Northeastern University
College *of* Engineering

DATA WAREHOUSING & BUSINESS INTELLIGENCE (INFO 7290)

Group 22

Group Member Name	**NEU ID**
Abhay Kurian	001491312
Smit Patil	001378302
Shivam Verma	001344145
Nikhil Kukreja	001055376

Table of Contents

1. Objective
2. Tools Used
3. Introduction
4. Revision History
5. Overview of the Data Model
6. ER Diagram
7. Data Flow & Implementation
    - 7.1 Data Flow diagram
    - 7.2 Loading Data into Staging Tables
    - 7.3 Error Handling and Data Validation
    - 7.4 OLAP Cube Design
    - 7.5 Challenges faced along with solutions
    - 7.6 Loading data into Fact Table
8. Data Analysis & Visualizations
9. Conclusion & Future Scope

Objective / Scope of the Document

Good Reads dataset is a comprehensive dataset containing details about books, authors, genres, ratings and other measures that readers across the globe are interested in and would help them make a better decision about which book to read next. Our Objective here is to create a tool that can help us analyze the Goodreads Data and help users slice and dice this data as per convenience. We will be using Cubes and Dashboards to help the user achieve this goal. We want to complete the following objectives on completion of this project:

- An exhaustive data warehouse, easily available, clean and understood
- Dashboard, with easily interpretable information and interactive
- OLAP Cubes that can help drill down and up different dimensions to create useful measure analysis
- Proper understanding of distribution of Authors Ratings and Books Ratings.
- Know which books are read the most, have the most reviews and best ratings
- Be able to analyze the above measure based on different timeline, like year or months or day of week
- Understand how different publishers are rated and reviewed
- Analyze description of books to know which words or genres are most sort after

Tools Used

1. SQL Server Integration Services (SSIS)
2. SQL Server Management Studio (SSMS)
3. SQL Server Data Tools (SSDT)
4. Visual Studio
5. ER Studio
6. Tableau Desktop
7. Power BI Desktop
8. R studio

Introduction

Goodreads Books Dataset

Goodreads is the world's largest site for readers and book recommendations. UCSD Book Graph collected the datasets in 2019. Goodreads API was used by the developers to retrieve the data from the database. The users' public shelves, i.e., the information available to any other user without logging in, were scrapped. The scrapped dataset was stored in a JSON file format, transformed into a CSV format before beginning the data integration process. There are five different files containing information about the books, authors, works, series, and genres.

Tableau Books Dataset

The Tableau datasets include two files containing information regarding the Bookshops and the Libraries. Both these files are in XLSX file format, and the bookshop file has 13 Excel sheets present in it. The central concept for this data set is the idea of a book versus an edition. A book is a concept with attributes such as author, title, and genre. An edition is a physical version of the book, with attributes such as format (hardcover, paperback), publication date, and page count. The libraries dataset contains information regarding various libraries and their books catalogue.

1) Description of the data to be stored and moved into the analytics system

1. Details description of each file and its source:
   - **I. Books**
     a. Description: The books data file contains information regarding all the books available on the Goodreads database.
        - The books data file includes columns like the title, description, num_pages, publisher, language_code, ratings.
        - The file includes book_id as its primary key.
        - This file also includes author_id, work_id to join authors, and works files respectively to the books data file.
        - The size of this file is about 2GB zipped with more than 2.3 million books.

     b. Metadata:

     c.

Column Name	Data Type	Length
isbn	DT_STR	255
text_reviews_count	DT_STR	255
Country_code	DT_STR	255
language_code	DT_STR	255
asin	DT_STR	255
is_ebook	Int	4- byte
average_rating	DT_STR	255
Kindle_asin	DT_STR	255

description	DT_STR	255
format	DT_STR	255
link	DT_STR	255
publisher	DT_STR	255
num_pages	Int	4- byte
Publication_day	Date	
Isbn13	DT_STR	255
Publication-month	Date	
Edition_information	DT_STR	255
Publication_year	Date	
url	DT_STR	255
Image_url	DT_STR	255
Book_id	Int	4- byte
Ratings_count	Int	4- byte
Work_id	Int	4- byte
title	DT_STR	255
Title_without_series	DT_STR	255

d. Source:
https://drive.google.com/uc?id=1LXpK1UfqtP89H1tYy0pBGHjYk8IhigUK

II. Authors

a. Description: The author's file contains details about the author's name and other facts regarding the authors.
- The authors data file contains columns about the author name, text_review_count, rating_count, and average_rating.
- author_id is the primary key for this data file.
- The size of this file is about 17MB zipped with more than 800 thousand authors.

b. Metadata:

Column Name	Data type	Length
Author_id	Int	4- byte
Average rating	Int	4- byte
Text_reviews_count	Int	4- byte
Name	DT_STR	255
Ratings_count	Int	4- byte

c. Source:
https://drive.google.com/uc?id=19cdwyXwfXx_HDIgxXaHzH0mrx8nMyLvC

III. Works

a. Description: The works data file is an abstract version of a book regardless of any specific edition.

The works file contains information about the original_title, original_publication_year, original_language_id, reviews_count, and rating_sum.

- The primary key for this dataset is work_id
- The file size is about 73MB zipped with about 1.5 million records.

b. Metadata:

Column Name	Data Type	Length
Books_count	Int	4- byte
Reviews_count	Int	4- byte
Original_publication_month	Date	
Default_description_language_code	DT_STR	255
Text_reviews_count	Int	4- byte
Best_book_id	DT_STR	255
Original_publication_year	Date	
Original_title	DT_STR	255
Rating_dist	DT_STR	255
Default_chaptering_book_id	Int	4- byte
Original_publication_day	Date	
Original_language_id	Int	4- byte
Ratings-count	Int	4- byte
Media_type	DT_STR	255
Ratings_sum	Int	4- byte
Work_id	Int	4- byte

c. Source: https://drive.google.com/uc?id=1TLmSvzHvTLLLMjMoQdkx6pBWon-4bli7

IV. Series

a. Description: The series contains information about the book series and has details on the entire series.
   - The series data file contains columns such as title, description, series_works_count.
   - series_id is the primary key for the series data file.
   - The size of this file is 27MB zipped, with 400 thousand plus unique book series.

b. Metadata:

Column Name	Data Type	Length

numbered	Int	4- byte
note	DT_STR	255
description	DT_STR	255
title	DT_STR	255
Series_works_count	Int	4- byte
Series_id	Int	4- byte
Primary_work_count	Int	4- byte

c. Source: https://drive.google.com/uc?id=1op8D4e5BaxU2JcPUgxM3ZqrodajryFBb

V. Genres

a. Description: The genre file contains a subset of genres for each book along with the count of the number of times it was tagged.
   - The genres data file contains only two columns the book_id and the genre.
   - There is no primary key like a genre_id to the genre data, but book_id references all the genres.
   - The size of this file is 23MB zipped, with 2.3 million lines of data.

b. Metadata:

Column Name	Data Type	Length
Book_id	Int	4- byte
History, historical fiction	DT_STR	255
fiction	DT_STR	255
Fantasy, paronormal	DT_STR	255
Mystery, thriller, crime	DT_STR	255
poetry	DT_STR	255
romance	DT_STR	255
Non-fiction	DT_STR	255
children	DT_STR	255
Young_adult	DT_STR	255
Comics, graphic	DT_STR	255

c. Source: https://drive.google.com/uc?id=1ah0_KpUterVi-AHxJ03iKD6O0NfbK0md

2. Joining all these different datasets for the creation of a single data warehouse:
   - The Goodreads dataset contains five files, namely books, authors, works, series, and genres.
   - The book's data file had book_id as its primary key and author_id, works_id, and series as the foreign keys.
   - The authors table can be joined to the books table with author_id as its key.
   - The works table can be joined to the books table with work_id as its key.

- The series table has series_id as its primary key, which can be joined to the series column in the books table.

The genre table doesn't have its primary key but has book_id as its foreign, which also can be used to join the books table to the genre table.

3. Data transformation tasks
    - The name column in the authors' table contains the author's full name, which can be converted into its distinct values as first_name, middle_initial, and last_name.
    - The book publishing details are separated based on publication_year, publication_month, and publication_day. These columns can be consolidated into a single derived column of published_date.

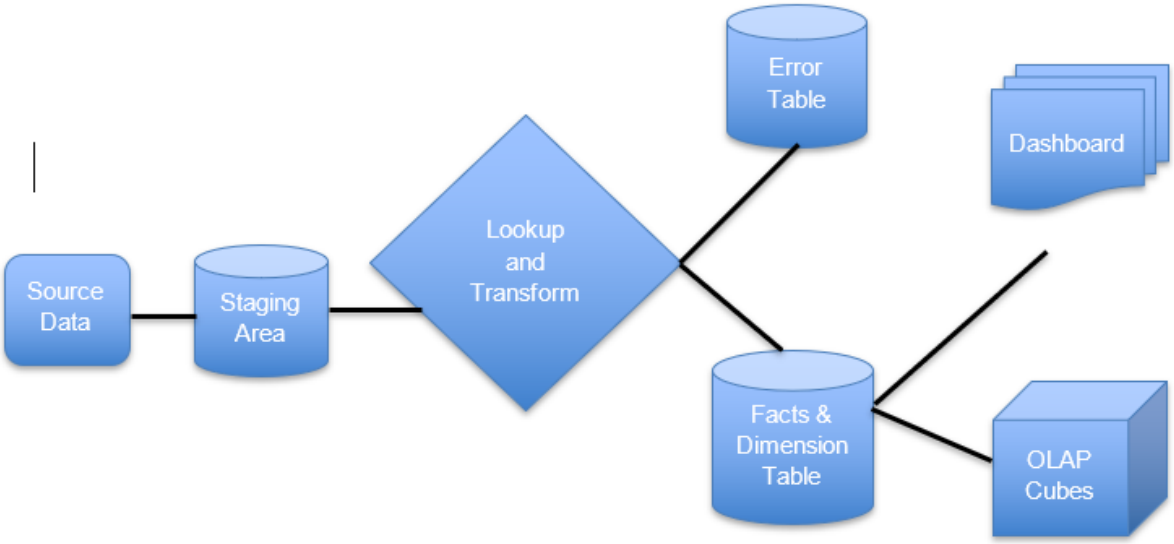
4. High-level dimension for the Goodreads dataset
    - This dataset's dimension will mainly contain four categories: books, authors, works, and series.
    - These dimensions will mainly contain all the data regarding the book title, author name, works original_title, series title, etc.
    - There will also be a fifth dimension called the date dimension. The date dimension will contain all the dates regarding the book release_date, publication_year, etc.
    - Along with all these dimensions, there will also be a fact table containing all the primary keys of all the dimensions along with a surrogate key of the fact table.
    - The fact table will contain all the facts of the dataset, which may include the num_pages, reviews_count, average_rating, etc

Revision History

VERSION NUMBER	REVISION DATE	PROFESSOR COMMENTS	REVISION DESCRIPTION
Version 3	12/04/2020	Column naming isn't consistent… Large datatypes should break tables apart if columns can be that large You have a lot of work here, but this is not a design document … it is a collection of ddl and screenshots	-Added revision history to the design doc -Developed consistent naming for all column's names and table names in staging and Dimension/Fact tables. -Changed the Data model design to break up the data further. -Changed design doc to include ER diagram of Staging area and Datawarehouse model (Star schema). -Also, Developed OLAP CUBES for better reporting -Improved visualization and included Tableau visualizations.
Version 2	11/15/2020	Add a revision history Since all this data is from a single source there is very little difficulty joining or cleaning the data You will need to make some error and split files so you can load data Can you bring in some other data? weather / state income some other source to mash up Very good but please add something into it to increase ethe complexity of joining the data	-Researched to find other data sets that could provide meaning or connect with the Good Reads data. Added Tableau's data (bookshop and libraries data). Also added Lookup Functions in SSIS package to check for certain errors.
Version 1	11/5/2020		Built first draft of project proposal based on the GoodReads Dataset and tried to include as many points covered by professor

