001 XYZ ROSSES	MAIN CENTRAL		1	570001 EXC		1 18/11/2019 0:28	18/11/2019 0:41		1	0
1002 XYZ INS HO	POLE CENTRAL		2	570002 EXC		2 18/11/2019 0:28	18/11/2019 0:41		1	0
1003 XYZ LISTOWEL	MAIN STREET CENTRAL		3	570003 EXC		3 18/11/2019 0:28			1	1
1004 HEGARTY HO	POLE STREET CENTRAL		4	570004 EXC		4 18/11/2019 0:28			1	1
1005 LONDON HO	LONDON CENTRAL		5	570005 EXC		5 18/11/2019 0:28			1	1
1006 XYZ AREA 1	#52, TALL TOWERS		1	570006 IND		6 18/11/2019 0:28			1	1
1007 XYZ AREA 1	#8, BANK ARCADE		2	570007 IND		7 18/11/2019 0:28			1	1
1008 HEGARTY AREA 1	#51, 22 BUILDING		3	570008 IND		8 18/11/2019 0:28			1	1
1009 XYZ AGENT AREA 1	#42, MAINSTREET		4	570009 IND		9 18/11/2019 0:28			1	1
2003 LONDON AREA 1	#6, BIG BEN BUILDING		5	570009 IND		10 18/11/2019 0:34	18/11/2019 0:34		1	0
2003 LONDON AREA 1	#6, BIG BEN BUILDING,CENTRAL			570009 IND		11 18/11/2019 0:34			2	1
1001 XYZ ROSSES	MAIN STREET		1	570001 EXE		12 18/11/2019 0:41			2	1
1002 XYZ INS HO	POLE STREET		2	570002 EXE		13 18/11/2019 0:41			2	1

Figure 5.14: Dimension table after agent U

5.4 Transaction Detail

A	B	C	D	E	F	G	H	I
TRANSACTION_ID	TRANSACTION_TYPE	TRANSACTION_AMT	DATE	SALES_CHANNEL	PAYMENT_METHOD	CUSTOMER_ID	AGENT_ID	ready_indicator
9036 NB		50000	04/05/2012 0:00	ONLINE	PAYPAL	8888	2010	0
9037 NB		4000	05/05/2012 0:00	ONLINE	PAYPAL	9999	1006	0
9001 REN		4500	05/05/2012 0:00	AGENT	PAYPAL	12345	1001	0
9002 CANCEL		1000	05/05/2012 0:00	AGENT	DD	12345	1006	0
9003 REN		6000	08/07/2011 0:00	AGENT	DD	678910	1002	0
9004 CANCEL		2000	05/06/2012 0:00	AGENT	PAYPAL	12345	1001	0
9005 CANCEL		4000	03/10/2010 0:00	AGENT	DD	11121314	1008	0
9006 NB		5000	03/04/2011 0:00	AGENT	CASH	12345	1001	0
9007 NB		9000	06/05/2012 0:00	AGENT	DD	15161718	1004	0
9008 CANCEL		1500	08/08/2011 0:00	AGENT	DD	678910	1007	0
9009 NB		10000	05/05/2011 0:00	AGENT	DD	12345	1001	0
9010 CANCEL		5000	08/07/2011 0:00	AGENT	DD	12345	1006	0
9011 NB		3400	10/10/2011 0:00	AGENT	DD	678910	1002	0
9012 CANCEL		1200	10/11/2011 0:00	AGENT	DD	678910	1002	0
9013 REN		6500	12/12/2011 0:00	AGENT	DD	678910	1002	0
9014 CANCEL		500	13/12/2011 0:00	AGENT	DD	678910	1007	0
9015 CANCEL		2500	15/12/2011 0:00	AGENT	DD	678910	1007	0
9016 NB		5000	17/12/2011 0:00	AGENT	DD	678910	1002	0
9017 CANCEL		8000	31/12/2011 0:00	AGENT	DD	678910	1002	0
9018 REN		4200	08/12/2011 0:00	AGENT	DD	11121314	1003	0
9019 REN		2500	14/08/2011 0:00	AGENT	DD	11121314	1003	0
9020 CANCEL		1600	14/08/2011 0:00	AGENT	DD	11121314	1008	0
9021 REN		8000	17/09/2011 0:00	AGENT	DD	11121314	1003	0
0003 CANCEL		3000	18/09/2011 0:00	AGENT	DD	11121314	1003	0

