Northeastern University
College of Engineering

INFO7290: DATA WHOUSE & BUS INTELLIGENCE

Final Group Project- Design Document

Group 1-Victory

Healthcare Expenditure and Coverage Analysis

Group-1 Members

Sreya Pedamalla

Priyanka Sharma

INFO7290 Instructor: Vincent Lattuada

Table of Contents

Introduction

National health expenditure is the official estimate of health care expenses in the United States. We have data for three categories of data included:

1. Medicare
2. Medicaid
3. Private Health Insurance

Medicare is a federal program that provides health coverage if you are 65+ or under 65 and have a disability, no matter your income.
Medicaid is a state and federal program that provides health coverage if you have a very low income.
Private health insurance refers to any health insurance coverage that is not offered by a state or federal government.
Health expenditure data files (1991-2014) present the aggregate estimates for health care consumption, including the establishment of delivering care (hospitals, physicians and clinics, nursing homes, etc.) and medical products (prescription drugs, over-the-counter medicines) purchased in retail outlets.
Hospital Information comprises all the hospital related details like unique Id, Hospital Type, Hospital Ownership and ratings in various states.
Physician Information consists of all the physicians related details like unique ID, Physician Field, Specialty and location in various states.

Objectives/Scope

A data warehouse is built by populating enrollment and expenditure data of healthcare by state and region, US population data, hospitals registered, and physician profiles associated with the same.

The datasets in our data warehouse will help us to perform different analyses like how the U.S. healthcare expenditure has changed over time. It will allow us to draw insights into the percentage of state spending as a share of the national total and various other population and location-specific analysis.
We want to analyze how healthcare service is developed in each state and how it can be improved in the future.

Tools used

Following tools are used in this project:

- ER Studio for data modeling and Conceptual design process.
- Visual Studio (SSDT - SQL Server Data Tools, SSIS - SQL Server Integration Services) for data loading and designing a data warehouse
- Talend and Microsoft SQL Server for data profiling and quick analysis
- SSIS and SQL Server for data transformation.
- Tableau for data visualization

Data Preparation

Data preparation is a process in which raw data from one or multiple sources are cleaned, structured and transformed into a desired output for business purposes. Data preparation steps include:

Gather and extract data from source systems. For the sake of this project data is collected from multiple sources and files for health expenditure, enrollment data and the physician data is downloaded from the CMS.gov website (link provided in the Appendix), hospital information from data.gov (link in appendix) and Zip Code information from USPS open source website (link in appendix).

Reformat Data. While loading data into the staging tables data is formatted for all the columns to be varchar of size 100-300. While loading the data from stage to data warehouse data conversion is performed to have an appropriate data type for each column with a few limitations on the data sizes and NOT NULL constraints wherever appropriate.

Consolidate and Validate Data. In this project, there are different files being extracted for health insurance expenditure, Medicare Aggregate, Medicaid Aggregate and PHI aggregate. All these files have the same schema, so they are merged together by doing union of all the files into one aggregate table in the data warehouse. Similarly, as Medicare Enrollment, Medicaid Enrollment and PHI Enrollment had the same schema these three files are merged together by doing a union into one Enrollment table in the data warehouse.

Transform data. Normalized the tables to create Zip Code, License State code and geography tables. The multiple year columns are transformed by creating a single year column and doing an unpivot to load all aggregate and enrollment data corresponding to every year in multiple rows rather than a single row.

Data Cleaning. Upon examining the data, we transformed a few columns to make them uniform and readable. For example, the phone number column is transformed to have a uniform format of (xxx)-xxx-xxxx, zip code column to have 5 characters as a few of zip code information from the source also included an extension (xxxx-xxxx)

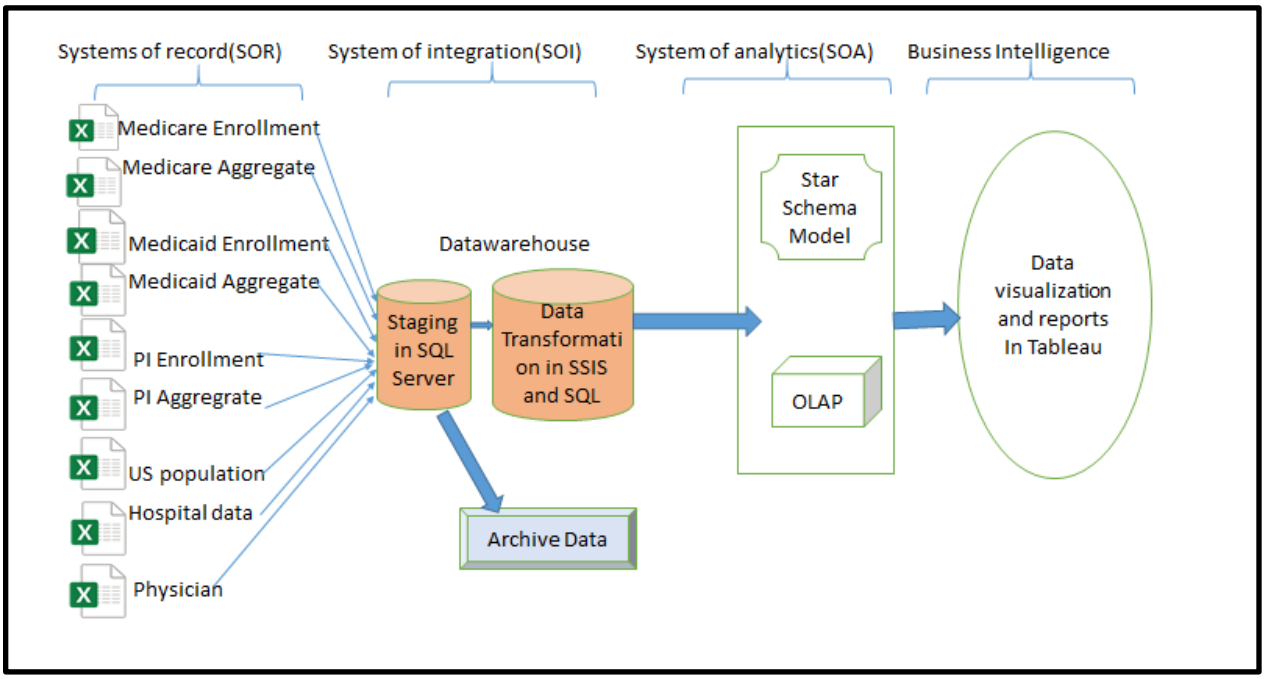
Store Data. Data is moved from the source to stage tables, then it is moved to the DataWarehouse. Then transformed into star schema

Data Architecture

Hybrid Dimensional-Normalized Model: This model leverages both normalization and dimensional model design. The project applies this data architecture as detailed below:

- The architecture includes EDW which allows consolidating of entities like Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate into DWHealthExpenditureAggregate and satisfy 3NF alongside.
- Building the EDW data model to track data integration from various stage tables, extending it to the Dimensional model and then supporting the analytical processes.

- The architecture implements a standard dimensional data model in the form of snowflake schema which is explained under the data load process in detail.
- Used advanced dimensional modelling constructs like:
    1) Implemented a year dimension and added year dimension foreign keys for each of the year attributes in fact tables (Fact_Aggregate, Fact_Enrollment and Fact_Population) while retaining the original data attributes.
    2) Created slowly changing dimension (SCD) data models to track historical changes to dimensional values in EDW which is explained under the data load process in detail.
    3) Implemented hierarchies while creating OLAP cubes rather than using separate tables for each hierarchy level.
    4) Created surrogate keys like SKEnrollID, SKAggregateID independent of source systems' natural keys, for required tables.



Graph 1: data methodology diagram

Data Description

- **Medicare Aggregate** - Total health care expenditure for Medicare by state and by service for years 1991-2014
- **Medicare Enrollmen**t - The total enrollment count for Medicare by state for years 1991-2014
- **Medicaid Aggregate** - Total health care expenditure for Medicaid by state and by service for the years 1991-2014
- **Medicaid Enrollmen**t - The total enrollment count for Medicaid by state for years 1991-2014

- **PHI Aggregate** - Total private health insurance personal health care spending by state and by service, 2001-2014
- **PHI Enrollment** - The total enrollment count for Private Health Insurance by state for years 2001-2014
- **US Population** - US Population by State for years 1991-2014
- **Hospital General Information**- A list of all Hospitals that have been registered with the above-mentioned insurance services.
- **Physician Details** - A list of all physicians registered for the above mentioned insurance services

Table 1: Variable Names and Descriptions for Health Expenditure Data Files

VARIABLE NAME	DESCRIPTION	DATA TYPE	DATA SIZE
Code	Numerical code assigned to each Item. It has 11 categories included. It does not have any NULL values.	Integer	~2 bytes
Item	Item is a text description of a Code that represents the type of service or care for which the insurance is used. Code 11 represents the enrollee population which is used in the enrollment data files. It does not have any NULL values.	Varchar	~50bytes-~100 bytes
Group	Level of aggregation by geography. This column has three levels (United states, Region, state).It does not have any NULL values.	Varchar	~10 byte
Region_Number	Numerical code assigned to each Region. It does not have any NULL values.	Integer	~1 byte
Region_Name	Region Name has the 8 categories into which each state is assigned to. It does not have any NULL values.	Varchar	~10 byte
State_Name	U.S. State Name.	Varchar	~15 byte
Y1991-Y2014	Expenditure, population, or enrollment counts for years 1991-2014. It does not have any NULL values.	Integer	~10 byte
Average_Annual_ Percent_ Growth	The average annual growth rate for spending, population, or enrollment, 1991-2014.It does not have any NULL values.	Integer	~3 byte

Table 2: Variable Names and Descriptions for Hospital Information Data Files

VARIABLE NAME	DESCRIPTION	DATA TYPE	DATA SIZE
Facility ID	Unique ID for each hospital.	Integer	~6 bytes
Facility Name	Name of the hospital.	Varchar	5 bytes- 53 bytes
Address	Street name for the location of the hospital.	Varchar	~7 bytes-~51 bytes
City	City in which the hospital is located.	Varchar	~3 bytes-~20 bytes
State	US state.	Varchar	2 bytes
ZIP Code	Zip Code in which the hospital is located	Integer	5 bytes
Phone Number	10-digit phone number formatted as (xxx)-xxx-xxxx	Integer	14 bytes
Hospital Type	Types of hospitals (5 categories description present)	Varchar	9 byte-34 byte
Hospital Ownership	Describes the ownership types which include profit, not--profit owned etc.. Contains 11 types of ownerships.	Varchar	6 byte-43 byte
Emergency Services	A Boolean (True/False) indicating it is present or not	Varchar	4 bytes- 5bytes
Hospital overall rating	Hospital Rating from 1-5 summarizes a variety of measures across 7 areas of quality into a single star rating for each hospital. Contains NULL values	Integer	1 byte
Mortality national comparison	A comparison with National Average for mortality rating (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes

Safety of care national comparison	A comparison with National Average for Safety of Care (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Readmission national comparison	A comparison with National Average for Readmission (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Patient experience national comparison	A comparison with National Average for patient Experience (Values like below, above, same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Effectiveness of care national comparison	A comparison with National Average for Effectiveness of Care (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Timeliness of care national comparison	A comparison with National Average for Timeliness of Care (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Efficient use of medical imaging national comparison	A comparison with National Average for Medical Imaging (Values include Below, Above and Same as national average). Contains NULL values	Varchar	13 bytes-28 bytes
Location	latitude and longitude of the location of the hospital. Contains NULL values	Varchar	29 bytes

Table 3: Variable Names and Descriptions for Physician Information Data Files

VARIABLE NAME	DESCRIPTION	DATA TYPE	DATA SIZE
PhysicianID	Unique ID number ranging from 1-5706079	Integer	1 bytes- 7 bytes

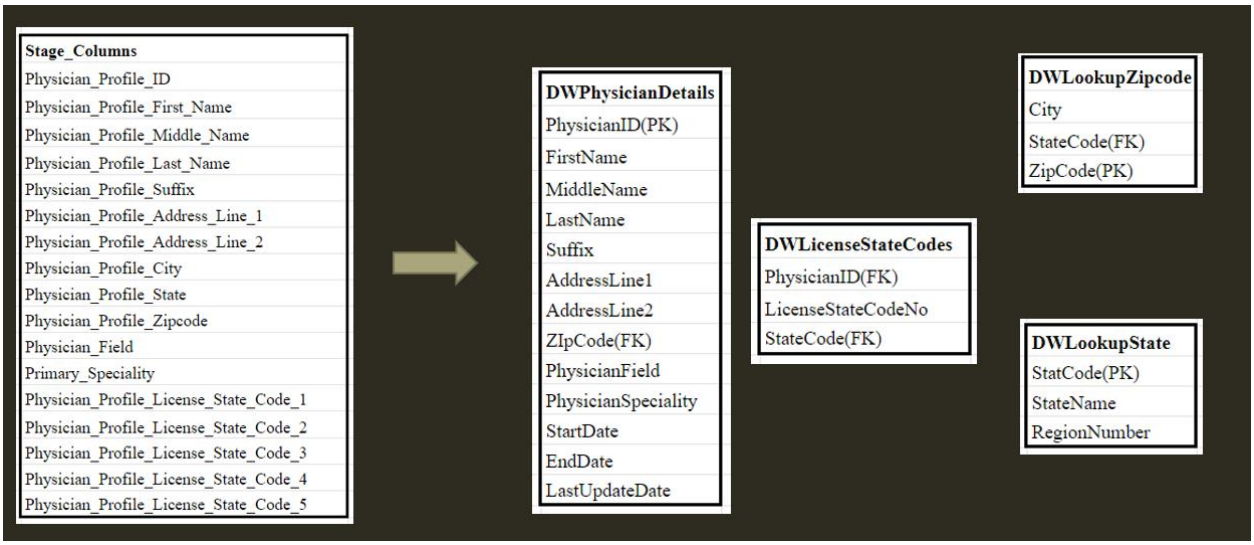
FirstName	Physician First Name	Varchar	1 byte-25 byte
MiddleName	Physician Middle Name. Contains Null values	Varchar	0 byte-25 byte
LastName	Physician Last Name. Contains Null values	Varchar	0 byte-30 byte
Suffix	Physician Suffix. Contains Null values	Varchar	0 byte-10 byte
Alternate First Name	Alternate Physician First Name Contains Null values	Varchar	1 byte-25 byte
Alternate Middle Name	Alternate Physician Middle Name. Contains Null values	Varchar	0 byte-25 byte
Alternate Last Name	Alternate Physician Last Name. Contains Null values	Varchar	0 byte-30 byte
Alternate Suffix	Alternate Physician Suffix. Contains Null values	Varchar	0 byte-10 byte
AddressLine1	Physicians Address line 1 Contains Null values	Varchar	2 byte-55 byte
AddressLine2	Physicians Address line 2 Contains Null values	Varchar	2 byte-55 byte
City	Physicians City	Varchar	0 byte-55 byte
State	Physicians State	Varchar	1 byte-20 byte
Zip Code	Zip code	Integer	5 byte
PhysicianField	Physician field of medicine	Varchar	0 byte-46 byte
PhysicianSpeciality	Physician specific department of medicine	Varchar	0 byte-50 byte
LicenseStateCode1	State code 1 for which physician has a license	Varchar	2 byte

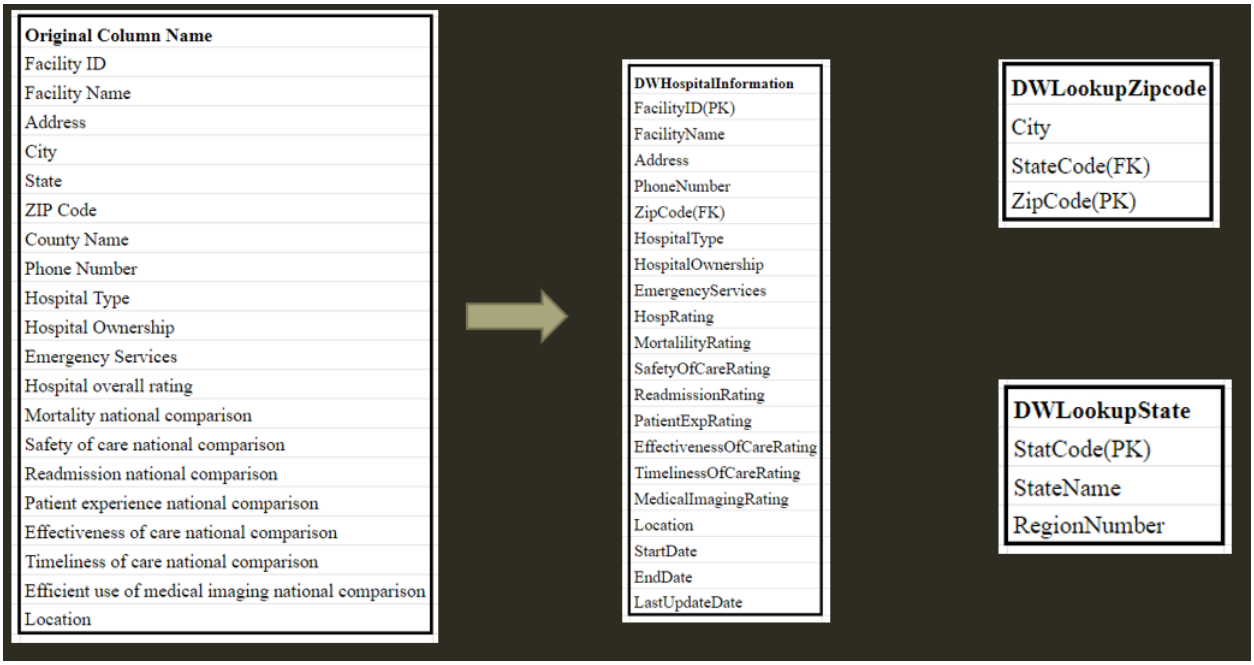
LicenseStateCode2	State code 2 for which physician has a license. Contains Null values	Varchar	2 byte
LicenseStateCode3	State code 3 for which physician has a license. Contains Null values	Varchar	2 byte
LicenseStateCode4	State code 4 for which physician has a license. Contains Null values	Varchar	2 byte
LicenseStateCode5	State code 5 for which physician has a license. Contains Null values	Varchar	2 byte

Normalization

In this project, the ER model for Data warehouse is built out as a normalized model by applying 3NF design principles. It helps to minimize data redundancy and ensures data integrity which can be used for transactional systems and business intelligence.

- **First normal form (1NF)**: There were repeating groups present in the physician table for license state code, performing 1 NF, LicenseStateCode table is built by following referential integrity constraints.
- **Second normal form (2NF)**: There were partial key dependencies for zip code on city in hospital and physician table, so lookup tables are created as DWLookupZipcode and DWLookupState with zipcodeID and state code as primary key.
- **Third normal form (3NF)**: The tables now satisfy 3NF as well.





EL-TL

We are performing EL-TL. Since we are loading all the data first into staging tables and the transforming and loading into data warehouse.

The advantages of choosing this method is it gives a pre-structured and transformed data warehouse which allows a speedier and efficient analysis of data.

Data Workflow

Staging Tables

Staging is done as a part of the ETL process where raw data from the source is loaded before loading it into the target/destination files. This is done to prevent unformatted data in the target tables which might cause issues/errors in the business processes.
A stage typically has:

No restrictions/ constraints

Loose datatypes (example varchar which accepts all the data)

No quality checks on the data

and 1 stage per source.

Thus a 1:1 mapping is created for all the data source files and loaded into a stage table.

