

## Vehicle Insurance Data Warehouse

SHRUTI SIKRI

#### Company Description

- The insurance company XYZ provides various insurances to different Automobiles across USA.
- The company has different covers for each vehicle depending on Driver's driving history, points on license, vehicle details and type etc.



## Data Warehouse mission statement



The objective behind building a data warehouse is connecting data from source systems, applying transformations and storing into one repository for better data flow across the organization.



This finally helps in better analytics considering vast data coming from different sources, which results in data-based decision making across the organization.



The data warehouse repository will also be used across different departments to serve their data and reporting needs

#### Business case

- The need of central data warehouse repository is imminent because data is spread across various sources, causing inconsistent data flow throughout the organization.
- This results in poor analytical operations being performed across organization, ultimately resulting in slow growth of organization.

#### Project Scope

The scope consists of four major activities:

- 1. Creating data warehouse
- 2. Breaking the DW down into data marts
- 3. Creating BI plan
- 4. Maintenance of DW

#### Project Stakeholders



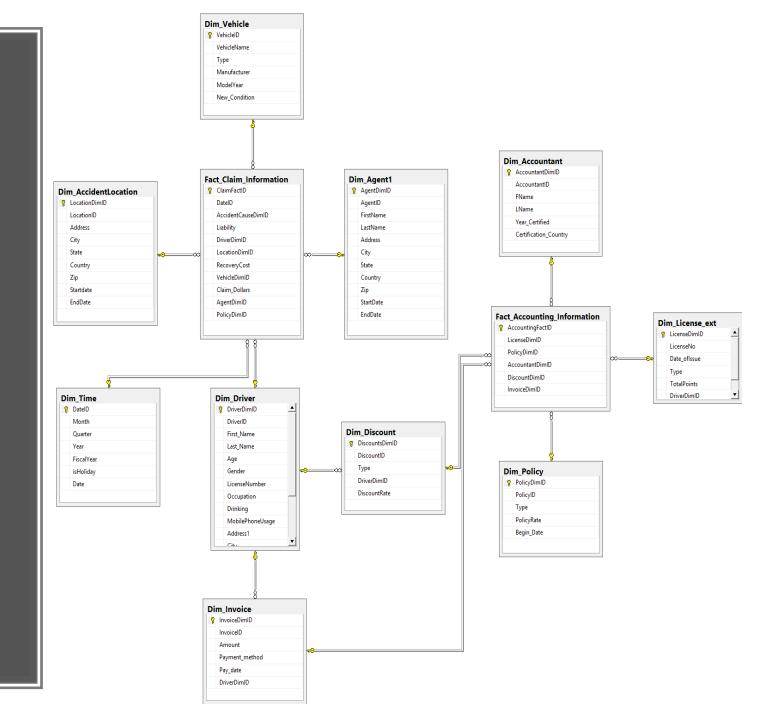
- Project Sponsor: Director of XYZ company who is funding the project
- Project Team: Team responsible for implementing the data warehouse
- End Users: Employees using the DW for data and analytical purposes
- Review board: The team responsible to check whether the DW meets the requirements

#### Interview Questions



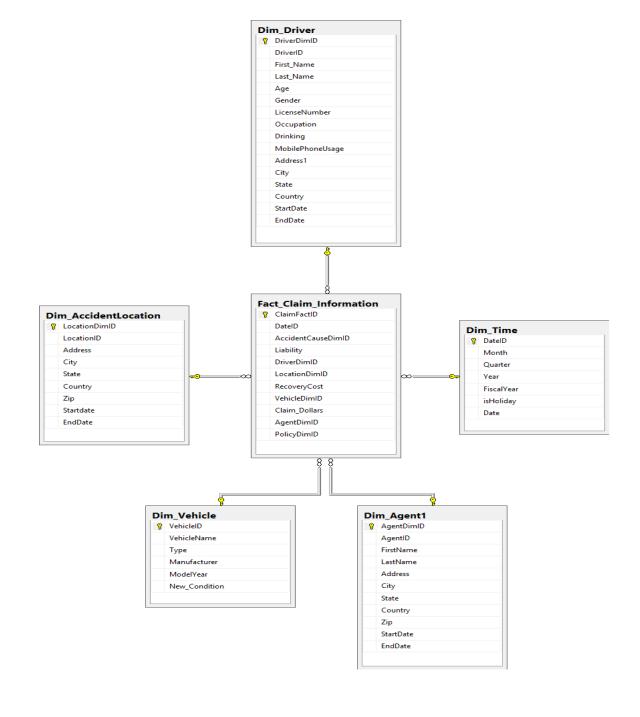
- What are the data sources and type of data?
- What regulations do we need to adhere to?
- What kind of reporting and analytical services it should provide?
- Do we need Data marts or just one consolidated DW?
- What are the existing data security measures?
- Do you need the team to support the data warehouse or it can be done in house?
- What are current problems while extracting data from different sources for analytical purposes?