Figure 5.15: Staging table for Transaction

A	B	C	D	E	F	G	H	I
TRANSACTION_ID	TRANSACTION_TYPE	TRANSACTION_AMT	DATE	SALES_CHANNEL	PAYMENT_METHOD	CUSTOMER_ID	AGENT_ID	ready_indicator
9037 NB		4000	05/05/2012 0:00	ONLINE	PAYPAL	9999	1006	0
9036 NB		50000	04/05/2012 0:00	ONLINE	PAYPAL	8888	2010	0
9035 NB		3000	06/12/2012 0:00	AGENT	CASH	19202122	1005	0
9034 CANCEL		8000	06/10/2012 0:00	AGENT	CHEQUE	19202122	1005	0
9033 NB		5000	06/08/2012 0:00	AGENT	CASH	19202122	1005	0
9032 NB		5000	06/07/2012 0:00	AGENT	CASH	19202122	1005	0
9031 NB		5000	06/06/2012 0:00	AGENT	CASH	19202122	1005	0
9030 CANCEL		4000	06/08/2011 0:00	AGENT	CHEQUE	15161718	1009	0
9029 CANCEL		1000	06/07/2012 0:00	AGENT	CHEQUE	15161718	1009	0
9028 REN		8000	06/06/2012 0:00	AGENT	DD	15161718	1004	0
9027 CANCEL		4000	10/10/2012 0:00	AGENT	DD	12345	1006	0
9026 NB		5000	09/09/2012 0:00	AGENT	DD	12345	1001	0
9025 NB		9000	08/08/2012 0:00	AGENT	DD	12345	1001	0
9024 NB		7000	22/09/2012 0:00	AGENT	DD	11121314	1003	0
9023 CANCEL		4000	20/09/2012 0:00	AGENT	DD	11121314	1003	0
9022 CANCEL		2000	18/09/2011 0:00	AGENT	DD	11121314	1003	0
9021 REN		8000	17/09/2011 0:00	AGENT	DD	11121314	1003	0
9020 CANCEL		1600	14/08/2011 0:00	AGENT	DD	11121314	1008	0
9019 REN		2500	14/08/2011 0:00	AGENT	DD	11121314	1003	0
9018 REN		4200	08/12/2011 0:00	AGENT	DD	11121314	1003	0
9017 CANCEL		8000	31/12/2011 0:00	AGENT	DD	678910	1002	0
9016 NB		5000	17/12/2011 0:00	AGENT	DD	678910	1002	0
9015 CANCEL		2500	15/12/2011 0:00	AGENT	DD	678910	1007	0
9014 NB		5000	17/12/2011 0:00	AGENT	DD	678910	1007	0