Staging Tables are as follows:

1. **Stage_Medicare_Aggregate:**

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)

Region_Name	varchar(100)
State_Name	varchar(100)
Y1991	varchar(100)
Y1992	varchar(100)
Y1993	varchar(100)
Y1994	varchar(100)
Y1995	varchar(100)
Y1996	varchar(100)
Y1997	varchar(100)
Y1998	varchar(100)
Y1999	varchar(100)
Y2000	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)
Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)

Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

2. Stage_Medicare_Enrollment:

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y1991	varchar(100)
Y1992	varchar(100)
Y1993	varchar(100)
Y1994	varchar(100)
Y1995	varchar(100)
Y1996	varchar(100)
Y1997	varchar(100)
Y1998	varchar(100)
Y1999	varchar(100)
Y2000	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)

Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)
Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

3. Stage_Medicaid_Aggregate

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y1991	varchar(100)
Y1992	varchar(100)

Y1993	varchar(100)
Y1994	varchar(100)
Y1995	varchar(100)
Y1996	varchar(100)
Y1997	varchar(100)
Y1998	varchar(100)
Y1999	varchar(100)
Y2000	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)
Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

4. Stage_Medicare_Enrollment

Stage_Column Name	Datatype, Size and Constraints
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y1991	varchar(100)
Y1992	varchar(100)
Y1993	varchar(100)
Y1994	varchar(100)
Y1995	varchar(100)
Y1996	varchar(100)
Y1997	varchar(100)
Y1998	varchar(100)
Y1999	varchar(100)
Y2000	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)

Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

5. Stage_PHI_Aggregate

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)

Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

6. Stage_PHI_Enrollment

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)

Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

7. Stage_US_Population

Stage_Column Name	**Datatype, Size and Constraints**
Code	varchar(100)
Item	varchar(100)
Group_Type	varchar(100)
Region_Number	varchar(100)
Region_Name	varchar(100)
State_Name	varchar(100)
Y1991	varchar(100)
Y1992	varchar(100)
Y1993	varchar(100)
Y1994	varchar(100)
Y1995	varchar(100)
Y1996	varchar(100)

Y1997	varchar(100)
Y1998	varchar(100)
Y1999	varchar(100)
Y2000	varchar(100)
Y2001	varchar(100)
Y2002	varchar(100)
Y2003	varchar(100)
Y2004	varchar(100)
Y2005	varchar(100)
Y2006	varchar(100)
Y2007	varchar(100)
Y2008	varchar(100)
Y2009	varchar(100)
Y2010	varchar(100)
Y2011	varchar(100)
Y2012	varchar(100)
Y2013	varchar(100)
Y2014	varchar(100)
Average_Annual_Percent_Growth	varchar(100)

8. Stage_Facilities

Stage_Column Name	**Datatype, Size and Constraints**
Facility_ID	varchar(100)
Facility_Name	varchar(100)
Address	varchar(100)

City	varchar(100)
State	varchar(100)
ZipCode	varchar(100)
County_Name	varchar(100)
Phone	varchar(100)
Hospital_Type	varchar(100)
Hospital_Ownership	varchar(100)
Emergency_Services	varchar(100)
Hospital_Overall_Rating	varchar(100)
Safety_Of_Care_National_Comparison	varchar(100)
Readmission_National_Comparison	varchar(100)
Patient_Experience_National_Comparison	varchar(100)
Effectiveness_Of_Care_National_Comparison	varchar(100)
Timeliness_Of_Care_NationalComparison	varchar(100)
Medical_Imaging_National_Comparison	varchar(100)
Location	varchar(100)

9. Stage_Physician

Stage_Column Name	**Datatype, Size and Constraints**
Profile_ID	varchar(100)
First_Name	varchar(100)
Last_Name	varchar(100)

Middle_Name	varchar(100)
Suffix	varchar(100)
Alternate_First_Name	varchar(100)
Alternate_Last_Name	varchar(100)
Alternate_Middle_Name	varchar(100)
Alternate_Suffix	varchar(100)
City	varchar(100)
State	varchar(100)
Zipcode	varchar(100)
Country	varchar(100)
Province_Name	varchar(100)
Physician_Field	varchar(100)
Physician_Speciality	varchar(100)
License_State_Code_1	varchar(100)
License_State_Code_2	varchar(100)
License_State_Code_3	varchar(100)
License_State_Code_4	varchar(100)
License_State_Code_5	varchar(100)

Data Profiling

The data profiling is performed on all staging tables examining the data available, to summarize the data and to collect statistics. This helps us perform data quality checks and perform data reformatting, whenever necessary.

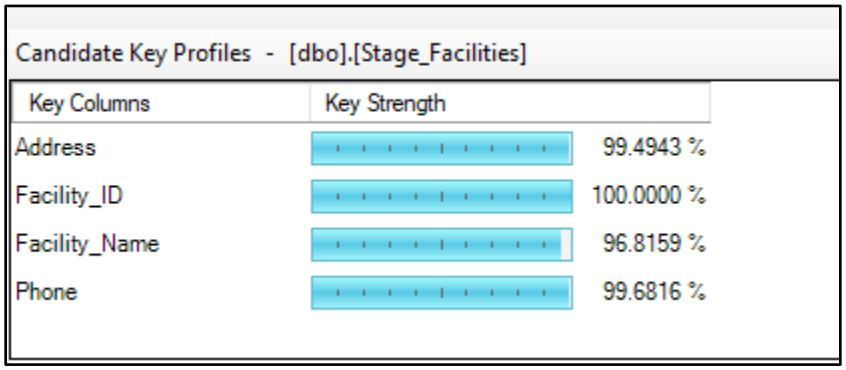
The different types of analysis performed are mentioned below:

- **Candidate key profiles**
- **Column Length Distribution profiles**
- **Column null ratio profiles**
- **Column value distribution profiles**

- **Column Pattern Profiles**

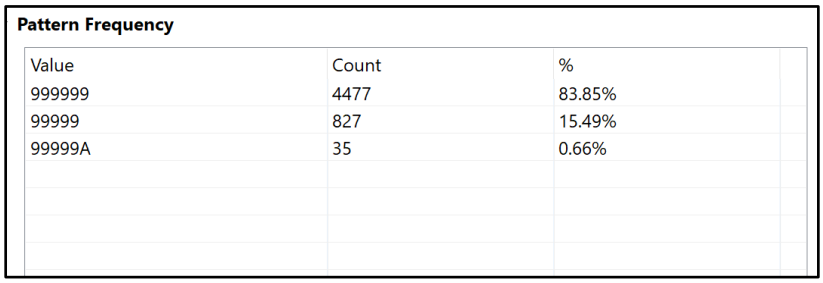
Stage_Facilities

Facility_ID shows 100% key Strength to act as a primary key. So,it gives an initial analysis of considering Facility_ID to act as primary key in the DW_Facilities table.


Candidate Key Profiles - [dbo].[Stage_Facilities]

Key Columns	Key Strength
Address	99.4943 %
Facility_ID	100.0000 %
Facility_Name	96.8159 %
Phone	99.6816 %

83.85 % of the Facility_IDs are formatted in 6-length digits. There are 15.49% Facility_IDs are formatted as 5 digits. Only 0.66% Facility_IDs are combined by 5 digits and one letter as tail.


Pattern Frequency

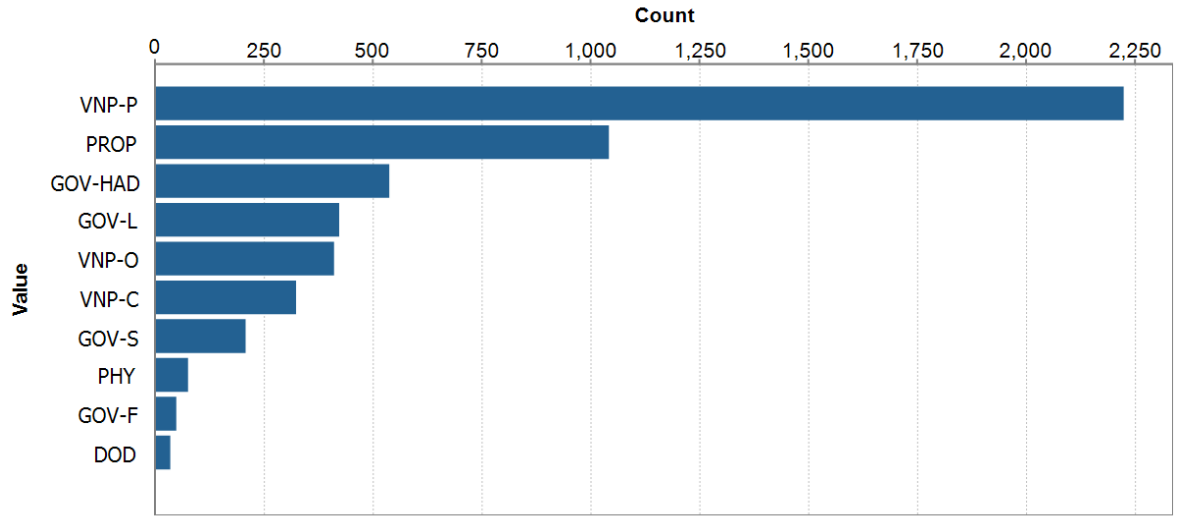
Value	Count	%
999999	4477	83.85%
99999	827	15.49%
99999A	35	0.66%

Hospital_Ownership has these unique values; it has 11 categories. Most frequent data is “Voluntary non-profit-Private” , which means 2224 hospitals are voluntary nonprofit..
The ownership with least no of hospitals is “Tribal” with only 9 hospitals associated with it.

Frequent Value Distribution (0.1000 %) - Hospital_Ownership — Encrypted Connection — 1000 Rows

Value	Count	Percentage
Proprietary	1042	19.5168 %
Government - State	208	3.8959 %
Tribal	9	0.1686 %
Government - Hospital District or Authority	538	10.0768 %
Department of Defense	35	0.6556 %
Voluntary non-profit - Private	2224	41.6557 %
Physician	76	1.4235 %
Government - Local	423	7.9228 %
Government - Federal	49	0.9178 %

Hospital OwnershipCode is an abbreviation for Hospital Ownership. It is observed every hospital ownership type has its own code.

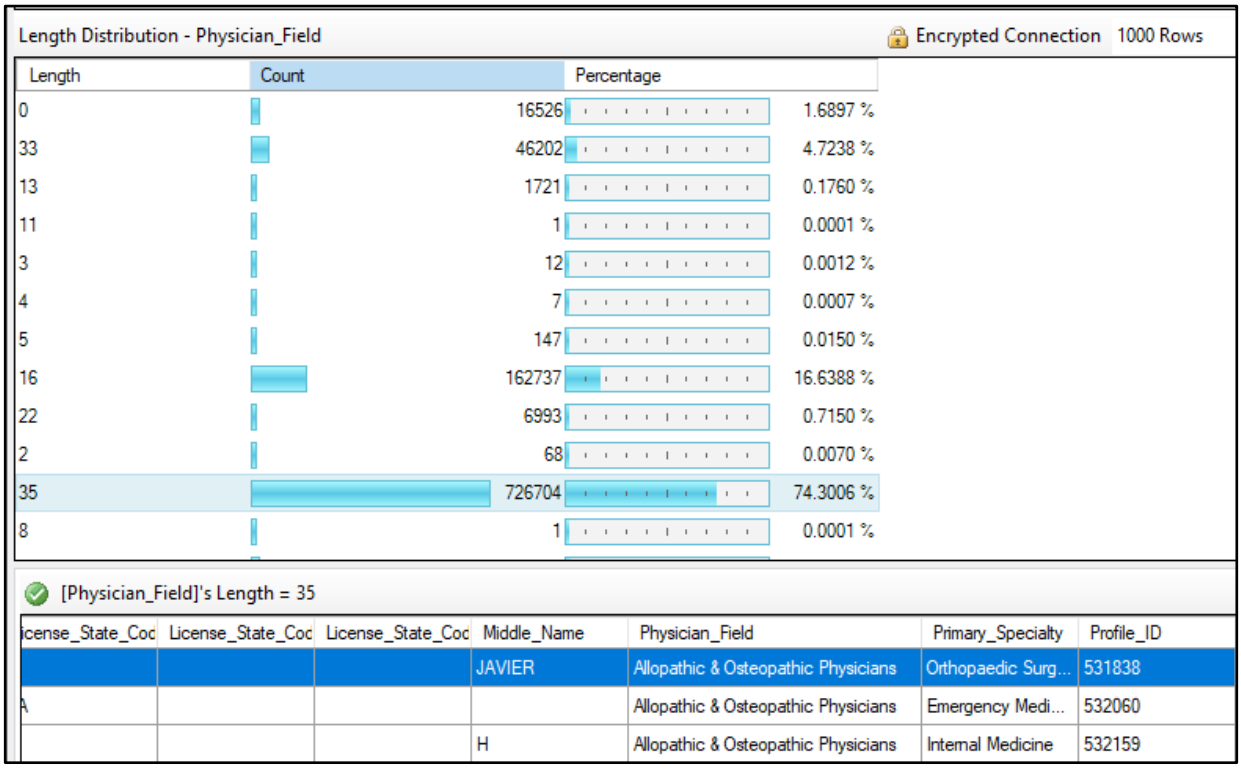


Stage_Physician

Physicians ID are all digits and its length is between 1 and 7 digits.

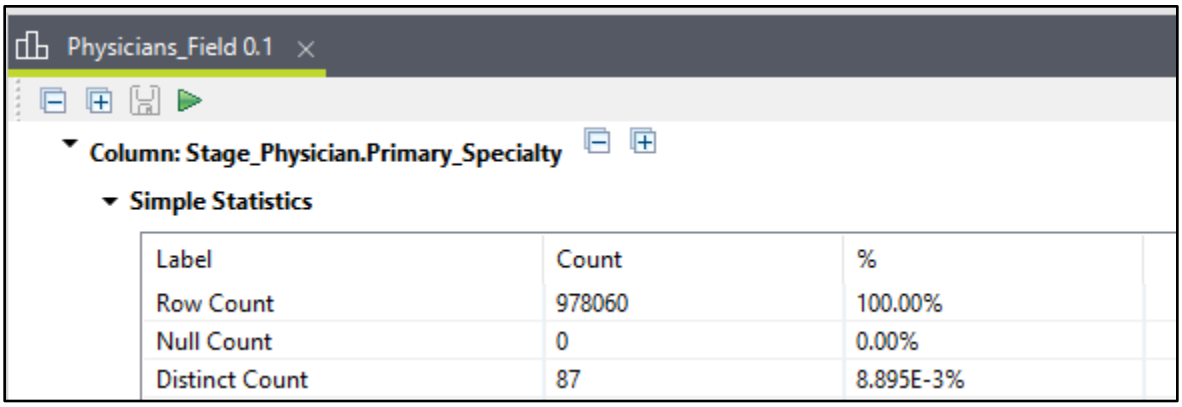
Value	Count	%
99	90	N/A
999999	635004	N/A
99999	89942	N/A
999	899	N/A
9	9	N/A
9999	8995	N/A
9999999	300594	N/A

Physician field has maximum length data as 35 chars and the value for this maximum length field is "Allopathic & Osteopathic Physicians". It gives an initial analysis of minimum and maximum length fields to determine the column length size in the data warehouse.



Pattern Analysis for Physician Specialty

This shows us that physician specialty has 87 unique values/ categories. There are no null values and



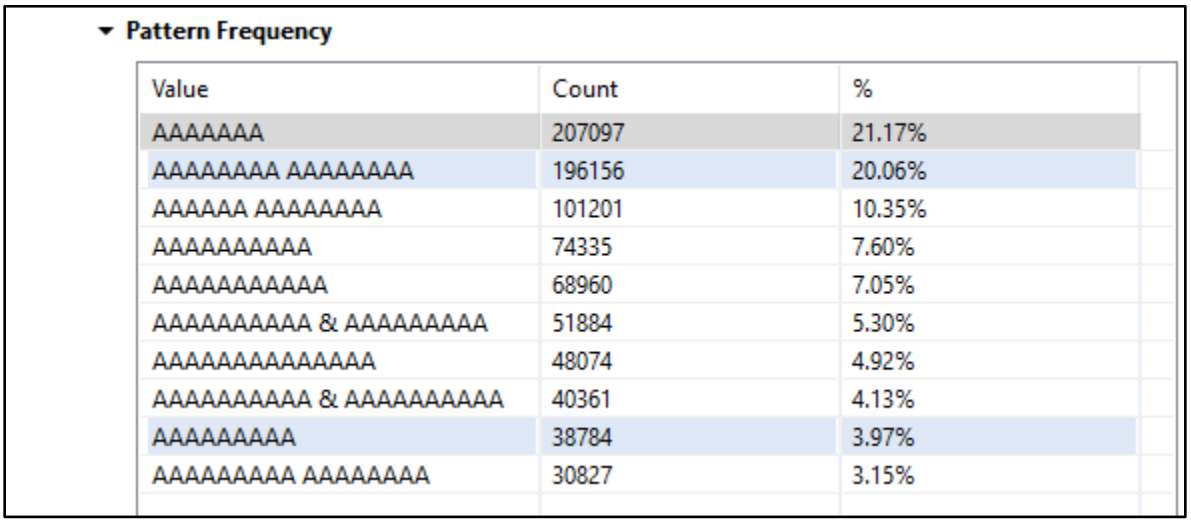
Physicians_Field 0.1

Column: Stage_Physician.Primary_Specialty

Simple Statistics

Label	Count	%
Row Count	978060	100.00%
Null Count	0	0.00%
Distinct Count	87	8.895E-3%

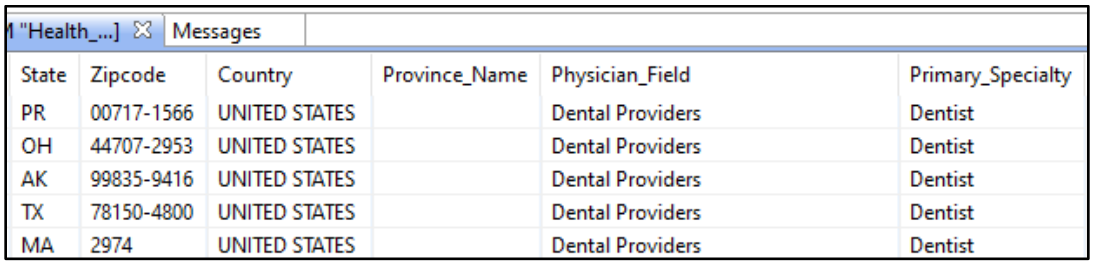
It shows different types of patterns and their counts for Physician specialty. The analysis shows the percentage of every pattern existing in data. So, it gives a basic idea of possible pattern values.



Pattern Frequency

Value	Count	%
AAAAAAA	207097	21.17%
AAAAAAAA AAAAAAAA	196156	20.06%
AAAAAA AAAAAAAA	101201	10.35%
AAAAAAAAAA	74335	7.60%
AAAAAAAAAAA	68960	7.05%
AAAAAAAAAA & AAAAAAAAA	51884	5.30%
AAAAAAAAAAAAAA	48074	4.92%
AAAAAAAAAA & AAAAAAAAAA	40361	4.13%
AAAAAAAAA	38784	3.97%
AAAAAAAAA AAAAAAAA	30827	3.15%

The records having the first pattern value “AAAAAAA” and the corresponding data “Dentist” for specialty are:



1 "Health_...] Messages

State	Zipcode	Country	Province_Name	Physician_Field	Primary_Specialty
PR	00717-1566	UNITED STATES		Dental Providers	Dentist
OH	44707-2953	UNITED STATES		Dental Providers	Dentist
AK	99835-9416	UNITED STATES		Dental Providers	Dentist
TX	78150-4800	UNITED STATES		Dental Providers	Dentist
MA	2974	UNITED STATES		Dental Providers	Dentist

Stage_Medicaid_Enrollment

Group has only three types which are United States, Region and State in all Enrollment and Aggregate datasets.

Value Frequency

Value	Count	%
State	51	85.00%
Region	8	13.33%
United States	1	1.67%

Data Warehouse Tables:

- Implemented data cleansing, data preprocessing on raw data.
- The data warehouse is built out by following 3NF design principles representing the entities, attributes and relationships from the staging tables that are relevant to BI.
- Created slowly changing dimension (SCD) data models to track changes to data warehouse tables.

The list of DW tables in our project:

DWLookupState

DWLookupstate contains the Statecode, StateName and Region Number data for 62 states in the United States(50 States, Federal District and Territories). This data is collected from Wikipedia and has been loaded into the database using the SSIS package.

Column Name	Datatype, Size and Constraints	Keys
StateCode	varchar (2), NOTNULL	PrimaryKey
StateName	varchar (50), NOT NULL	
RegionNumber	int, NOTNULL	

DWHealthInsuranceEnrollment

DWHealthInsuranceEnrollment table has data coming from three stage tables (Stage_Medicare_Enrollment,Stage_Medicaid_Enrollment, Stage_PHI_Enrollment). These three tables are merged together by doing a union of all the data. These three tables are combined because they all have the same schema and all of them contain enrollment data. To identify the enrollment data appropriately, a new column Insurance Type is introduced. This column contains categorical information (Medicare, Medicaid, Private). A auto-incremented surrogate key is introduced and it behaves as the primary key of this table.

Source Table	**Source Column**	**Destination Column**	**DataType, Size, Constraint**	**Key**	**Transformation**
		SKEnrollID	int , NOT NULL	Primary Key	Auto-Incremented
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	Code	Code	int , NOT NULL		Data Type Conversion
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	Item	Item	varchar(75), NOT NULL		Length Conversion
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	GroupType	GroupType	varchar(15), NOT NULL		Length Conversion
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	RegionName	RegionName	varchar(50),NOTNULL		Length Conversion
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	StateName	StateCode	varchar(2), NOTNULL	Foreign Key	Lookup performed on DWLookupState using StateName to obtain StateCode
		Insurance Type	varchar(8), NOTNULL		Used a Derived column to get the type of insurance . It has three types Medicare, Medicaid, Personal
Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	Column Names from Y1991-Y2014	EnrollYear	int, NOTNULL		Transformation using Unpivot to convert each year column into one Enrollyear column which carries the previous year columns names as values

Stage_Medicare_Enrollment, Stage_Medicaid_Enrollment, Stage_PHI_Enrollment	Values from Y1991-Y2014	EnrollCount	int, NOTNULL		Transformation using unpivot to get count from multiple columns year columns into one Count column
		LoadDate	Date, NOT NULL		Used GetDate() function in Derived column to get the date on load of data

DWHealthExpenditureAggregate

DWHealthInsuranceAggregate table has data coming from three stage tables (Stage_Medicare_Aggregate,Stage_Medicaid_Aggregate, Stage_PHI_Aggregate). These three tables are merged together by doing a union of all the data. These three tables are combined because they all have the same schema and all of them contain aggregate expenditure data. To identify the expenditure data appropriately, a new column Insurance Type is introduced. This column contains categorical information (Medicare, Medicaid, Private). An auto-incremented surrogate key is introduced and it behaves as the primary key of this table.