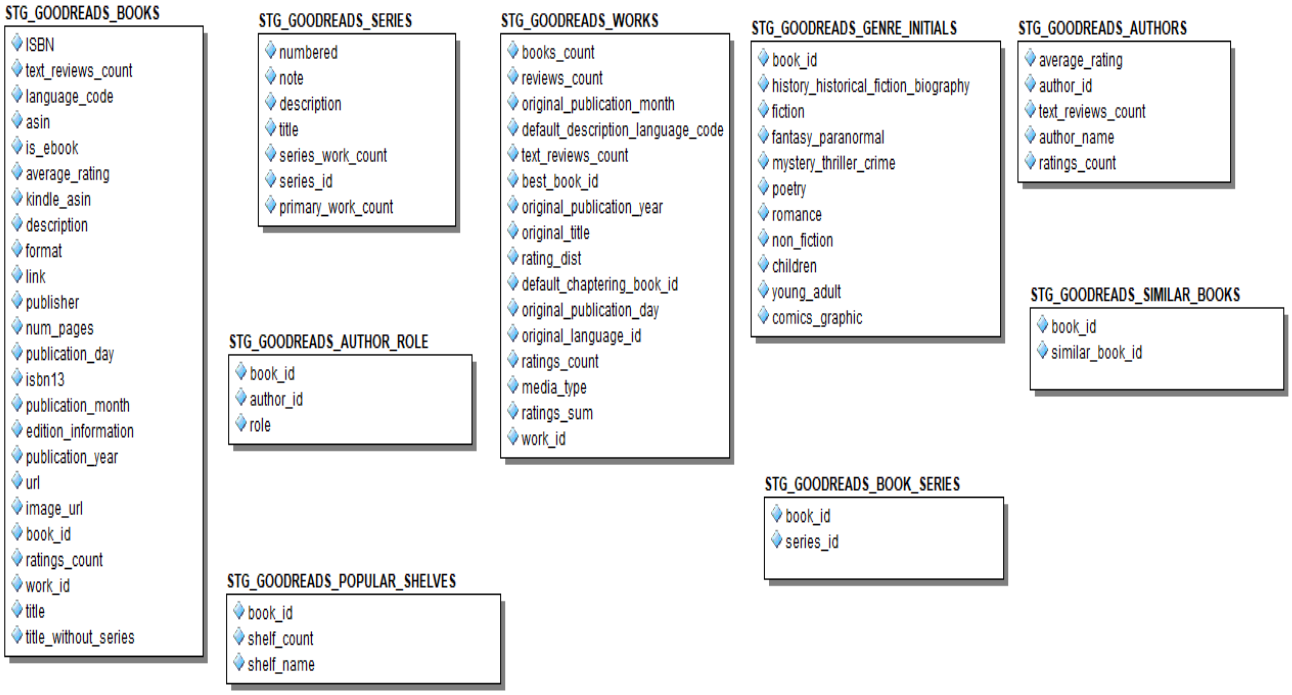
Data Model

OVERVIEW OF DATA MODEL/DATA FLOW

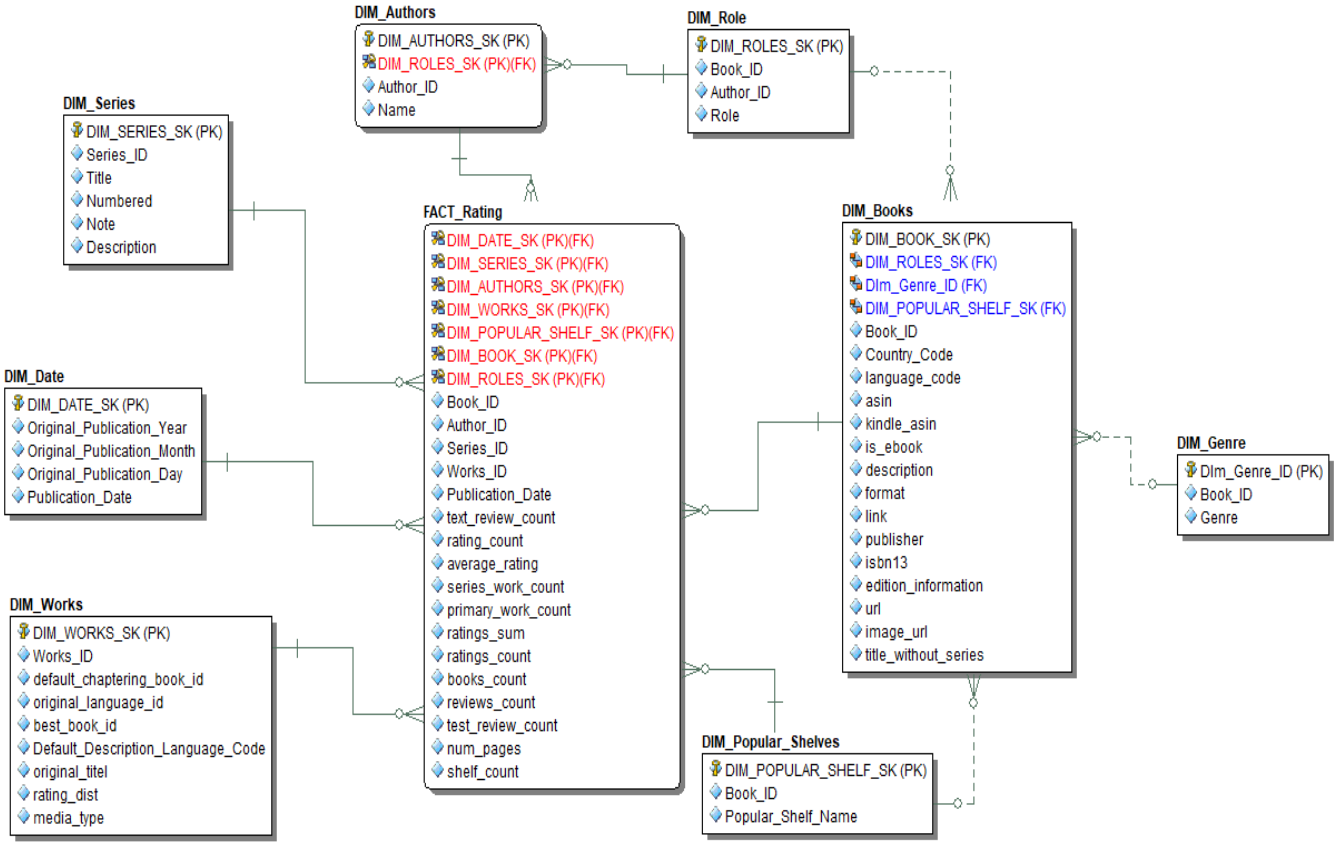


ER Diagram

Staging Area ER Diagram



ER Diagram of Dimension/Fact Table (Star Schema)



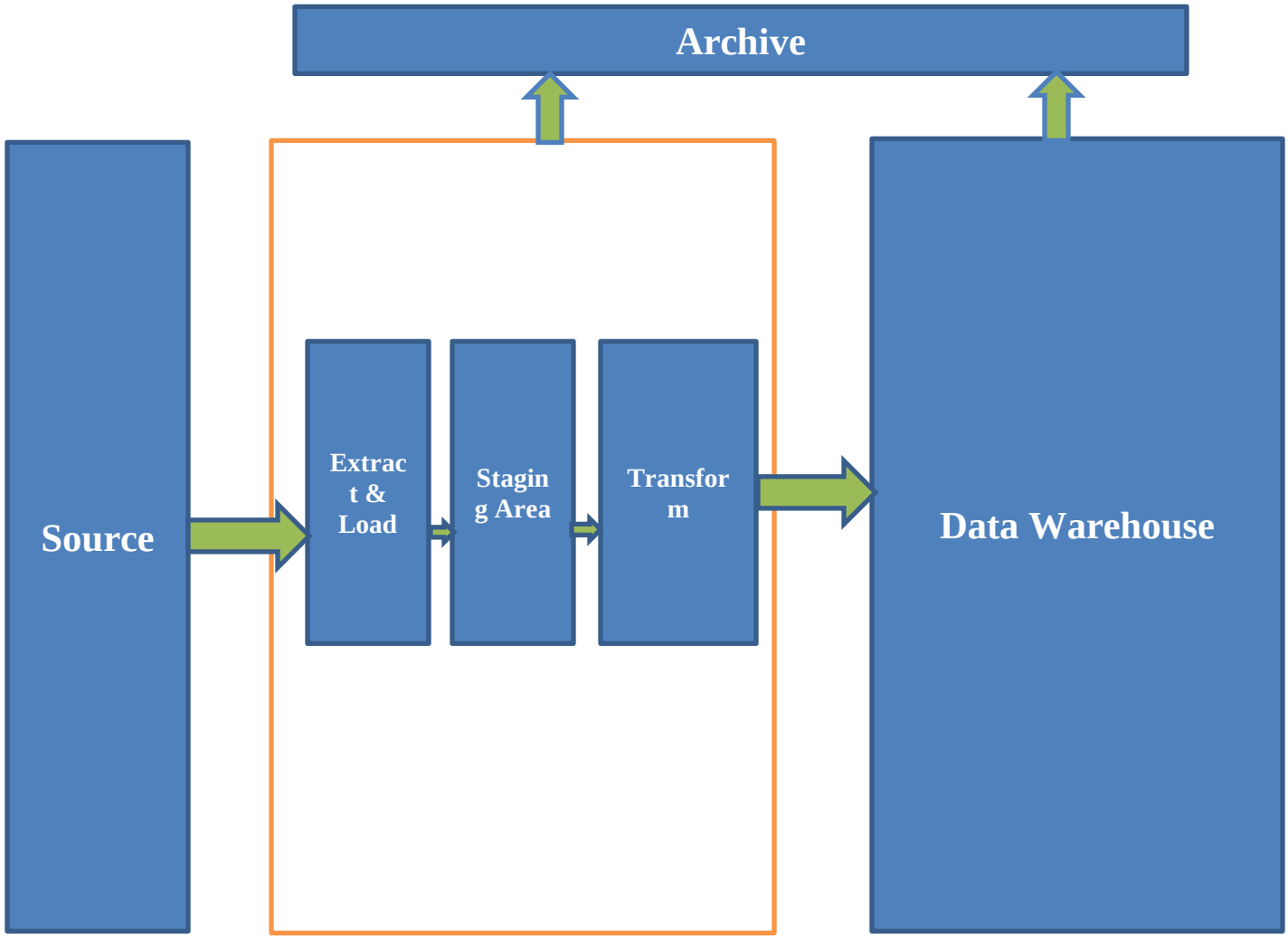
We are using a Star schema model for the data warehouse, with the fact tables at the center and the dimension tables branching out of it. With the Fact tables containing measures regarding Ratings and its aggregates, the book counts, number of pages and shelf count in it. The fact table gives a measure of the ratings given to books, along with numerical data about the ratings and books being analyzed. The corresponding dimension tables which include details regarding authors, book genres, shelf details, work and series details and a dimension for the date. The dimension tables provide business context about the who, what, where and why of the books and the ratings being provided to them.

Although all the dimension tables have natural primary keys in them, we are generating surrogate identity primary keys, so that when we need to perform slowly changing dimension in the dimensions, redundant data is not created. The Primary key for the fact table is a combination of surrogate primary keys from all the dimension tables.

All the Dimension tables have a one-to-many relationships with the fact tables. The book and genre tables have a many to many relationships and are connected through a bridge table. The popular shelves and Book Dimensions are connected through an optional one to many relationships. Also, the Role dimension is connected to the authors dimension and books dimension by a one to many and one to one relationship.