Figure 5.16: dimension table for Transaction

5.5 Business Rule Detail

A	B	C	D	E	F	G	H	I
TRANSACTION_ID	TRANSACTION_TYPE	TRANSACTION_AMT	DATE	SALES_CHANNEL	PAYMENT_METHOD	CUSTOMER_ID	AGENT_ID	ready_indicator
9001 REN		4500	05/05/2012 0:00	AGENT	PAYPAL	12345	1001	1
9002 CANCEL		1000	05/05/2012 0:00	AGENT	DD	12345	1006	1
9003 REN		6000	08/07/2011 0:00	AGENT	DD	678910	1002	1
9004 CANCEL		2000	05/06/2012 0:00	AGENT	PAYPAL	12345	1001	1
9005 CANCEL		4000	03/10/2010 0:00	AGENT	DD	11121314	1008	1
9006 NB		5000	03/04/2011 0:00	AGENT	CASH	12345	1001	1
9007 NB		9000	06/05/2012 0:00	AGENT	DD	15161718	1004	1
9008 CANCEL		1500	08/08/2011 0:00	AGENT	DD	678910	1007	1
9009 NB		10000	05/05/2011 0:00	AGENT	DD	12345	1001	1
9010 CANCEL		5000	08/07/2011 0:00	AGENT	DD	12345	1006	1
9011 NB		3400	10/10/2011 0:00	AGENT	DD	678910	1002	1
9012 CANCEL		1200	10/11/2011 0:00	AGENT	DD	678910	1002	1
9013 REN		6500	12/12/2011 0:00	AGENT	DD	678910	1002	1
9014 CANCEL		500	13/12/2011 0:00	AGENT	DD	678910	1007	1
9015 CANCEL		2500	15/12/2011 0:00	AGENT	DD	678910	1007	1
9016 NB		5000	17/12/2011 0:00	AGENT	DD	678910	1002	1
9017 CANCEL		8000	31/12/2011 0:00	AGENT	DD	678910	1002	1
9018 REN		4200	08/12/2011 0:00	AGENT	DD	11121314	1003	1
9019 REN		2500	14/08/2011 0:00	AGENT	DD	11121314	1003	1
9020 CANCEL		1600	14/08/2011 0:00	AGENT	DD	11121314	1008	1
9021 REN		8000	17/09/2011 0:00	AGENT	DD	11121314	1003	1
9022 CANCEL		2000	18/09/2011 0:00	AGENT	DD	11121314	1003	1
9023 CANCEL		4000	20/09/2012 0:00	AGENT	DD	11121314	1003	1
9024 NB		7000	22/09/2012 0:00	AGENT	DD	11121314	1003	1

Figure 5.17: Business Rule

5.6 Calendar Detail

Date_Key	WEEKDAY_NUM	Year	Month	Day	Weekday	Month_words	Date
1	5	2010	1	1	FRI	JAN	01/01/2010 0:00
2	6	2010	1	2	SAT	JAN	02/01/2010 0:00
3	7	2010	1	3	SUN	JAN	03/01/2010 0:00
4	1	2010	1	4	MON	JAN	04/01/2010 0:00
5	2	2010	1	5	TUE	JAN	05/01/2010 0:00
6	3	2010	1	6	WED	JAN	06/01/2010 0:00
7	4	2010	1	7	THU	JAN	07/01/2010 0:00
8	5	2010	1	8	FRI	JAN	08/01/2010 0:00
9	6	2010	1	9	SAT	JAN	09/01/2010 0:00
10	7	2010	1	10	SUN	JAN	10/01/2010 0:00
11	1	2010	1	11	MON	JAN	11/01/2010 0:00
12	2	2010	1	12	TUE	JAN	12/01/2010 0:00
13	3	2010	1	13	WED	JAN	13/01/2010 0:00
14	4	2010	1	14	THU	JAN	14/01/2010 0:00
15	5	2010	1	15	FRI	JAN	15/01/2010 0:00
16	6	2010	1	16	SAT	JAN	16/01/2010 0:00
17	7	2010	1	17	SUN	JAN	17/01/2010 0:00
18	1	2010	1	18	MON	JAN	18/01/2010 0:00
19	2	2010	1	19	TUE	JAN	19/01/2010 0:00
20	3	2010	1	20	WED	JAN	20/01/2010 0:00
21	4	2010	1	21	THU	JAN	21/01/2010 0:00
22	5	2010	1	22	FRI	JAN	22/01/2010 0:00
23	6	2010	1	23	SAT	JAN	23/01/2010 0:00
24	7	2010	1	24	SUN	JAN	24/01/2010 0:00