Source Table	**Source Column**	**Destination Column**	**DataType, Size, Constraint**	**Key**	**Transformation**
		SKAggregateID	int , NOT NULL	Primary Key	Auto-Incremented
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	Code	Code	int , NOT NULL		Data Type Conversion
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	Item	Item	varchar(75), NOT NULL		Length Conversion
Stage_Medicare_Aggregate,	GroupType	GroupType	varchar(15), NOT NULL		Length Conversion

Stage_Medicaid_Aggregate, Stage_PHI_Aggregate					
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	RegionName	RegionName	varchar(50), NOTNULL		Length Conversion
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	StateName	StateCode	varchar(2), NOTNULL	Foreign Key	Lookup performed on DWLookupState using StateName to obtain StateCode
		InsuranceType	varchar(8), NOTNULL		Used a Derived column to get the type of insurance . It has three types Medicare, Medicaid, Personal
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	Column Names from Y1991-Y2014	AggregateYear	int, NOTNULL		Transformation using Unpivot to convert each year column into one AggregateYear column which carries the previous year columns names as values
Stage_Medicare_Aggregate, Stage_Medicaid_Aggregate, Stage_PHI_Aggregate	Values from Y1991-Y2014	Aggregate Count	int, NOTNULL		Transformation using unpivot to get count from multiple columns year columns into one Count column
		LoadDate	Date, NOT NULL		Used GetDate() function in Derived column to get the date on load of data

DWUSPopulation

This table contains the US Population counts for the years 1991-2014. A surrogate key is introduced which is auto incremented and this acts as a primary key.

Source Table	Source Column	Destination Column	DataType, Size, Constraint	Key	Transformation
		SKPOPID	int , NOT NULL	Primary Key	Auto-Incremented
Stage_US_ Population	Code	Code	int , NOT NULL		Data Type Conversion
Stage_US_ Population	Item	Item	varchar(75), NOT NULL		Length Conversion
Stage_US_ Population	GroupType	GroupType	varchar(15), NOT NULL		Length Conversion
Stage_US_ Population	RegionName	RegionName	varchar(50), NOTNULL		Length Conversion
Stage_US_ Population	StateName	StateCode	varchar(2), NOTNULL	Foreign Key	Lookup performed on DWLookupState using StateName to obtain StateCode
Stage_US_ Population	Column Names from Y1991-Y2014	Year	int, NOTNULL		Transformation using Unpivot to convert each year column into one Year column which carries the previous year columns names as values
Stage_US_ Population	Values from Y1991-Y2014	PopCount	int, NOTNULL		Transformation using unpivot to get count from multiple columns year columns

					into one Count column
		LoadDate	Date, NOT NULL		Used GetDate() function in Derived column to get the date on load of data

DWLookupZipCode

DWLookupZipCode table contains all geographic information related to zip code in three staging tables (Stage_Medicare_Enrollment,Stage_Medicaid_Enrollment, Stage_PHI_Enrollment). A surrogate key ZipCode_ID is the identity primary key which is automatically generated while loading this table. StateCode is a foreign key from the DWLookupState table. The raw data file for this table comes from the USPS website.

Column Name	**Datatype, Size and Constraints**	**Keys**
ZipCode_ID	int, NOTNULL, AutoIncremented	PrimaryKey
Zipcode	varchar(5), NOTNULL	
City	varchar(50),NOTNULL	
StateCode	varchar(2),NOTNULL	ForeignKey

DWHospitalInformation

This has all the hospital data; this data is moved over from the Stage_Facilities with transformations as mentioned below. A few columns that had repetitive and non-uniform data are not moved to this table from the stage table.

Source Table	**Source Column**	**Destination Column**	**DataType, Size, Constraint**	**Key**	**Transformation**
Stage_Facilities	Facility_ID	FacilityID	varchar(10),NOT NULL	PrimaryKey	Length Conversion

Stage_Facilities	Facility_Name	FacilityName	varchar(100),NOTNULL		Copy Over
Stage_Facilities	Address	Address	varchar(100),NOTNULL		Copy Over
Stage_Facilities	Phone	PhoneNumber	varchar(14), NOTNULL		Length Conversion
Stage_Facilities	Zipcode	ZipcodeID	int	ForeignKey	Lookup on DWLookupZipcode using ZipCode to get ZipcodeID
Stage_Facilities	Hospital_Type	HospitalType	varchar(35), NOTNULL		Length Conversion
Stage_Facilities	Hospital_Ownership	HospitalOwnership	varchar(50), NOTNULL		Length Conversion
Stage_Facilities	Emergency_Services	EmergencyServices	varchar(6)		Length Conversion
Stage_Facilities	Hospital_Overall_Rating	HospRating	varchar(5)		Length Conversion
Stage_Facilities	Safety_Of_Care_National_Comparison	SafteyofCareRating	varchar(30)		Length Conversion
Stage_Facilities	Readmission_National_Comparison	ReadmissionRating	varchar(30)		Length Conversion
Stage_Facilities	Patient_Experience_National_Comparison	PatientExpRating	varchar(30)		Length Conversion
Stage_Facilities	Effectiveness_Of_Care_National_Comparison	EffectivenessOfCareRating	varchar(30)		Length Conversion
Stage_Facilities	Timeliness_Of_Care_NationalComparison	TimelinessOfCareRating	varchar(30)		Length Conversion

Stage_Facilities	Medical_Imaging_National_Comparison	MedicalImagingRating	varchar(30)		Length Conversion
Stage_Facilities	Location	Location	varchar(30)		Length Conversion
		StartDate	Date		Used GetDate() function in Derived column to get the date on load of data
		EndDate	Date		updated by SCD during historical changes
		LastUpdateDate	Date		updated by SCD on changes

DWPhysicianDetails

DW PhysicianDetails contains all the physician personal information and medical field and speciality. This information is moved over from stage_Physician. Columns like the country name is removed because all this information is for the United States and the country name contains “United States”.

Source Table	Source Column	Destination Column	DataType, Size, Constraint	Key	Transformation
Stage_Physician	Profile_ID	PhysicianID	int	Primary key	Length Conversion
Stage_Physician	First_Name	FirstName	varchar(50)		Length Conversion
Stage_Physician	Middle_Name	MiddleName	varchar(50)		Length Conversion
Stage_Physician	Last_Name	LastName	varchar(50)		Length Conversion
Stage_Physician	Suffix	Suffix	varchar(5)		Length Conversion

Stage_Physician	Address_Line_1	AddressLine1	varchar(100)		Copy Over
Stage_Physician	Address_Line_2	AddressLine2	varchar(100)		Copy Over
Stage_Physician	Zipcode	Zipcode	varchar(5)	Foreign key	Length Conversion
Stage_Physician	Physician_Field	PhysicianField	varchar(50)		Length Conversion
Stage_Physician	Primary_Specialty	PhysicianSpeciality	varchar(50)		Length Conversion
Stage_Physician		StartDate	Date		Used GetDate() function in Derived column to get the date on load of data
Stage_Physician		EndDate	Date		Updated by SCD during historical changes
Stage_Physician		LastUpdateDate	Date		Updated by SCD on all changes

DWLicensceStateCode

DWLicenseStateCode holds the data of state codes for which the physicians have licenses for. This is moved over from the Stage_Facilities table. Each physician can have licenses for upto 5 license states. The unpivot transformation is applied on DWPhysician Table to load this table

Column Name	**Datatype, Size and Constraints**	**Keys**
PhysicianID	int	Foreign Key
LicenseStateCodeNumber	int	
StateCode	varchar(2)	Foreign Key

DIM and FACT Tables for Dimensional Model

1. DIM ItemCode

DIM ItemCode contains Item and ItemCode data. This comes from DWHealthInsuranceEnrollment and DW HealthExpenditureAggregate. We populate Item by selecting the unique category values from the mentioned DW tables.

Source Table	Source Column	Column Name	Data Type and Constraints	Keys	Transformation
		ItemCode	Int, NOT NULL	Primary Key, AutoIncrement	
DWHealthInsuranceEnrollment and DWHealthExpenditureAggregate	Item	Item	varchar(100),NOT NULL, UNIQUE		DISTINCT Item values from the mentioned source tables.

2. DIM Geography

DIM Geography contains the State and Region associations data.This data is coming from DWLookupState table andDWHealthInsuranceEnrollment. A left outer join on statecode is done on the mentioned tables to combine the state and region information. This merge will help make appropriate state and region relations.

Source Table	Source Column	Column Name	Data Type and CONSTRAINTS	Keys	Transformation
DWLookupState	StateCode	StateCode	varchar(5), NOT NULL	Primary Key	Copy Over
DWLookupState	StateName	StateName	varchar(50), NOT NULL		Copy Over
DWLookupState	Region Number	Region Number	int, NOT NULL		Copy Over
DWHealthInsuranceEnrollment	Region_Name	Region Name	varchar(50), NOT NULL		A left outer join on StateCode on DWLookupState and DWHealthInsuranceEnrollment is done and DISTINCT Region Name are used to populate this column

3. DIM Zipcode

DIM Zipcode table comes from the DWLookupZipcode table, it is copied over as is.

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
DWLookupZipcode	ZipcodeID	ZipcodeID	int, NOT NULL	PrimaryKey, AutoIncrement	Copy Over
DWLookupZipcode	Zipcode	Zipcode	varchar(5),NOT NULL,UNIQUE		Copy Over
DWLookupZipcode	City	City	varchar(50), NOT NULL		Copy Over
DWLookupZipcode	StateCode	StateCode	varchar(2), NOT NULL	Foreign Key	Copy Over

4. DIM Date

DIM Date table contains all the Years used across multiple tables in the data warehouse. DateKey is a primary key and is auto incremented on load.

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
		DateKey	int, NOT NULL	PrimaryKey, AutoIncrement	
DW	Year Columns	Year	varchar(5), NOT NULL		copy over

5. DIM PhysicianField

DIM PhysicianField is from PhysicianField in DWPhysicianDetails table and contains all Physician Field of physicians.

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
		PhysicianFieldID	int	PrimaryKey, AutoIncrement	
DWPhysicianDetails	PhysicianField	PhysicianField	varchar(35)		DISTINCT PhysicianField

6. DIM PhysicianSpeciality

DIM PhysicianSpeciality is from PhysicianSpeciality in DWPhysicianDetails data file and contains all Physician Speciality of physicians.

Source Table	Source Column	DimColumn Name	Data Type	Keys	Transformation
		PhysicianSpecialityID	int	PrimaryKey, AutoIncrement	
DWPhysicianDetails	PhysicianSpeciality	PhysicianSpeciality	varchar(50)		DISTINCT PhysicianDetails

7. DIM PhysicianDetails

DIM PhysicianDetails is from DWPhysicianDetails table. It contains all information of physicians in detail. It is populated by performing a lookup on Physician Field ID and Physician Speciality ID from the respective tables.

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
DWPhysicianDetails	Profile_ID	PhysicianID	int	PrimaryKey, AutoIncrement	
DWPhysicianDetails	First_Name	FirstName	varchar(50)		
DWPhysicianDetails	Middle_Name	MiddleName	varchar(50)		
DWPhysicianDetails	Last_Name	LastName	varchar(50)		

DWPhysicianDetails	Suffix	Suffix	varchar(5)		
DWPhysicianDetails	Address_Line_1	AddressLine1	varchar(100)		
DWPhysicianDetails	Address_Line_2	AddressLine2	varchar(100)		
DWPhysicianDetails	Zipcode	ZipcodeID	int	Foreign Key	
DWPhysicianDetails	Physician_Field	PhysicianFieldID	int	Foreign Key	
DWPhysicianDetails	Primary_Specialty	PhysicianSpecialityID	int	Foreign Key	

8. **DIM License Code:**

It contains License Information for all the physicians and it is copied over from DWLicenseStateCode.

Source Table	**Source Column**	**Column Name**	**Data Type**	**Keys**	**Transformation**
DWLicenseStaeCode	PhysicianID	PhysicianID	int	ForiegnKey	
DWLicenseStaeCode	LicenseStateCOdeNo	LicenseStateCOdeNo	varchar(10)		
DWLicenseStaeCode	LicenseSateCode	LicenseSateCode	int	FOreignKKey	

9. **DIM HospitalInfo:** It contains all the hospital information and is populated from DWHospitalInfo by performing the lookup on HospitalTypeID and HospitalOwnership.

Source Table	**Source Column**	**Column Name**	**Data Type**	**Keys**	**Transformation**
DWHospitalInformation	FacilityID	FacilityID	varchar(10)	Primary Key	

DWHospitalInformation	FacilityName	FacilityName	varchar(50)		
DWHospitalInformation	Address	Address	varchar(50)		
DWHospitalInformation	PhoneNumber	PhoneNumber	varchar(14)		
DWHospitalInformation	ZipCodeID	ZipCodeID	int	Foreign Key	
DWHospitalInformation	HospitalType	HospitalTypeID	int	Foreign Key	Lookup from DIM hospitalType
DWHospitalInformation	HopitalOwnership	HopitalOwnershipID	int	Foreign Key	Lookup fromDIM hospital ownership
DWHospitalInformation	EmergencyServices	EmergencyServices	varchar(5)		
DWHospitalInformation	HospRating	HospRating	varchar(6)		
DWHospitalInformation	MortalilityRating	MortalilityRating	varchar(30)		
DWHospitalInformation	SafetyOfCareRating	SafetyOfCareRating	varchar(30)		
DWHospitalInformation	ReadmissionRating	ReadmissionRating	varchar(30)		
DWHospitalInformation	PatientExpRating	PatientExpRating	varchar(30)		
DWHospitalInformation	EffectivenessOfCareRating	EffectivenessOfCareRating	varchar(30)		
DWHospitalInformation	TimelinessOfCareRating	TimelinessOfCareRating	varchar(30)		
DWHospitalInformation	MedicalImagingRating	MedicalImagingRating	varchar(30)		

DWHospitalInformation	Location	Location	varchar(10)		

10. FACT Aggregate:

This table contains the Expenditure details and is populated by performing a lookup on year and InsuranceTypeID

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
DWHEalthInsuranceAggregate	SKAggregateID	SKAggregateID	int	PrimaryKey	
DWHEalthInsuranceAggregate	Code	ItemCode	int	ForeignKey	
DWHEalthInsuranceAggregate	StateCode	StateCode	varchar(2)	ForeignKey	
DWHEalthInsuranceAggregate	InsuranceType	InsuranceTypeID	int	ForeignKey	Lookup on Insurance type
DWHEalthInsuranceAggregate	AggegateYear	DateKEy	int	ForeignKey	Lookup on DIm Date
DWHEalthInsuranceAggregate	AggregateCount	AggregateCount	int		

11. FACT Enrollment:

This table gives the enrollment count for all the states and is populated by doing a lookup on year and InsuranceTypeID

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
DWHealthinsuranceEnrollment	SKEnrollID	SKEnrollID	int	PrimaryKey	
	Code	ItemCode	int	ForeignKey	
	InsuranceType	InsuranceTypeID	int	ForeignKey	Lookup on Insurance type
	StateCode	StateCode	varchar(2)	ForeignKey	
	EnrollCount	EnrollCount	int		
	EnrollYear	DateKey	int	ForeignKey	Lookup on DIm Date

12. FACT Population:

This table contains the US population information and is populated by doing a lookup on year.

Source Table	Source Column	Column Name	Data Type	Keys	Transformation
DWUSPopulation	SKPopID	SKPopID	int	PrimaryKey	
DWUSPopulation	Code	ItemCode	int	ForeignKey	
DWUSPopulation	StateCode	StateCode	varchar(2)	ForeignKey	
DWUSPopulation	Year	DateKEy	int	ForeignKey	Lookup on DIm Date
DWUSPopulation	PopCount	PopCount	int		

Data Load Process

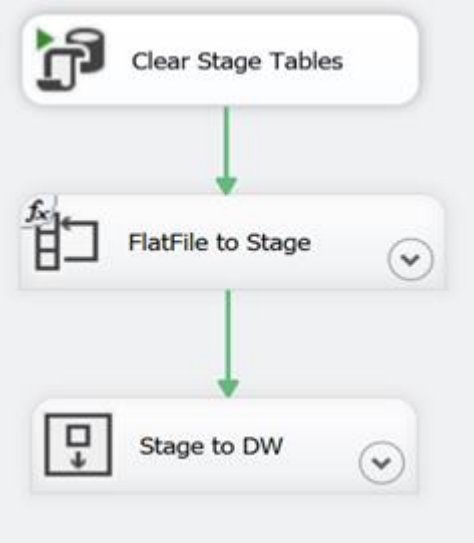
The data load is performed in following three steps:

1) **Initial Load:** In this step, first a clear stage step is performed to ensure we do not have any old data everytime we load stage data.Staging tables are loaded with 1:1 mapping from all the csv files. Thereafter, DW tables are loaded with a load date column by applying mapping transformations and performing data preprocessing and cleansing to staging data
2) **Incremental load:** In this step, Slowly changing Dimensions(SCD) transformations are applied to DW tables to track the historical changes with a last updated date during every load process.
3) **EDW to Dimensional Model(Snowflake Schema):** In this step, a snowflake dimensional model is built out and the dim and fact tables are loaded by applying joins and appropriate transformations as required for the design.

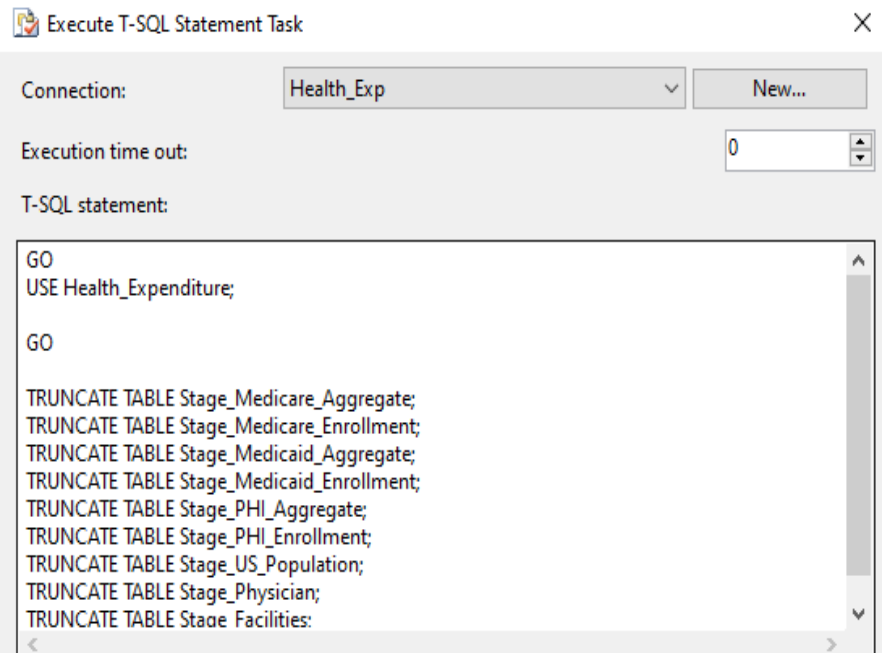
The detailed explanation of above 3 steps is shown below with snapshots from SSIS workflow:

An overview snapshot of Initial Load below: The stage tables are truncated first and then the stage tables are loaded by using for each loop and then DW tables are loaded by using required mapping transformations. Every Data flow is explained in detail below:

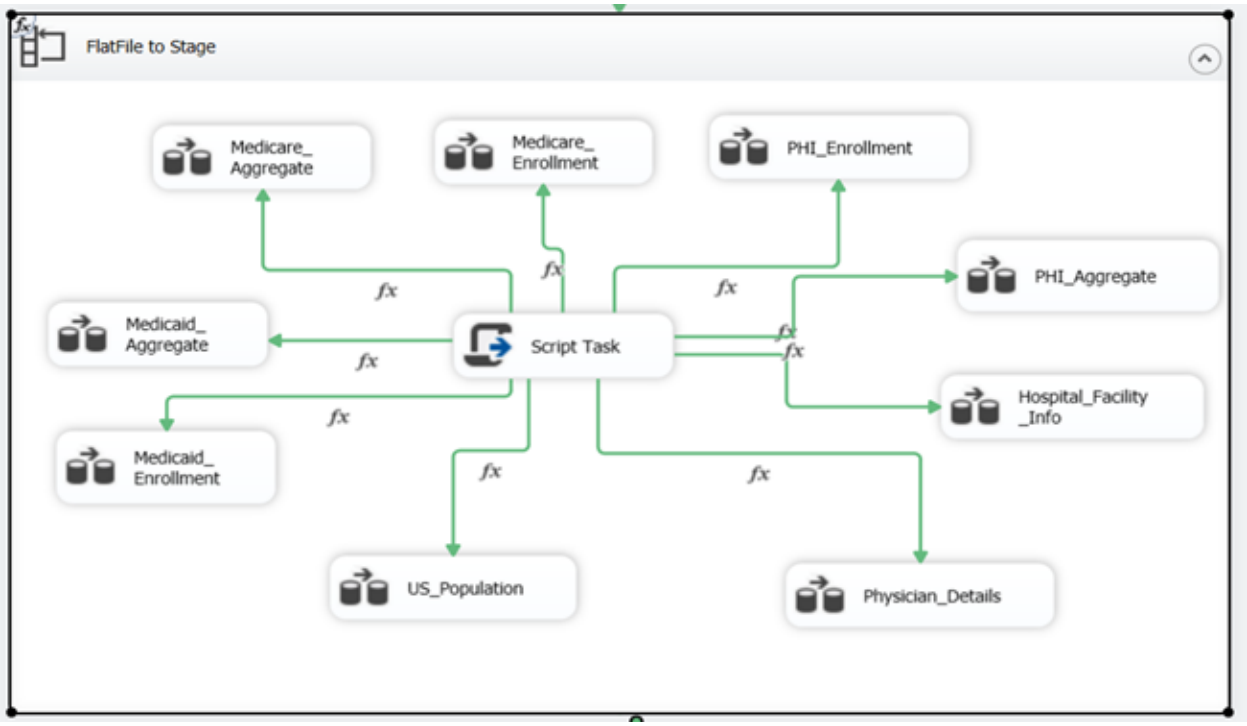
Step 1: Initial Data Load:



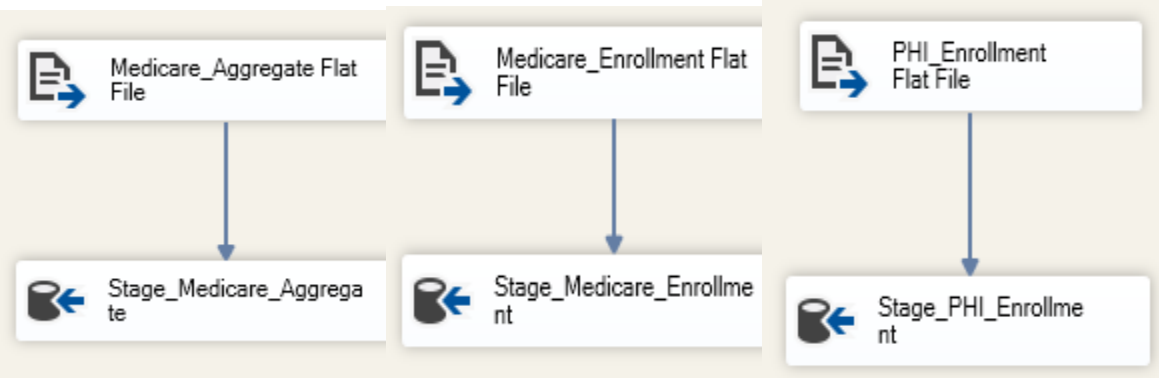
1. **Truncate the Stage Tables using T-SQL query:** Truncating all the stage tables before a new load process starts. This is done to remove all the old data in the staging area before we process the new data and load into the data warehouse.



2. Populate all stage tables using For-each loop: The For each loop is used to populate all the stage tables in a loop so that individual execution is not required for every data flow. The script task is used to define the expression where the file name is given for every flat file corresponding to a stage table.

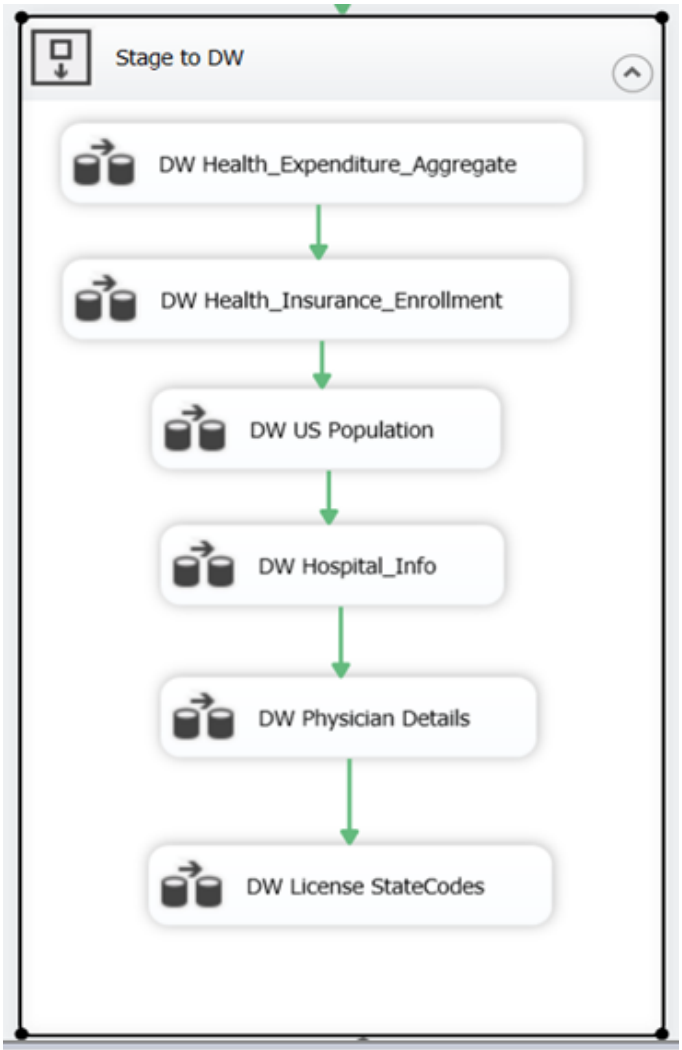


Data flow task for each stage table is similar to shown below:



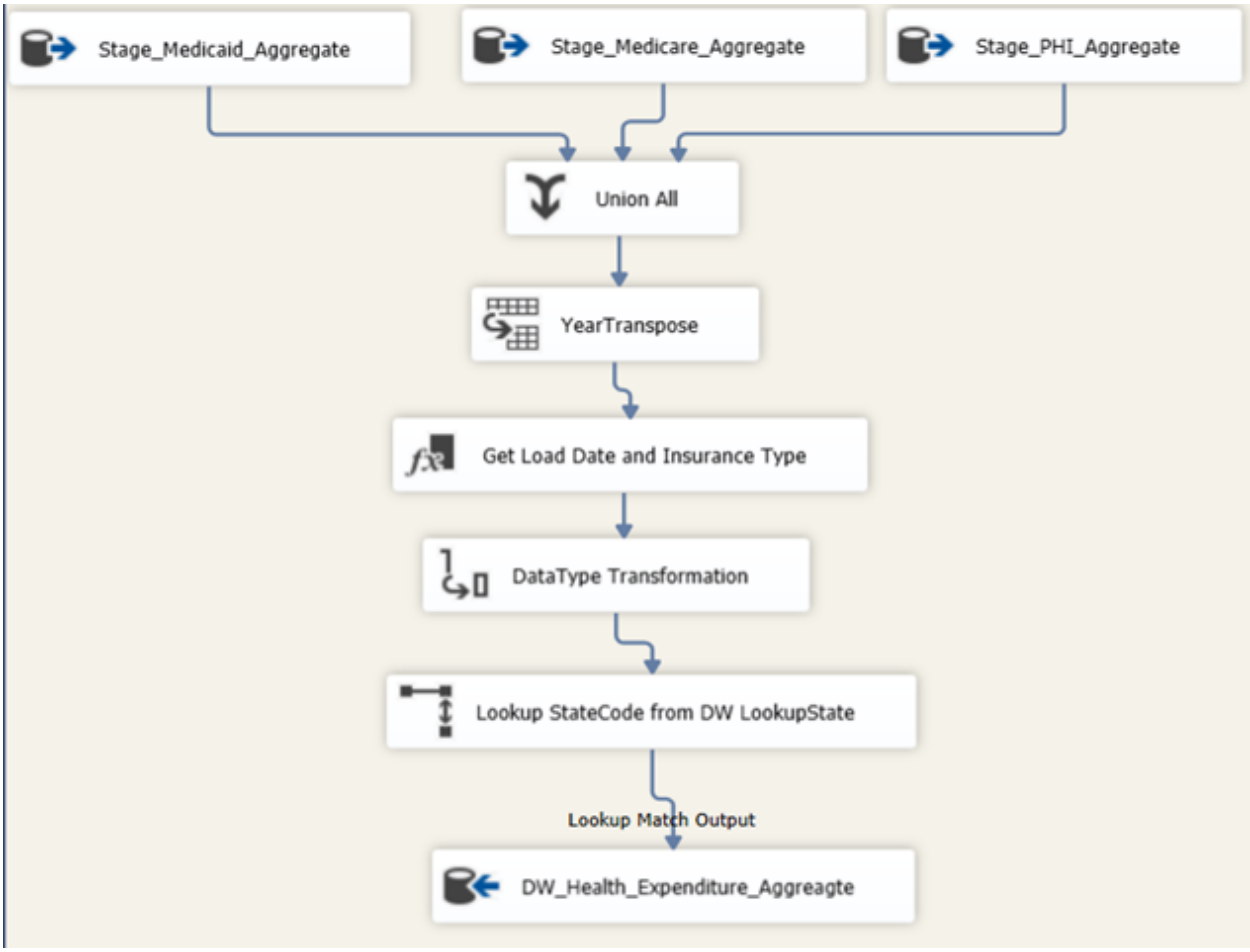
3. Load an Integrated Data warehouse by Data conversions and Data transformations

The DW tables are loaded in a sequence container as shown below:The detailed data flow is also explained below.



DW Health_Expenditure_Aggregate is transformed and populated with a load date in the following steps:

1. A union all is performed on stage_medicaid_aggregate, stage_medicare_aggregate and stage_phi_aggregate.
2. Unpivot transformation is applied on union output to transform the data so as to have multiple years data in different rows instead of a single row and build a robust data warehouse.
3. Load date is added as a derived column using getdate() method and Insurance type is also added as a derived column by extracting the substring from Item column.
4. The data conversion is used to make group_type and region name of appropriate length.
5. The Lookup Transformation is used to lookup state names from the DWlookupState table and then matched output records are loaded into the DW table.



DW Health Insurance Enrollment is transformed and populated with a load date as follows:

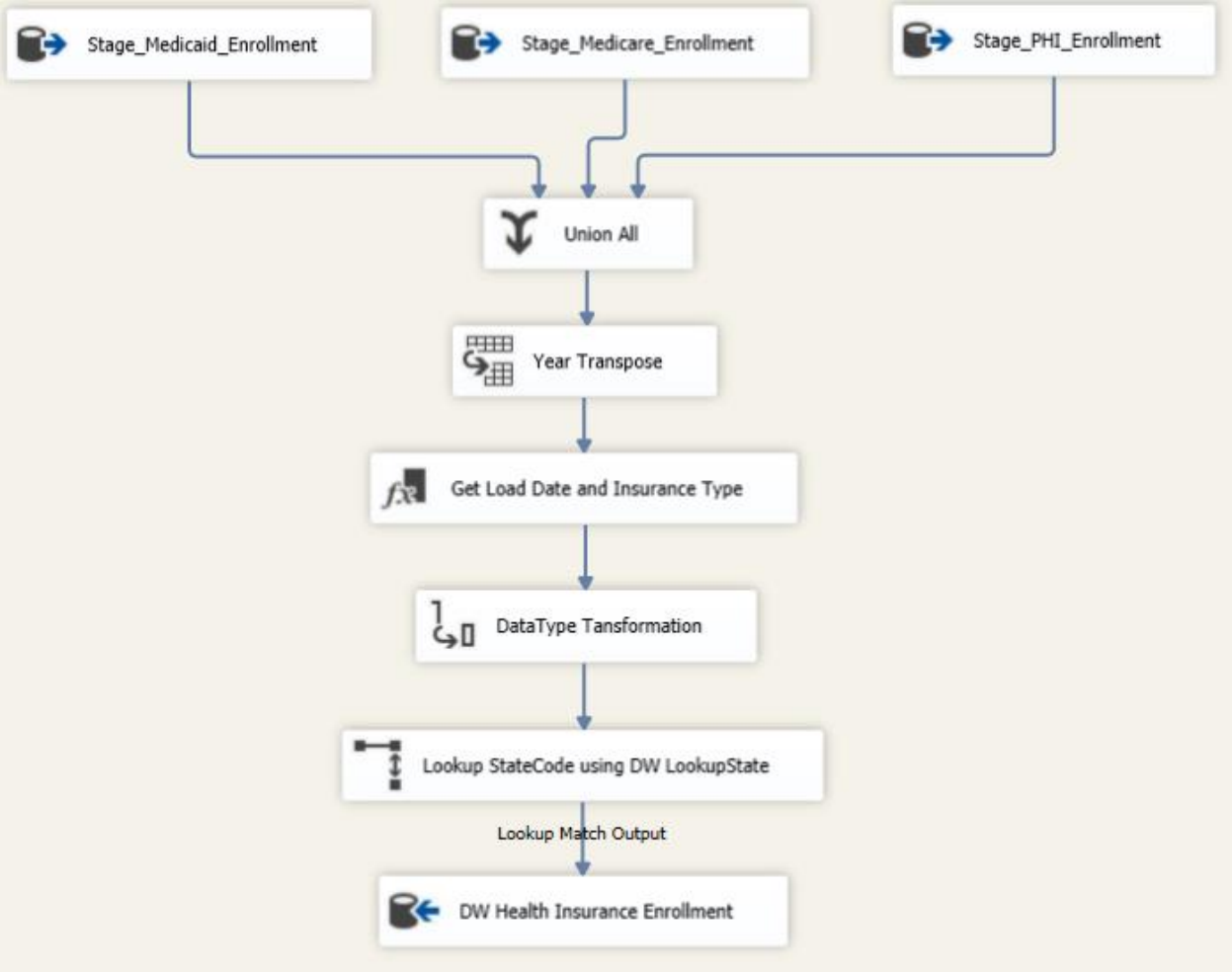
1. A union all is performed on stage_medicaid_enrollment, stage_medicare_enrollment and stage_phi_enrollment.

2. Unpivot transformation is applied on union output to transform the data so as to have multiple years data in different rows instead of a single row and build a robust data warehouse.

3. Load date is added as a derived column using getdate() method and Insurance type is also added as a derived column by extracting the substring from Item column.

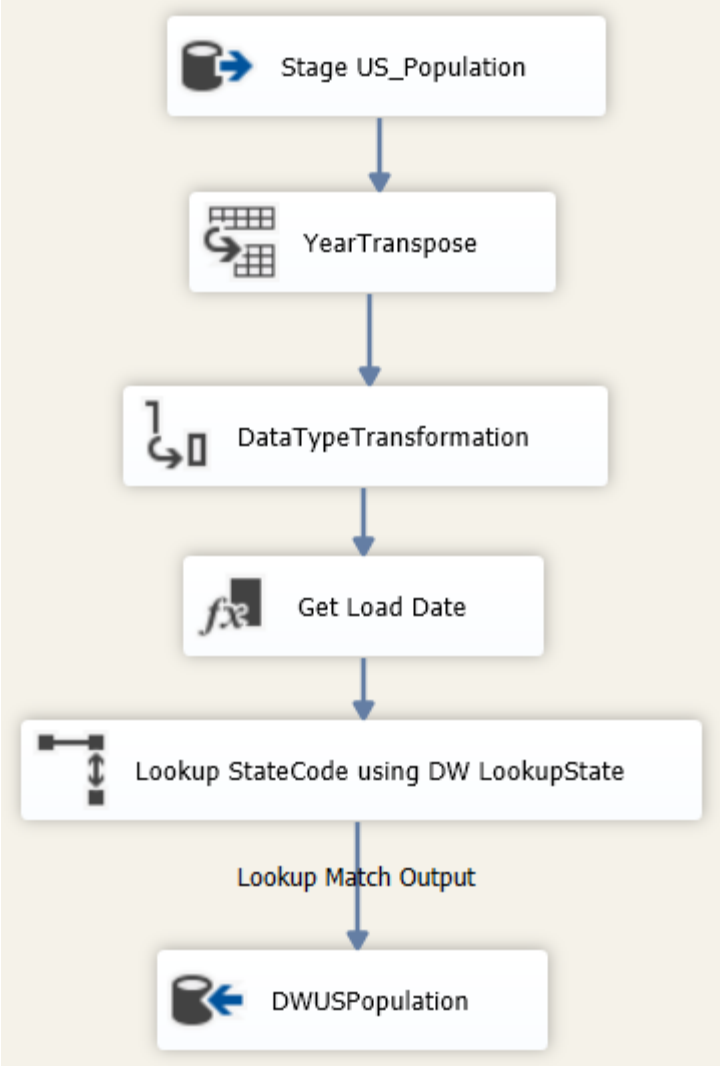
4. The data conversion is used to make group_type and region name of appropriate length.

5. The Lookup Transformation is used to lookup state names from the DWlookupState table and then matched output records are loaded into the DW table.

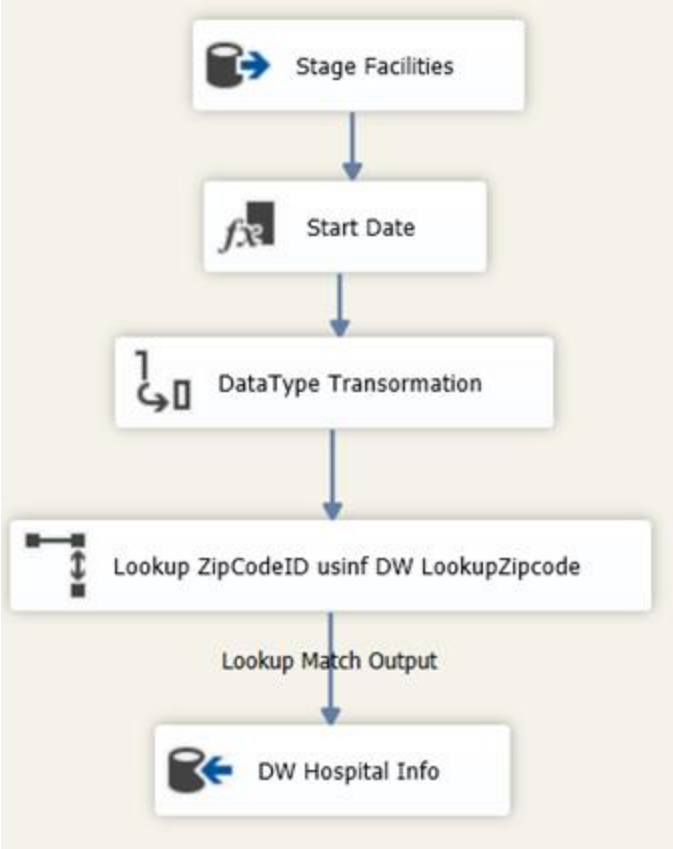


DW US Population is transformed and populated with a load date as follows:

1. Unpivot transformation is applied on stageUS_population to transform the data so as to have multiple years data in different rows instead of a single row and build a robust data warehouse.
2. Load date is added as a derived column using getdate() method.
3. The data conversion is used to make the region name of appropriate length.
4. The Lookup Transformation is used to lookup state names from the DWlookupState table and then matched output records are loaded into the DW table.

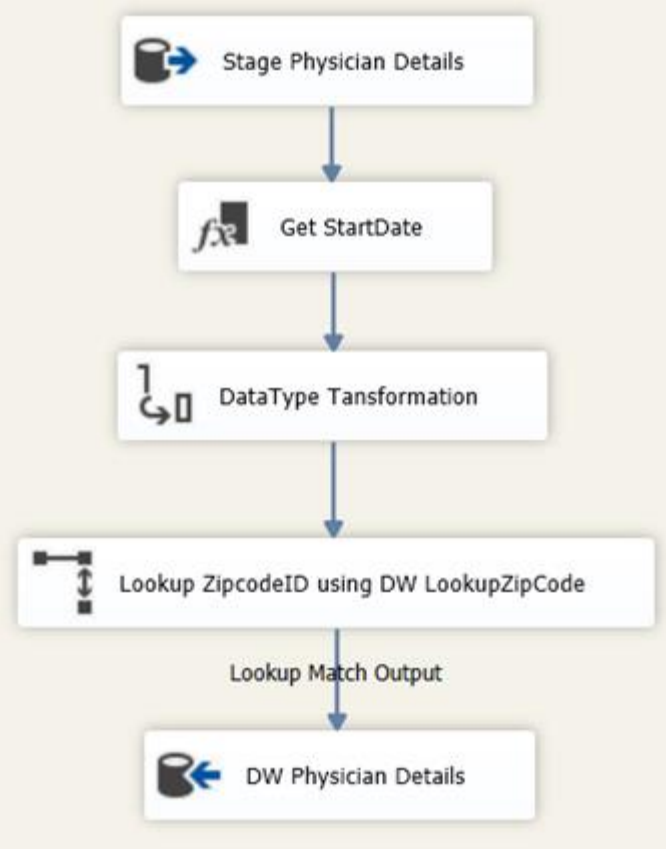


DW Hospital Info is transformed and populated with a load date as follows:
1. Load date is added as a derived column using getdate() method.
2. The data conversion transformation is used to make all the columns of appropriate length.
3. The Lookup Transformation is used to look up Zip Codes from the DWLookupZipCode table and then matched output records are loaded into the DWHospitalInfo table.