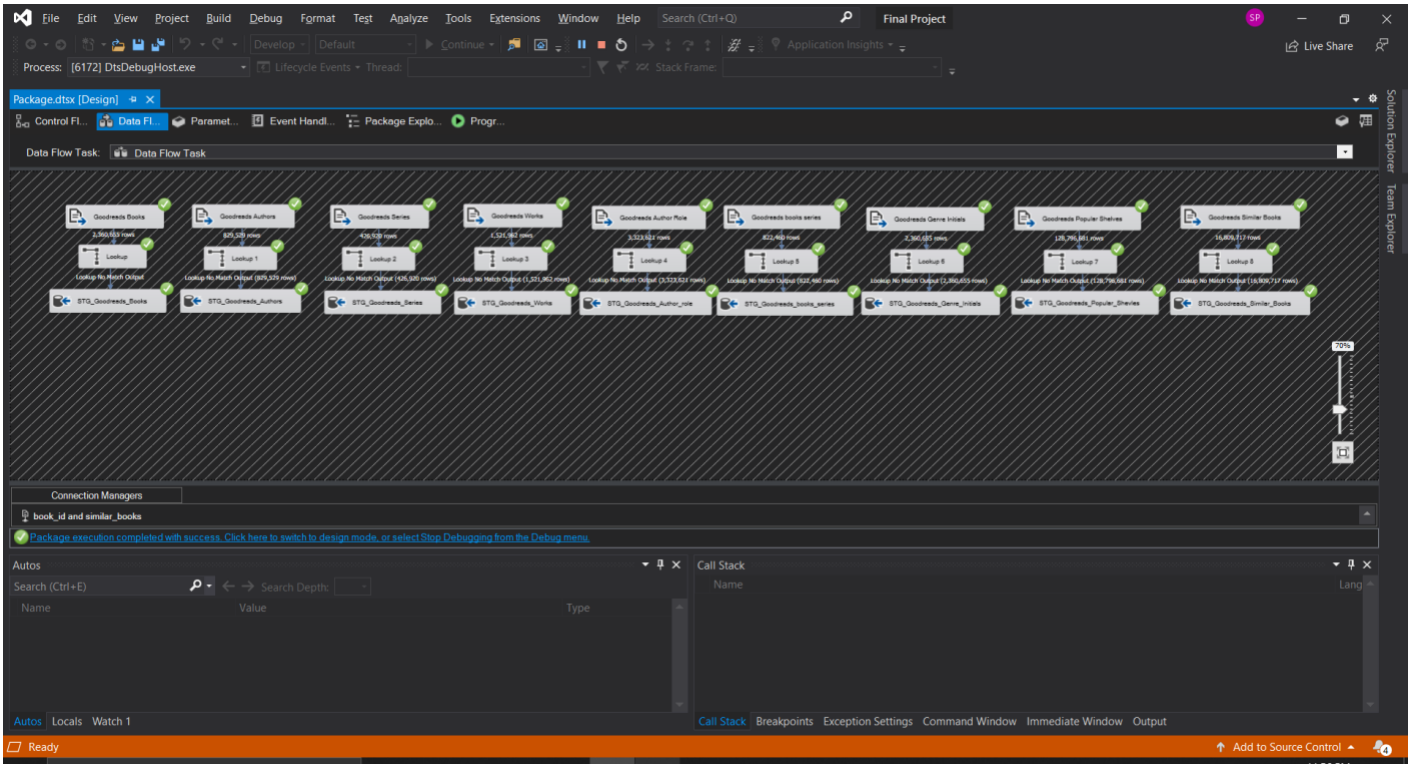
Data Flow & Implementation

Data Flow Diagram



Loading data into Staging Tables

We have collected number of files containing data from the Goodreads website, which includes data regarding the authors, books, series, popular shelves, ratings, roles of authors, genre of book, etc. We load the data from these files into the staging area tables, which are all varchar datatype, to ensure ease of loading. After this, we use SSIS to create packages that will transform the data in the staging area to appropriate data types and load it into the Dimension/Fact tables.



Error Handling and Data Validation

Data testing and validation needs to be done to ensure data integrity, accuracy and consistency to comply with data standards. This is done to make sure that the data inside the data warehouse is integrated to be reliable enough for an enterprise to decide on.
Error handling and Data Validation is divided into the following parts:

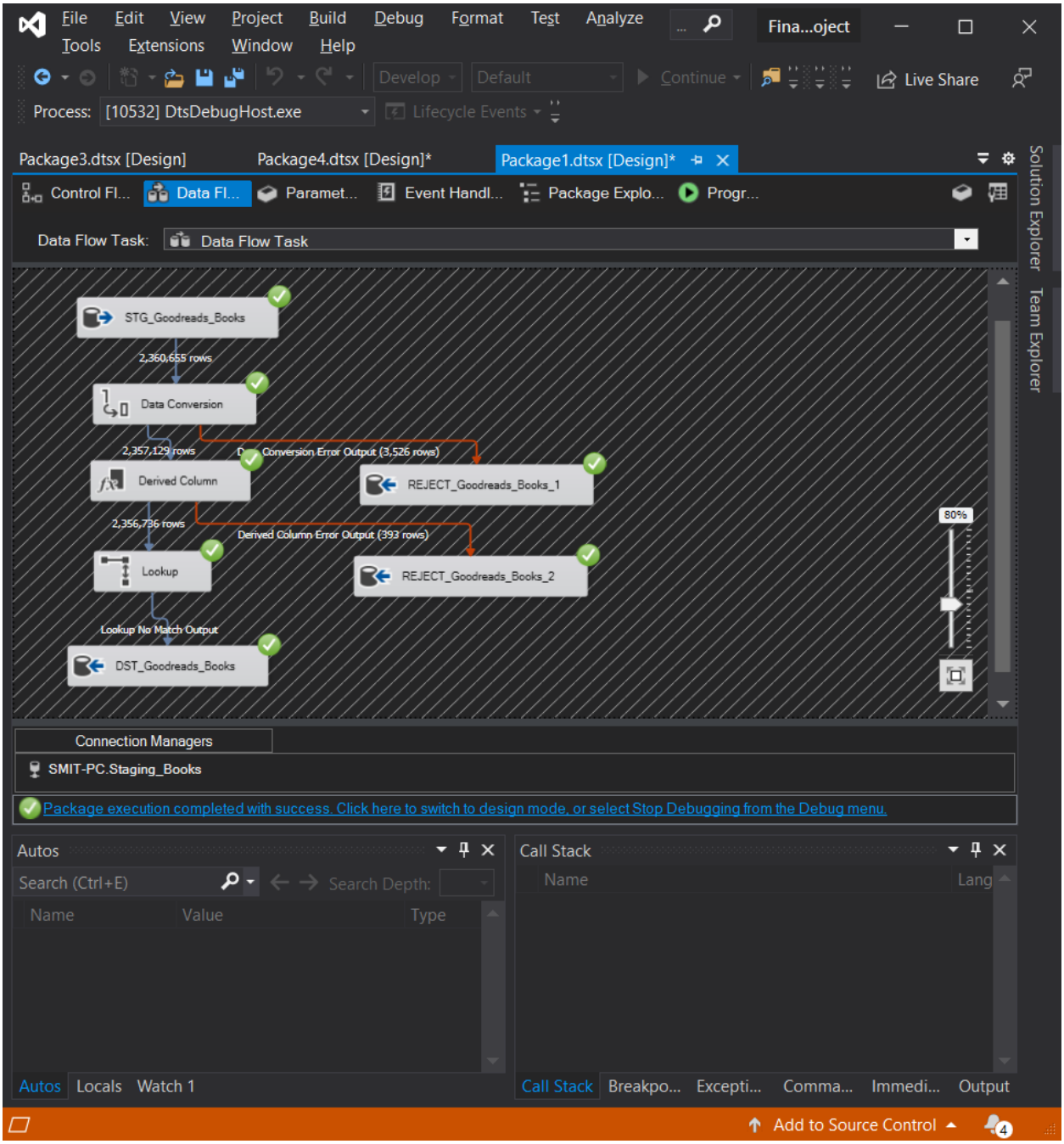
- Cleaning and Data conversion
- Gathering Data into multiple Dimension tables and Fact table, with generation of Surrogate keys for each Dimension Table.
- Logging and storing errors into error tables

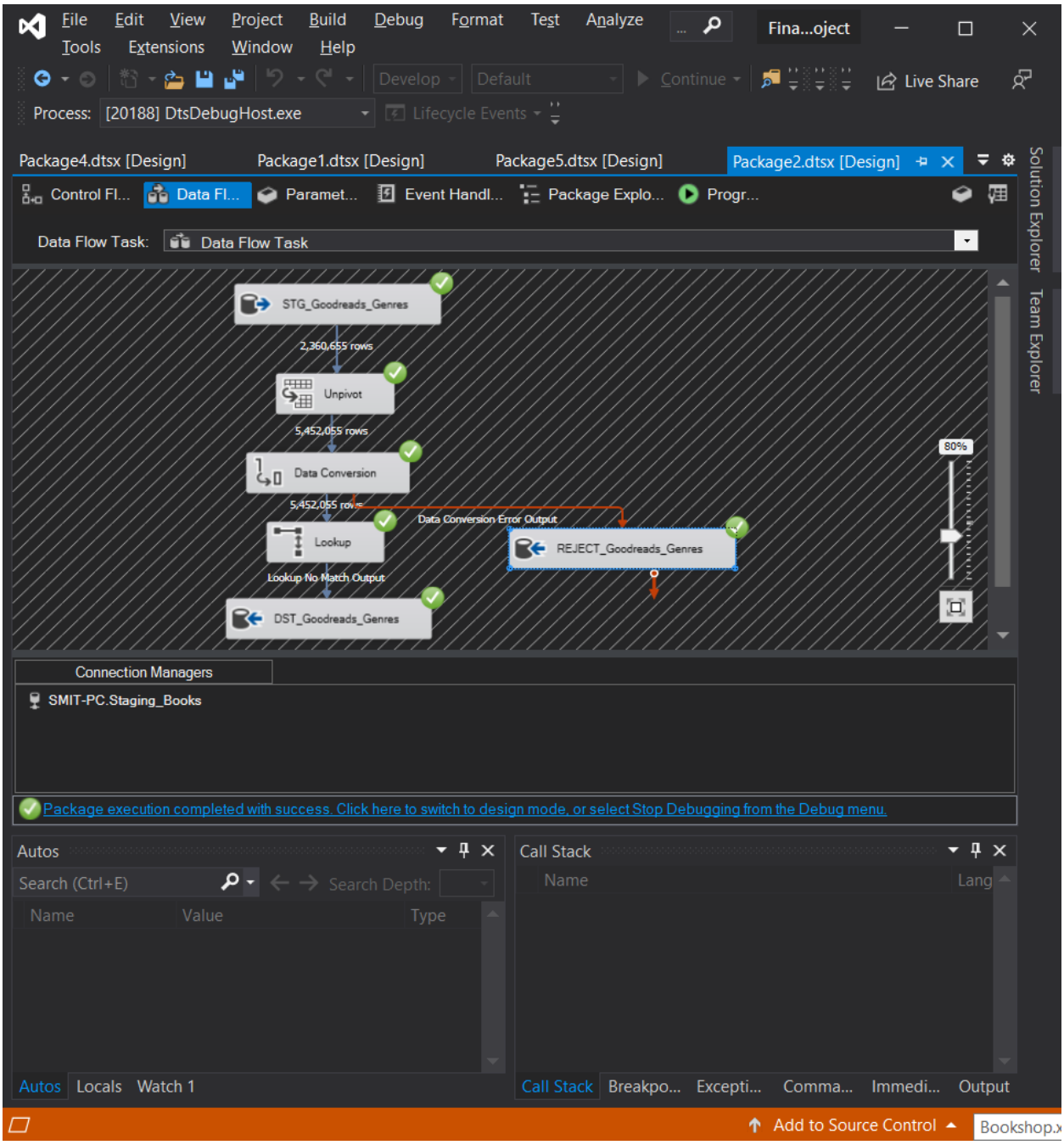
The SSIS package combines LOOKUP and Derived Columns for error handling, which then moves error rows directly to ERROR tables with error type and error log. We used Data Conversion to convert the data in the staging area from VARCHAR to the essential data types in each Dimension Table.
The types of errors handled by the SSIS package are:

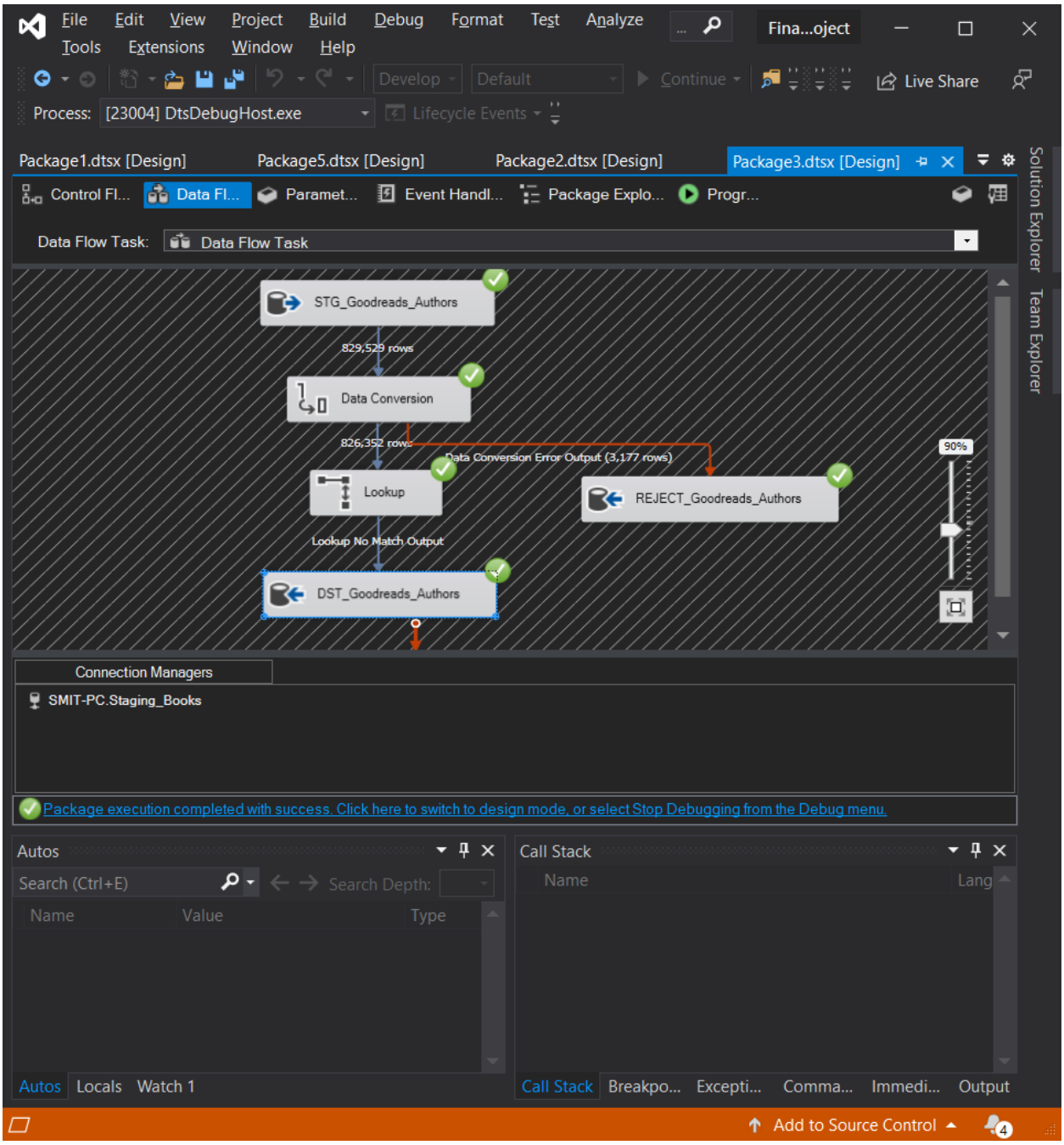
- Check for errors in data types i.e., ensure that integer type values are not entered into Attributes like Name in DIM_Authors and Genre in Dim_Genre.
- Check that Dates developed by the package are proper and not random.

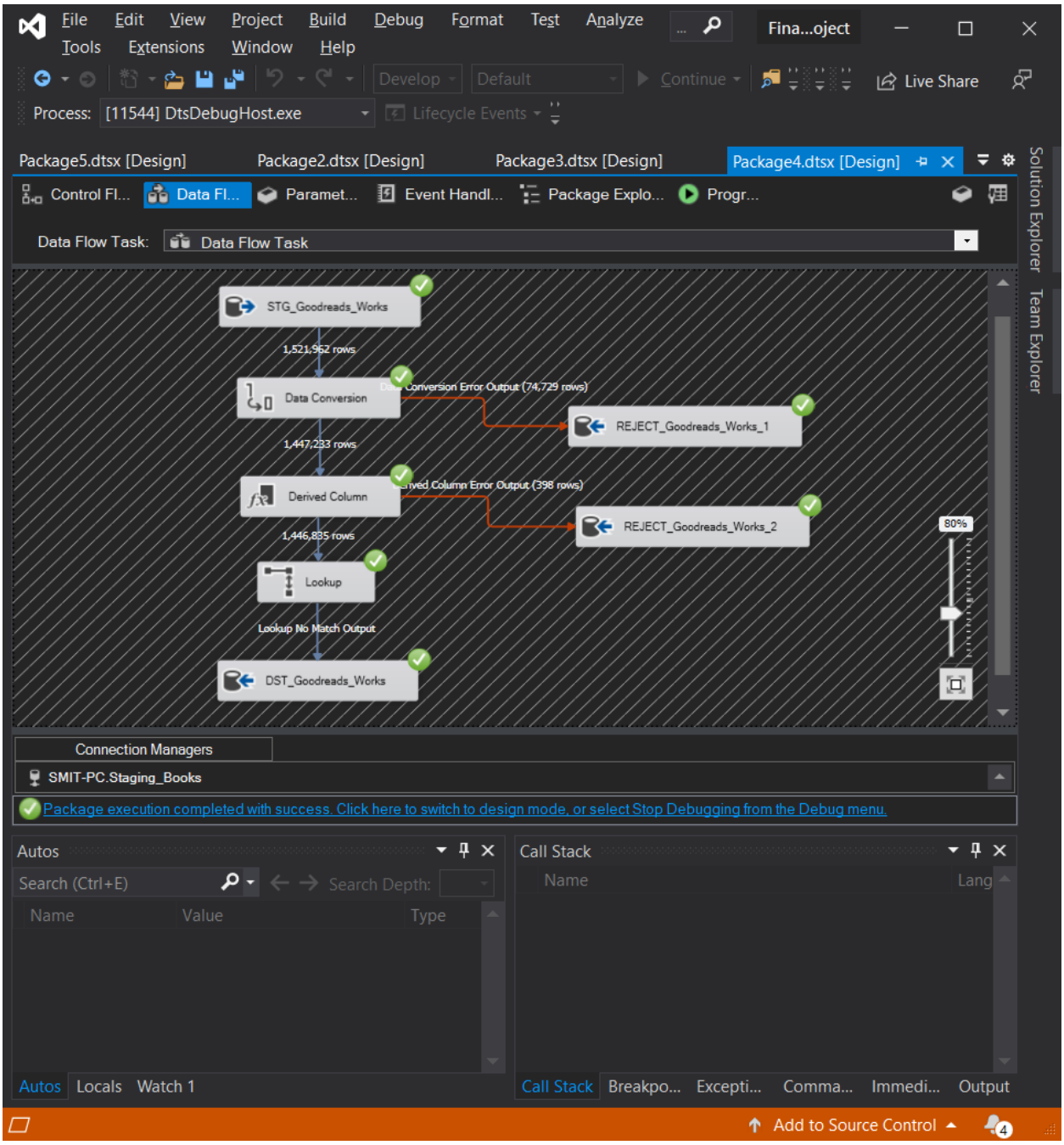
- All the integers should be positive values or blank as there are some negative values which does not make sense for the data
- Check for duplicate entries in all the dimension tables.

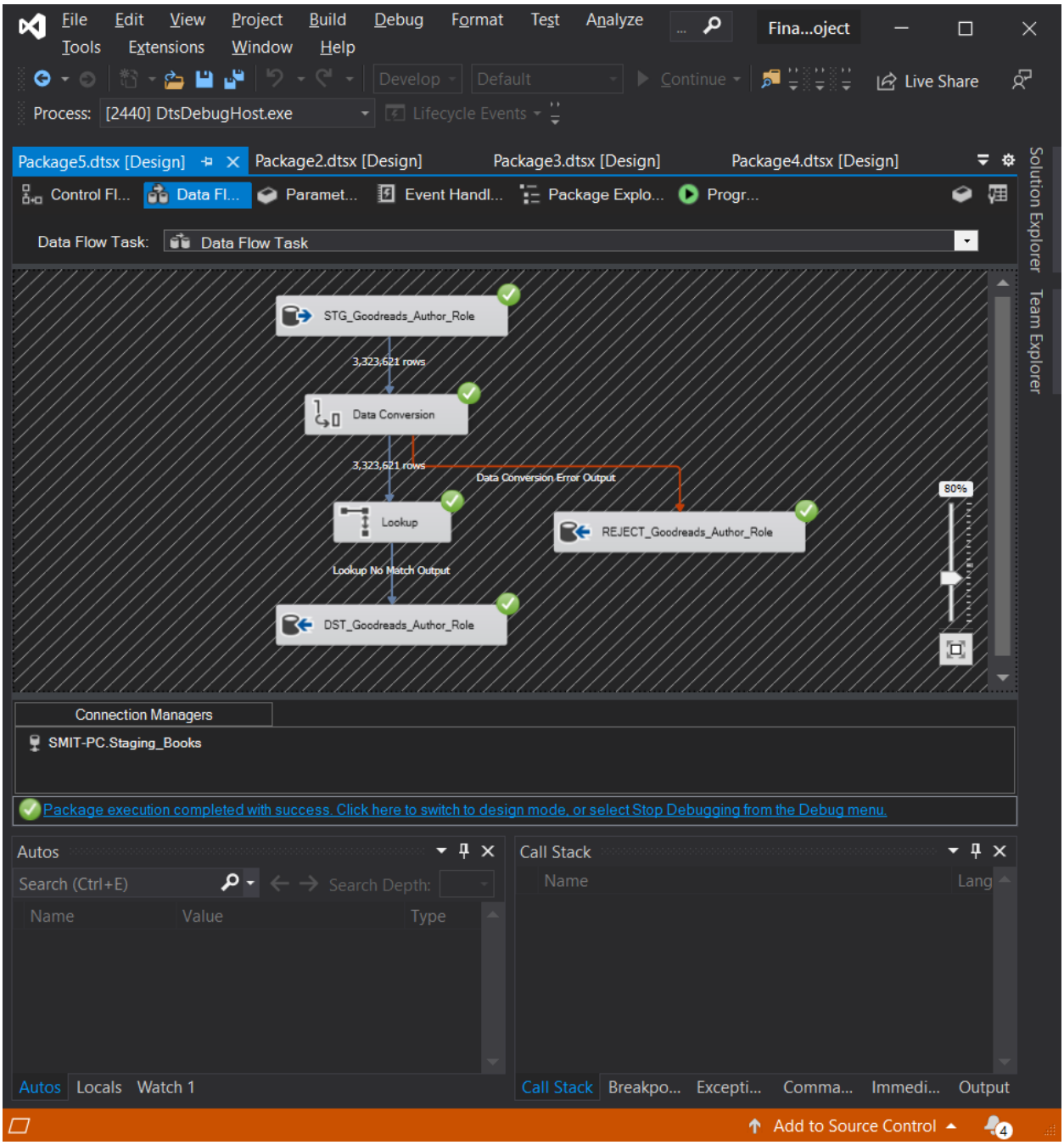
Screenshots of error handling form actual project:




File Edit View Project Build Debug Format Test Analyze
Tools Extensions Window Help
Fina...oject
Develop
Default
Continue
Live Share
Process: [20188] DtsDebugHost.exe
Lifecycle Events
Package4.dtsx [Design]
Package1.dtsx [Design]
Package5.dtsx [Design]
Package2.dtsx [Design]
Control Fl...
Data Fl...
Paramet...
Event Handl...
Package Explo...
Progr...
Data Flow Task:
Data Flow Task
STG_Goodreads_Genres
2,360,655 rows
Unpivot
5,452,055 rows
Data Conversion
5,452,055 rows
Data Conversion Error Output
Lookup
REJECT_Goodreads_Genres
Lookup No Match Output
DST_Goodreads_Genres
80%
Solution Explorer
Team Explorer
Connection Managers
SMIT-PC.Staging_Books
Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.
Autos
Search (Ctrl+E)
Search Depth:
Name
Value
Type
Autos
Locals
Watch 1
Call Stack
Name
Lang
Call Stack
Breakpo...
Excepti...
Comma...
Immedi...
Output
Add to Source Control
Bookshop.x


File Edit View Project Build Debug Format Test Analyze
Tools Extensions Window Help
Fina...oject
Develop
Default
Continue
Live Share
Process: [23004] DtsDebugHost.exe
Lifecycle Events
Package1.dtsx [Design]
Package5.dtsx [Design]
Package2.dtsx [Design]
Package3.dtsx [Design]
Control Fl...
Data Fl...
Paramet...
Event Handl...
Package Explo...
Progr...
Data Flow Task:
Data Flow Task
STG_Goodreads_Authors
829,529 rows
Data Conversion
826,352 rows
Data Conversion Error Output (3,177 rows)
Lookup
REJECT_Goodreads_Authors
Lookup No Match Output
DST_Goodreads_Authors
90%
Solution Explorer
Team Explorer
Connection Managers
SMIT-PC.Staging_Books
Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.
Autos
Search (Ctrl+E)
Search Depth:
Name
Value
Type
Call Stack
Name
Lang
Autos
Locals
Watch 1
Call Stack
Breakpo...
Excepti...
Comma...
Immedi...
Output
Add to Source Control