Figure 5.18: Calendar dimension table

5.7 Policy Detail

A	B	C	D	E	F	G	H
TRANSACTION_ID	TRANSACTION_TYPE	TRANSACTION_AMT	CUSTOMER_ID_SK	AGENT_ID_SK	Date_Key	SALES_CHANNEL	PAYMENT_METHOD
9001 REN		4500	6	12	856 AGENT	PAYPAL	
9002 CANCEL		1000	6	6	856 AGENT	DD	
9003 REN		6000	2	13	554 AGENT	DD	
9004 CANCEL		2000	6	12	887 AGENT	PAYPAL	
9005 CANCEL		4000	3	8	276 AGENT	DD	
9006 NB		5000	6	12	458 AGENT	CASH	
9007 NB		9000	4	4	857 AGENT	DD	
9008 CANCEL		1500	2	7	585 AGENT	DD	
9009 NB		10000	6	12	490 AGENT	DD	
9010 CANCEL		5000	6	6	554 AGENT	DD	
9011 NB		3400	2	13	648 AGENT	DD	
9012 CANCEL		1200	2	13	679 AGENT	DD	
9013 REN		6500	2	13	711 AGENT	DD	
9014 CANCEL		500	2	7	712 AGENT	DD	
9015 CANCEL		2500	2	7	714 AGENT	DD	
9016 NB		5000	2	13	716 AGENT	DD	
9017 CANCEL		8000	2	13	730 AGENT	DD	
9018 REN		4200	3	3	707 AGENT	DD	
9019 REN		2500	3	3	591 AGENT	DD	
9020 CANCEL		1600	3	8	591 AGENT	DD	
9021 REN		8000	3	3	625 AGENT	DD	
9022 CANCEL		2000	3	3	626 AGENT	DD	
9023 CANCEL		4000	3	3	994 AGENT	DD	
.....	

Figure 5.19: Policy Fact table

5.8 Business Report

The Business queries supplied in the requirement document is answered through sql queries and visualised in tableau.

1Q)What is the total revenue for New business in 2011?.

```
SELECT sum(transaction_amt) as total_new_business_revenue, TRANSACTION_TYPE
FROM insurance_dim.business_transaction_rule WHERE TRANSACTION_TYPE='NB'
and date like '%2011%'
```

	total_new_business_revenue	TRANSACTION_TYPE
▶	23400	NB

Figure 5.20: Business Query 1

Tableau Visualisation.