# Data Warehouse Schema Design



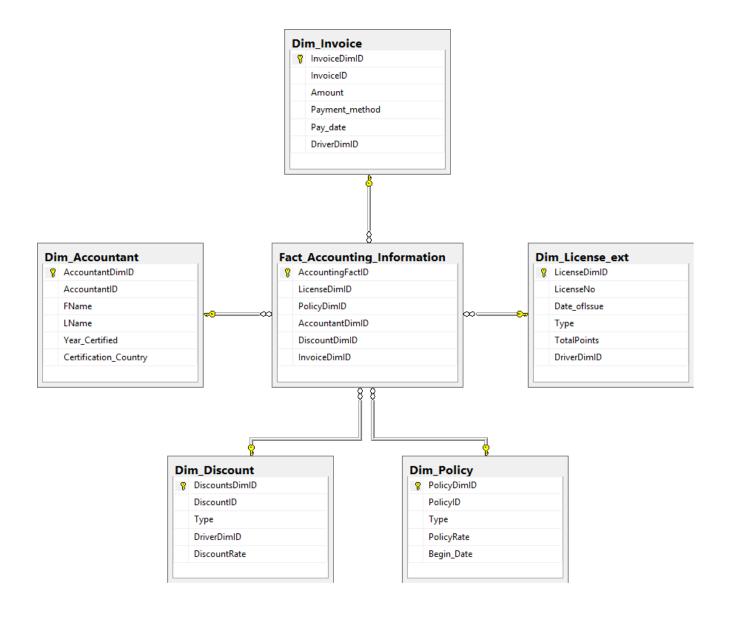
### Data Marts Schema Design

1. Claim Information DM



#### Data Marts Schema Design

2. Accounting DM



### SQL-Build (create)Dim\_Tables

```
SET ANSI_NULLS ON
SET QUOTED IDENTIFIER ON
∃CREATE TABLE [dbo].[Dim Driver](
    [DriverDimID] [int] NOT NULL,
    [DriverID] [nvarchar](50) NULL,
    [First_Name] [nvarchar](50) NULL,
    [Last_Name] [nvarchar](50) NULL,
    [Age] [int] NULL,
    [Gender] [nvarchar](50) NULL,
    [LicenseNumber] [nvarchar](50) NULL,
    [Occupation] [nvarchar](50) NULL,
    [Drinking] [bit] NULL,
    [MobilePhoneUsage] [text] NULL,
    [Address1] [nvarchar](50) NULL,
    [City] [nvarchar](50) NULL,
    [State] [nvarchar](50) NULL,
    [Country] [nvarchar](50) NULL,
    [StartDate] [datetime] NULL,
    [EndDate] [datetime] NULL,
 CONSTRAINT [PK_Dim_Driver] PRIMARY KEY CLUSTERED
 )WITH (PAD INDEX = OFF, STATISTICS NORECOMPUTE = OFF, IGNORE DUP KEY = OFF, ALLOW ROW LOCKS = ON, ALLOW PAGE LOCKS = ON) ON [PRIMARY]
 ON [PRIMARY] TEXTIMAGE ON [PRIMARY]
```