File Edit View Project Build Debug Format Test Analyze
Tools Extensions Window Help
Fina...oject
Develop
Default
Continue
Live Share
Process: [11544] DtsDebugHost.exe
Lifecycle Events
Package5.dtsx [Design]
Package2.dtsx [Design]
Package3.dtsx [Design]
Package4.dtsx [Design]
Control Fl...
Data Fl...
Paramet...
Event Handl...
Package Explo...
Progr...
Data Flow Task:
Data Flow Task
STG_Goodreads_Works
1,521,962 rows
Data Conversion
Conversion Error Output (74,729 rows)
REJECT_Goodreads_Works_1
1,447,233 rows
Derived Column
rived Column Error Output (398 rows)
REJECT_Goodreads_Works_2
1,446,835 rows
Lookup
Lookup No Match Output
DST_Goodreads_Works
80%
Solution Explorer
Team Explorer
Connection Managers
SMIT-PC.Staging_Books
Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.
Autos
Search (Ctrl+E)
Search Depth:
Name
Value
Type
Call Stack
Name
Lang
Autos
Locals
Watch 1
Call Stack
Breakpo...
Excepti...
Comma...
Immedi...
Output
Add to Source Control

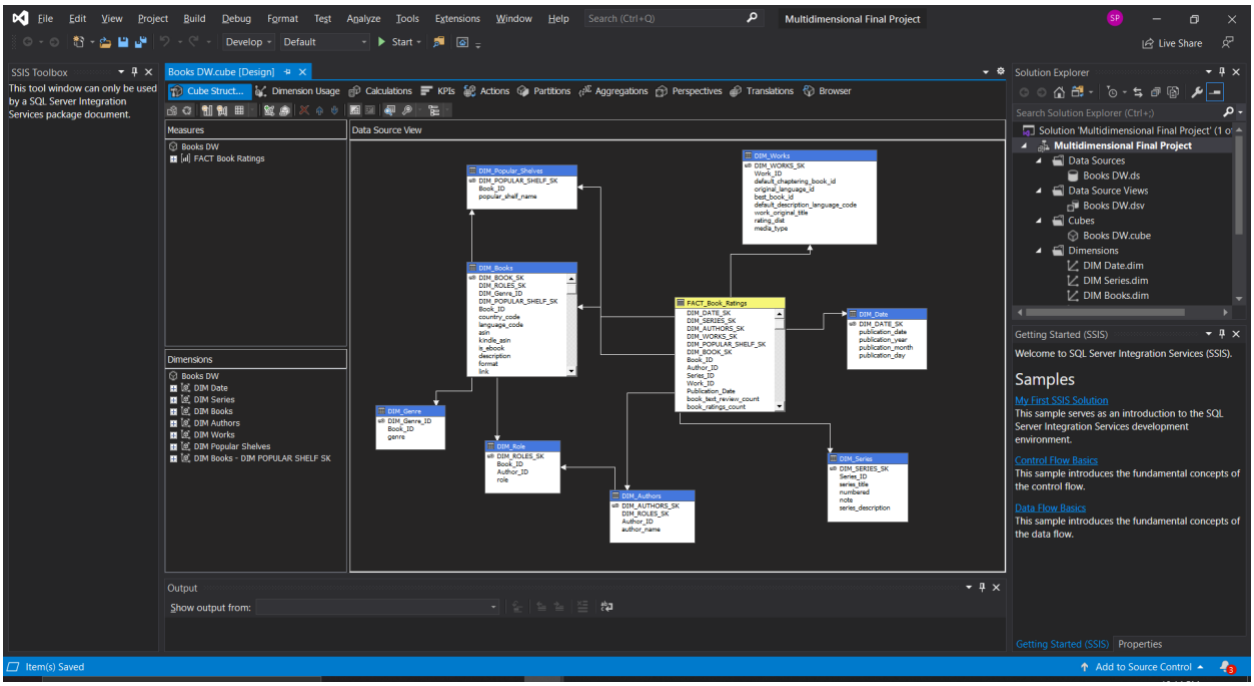


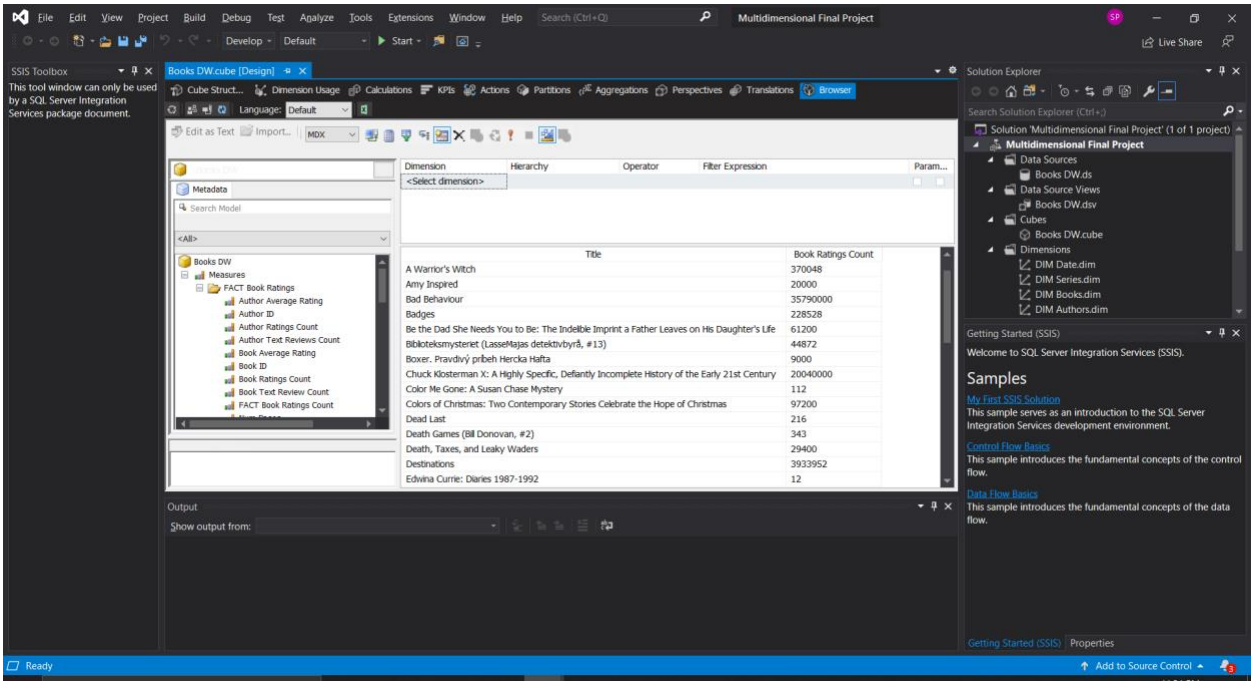
Cube Design (OLAP)

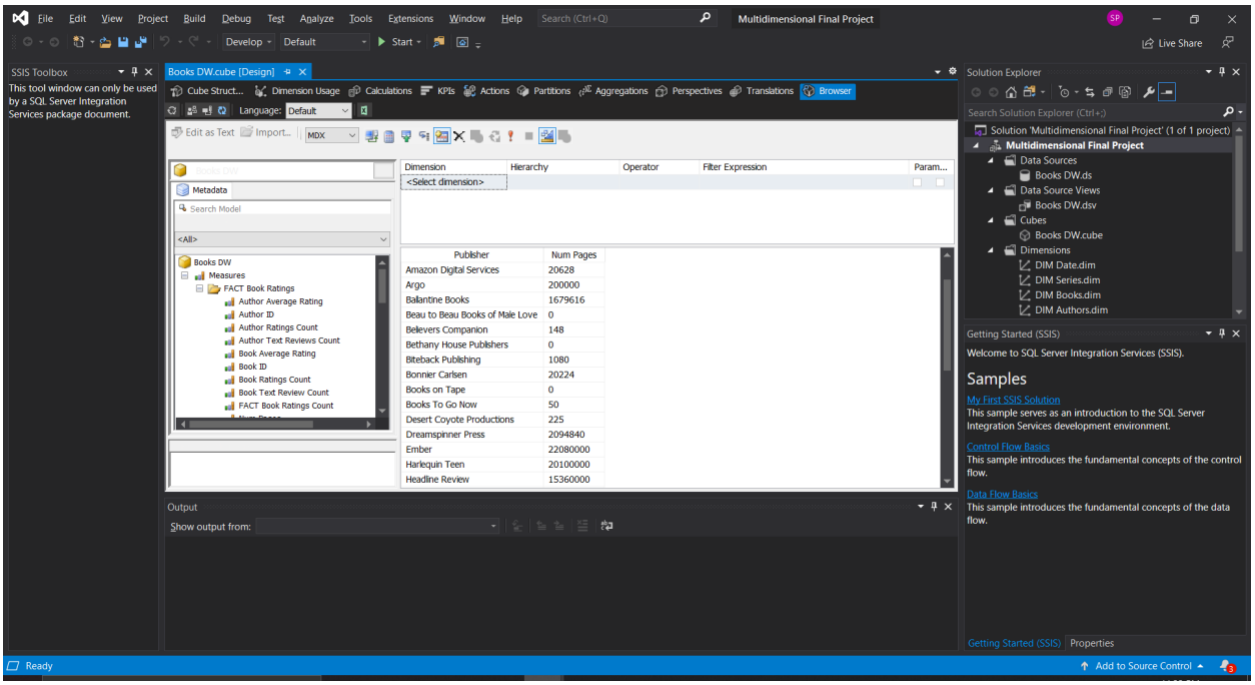
An OLAP cube is a multi-dimensional array of data. Online analytical processing (OLAP) is a computer-based technique of analyzing data to look for insights. OLAP data is typically stored in a star schema or snowflake schema in a relational data warehouse or in a special-purpose data management system. Measures are derived from the records in the fact table and dimensions are derived from the dimension tables. An OLAP cube is a data structure that overcomes the limitations of relational databases by providing rapid analysis of data. Cubes can display and sum large amounts of data while also providing users with searchable access to any data points. This way, the data can be rolled up, sliced, and diced as needed to handle the widest variety of questions that are relevant to a user's area of interest. The useful feature of an OLAP cube is that the data in the cube can be contained in an aggregated form. To the user, the cube seems to have the answers in advance because assortments of values are already precomputed.

OLAP CUBE

We created the OLAP cube so that, we can drill up and down and create various measures based on dimensional descriptions.