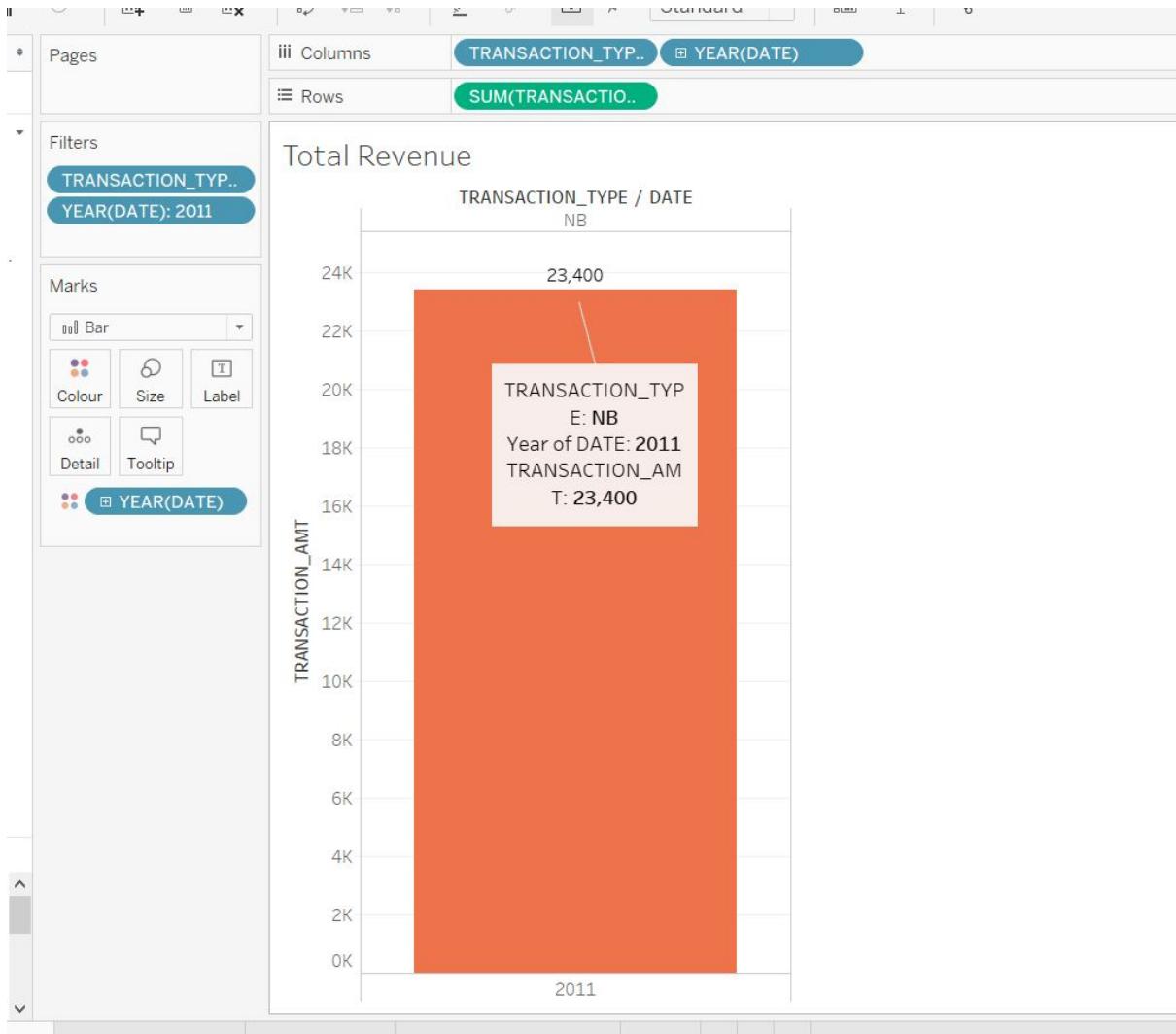


Figure 5.21: Tableau

2Q) What is the total new business transactions amount per Agent?.

```
select distinct a.agent_name,b.transaction_type,b.sum from agent_dim a join (select * ,sum(TRANSACTION_AMT) as sum from business_transaction_rule where transaction_type='NB' group by agent_id) b on a.agent_id=b.agent_id; .
```

	agent_name	transaction_type	sum
▶	XYZ ROSSES	NB	29000
	XYZ INS HO	NB	8400
	XYZ LISTOWEL	NB	7000
	HEGARTY HO	NB	9000
	LONDON HO	NB	18000
	XYZ AREA 1	NB	4000

Figure 5.22: Business Query 2

Tableau Visualisation.