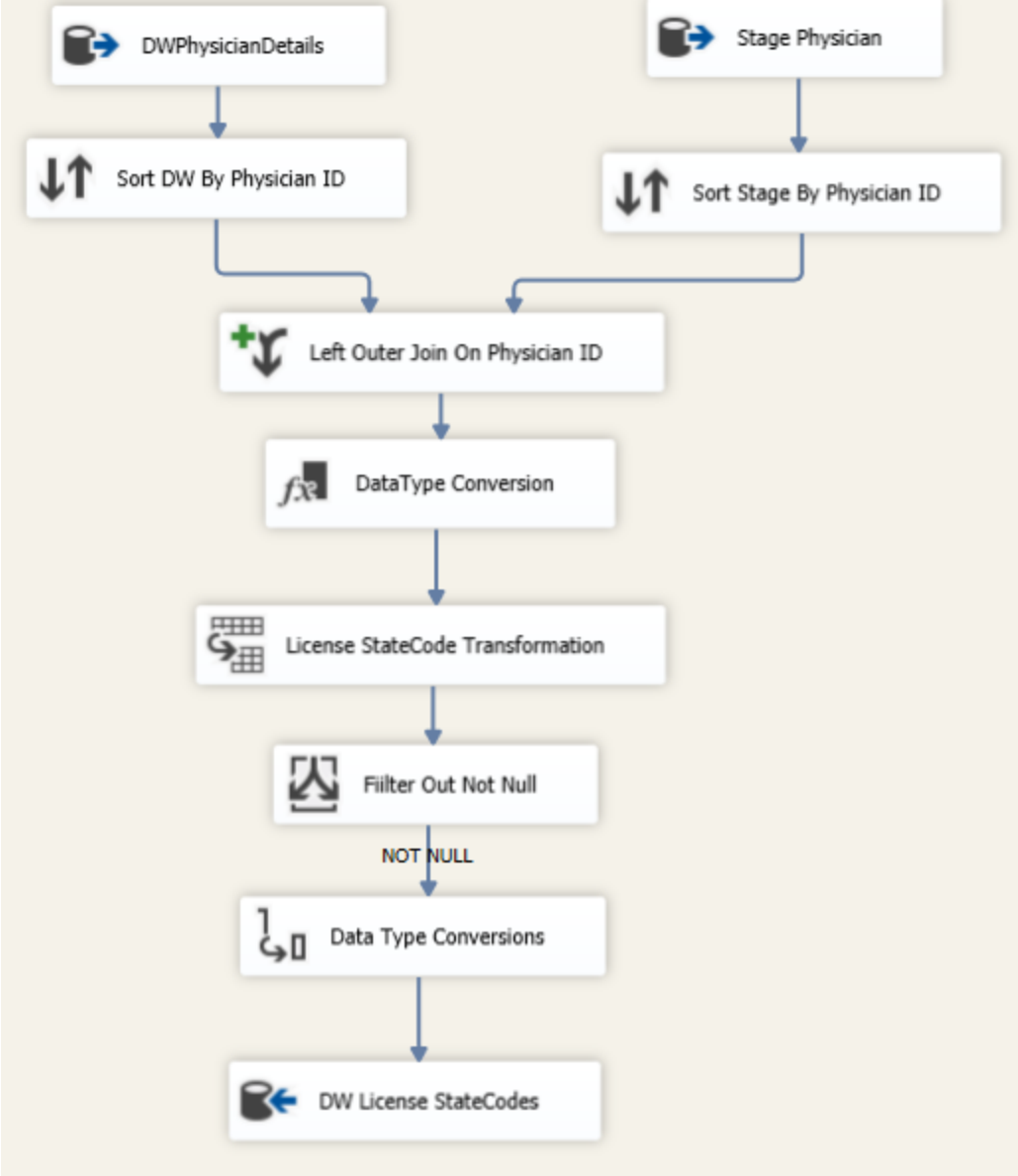
DW Physician Details is transformed and populated with load_date as follows:

1. Load date is added as a derived column using getdate() method.
2. The data conversion transformation is used to make all the columns of appropriate length.
3. The Lookup Transformation is used to look up Zip Codes from the DWLookupZipCode table and then matched output records are loaded into the DWPhysicianDetails table.



DW License State codes is transformed and populated with load_date as follows:

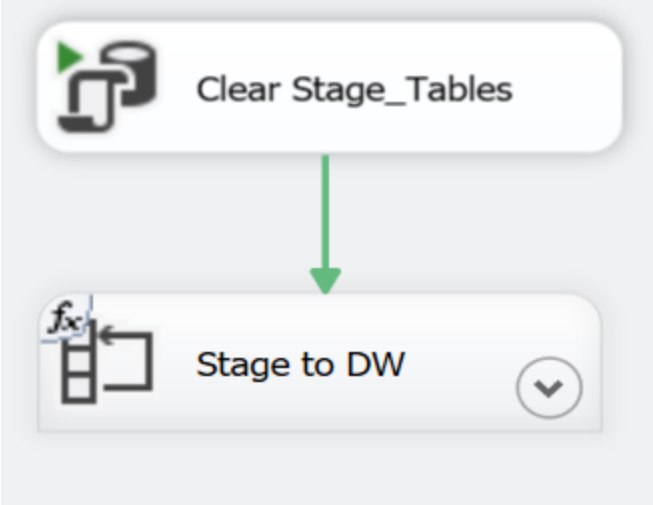
1. A left outer join is performed between DW physician table and stage physician table
2. Data conversion transformation is performed to extract derived columns for license state codes by cleaning up invalid state codes (length <2) replacing it with null and converting the valid license state codes to string data type of proper length.
3. Unpivot transformation is applied to generate a pivot column License_state_code with values of integer data type as 1,2,3,4,5 where value 1 means first License state code of physician and so on.
4. Conditional split is performed to load only not null state codes.
5. Data conversion is applied to convert statecode to length 2
6. Transformed data is loaded into DW License State codes.



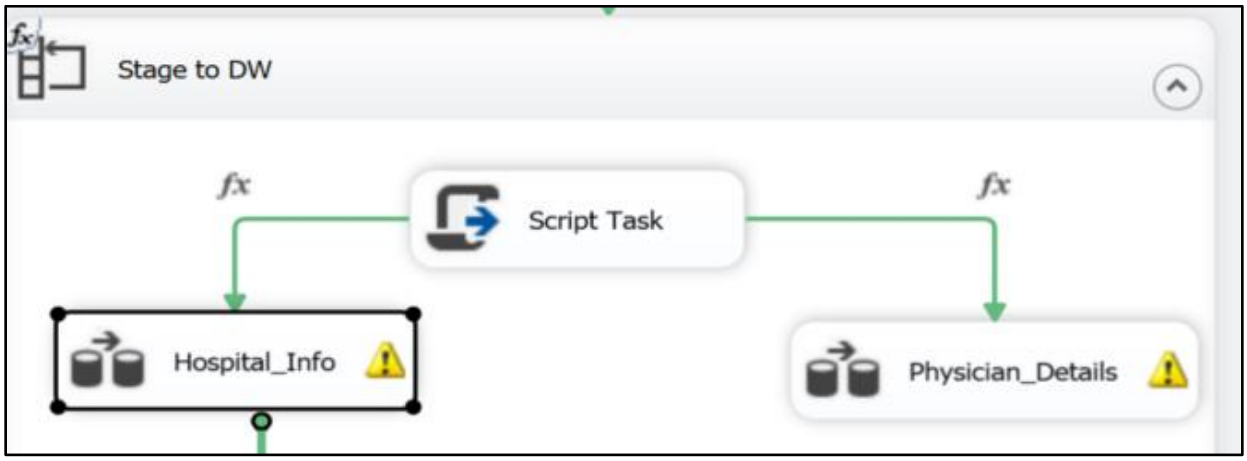
Step 2: Incremental Load

Once we have performed the initial load in the data warehouse, we will perform SCD while doing an incremental load in the data warehouse.

1. **Clear the stage tables and populate the stage tables again.**



SCD for Hospital and Physician Details



2. **SCD for Hospital:** The details of all the columns with changes tracked are shown below:

   Update In place This is done for Address, Zipcode and phone number.

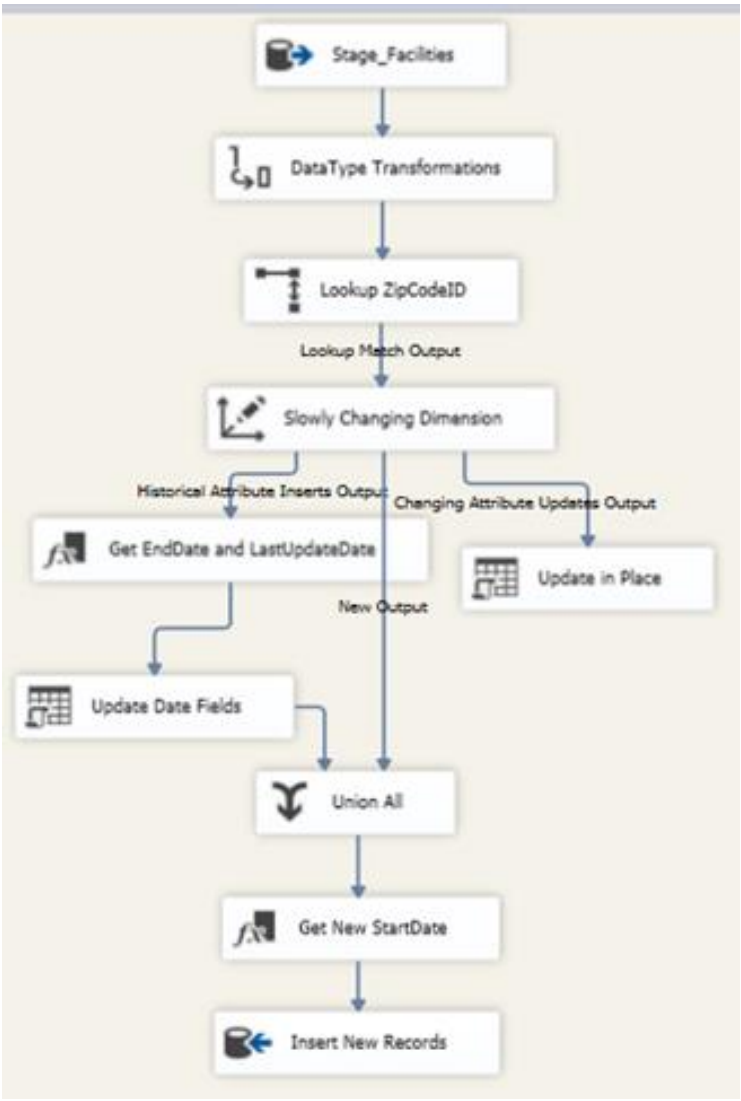
   The address of Hospital is very unlikely to change(changing the physical location of a hospital). Also zip code is a part of the address so update in place for zip code also.

   Also, the old phone number no longer keeps working, so tracking the history of old phone numbers is not of much importance.

   No Change This is done for Facility Id and Location, since the Facility ID is a unique value mapped to every hospital, it will never change. ALso the Location of the hospital is highly unlikely to change.

Historical Change This is done for Hospital Name, Type and Ownership and all other information related to the hospital so as to track the historical data of a hospital. It would help in future to perform the hospital's historical data analysis.

DW_Column_Name	SCD
FacilityID	No Change
FacilityName	Historical Change
Address	Update in Place
ZipCode	Update in Place
PhoneNumber	Update in Place
MortalilityRating	Historical Change
SafetyOfCareRating	Historical Change
ReadmissionRating	Historical Change
PatientExpRating	Historical Change
EffectivenessOfCareRating	Historical Change
Location	NoChange
StartDate	OnLoad
EndDate	OnHistorical Change
LastUpdateDate	On Change



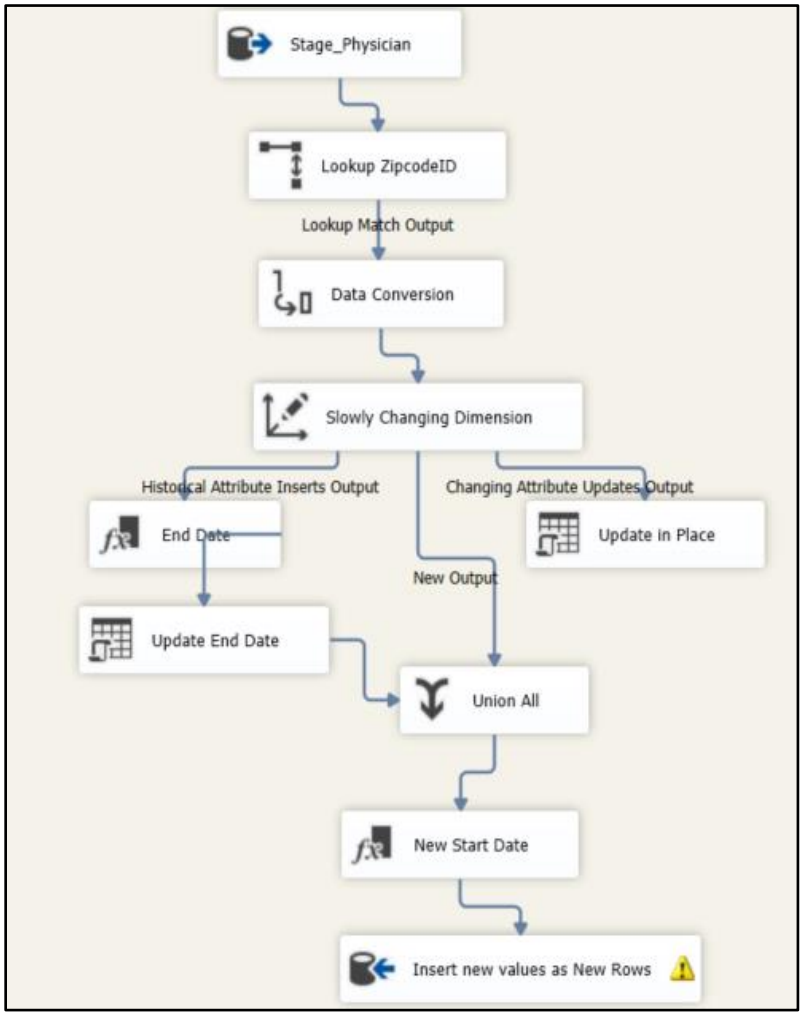
SCD for Physician Details The details of all the columns with changes tracked are shown below:

Update In place: The Name and suffix field of physician are updated in place, because tracking the historical data for physicians is not required.

No Change: There are no columns which fall under this category

Historical Change: The address, field,speciality are made historical change columns, so that this historical data can be used for analysis purposes.

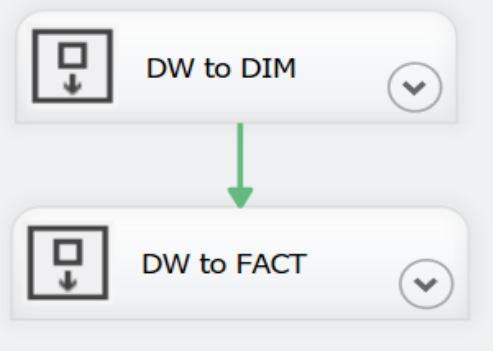
DW_Column Names	SCD
PhysicianID	No Change
FirstName	Update in Place
MiddleName	Update in Place
LastName	Update in Place
Suffix	Update in Place
AddressLine1	Historical Change
AddressLine2	Historical Change
City	Historical Change
State	Historical Change
Zipcode	Historical Change
PhysicianField	Historical Change
PhysicianSpeciality	Historical Change
StartDate	OnLoad
EndDate	OnHistorical Change
LastUpdateDate	On Change



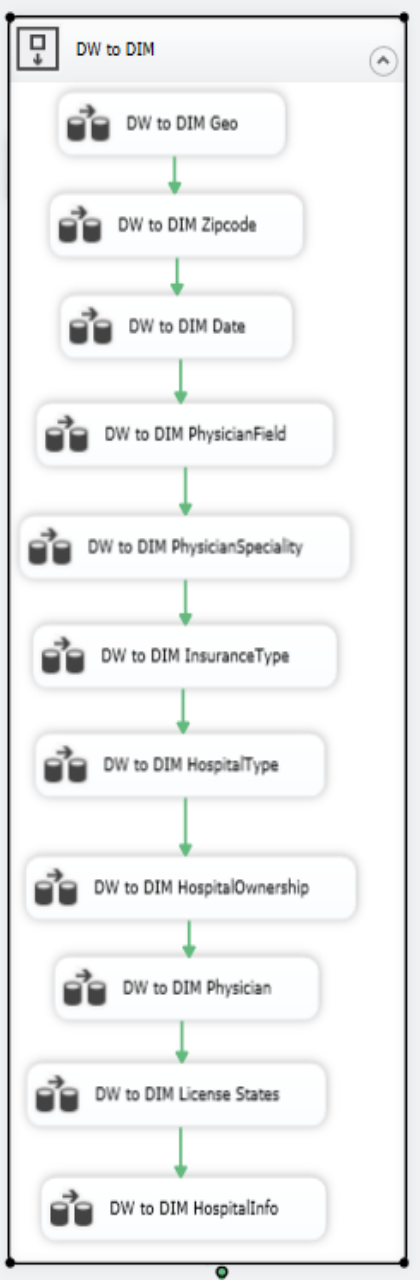
Step 3: Data Warehouse to DIM and FACT Dimensional model Load:

Once the data warehouse is loaded with all the mapping transformations and data cleaning, dimensional load processing starts to build the Snowflake schema.

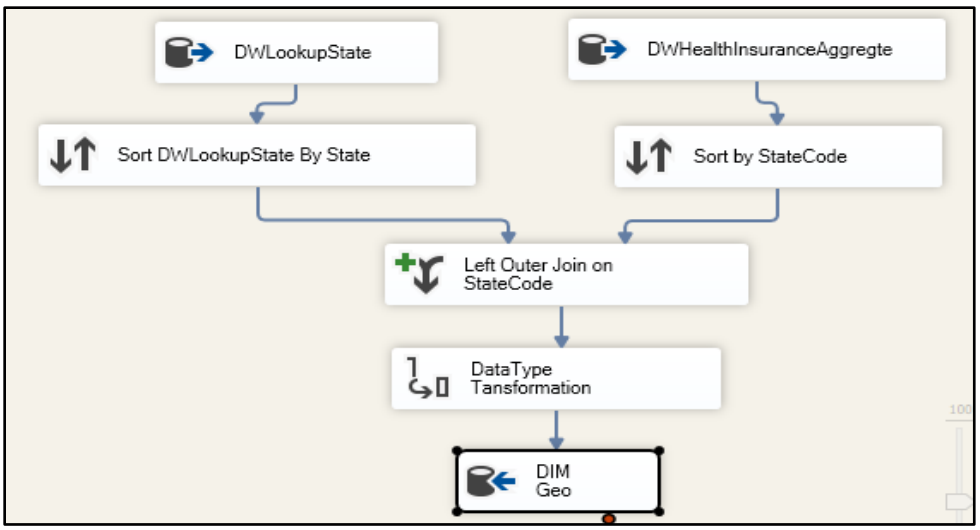
It is shown in detail as below:



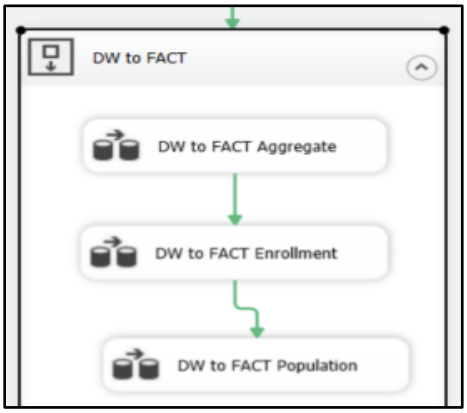
DW to DIM tables: All the dim tables are populated from the respective dw tables and by applying necessary transformations which is explained in detail below:



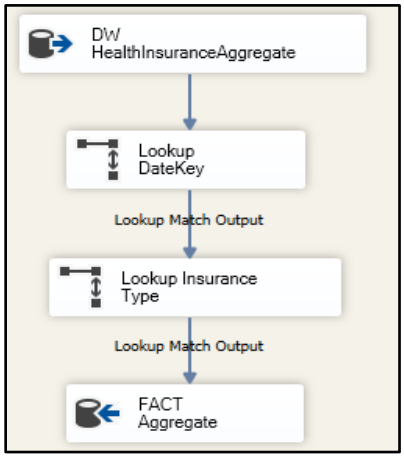
DIMGeo A left outer join is performed on DWLookup State & DWHealthInsurance to populate DIM Geo. Similarly all the other DIM tables are populated an example of the load process is shown below



DW to FACT Tables: Fact tables are populated using mapping transformations and performing lookups



Fact_aggregate is populated by first doing a lookup for the year column from Date table and then lookup for insurance type from Lookup Insurance table.