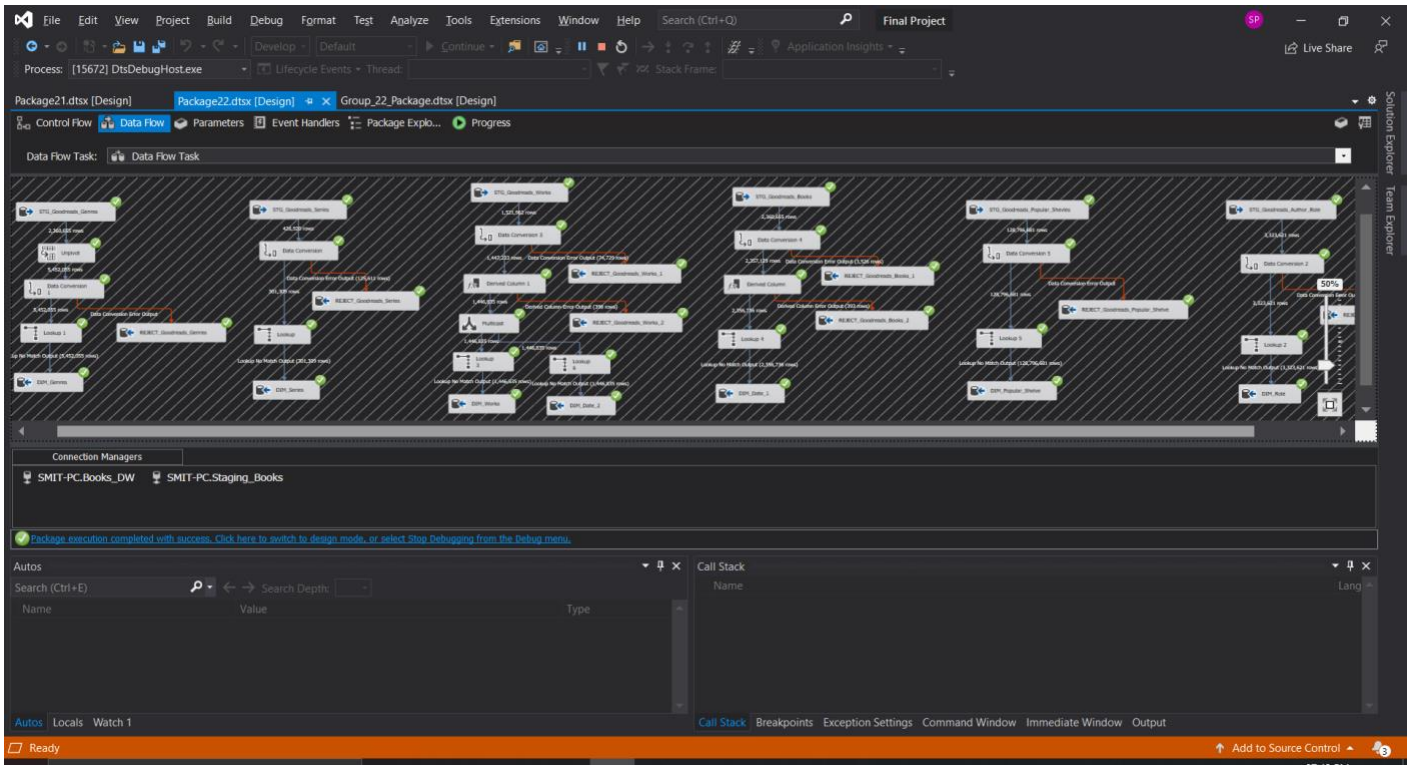



File Edit View Project Build Debug Test Analyze Tools Extensions Window Help
Multidimensional Final Project
Develop Default Start Live Share
SSIS Toolbox
This tool window can only be used by a SQL Server Integration Services package document.
Books DW.cube [Design]
Cube Struct... Dimension Usage Calculations KPIs Actions Partitions Aggregations Perspectives Translations Browser
Language: Default
Edit as Text Import... MDX
Metadata
Search Model
<All>
Books DW
Measures
FACT Book Ratings
Author Average Rating
Author ID
Author Ratings Count
Author Text Reviews Count
Book Average Rating
Book ID
Book Ratings Count
Book Text Review Count
FACT Book Ratings Count
Dimension Hierarchy Operator Filter Expression Param...
<Select dimension>
Title Book Ratings Count
A Warrior's Witch 370048
Amy Inspired 20000
Bad Behaviour 35790000
Badges 228528
Be the Dad She Needs You to Be: The Indelible Imprint a Father Leaves on His Daughter's Life 61200
Biblioteksmysteriet (LasseMajas detektivbyrå, #13) 44872
Boxer. Pravdivý príbeh Hercka Hafta 9000
Chuck Klosterman X: A Highly Specific, Defiantly Incomplete History of the Early 21st Century 20040000
Color Me Gone: A Susan Chase Mystery 112
Colors of Christmas: Two Contemporary Stories Celebrate the Hope of Christmas 97200
Dead Last 216
Death Games (Bill Donovan, #2) 343
Death, Taxes, and Leaky Waders 29400
Destinations 3933952
Edwina Currie: Diaries 1987-1992 12
Output
Show output from:
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Multidimensional Final Project' (1 of 1 project)
Multidimensional Final Project
Data Sources
Books DW.ds
Data Source Views
Books DW.dsv
Cubes
Books DW.cube
Dimensions
DIM Date.dim
DIM Series.dim
DIM Books.dim
DIM Authors.dim
Getting Started (SSIS)
Welcome to SQL Server Integration Services (SSIS).
Samples
My First SSIS Solution
This sample serves as an introduction to the SQL Server Integration Services development environment.
Control Flow Basics
This sample introduces the fundamental concepts of the control flow.
Data Flow Basics
This sample introduces the fundamental concepts of the data flow.
Getting Started (SSIS) Properties
Ready
Add to Source Control


File Edit View Project Build Debug Test Analyze Tools Extensions Window Help
Multidimensional Final Project
Develop Default Start Live Share
SSIS Toolbox
This tool window can only be used by a SQL Server Integration Services package document.
Books DW.cube [Design]
Cube Struct... Dimension Usage Calculations KPIs Actions Partitions Aggregations Perspectives Translations Browser
Language: Default
Edit as Text Import... MDX
Metadata
Search Model
<All>
Books DW
Measures
FACT Book Ratings
Author Average Rating
Author ID
Author Ratings Count
Author Text Reviews Count
Book Average Rating
Book ID
Book Ratings Count
Book Text Review Count
FACT Book Ratings Count
Dimension Hierarchy Operator Filter Expression Param...
<Select dimension>
Publisher Num Pages
Amazon Digital Services 20628
Argo 200000
Ballantine Books 1679616
Beau to Beau Books of Male Love 0
Believers Companion 148
Bethany House Publishers 0
Biteback Publishing 1080
Bonnier Carlsen 20224
Books on Tape 0
Books To Go Now 50
Desert Coyote Productions 225
Dreamspinner Press 2094840
Ember 22080000
Harlequin Teen 20100000
Headline Review 15360000
Output
Show output from:
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Multidimensional Final Project' (1 of 1 project)
Multidimensional Final Project
Data Sources
Books DW.ds
Data Source Views
Books DW.dsv
Cubes
Books DW.cube
Dimensions
DIM Date.dim
DIM Series.dim
DIM Books.dim
DIM Authors.dim
Getting Started (SSIS)
Welcome to SQL Server Integration Services (SSIS).
Samples
My First SSIS Solution
This sample serves as an introduction to the SQL Server Integration Services development environment.
Control Flow Basics
This sample introduces the fundamental concepts of the control flow.
Data Flow Basics
This sample introduces the fundamental concepts of the data flow.
Getting Started (SSIS) Properties
Ready
Add to Source Control

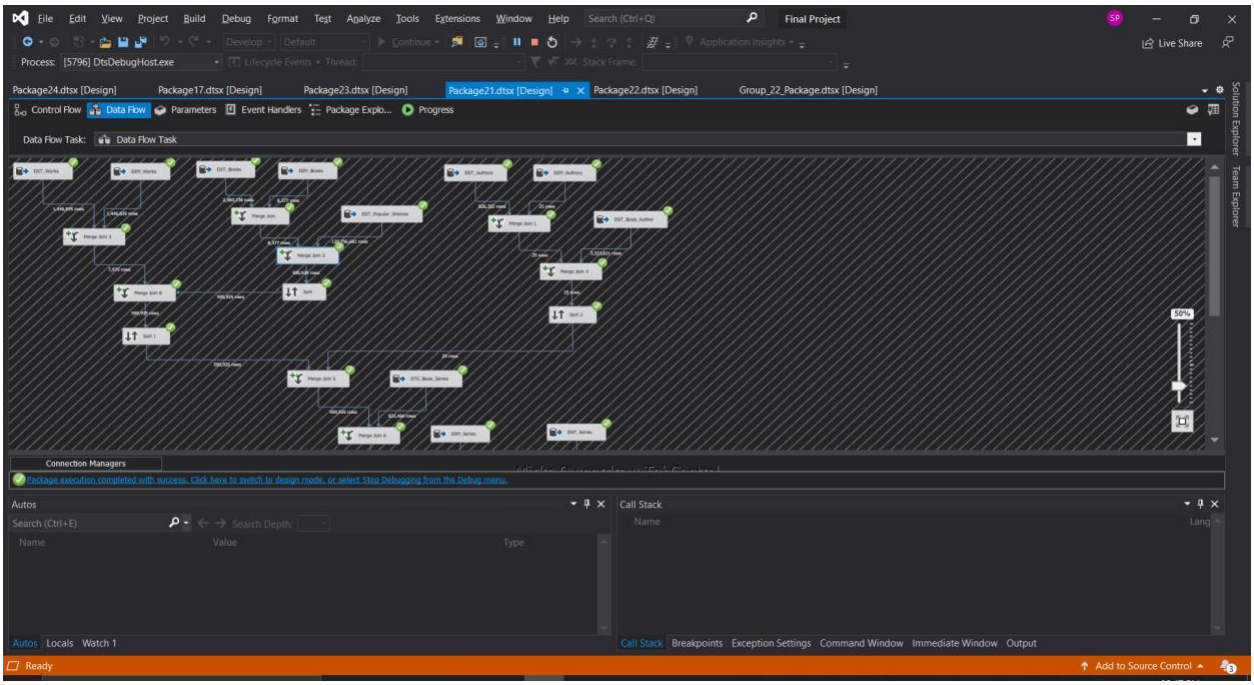
Challenges Faced and Solution for it

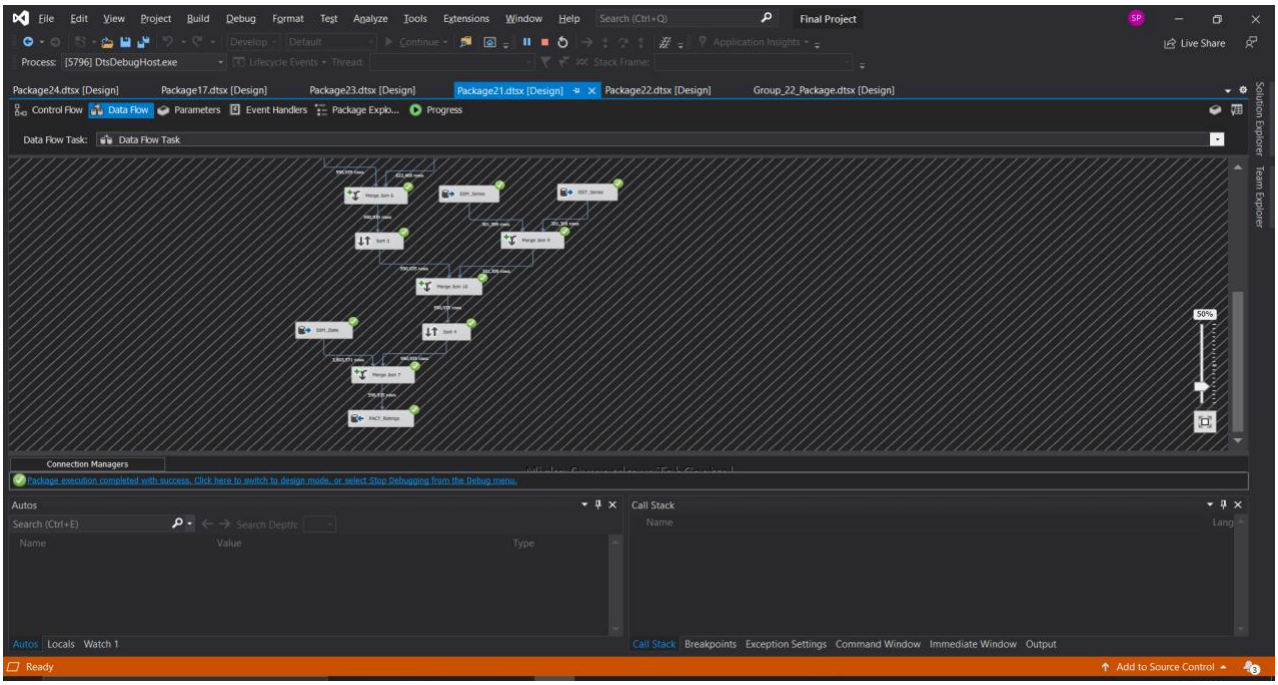
- Genre data from the source data is in the form of multiple columns having genre as attribute names, i.e., attribute names were Fiction, Fantasy, Paranormal, Poetry, Mystery, Thriller, Crime, Children, etc. These attributes had to be combined into a single attribute called Genre for the DIM_Genre dimension table. This challenge was solved using LOOKUP which would check for the type of Genre in staging data and add it to Genre attribute in DIM_Genre.
- The source data had date values separated into Publication_day, Publication_month and Publication_year attributes. So, we had to create a new attribute Publication_Date for DIM_Date dimension for each book. This challenge was solved by matching the bookid with work_id using LOOKUP and then using Derived Column to combine the three attributes into a single Date.
- The source data contained a table called Roles which links both the DIM_Authors and DIM_Books dimension. This challenge was solved by adding DIM_Role as a bridge table between DIM_Authors and DIM_Books dimension.



Loading data in Fact table

To load the Fact table, we have to match all the dimension with the measures from the staging table. We use, the MERGE JOIN function to join the dimension table surrogate keys with the measures like rating_count, ratings_sum, review_count, etc. pulled from the various staging tables.