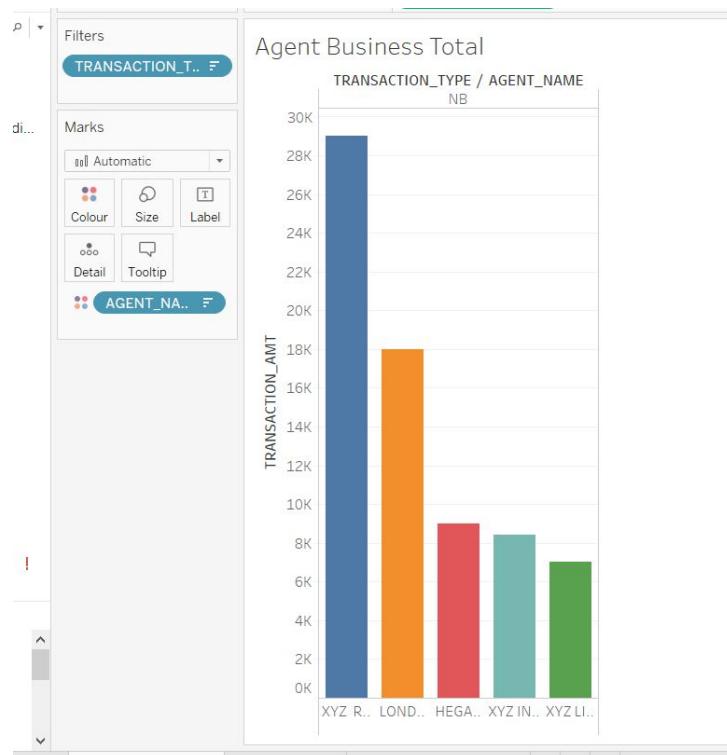


Figure 5.23: Tableau

3Q) Who are the top 3 performing agents for renewals?.

```
select distinct a.agent_name,transaction_type,renewals_total from agent_dim a join
```

```
(select *,sum(TRANSACTION_AMT) as renewals_total from business_transaction_rule
where transaction_type=REN group by agent_id ) b on a.agent_id=b.agent_id order by
renewals_total DESC limit 3;
```

	agent_name	transaction_type	renewals_total
▶	XYZ LISTOWEL	REN	14700
	XYZ INS HO	REN	12500
	HEGARTY HO	REN	8000

Figure 5.24: Business Query 3

Tableau Visualisation.