## SQL-Build (create) Fact\_Accounting

```
SET QUOTED_IDENTIFIER ON
□CREATE TABLE [dbo].[Fact_Accounting_Information](
    [AccountingFactID] [int] NOT NULL,
    [LicenseDimID] [int] NOT NULL,
    [PolicyDimID] [int] NOT NULL,
    [AccountantDimID] [int] NOT NULL,
    [DiscountDimID] [int] NOT NULL,
    [InvoiceDimID] [int] NOT NULL,
  CONSTRAINT [PK Fact Accounting Information] PRIMARY KEY CLUSTERED
 )WITH (PAD INDEX = OFF, STATISTICS NORECOMPUTE = OFF, IGNORE DUP KEY = OFF, ALLOW ROW LOCKS = ON, ALLOW PAGE LOCKS = ON) ON [PRIMARY]
  ON [PRIMARY]
□ALTER TABLE [dbo].[Fact_Accounting_Information] WITH CHECK ADD CONSTRAINT [FK_Fact_Accounting_Information_Dim_Accountant] FOREIGN KEY([Account
REFERENCES [dbo].[Dim_Accountant] ([AccountantDimID])
 ALTER TABLE [dbo].[Fact_Accounting_Information] CHECK CONSTRAINT [FK_Fact_Accounting_Information_Dim_Accountant]
□ALTER TABLE [dbo].[Fact Accounting Information] WITH CHECK ADD CONSTRAINT [FK Fact Accounting Information Dim Discount] FOREIGN KEY([DiscountD
REFERENCES [dbo].[Dim_Discount] ([DiscountsDimID])
```

```
ALTER TABLE [dbo]. [Fact_Accounting_Information] CHECK CONSTRAINT [FK_Fact_Accounting_Information_Dim_Discount]
∃ALTER TABLE [dbo].[Fact Accounting Information] WITH CHECK ADD CONSTRAINT [FK Fact Accounting Information Dim Invoice] FOREIGN KEY([InvoiceDim
 REFERENCES [dbo].[Dim_Invoice] ([InvoiceDimID])
 ALTER TABLE [dbo].[Fact_Accounting_Information] CHECK CONSTRAINT [FK_Fact_Accounting_Information_Dim_Invoice]
∃ALTER TABLE [dbo].[Fact Accounting Information] WITH CHECK ADD CONSTRAINT [FK Fact Accounting Information Dim License ext] FOREIGN KEY([Licens
 REFERENCES [dbo].[Dim_License_ext] ([LicenseDimID])
 ALTER TABLE [dbo].[Fact_Accounting_Information] CHECK CONSTRAINT [FK_Fact_Accounting_Information_Dim_License_ext]
∃ALTER TABLE [dbo].[Fact Accounting Information] WITH CHECK ADD CONSTRAINT [FK Fact Accounting Information Dim Policy] FOREIGN KEY([PolicyDimID
 REFERENCES [dbo].[Dim_Policy] ([PolicyDimID])
 ALTER TABLE [dbo].[Fact Accounting Information] CHECK CONSTRAINT [FK Fact Accounting Information Dim Policy]
```

## SQL-Build (insert)

ST-C	S-DW1.ist722Claim_Information X											
	ClaimFactID	DateID	AccidentCaus	Liability	DriverDimID	LocationDimID	RecoveryCost	VehicleDimID	Claim_Dollars	AgentDimlD	Po	
)	1	1	1	50	1	1	100	1	200	1	1	
	2	2	2	100	2	2	100	2	230	2	2	
	3	3	3	70	3	3	190	3	250	3	3	
	4	4	4	3000	4	4	200	4	2500	4	4	
	5	5	5	348	5	5	300	5	3445	5	5	
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NL	

IST-C	'S-DW1.ist722w	- dbo.Dim_Drive	Х								•
	DriverDimID	DriverID	First_Name	Last_Name	Age	Gender	LicenseNumber	Occupation	Drinking	MobilePhone	Ac
)	1	111	Tanmay	Atkekar	24	Male	166702	Data Scientist	False	3152228809	120
	2	112	Harsh	Takrani	24	Male	166701	Data Scientist	False	3151112309	245
	3	113	Farheen	Safoora	24	Female	166703	Data Scientist	False	3152226660	122
	4	114	Doug	Taber	50	Male	166700	Professor	False	3151123456	101
	5	115	Yun	Huang	36	Female	166704	Professor	False	3156666661	161