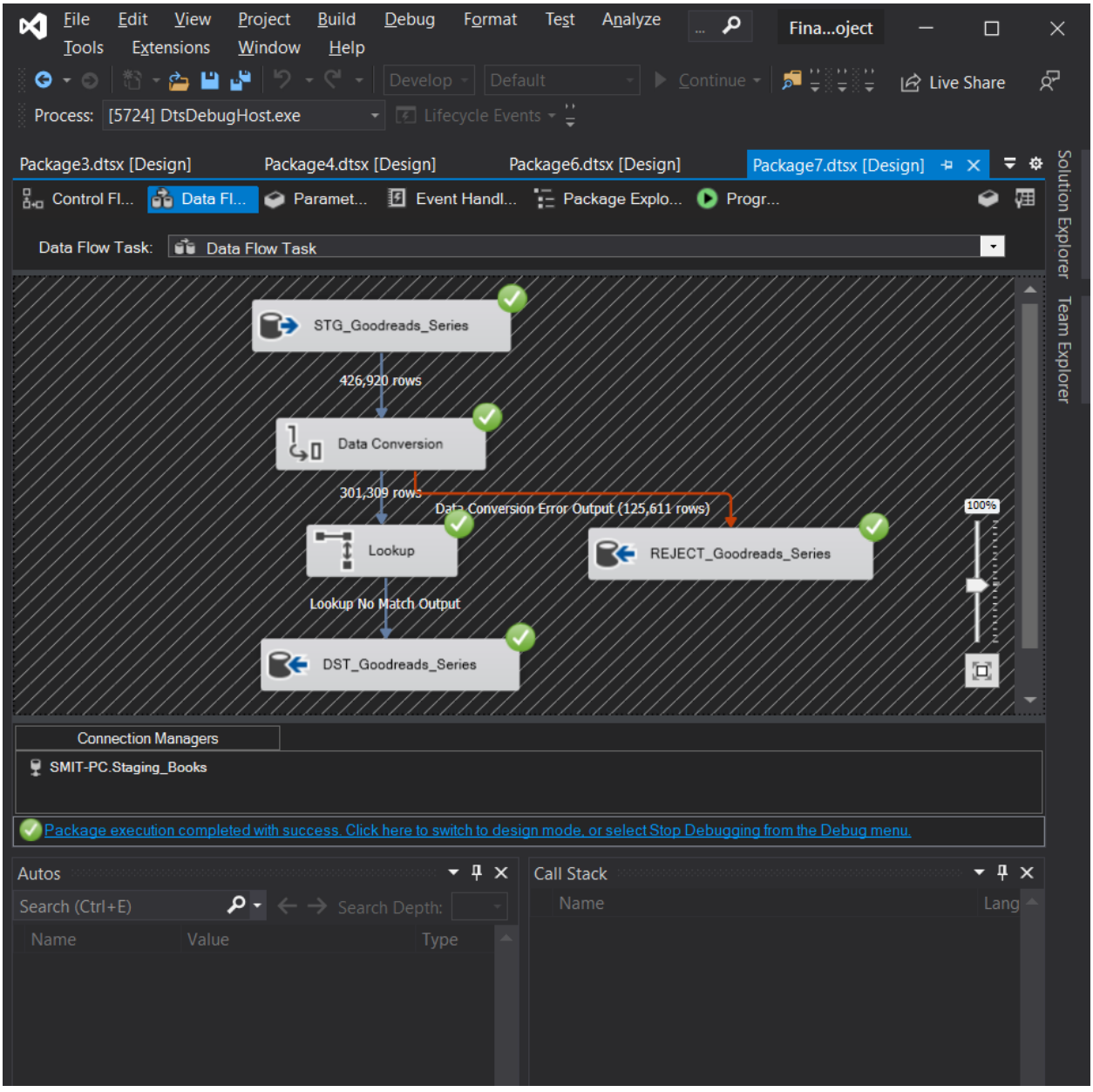




Logging

To maintain the entire ETL operation in a state of constant improvement, ETL with the correct logging process is important to help the team manage bugs and issues with data sources, data formats, transformations, destinations, etc. Errors can occur in between loading millions of rows of data and it can be difficult to pinpoint what exactly caused the error. Without logs, the whole process would have to be re-run to figure out each and every errors.
For logging, we have to create a new table with columns for storing data like error number, error message, rowed in which error was created and error type. We can use a conditional split task in SSIS to load the data into the error logging table whenever the conditional split fails. We can also use another method, wherein we use the SSIS log provider in the SQL server to create logs of error generated while running the SSIS package.

Screenshots of logging errors in the project:



First Time vs Monthly Load

The first load is generally called Full or Destructive load. During full load, there is generally no problem, because the tables in the Data warehouse are empty and there is no historic data that may need to be updated or deleted. The first full load is a simple transformation of the data, cleaning the data and direct load to the tables in the DW. Monthly load is generally knowing as Incremental load. Incremental load is a process of incrementally loading data. The destination is only loaded with new and modified data. Data that has not been changed will be left alone. In this method, data integrity can also be assured, but ETL can become complicated. An incremental load pattern will try to classify the data that has been generated or changed since the load process was last running. This is different from the traditional full data load, which copies from a given source the entire data collection. The selectivity of the incremental design generally decreases the overhead required for the ETL phase.
Incremental load in SSIS can be done using Change Data Capture (CDC) feature. We can use Lookup function within SSIS to check the date and time of last load and compare it to the date and time of current load. The above step is generally called source change detection. If the lookup has no match, then data can be inserted into the destination tables in the DW. If the Lookup has match it has two options – Delete or Update. We use the Conditional Split function in SSIS to decide what needs to be done to the data that matches with the historic data. Finally, we use Union All function in SSIS to merge the output from Insert, delete and Update into the destination tables in DW.

Data Warehouse Design along with history

The traditional data warehouse architecture had mainly three layers bottom layer contains all data ingestion logic and ETL processes. The ETL processes connect to data sources and extract data to local staging databases. The middle layer in OLAP layer that supports reporting and analytical logic, at this layer data can be transformed further and aggregated for BI processes.
Final layer, provides stakeholder web interfaces for viewing and querying reports or analytical data, as well as tools for visualization and market intelligence for end users conducting ad-hoc analysis. Data warehouses are relational databases related to conventional schemas, which are how data are represented and structured. Tables and their associations are arranged by a snowflake schema such that a representative entity-relationship diagram (ERD) resembles a snowflake. A centralized fact table refers to several tables of measurements, which refer to further tables of dimensions themselves, and so on. A particular case of the snowflake schema is the simplified star schema. The core fact table is only related to one level of dimension tables, resulting in ERDs with star forms. These dimension tables are denormalized, containing all the properties and data correlated with the unique record form they carry.

Data Mart Design with aggregates

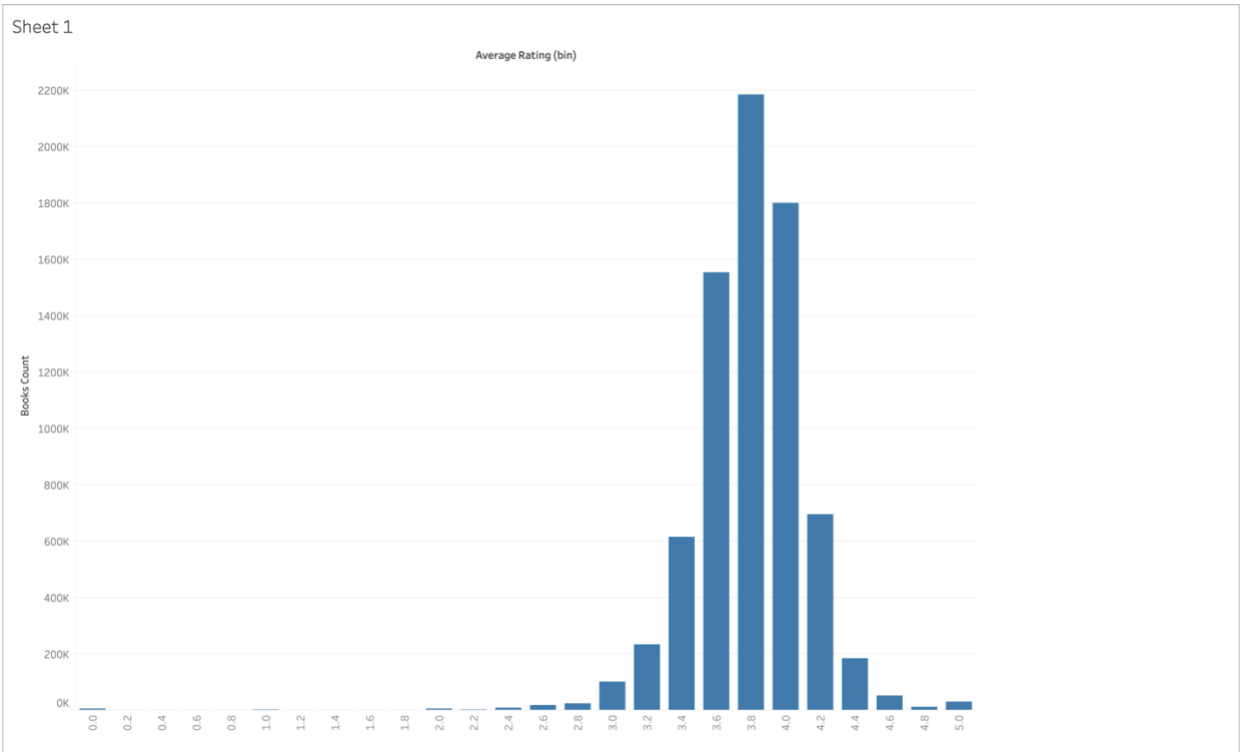
Data Marts will be designed to access subject-oriented information. For this project we will be creating a dependent Data Mart i.e., a subset of data will be used based on Publications. Physical Database design will be implemented using tables, indexes, views, etc. Populating the data mart will include Source data to target data Mapping, Cleaning and transformation operations on the data, Creating and storing metadata, etc. Finally, reporting would be done using queries

Data Analysis and Visualizations

(Visualizations made using Tableau and R Studio)

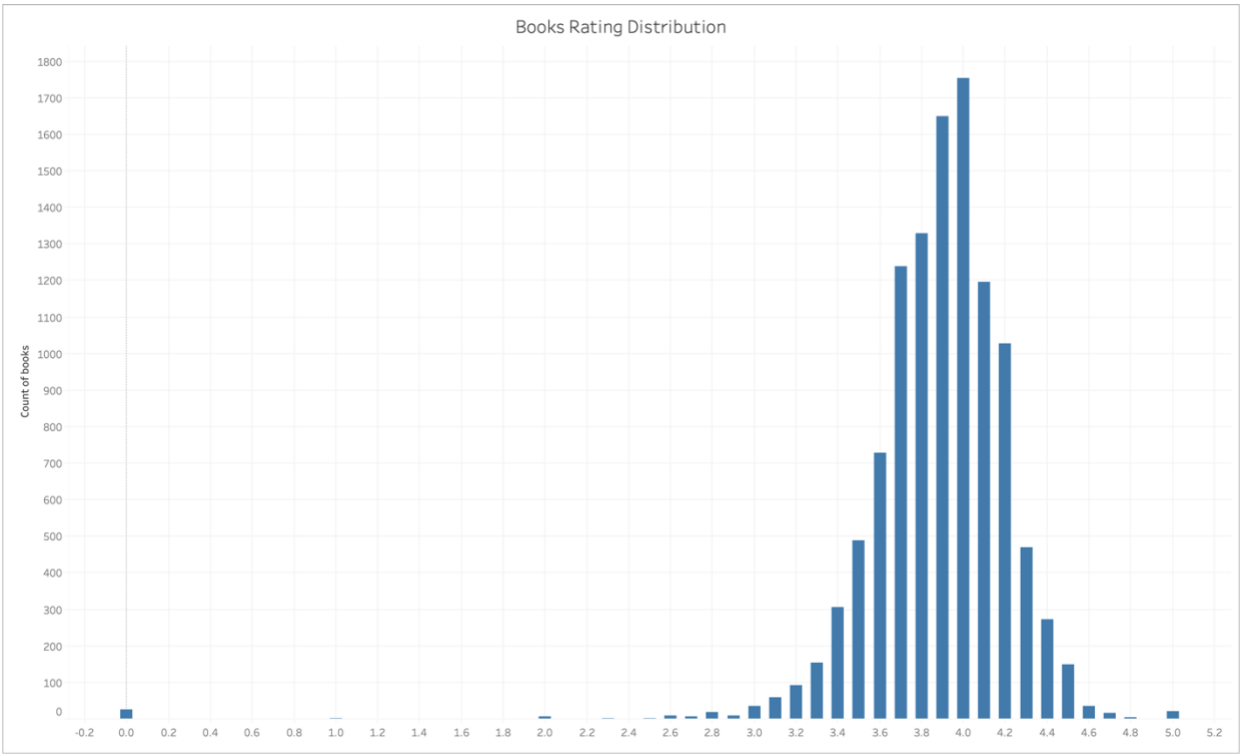
1. Authors Rating Distribution

Utilized the rating data to see what will motivate the user to rate a book.