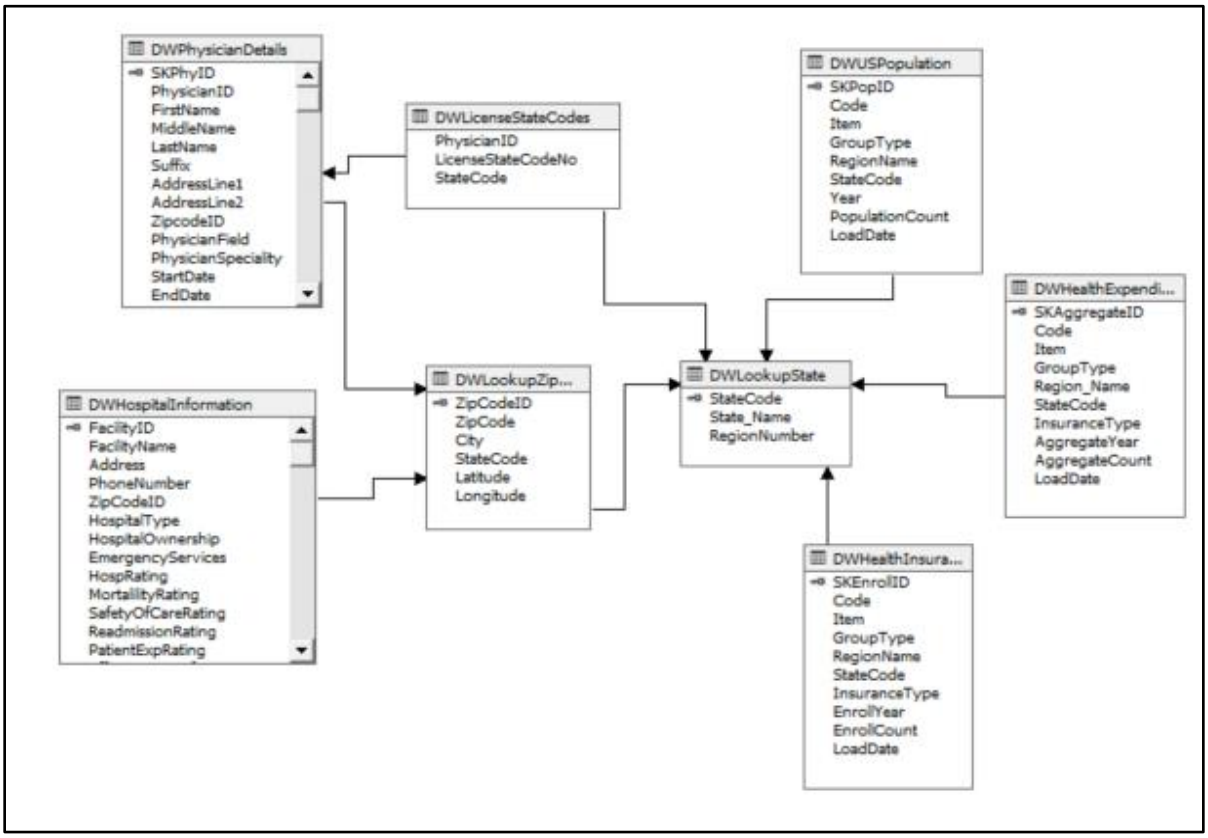
ER Diagram for Integrated Data warehouse Tables:

The primary key and foreign key relationships are shown in the physical model design

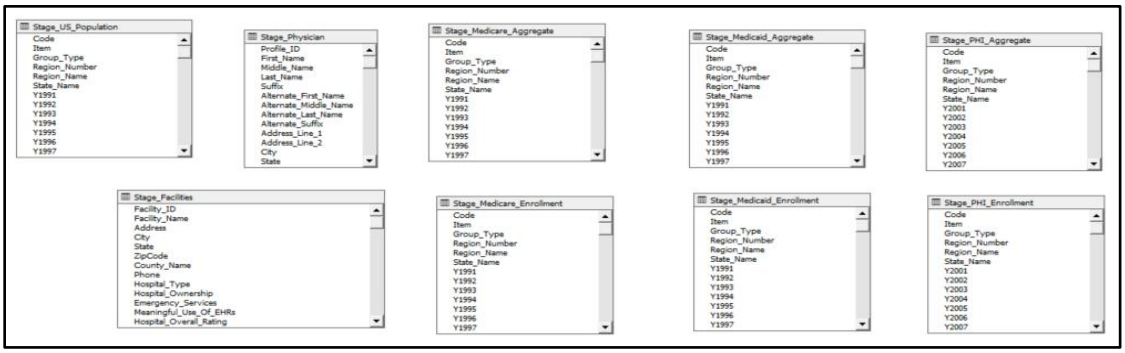
DWLookupZipCode has a one to many relationship DWHospital and Physician table.

DWLookup State table has a one to many relationship with DWHealthExpenditure,DWHealthInsurance, DWUSPopulation, DWLicenseStateCodes and DWLookupZipCode

DWLookupZipcode and DWLookupState are used to establish a relationship between all the tables and do various analysis state-wise.

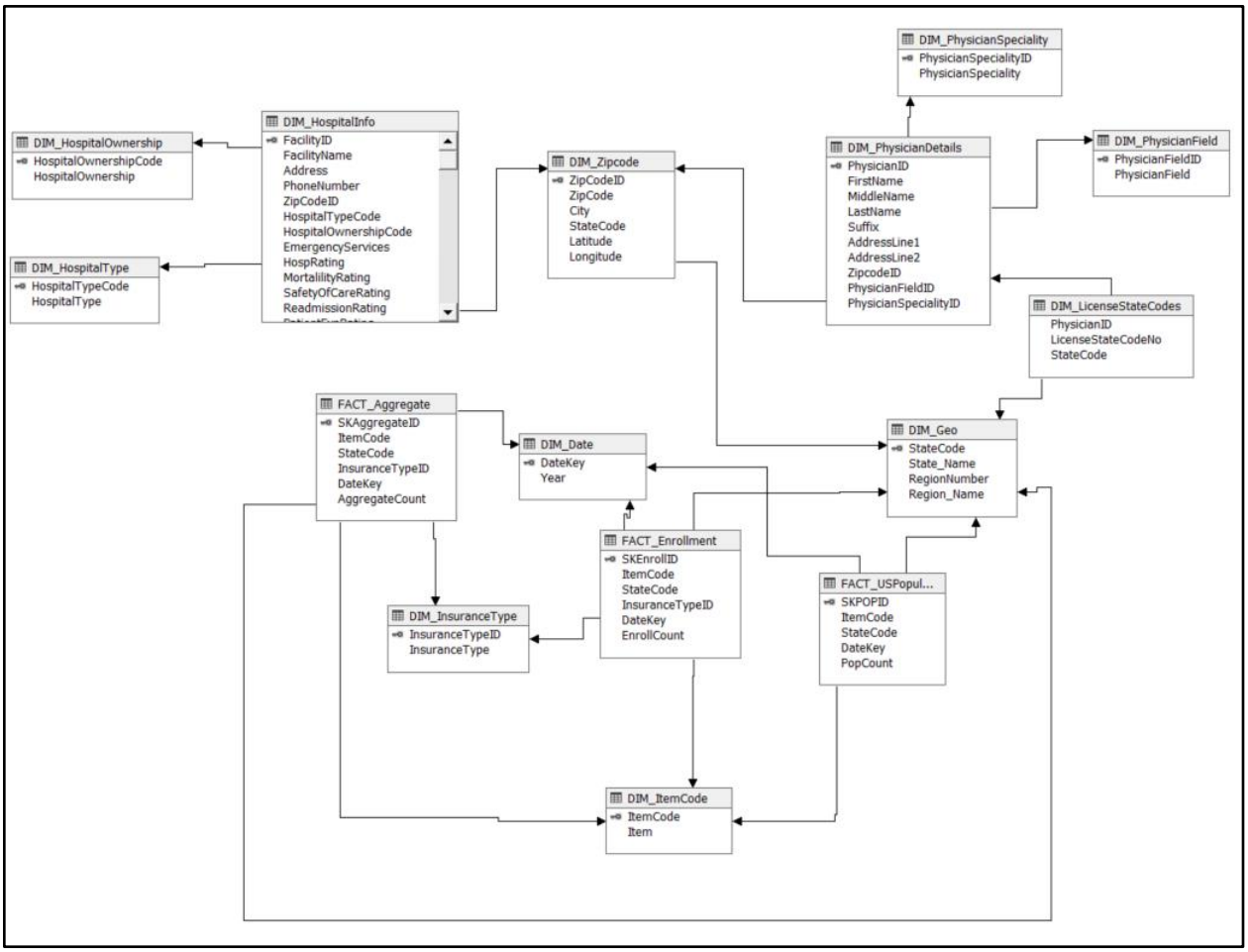


Stage Tables:

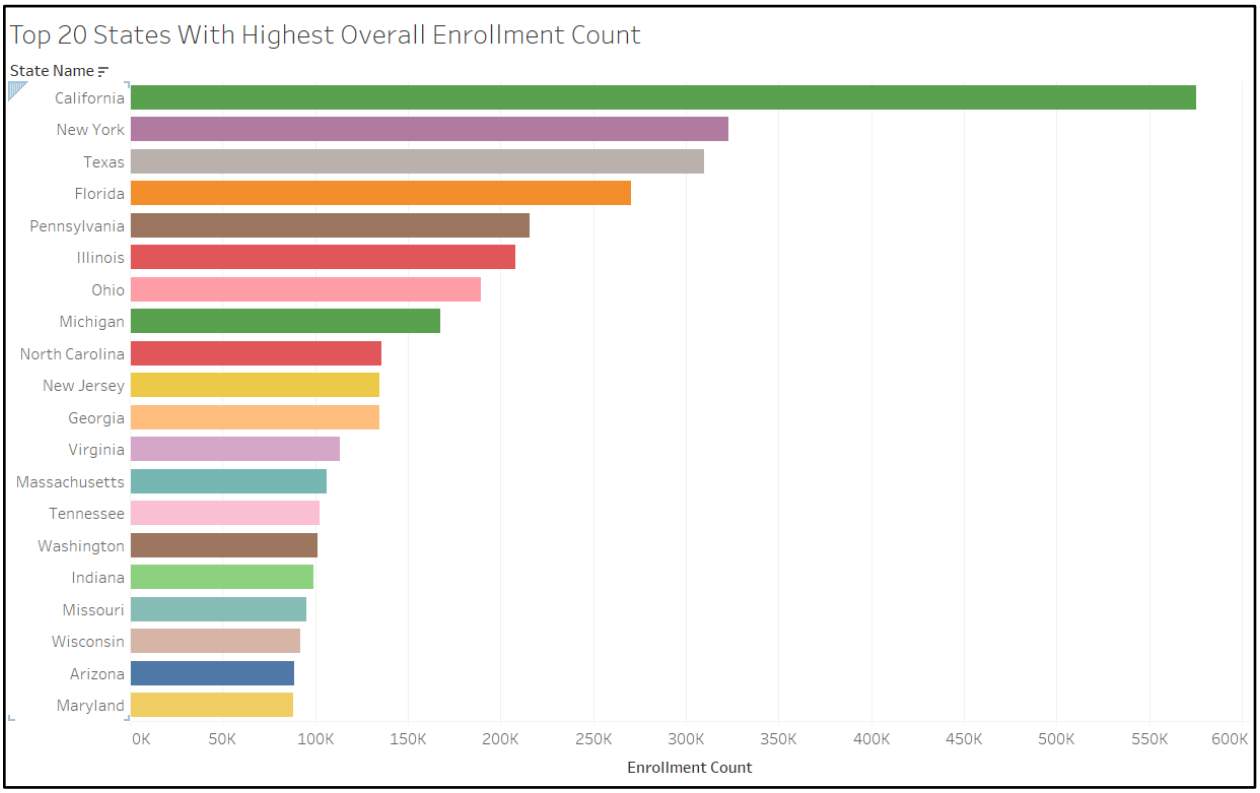


ER Diagram for Dimensional Model/Star Schema Design:

The primary key and foreign key relationships are shown in the physical model design

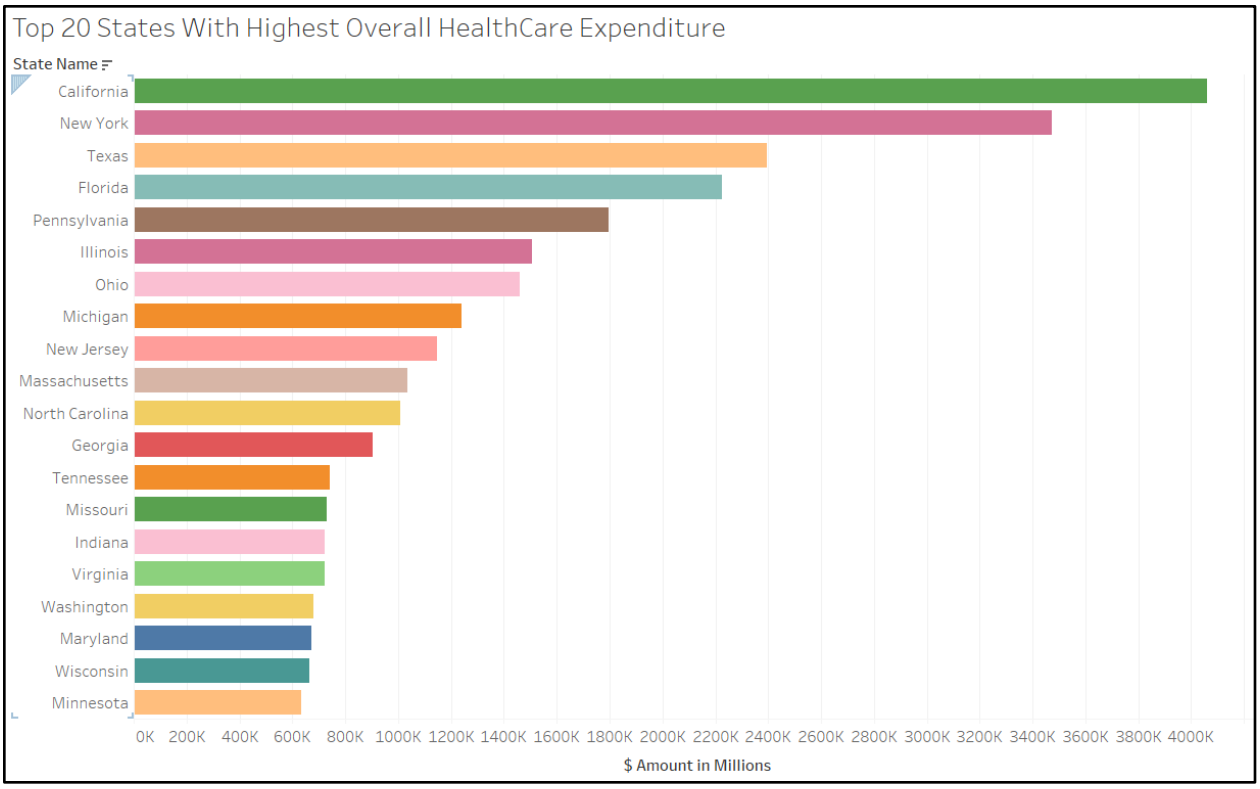


Data Analysis and Report visualizations:



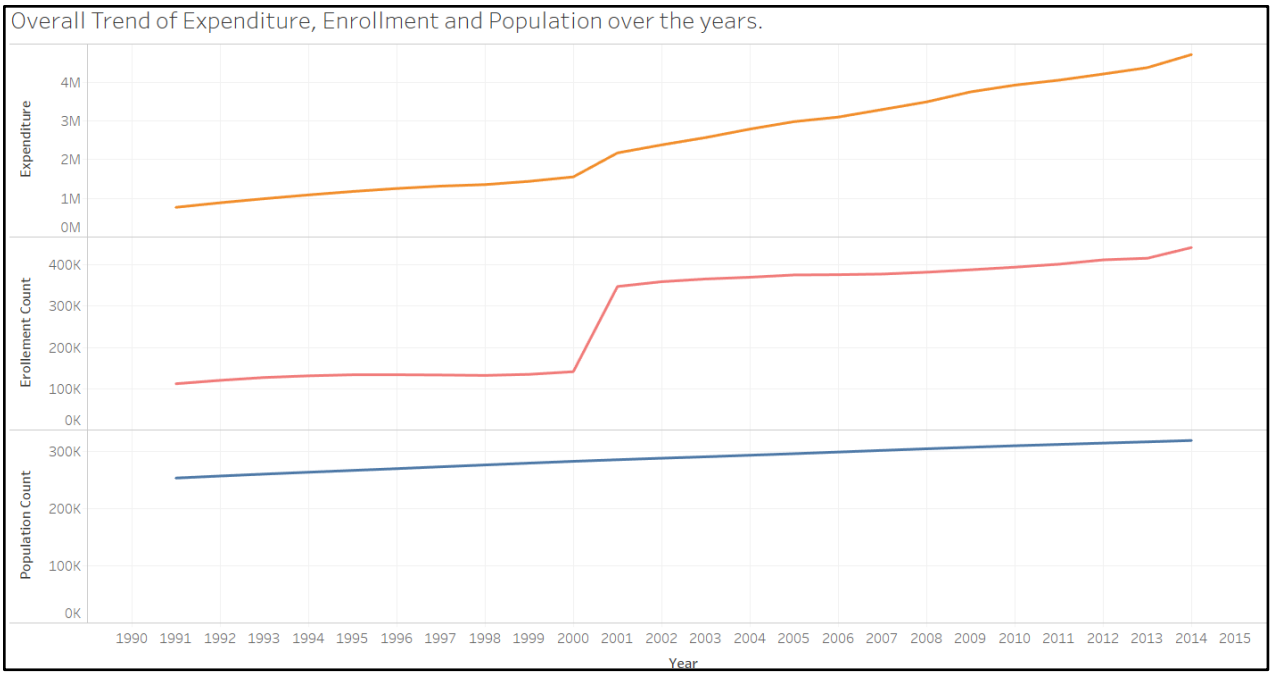
Graph 1: Top 20 states with Highest Overall Enrollment Count

The bar graph shows the top 20 states that have the highest share of enrollment to the health insurances. It can be noted that California has the highest share of enrollment count. It can be noted that there is a significant difference in the enrollment count of California and New York, even though New York is the next state to have the highest enrollment count. It can be noted that the states North Carolina, New Jersey and Georgia have almost the same number of enrollment counts (with a 1000 difference with each other). The last 9 states also show very less difference in the enrollment counts.



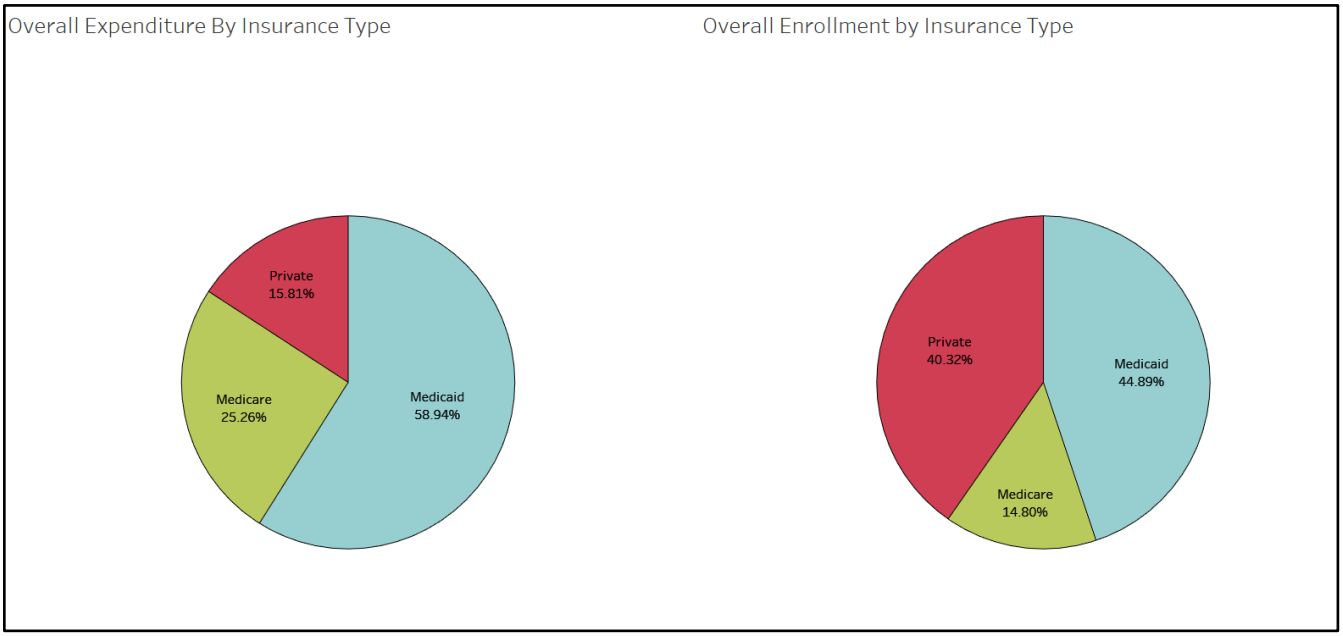
Graph 2: Top 20 states with Highest Overall Expenditure Count

The above bar graph shows the top 20 states with the highest expenditure in million dollars. California has the highest share of overall expenditure. It can be noted (from Graph 1) that there is a consequential relationship between the enrollment counts and the expenditure amounts in states. The above graph shows that New York has relatively higher expenditure even though there is a significant difference in enrollment from Graph.1. It is illustrated in this graph the top 8 state's expenditure is directly proportional to the enrollment information from Graph.1. The last 8 states have almost the same amount of expenditure.