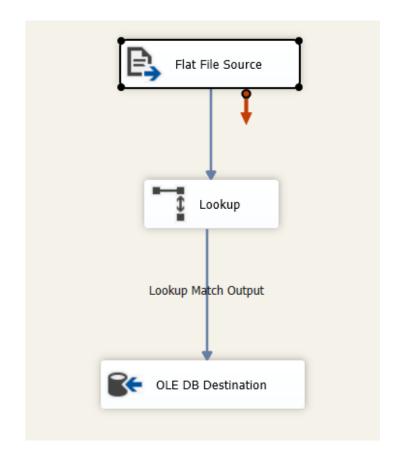
## ETL Script using Python

```
from auxiliary import *
directory = r"C:\Users\re70296\Downloads\IDR Project\sample\\"
filename = 'CF1 01 2018'
file_location = directory + filename + ".xlsx"
CF_Name = filename.split(' ',1)[0]
Quarter_Name = filename.split(' ',1)[1]
# Open the workbook and define the worksheet
book = xlrd.open_workbook(file_location)
SHEETS_index = {'Accountant': 'Dim_Accountant', 'Agent': 'Dim_Agent',
'Discount': 'Dim Discount', 'Driver': 'Dim Driver',
'Invoice': 'Dim_Invoice', 'License': 'Dim_Licence_Ext',
'Policy': 'Dim_Policy', 'Time': 'Dim_Time',
'Vehicle': 'Dim Vehicle'}
database, cursor = create_connection()
if(result==0):
    ###Update all tables
    for sheetname in SHEETS index:
        insert_values(book = book, sheetname = sheetname, SHEETS_index = SHEETS_index,
            database = database, cursor = cursor, file location = file location)
close_database(database, cursor)
input('Press ENTER to exit')
```

```
ort xlrd
     openpyxl import load workbook
import pymysql as MySQLdb
import numpy as np
import pandas as pd
def create connection():
   database = MySQLdb.connect(host="localhost", user = "root", passwd = "Tanmay9830!", db = "idr_db")
   # Get the cursor, which is used to traverse the database, line by line
   cursor = database.cursor()
   cursor.execute("USE idr_db") # select the database
   cursor.execute("SET SESSION sql_mode = ''")
   database.commit()
   print('Database Connection Opened')
    return database, cursor
def return_pk_index(tablename, database, cursor, pkid='PKID'):
   query_pk = """SELECT MAX(PKID) FROM """ + tablename
   cursor.execute(query_pk)
   query_pk_result = cursor.fetchone()[0]
   if(query_pk_result):
        return query_pk_result
        return 0
def insert_value_in_cell(cell, query, cursor, sheet):
   cursor.execute(query)
   val = cursor.fetchone()[0]
   sheet[cell] = val
```

	VehiclelD	VehicleName	Type	Manufacturer	ModelYear	New_Condition
•	1	Toyota	Sedan	Toyota	2017	NULL
	2	Q3	SUV	BMW	2015	NULL
	3	Model 3	Sedan	Tesla	2015	NULL
	4	Santro	Hatchback	Hyundai	2008	NULL
	5	WagonR	Hatchback	Maruti	2008	NULL
	NULL	NULL	NULL	NULL	NULL	NULL





#### Lookup Transformation in SSIS

#### ETL Implementation SCD-2

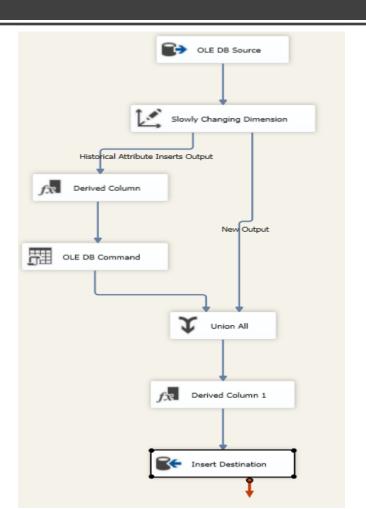
Before change in Policy Rate

After change in Policy Rate



	PolicyDimID	PolicyID	Type	PolicyRate	Begin_Date	End_Date
1	1	123	Lifelong	25	2019-01-01 00:00:00.000	2018-11-27 20:02:26.000
2	2	234	1 year	35	2018-09-18 00:00:00.000	2018-11-27 20:02:26.000
3	3	345	2 years	10	2014-10-03 00:00:00.000	NULL
4	4	567	6 months	5	2017-09-09 00:00:00.000	NULL
5	5	489	Lifelong	40	2016-01-13 00:00:00.000	2018-11-27 20:02:26.000