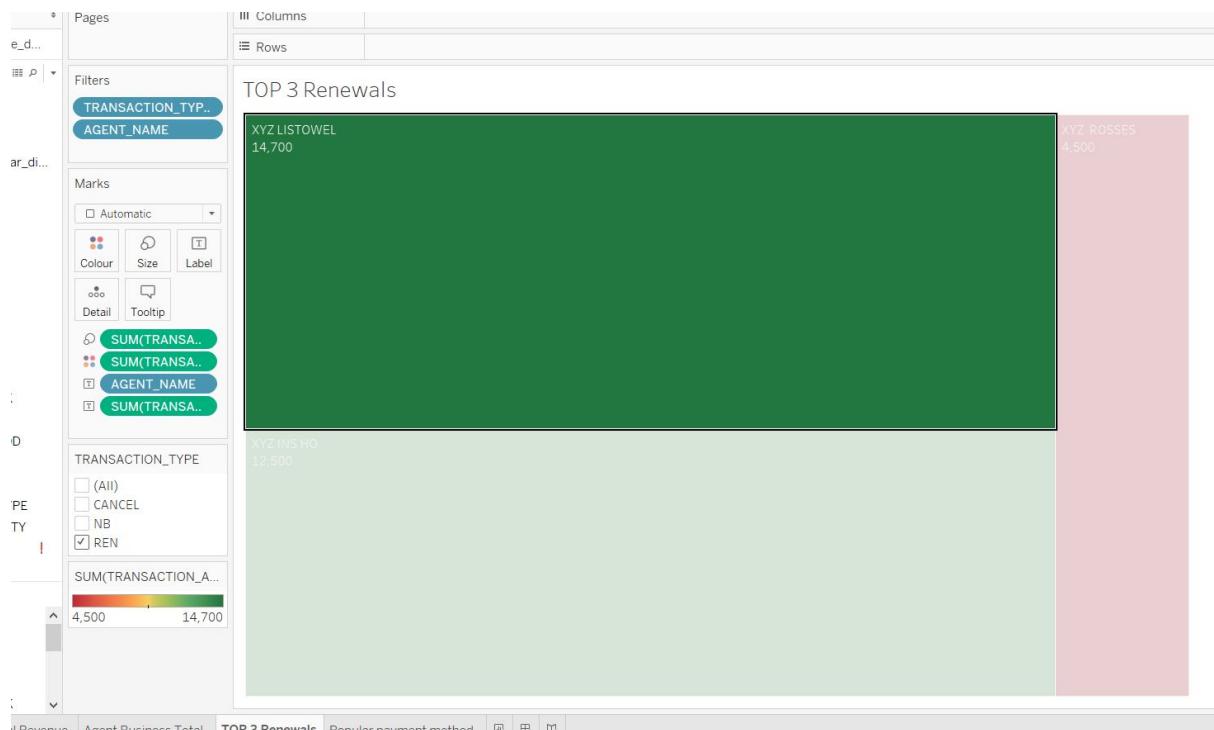


Figure 5.25: Tableau

4Q) Who are making the most in transaction total, exclusive or independent agents?.

```
select a.type,sum(transaction_amt) from agent_dim a join (select agent_id,transaction_amt,READY_
from business_transaction_rule ) b on a.agent_id=b.agent_id where a.type in (EXC,IND)
and b.ready_indicator=0 group by a.type; .
```

	type	sum(transaction_amt)
▶	EXC	136300
	IND	25100

Figure 5.26: Business Query 4

5Q)What is the most popular payment method?.

```
select payment_method,count(*) as TRANSACTION_COUNT from policy_fact group by PAYMENT_METHOD order by TRANSACTION_COUNT DESC limit 1; .
```

Tableau Visualisation

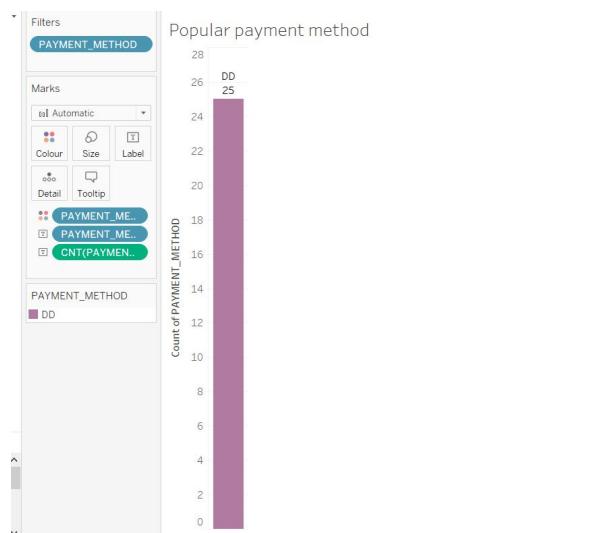


Figure 5.27: Business Query 5

Chapter 6

CONCLUSION

This was an assignment to create a fully functioning data warehouse for XYZ Insurance company. This is implemented using Talend open studio for data integration, MySQL workbench for storing the information. The project supplied with relevant requirements such as data model, Schemas, Staging and Dimension areas etc. The data was cleaned in staging area and loaded into dimension table and fact tables. After fully functioning data warehouse is created, the business queries are answered. The data was visualised through tableau and business insights were drawn. The main objective was to track the amount of business generated by each agent which was further answered by sql queries.

References

- [1] Lim, E., Chen, H., Chen, G.: Business intelligence and analytics: research directions. *ACM Trans. Manage. Inf. Syst.*, vol. 3, no. 4, Article 17, pp. 1–10 (2013)Google Scholar.
- [2] Brust, A.: Gartner releases 2013 BI Magic Quadrant 2013. <http://www.zdnet.com/gartner-releases-2013-bi-magic-quadrant-7000011264/>.
- [3] Tutorialspoint.com. (2019). Talend - Metadata - Tutorialspoint. [online] Available at: https://www.tutorialspoint.com/talend/talend_metadata.htm [Accessed 18 Nov. 2019].