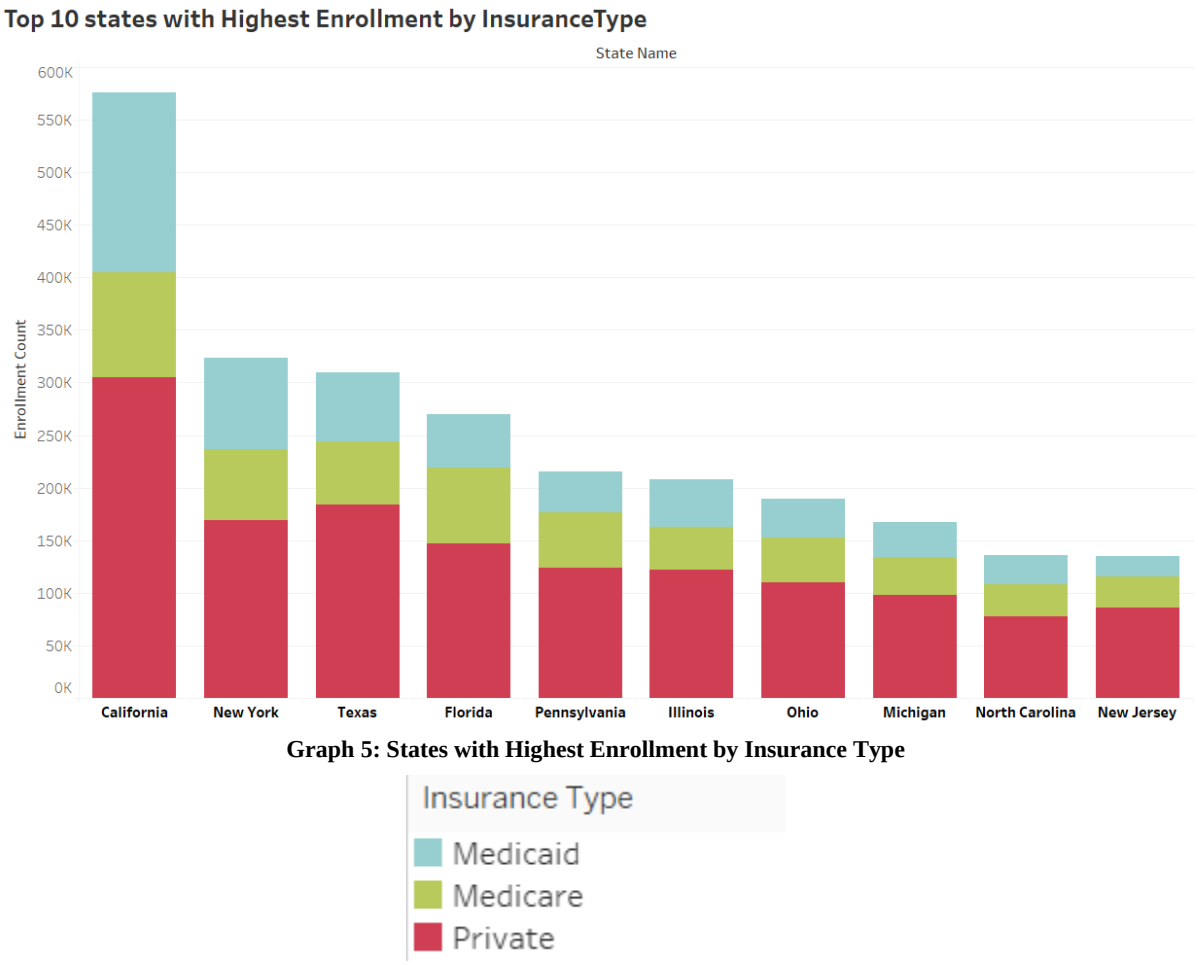
Graph 3: Overall Trend of Expenditure, Enrollment and Population over the years

The line graph shows the trends in the overall expenditure, enrollment and population over the years 1991-2014. The US population has been increasingly constantly each year from 1991-2014. Observing the overall enrollment count, there has been the constant slow increase in the enrollment count in the years 1991-2000. There is a sharp increase in the enrollment count in the years 2000-2001 and the constant growth over the next years continues. Similarly, with expenditure we see a sharp increase in the years 2000-2001.This line graph also evidently shows a relationship between the expenditure and enrollment.



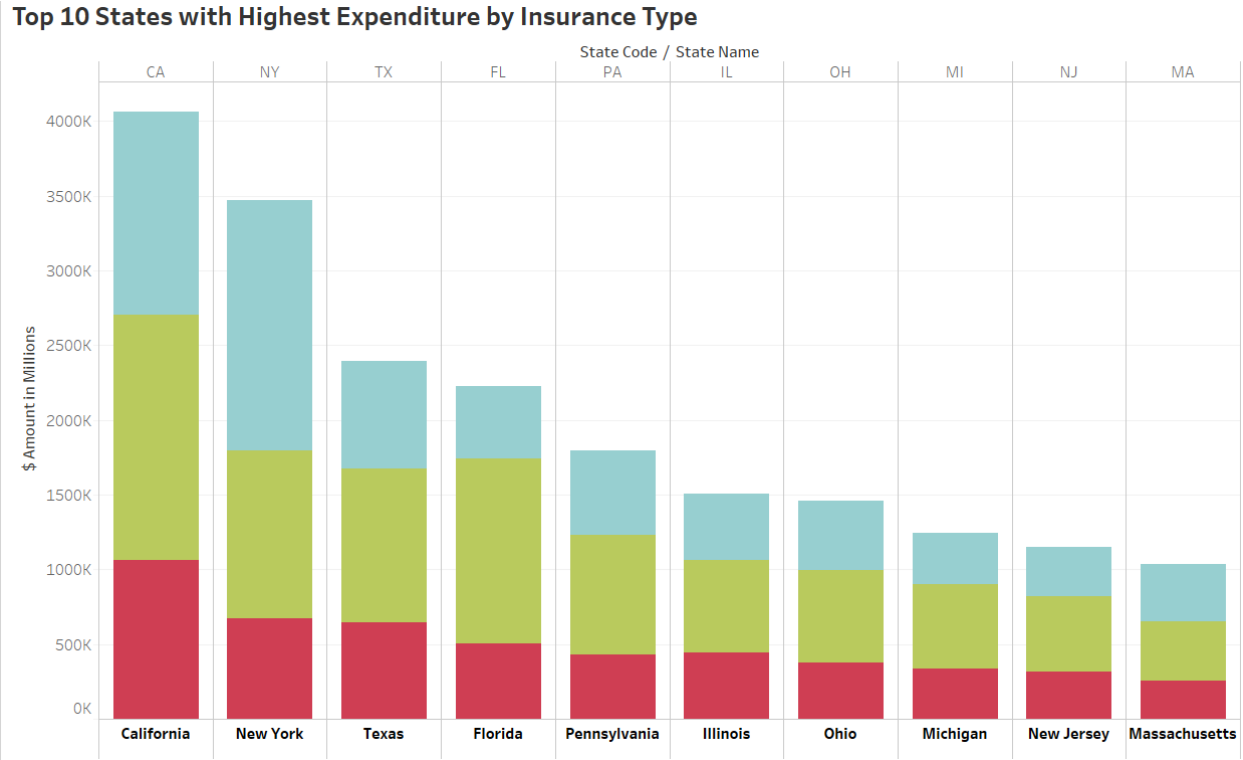
Graph 4: Health Expenditure, Enrollment by Insurance Type and State

The above pie chart shows the overall expenditure and enrollment by insurance type. The graph clearly indicates that Medicaid has the highest share of overall expenditure and also has the highest share of overall enrollment. Private health insurance has 40.32% of the overall enrollment but only contributes to 15.81% of the overall expenditure.



Graph 5: States with Highest Enrollment by Insurance Type

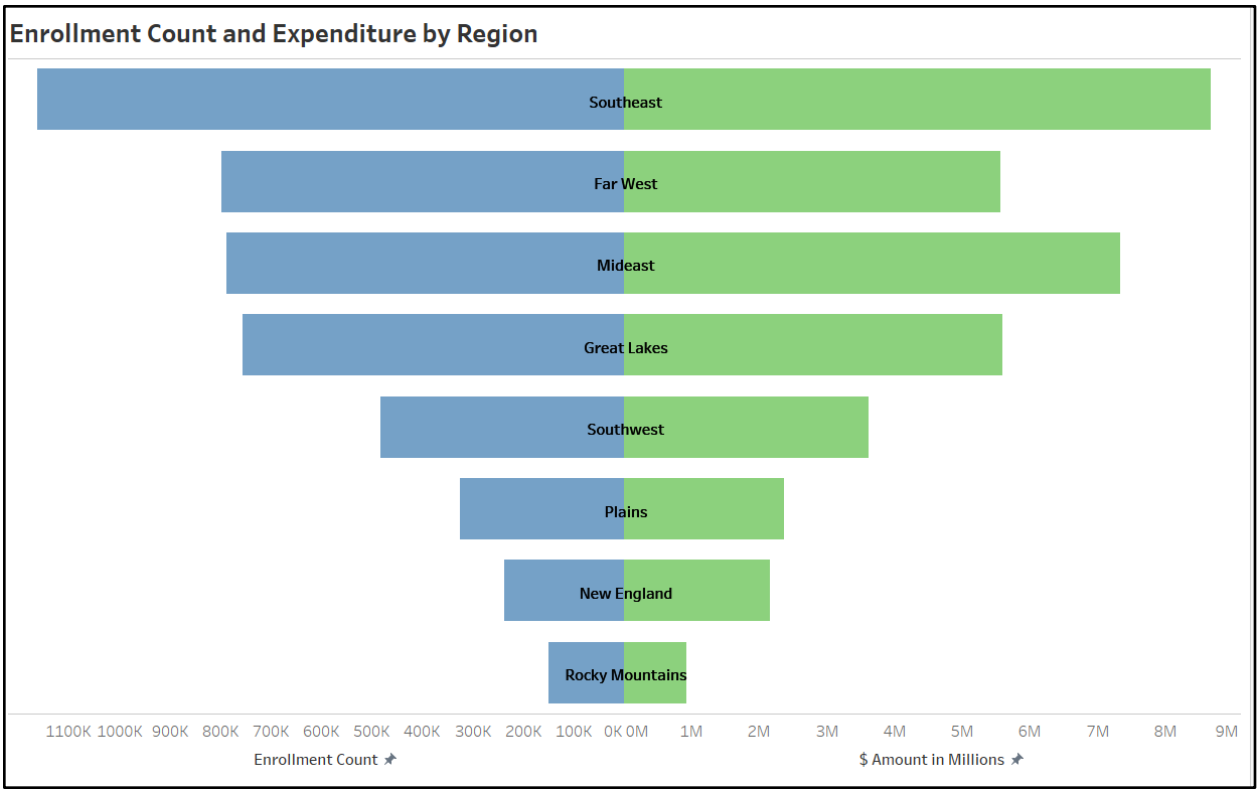
This stacked bar graph shows the top 10 states with highest enrollment count by insurance type. California has the highest share of enrollment (from Graph.1) it can be noted that out of these enrollments Private health insurance holds the greatest number of enrollments. It is evident that in every state private health insurance has the greatest number of enrollments. In the states California, New York and Texas, Medicaid has the next highest share of enrollment.



Graph 6: States with Highest Expenditure by Insurance Type

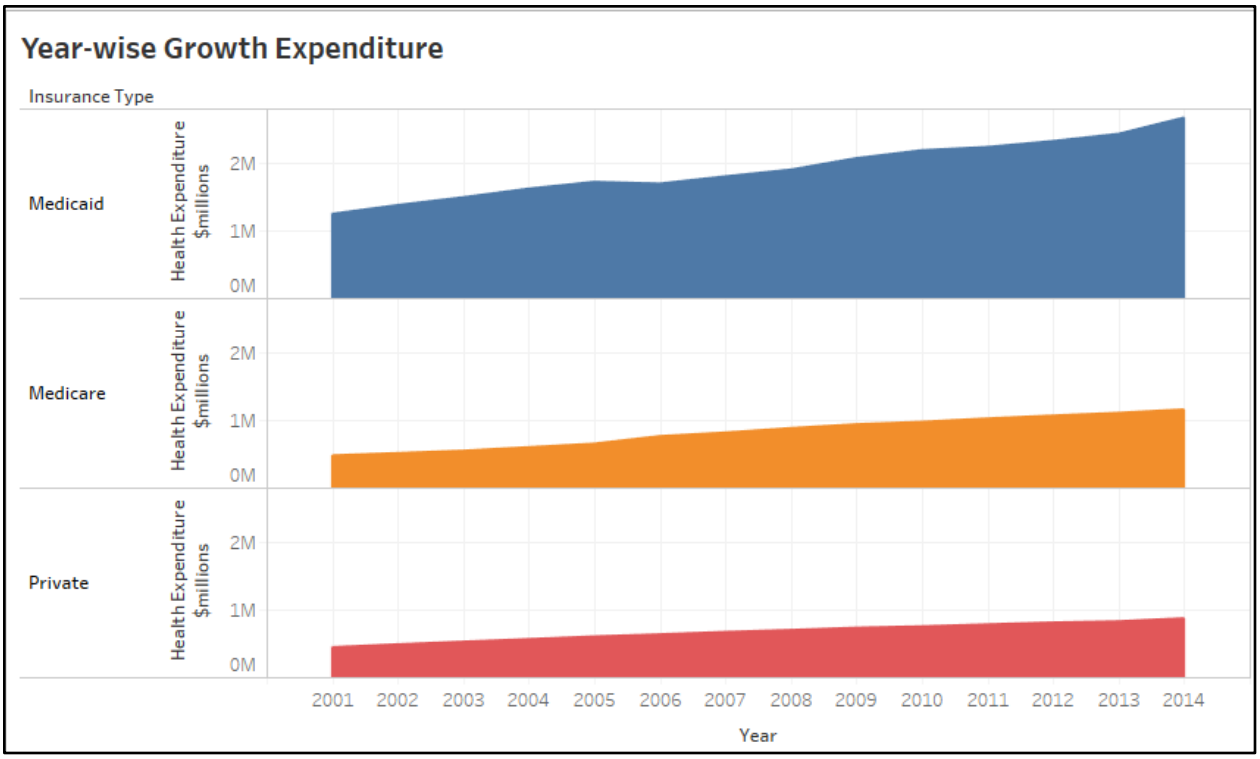
This grouped bar graph shows the Expenditure in top 10 states by Insurance Type.

It can be noted that California has the highest expenditure and out of the three Insurance Types, Medicare has the highest expenditure, whereas New York, the second highest state has highest expenditure in Medicaid insurance type. Texas and Florida, the next highest states have maximum expenditure in Medicare. So, it shows the Health Expenditure in various states do not follow the same pattern in diff Health Insurance Types. Medicare expenditure is high in most of the states.



Graph 7: Region-wise Enrollment and Expenditure Analysis

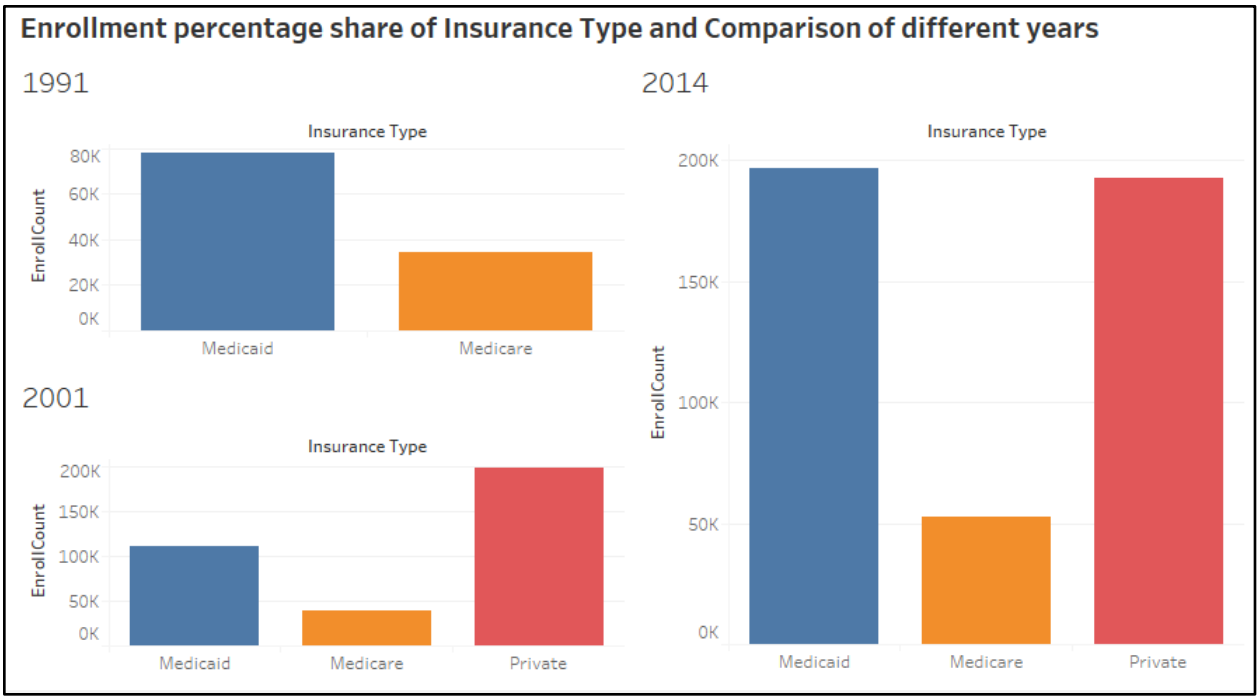
This butterfly chart shows Enrollment count and Expenditure by different regions of the United States. It can be noted from the graph that the Southeast region has highest enrollment as well as highest expenditure. Going down the graph we see that Far West, Mideast and Great lakes have almost the same enrollment count close to 800k, but Expenditure of Mideast is high in comparison to Great lakes and Far West. In general, we can see Regions follow a similar trend for Enrollment and Expenditure.