#### SCD-2 Dataflow in SSIS



#### ETL Implementation – SCD 3

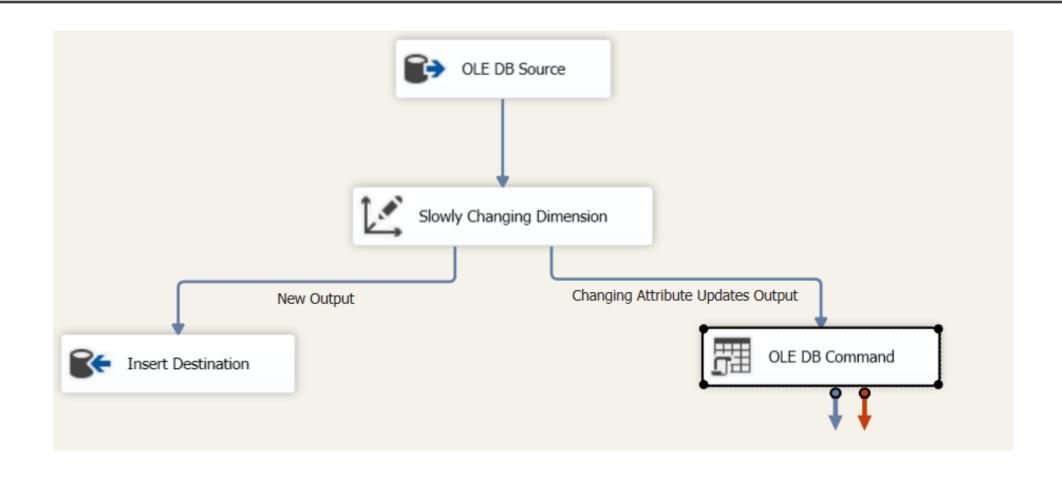
#### Before change in Address

	AgentID	FirstName	LastName	Address	City	State	Country	Zip	StartDate	EndDate
•	102	Adam	Paul	223 Oak Ave	Manhattan	New York	United States	10012	2018-09-11 00:0	NULL
	101	Eva	Dream	223 Oak Ave	Manhattan	New York	United States	10012	2018-08-13 00:0	NULL
	103	Jennifer	Dias	223 Oak Ave	Manhattan	New York	United States	10012	2018-09-13 00:0	NULL
	104	Noah	Williams	122 Adam St	Dallas	Texas	United States	10010	2018-09-13 00:0	NULL
	105	Rachel	Green	123 Brooklyn St	Brooklyn	New York	United States	10020	2018-09-13 00:0	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

#### After change in Address

AgentDimID	AgentID	FirstName	LastName	Address	City	State	Country	Zip	StartDate
1	102	Adam	Paul	504 Greenwood	Syracuse	NY	United States	10012	2018-09-11 00:
2	101	Eva	Dream	437 Columbus	Syracuse	NY	United States	10012	2018-08-13 00:
3	103	Jennifer	Dias	223 Oak Ave	Manhattan	New York	United States	10012	2018-09-13 00:
4	104	Noah	Williams	122 Adam St	Dallas	Texas	United States	10010	2018-09-13 00:
5	105	Rachel	Green	123 Brooklyn St	Brooklyn	New York	United States	10020	2018-09-13 00:
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

#### SCD 3 Data Flow





#### Assign

Assign data loading times into the data warehouse

#### Purge

Purge data: get rid of unwanted data in the data warehouse as it is not an unlimited repository

#### Tune

Tune the system:
Periodically review
how the data
warehouse is being
used and fine tune
the configuration to
optimize its
performance

#### Conclusion

 This newly formed central repository i.e. data warehouse is summarized data coming from various sources, without loss of any valuable information which is used for better decision making across the organization.



## Thank You