Graph 8: Year-wise Growth Expenditure for Insurance Types

Please Note: Private Health Insurance data only available from 2001

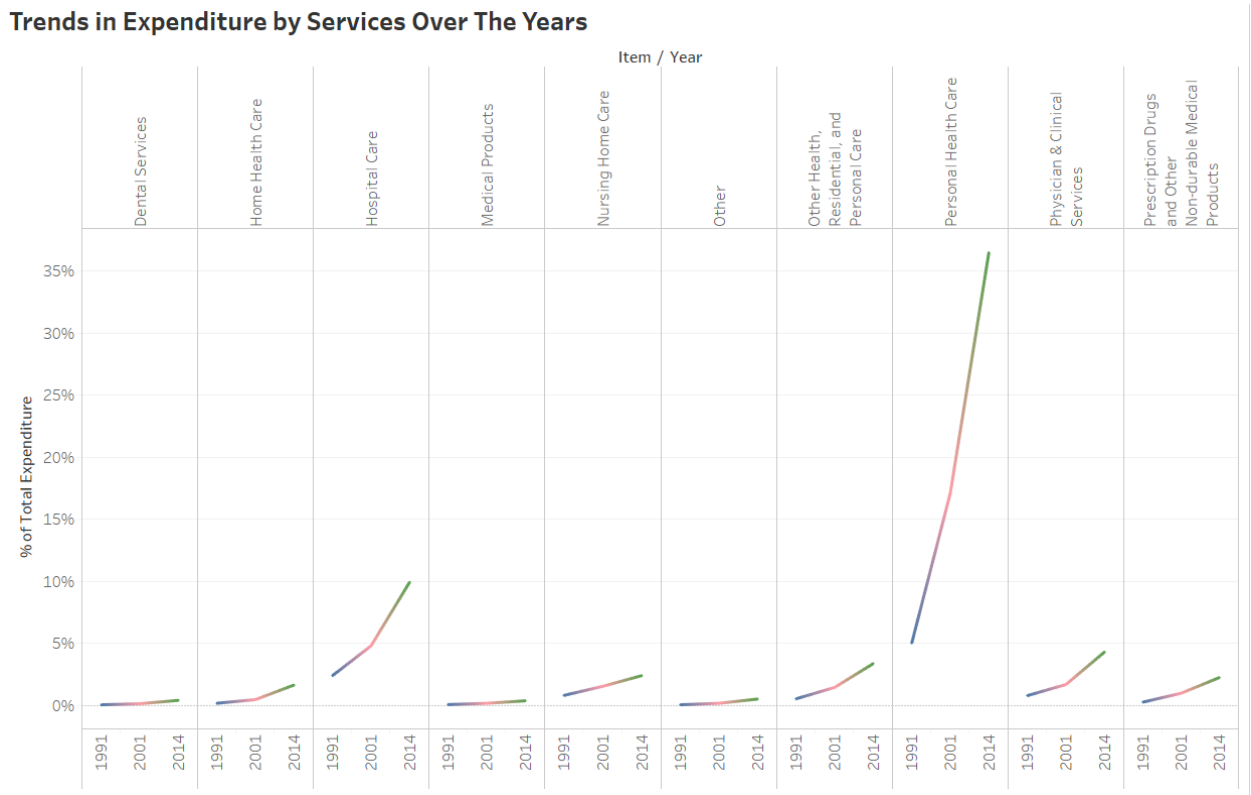
This area graph shows how health expenditure in millions dollars has changed over years(1991-2014) for Medicaid, Medicare and Private Health Insurance types. It can be seen from the graph that expenditure for Medicaid type rises rapidly from 1991 to 2014, for Medicare there is a slight increase from 1991 to 2001 but increases more after that. Private insurance type expenditure has a stable increase from 2001 to 2014. So, we can conclude by saying Private and Medicare insurance types spending is increasing at a low rate in comparison to Medicaid.



Graph 9: Total Enrollment in different Insurance Types and their Comparison between several years

Please Note: Private Health Insurance data only available from 2001

The above bar graph shows the percentage share of enrollment in different insurance types for years. Appx 80k people are enrolled in Medicaid in 1991 and only 38k people enrolled in Medicare. In 2001, there was a large number appx 200k enrolled in Private Health Insurance whereas 110k in Medicaid and small 40k in Medicare. In 2014 the percentage of people enrolled in Medicaid and private is significantly the same close to 190k and Medicare close to 50k. Thus, we can observe there is a huge trend change in people enrolling in Medicaid services from 1991 to 2014.

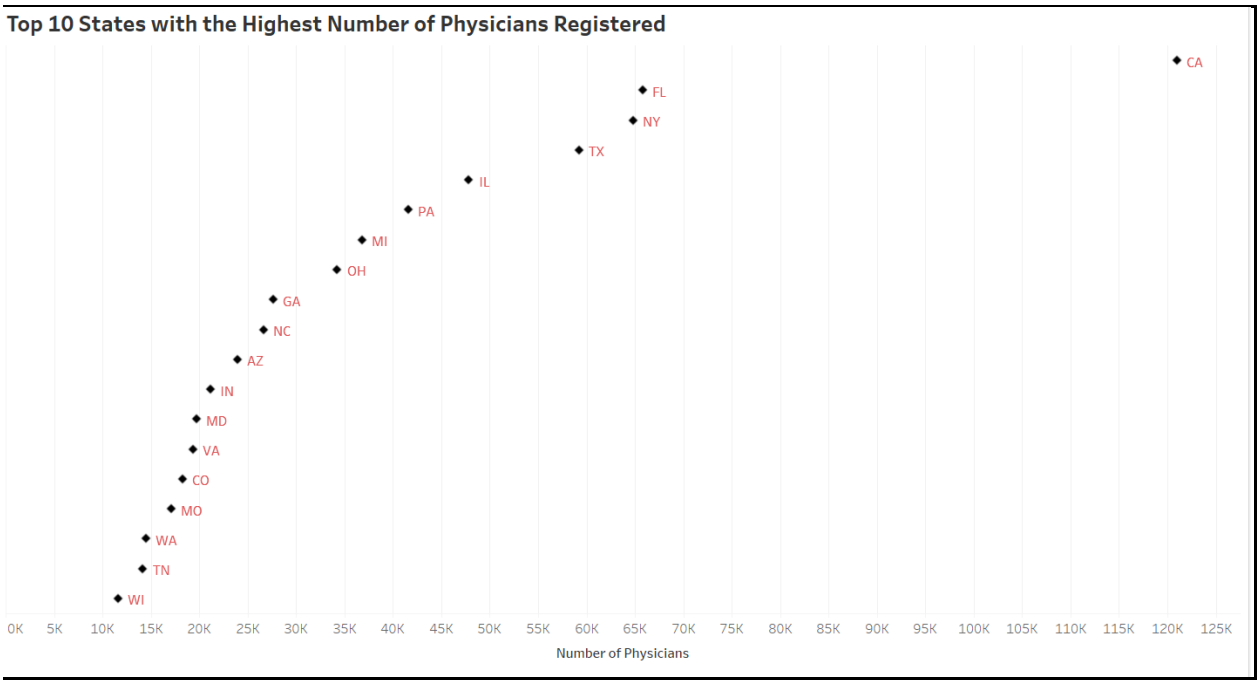


Graph 10: Trends in Expenditure for different Health Services

The above line chart shows the Trends in Expenditure for different Health Services.

It can be noted that personal Health care has a tremendous rise in Total expenditure. It has increased from 5% to 37% from year 1991 to 2014. Hospital care has the next highest increase.

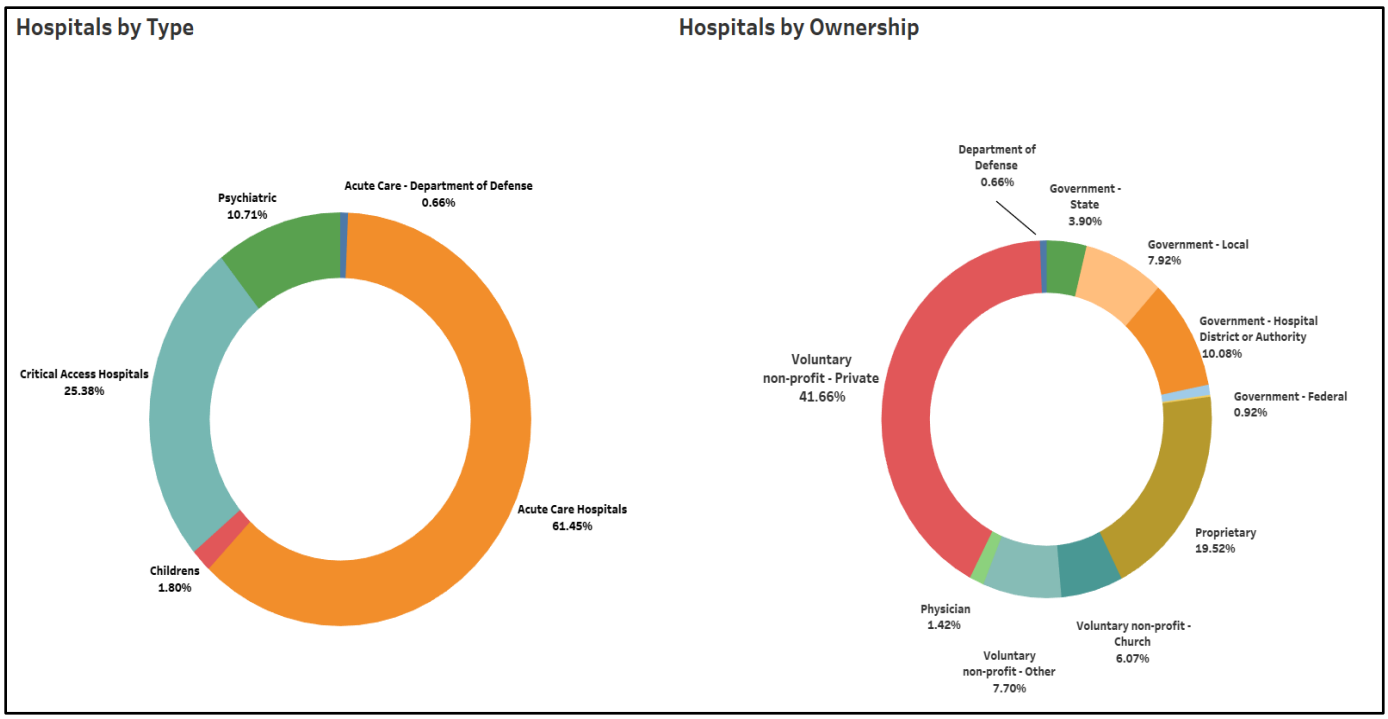
In comparison, Dental Services and Medical products are quite stable.



Graph 11: Top 10 states with highest number of physicians registered.

The above Grant Bar graph shows the count of physicians registered in every hospital.

It can be seen that California has a highest number of physicians registered and Florida and New York comes next.

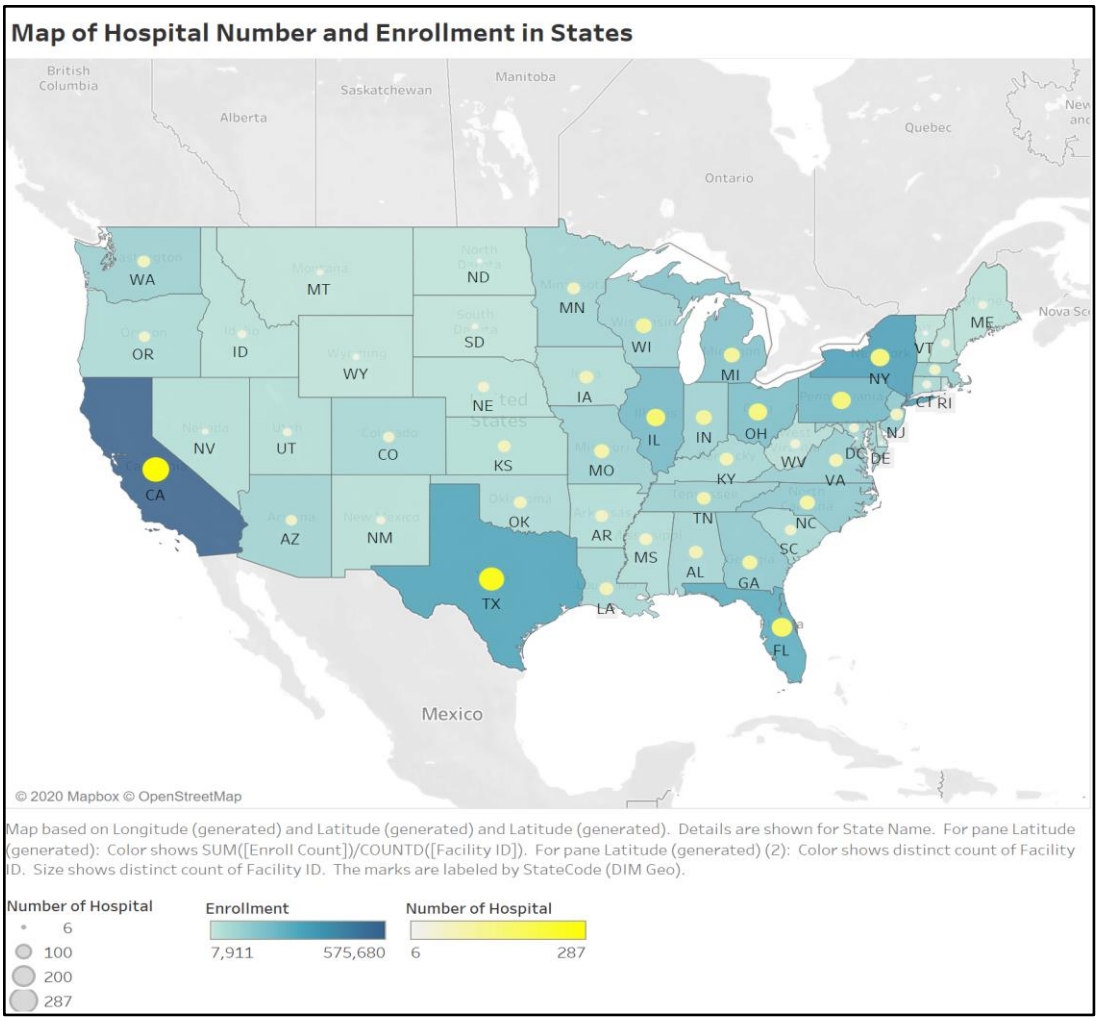


Graph 12: percentage share of Hospitals by Type and Ownerships

The above pie chart represents percentage share of Hospitals by Type and Ownerships.

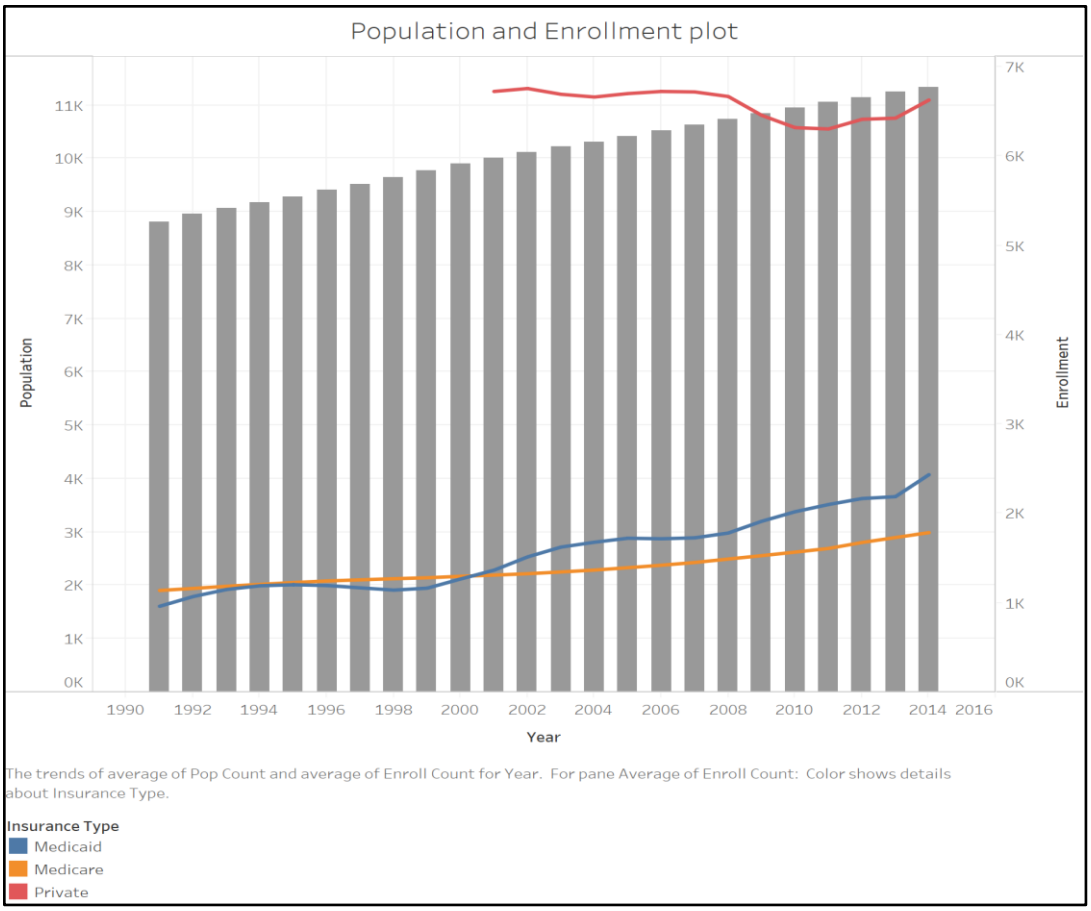
It can be noted that Acute care Hospitals have the highest percentage among all hospital types and Critical Access hospitals are the second highest.

Hospital Ownership Voluntary non-profit-Private form the highest share of percentage.



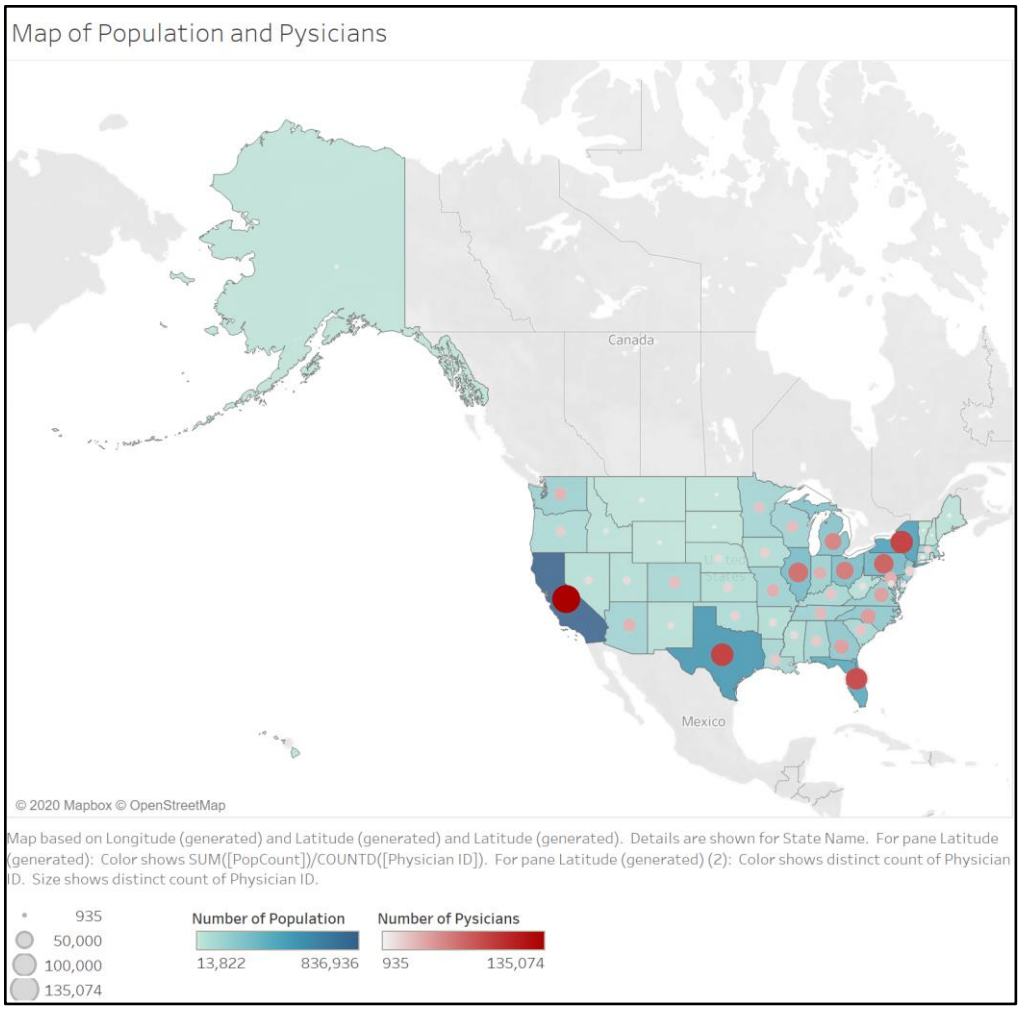
Graph 13: Relationship between Number of hospitals and enrollment count in diff states

From the map, we can see the state having larger enrollment will be in darker blue and if it has more hospitals, it will have larger and a brighter yellow spot on it. Thus, we can easily find that the state that has more enrollment always has more hospitals too.



Graph 14: Relationship between population and Health Insurance Enrollment for several years

In Population and Enrollment Plot, left y-axis stands for Population and right one is for Enrollment Numbers. The grey histogram shows the population in each year and the colored line shows the trend of enrollment numbers along with year. We can easily find the enrollment of Medicare and Medicaid increases with the increase in populations. However, the enrollment of Private insurance is more stable and it does not fluctuate with the increase in population.



Graph 15: Relationship of count of physicians with population

We can clearly find if the State has a larger population, it is higher possible for it to have more physicians.

Appendix A – Data Source References(Description Page:4)

A.1 Health Expenditures by the State of Residence

https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/NationalHealthExpendData/NationalHealthAccountsStateHealthAccountsResidence

A.2 Hospital General Information

https://catalog.data.gov/dataset/hospital-general-information

A.3 Physician Data

https://www.cms.gov/OpenPayments/Explore-the-Data/Dataset-Downloads

A.4 Zip Code Data

https://www.zip-codes.com/

Appendix B – Revision History

Date	Who	What	Version
3/15/2020	All members	Created initial proposal etc…	0.1
3/18/2020	All members	Document load process, created a picture to show the flow of data and added data model design	1.0
4/8/2020	All members	Requirement Analysis/Design document. Created all the staging,DW tables and built out a dimensional model from it.Added the SSIS data flow and logic	2.0
4/24/2020	All members	Final Project Report	3.0