2. Books Rating Distribution

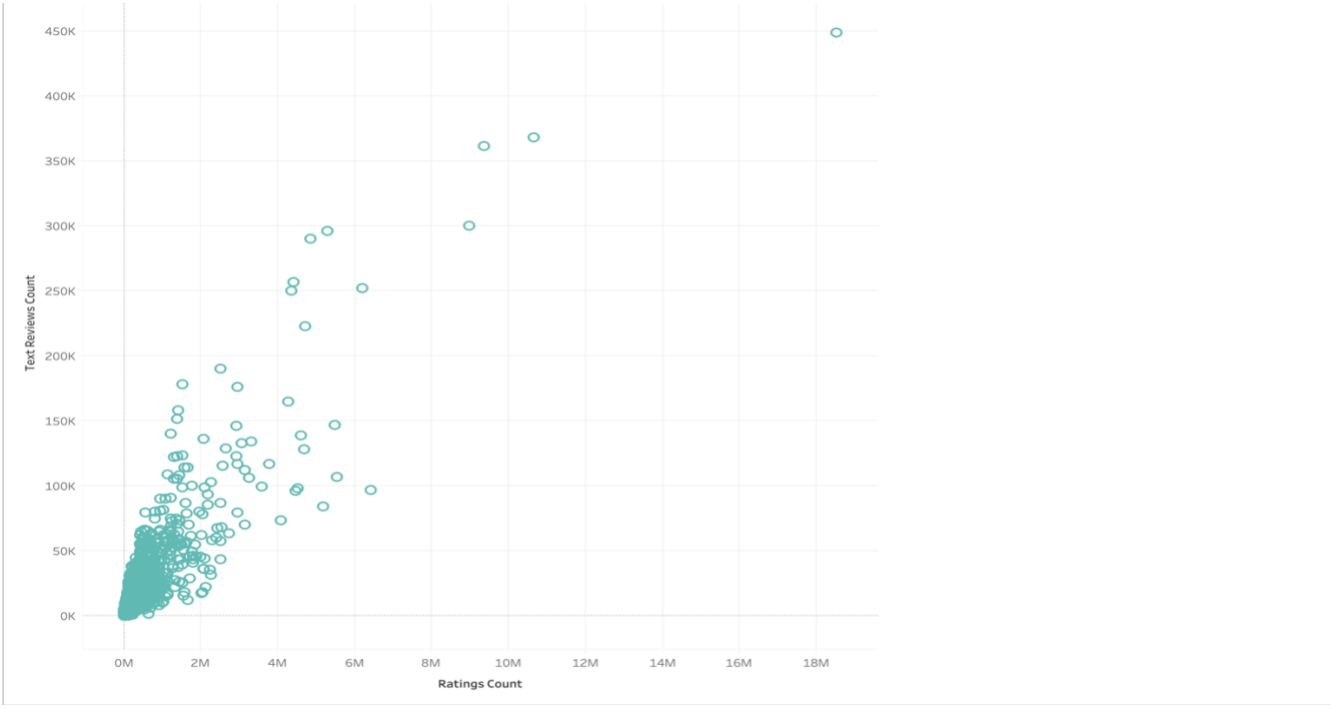
Observation: It is observed that majority of readers rate only the books they loved.



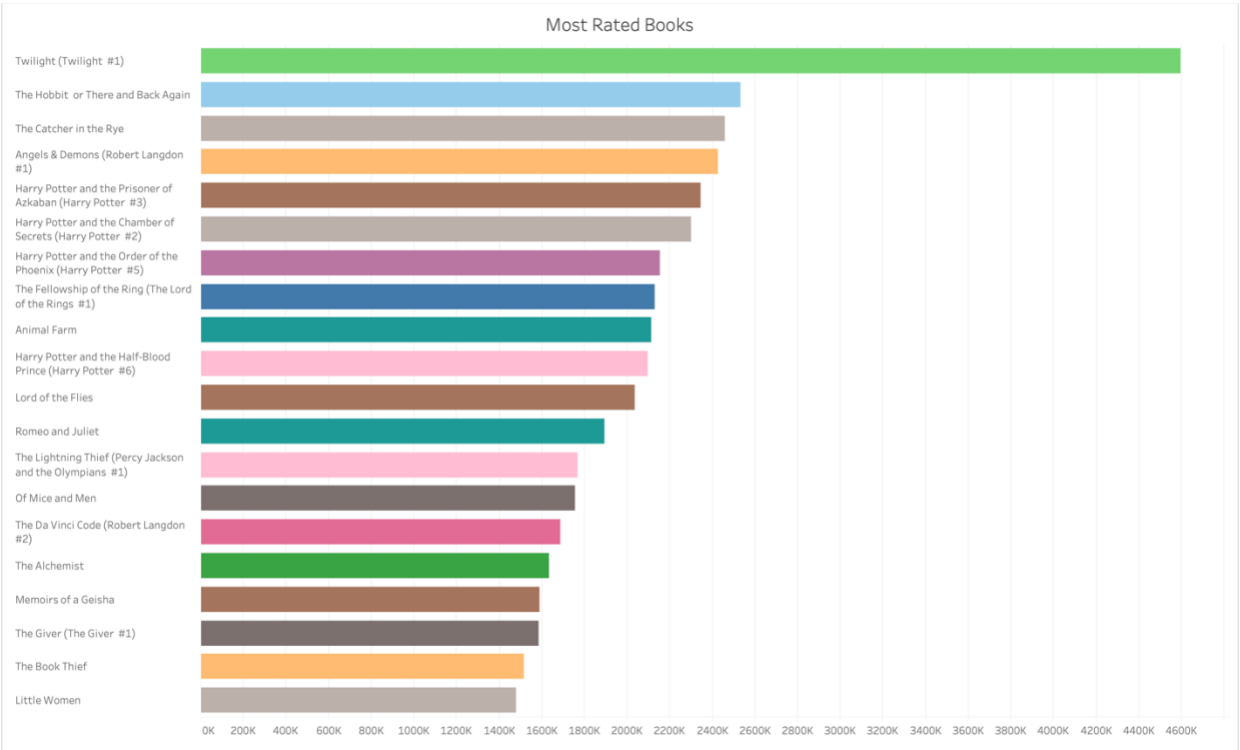
3. Books with Highest Ratings

	title	authors	average_rating	ratings_count	weighted_rating
2	Harry Potter and the Sorcerer's Stone (Harry P...	J.K. Rowling-Mary GrandPré	4.47	5629932	25165796.04
2000	The Hobbit or There and Back Again	J.R.R. Tolkien	4.26	2364968	10074763.68
4	Harry Potter and the Prisoner of Azkaban (Harr...	J.K. Rowling-Mary GrandPré	4.55	2149872	9781917.60
5301	Harry Potter and the Chamber of Secrets (Harry...	J.K. Rowling-Mary GrandPré	4.41	2115562	9329628.42
1	Harry Potter and the Order of the Phoenix (Har...	J.K. Rowling-Mary GrandPré	4.49	1996446	8964042.54
0	Harry Potter and the Half-Blood Prince (Harry ...	J.K. Rowling-Mary GrandPré	4.56	1944099	8865091.44
25	The Fellowship of the Ring (The Lord of the Ri...	J.R.R. Tolkien	4.35	2009749	8742408.15
4456	A Game of Thrones (A Song of Ice and Fire #1)	George R.R. Martin	4.45	1598396	7112862.20
8752	The Lightning Thief (Percy Jackson and the Oly...	Rick Riordan	4.24	1645445	6976686.80

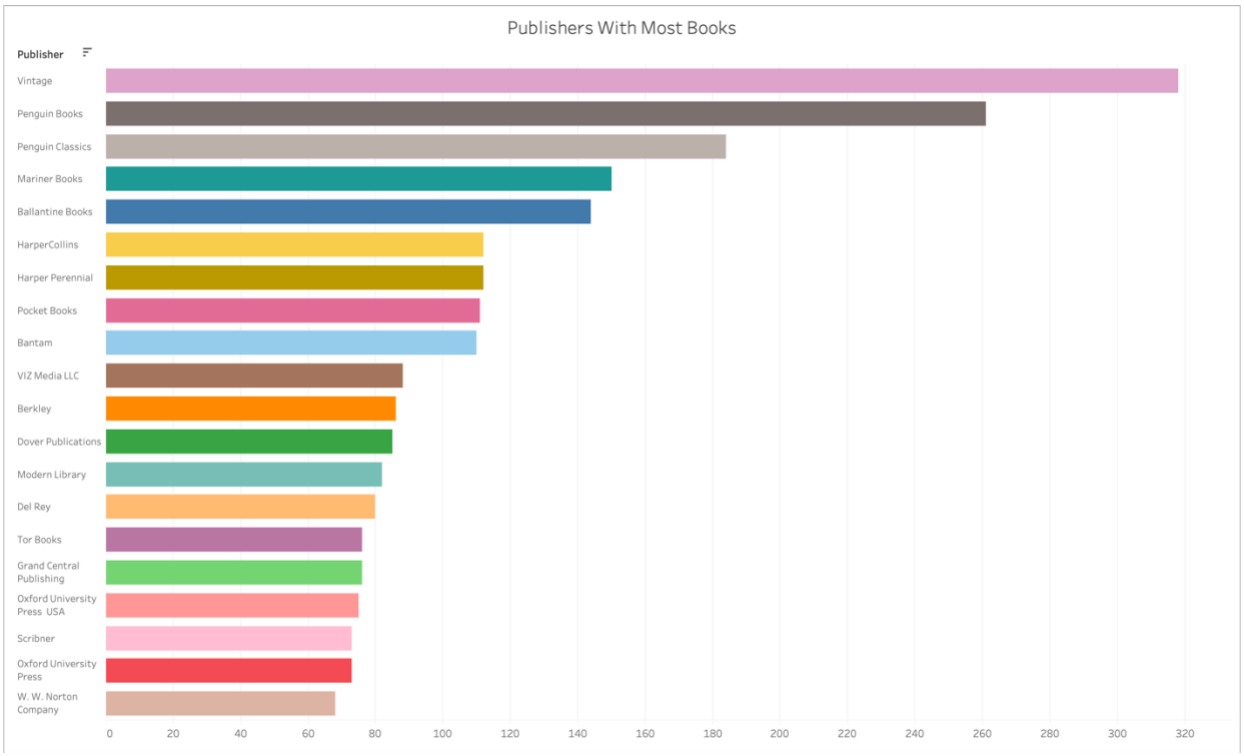
4. Distribution of Authors rated using Rating v/s Text reviews



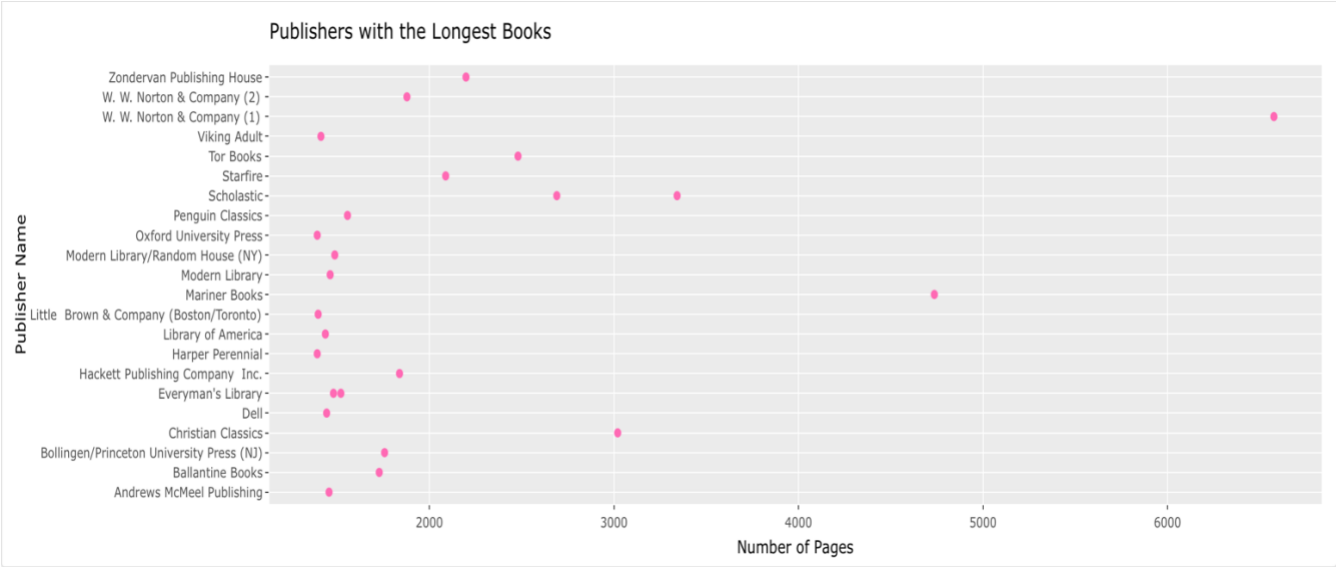
5. Most rated Books



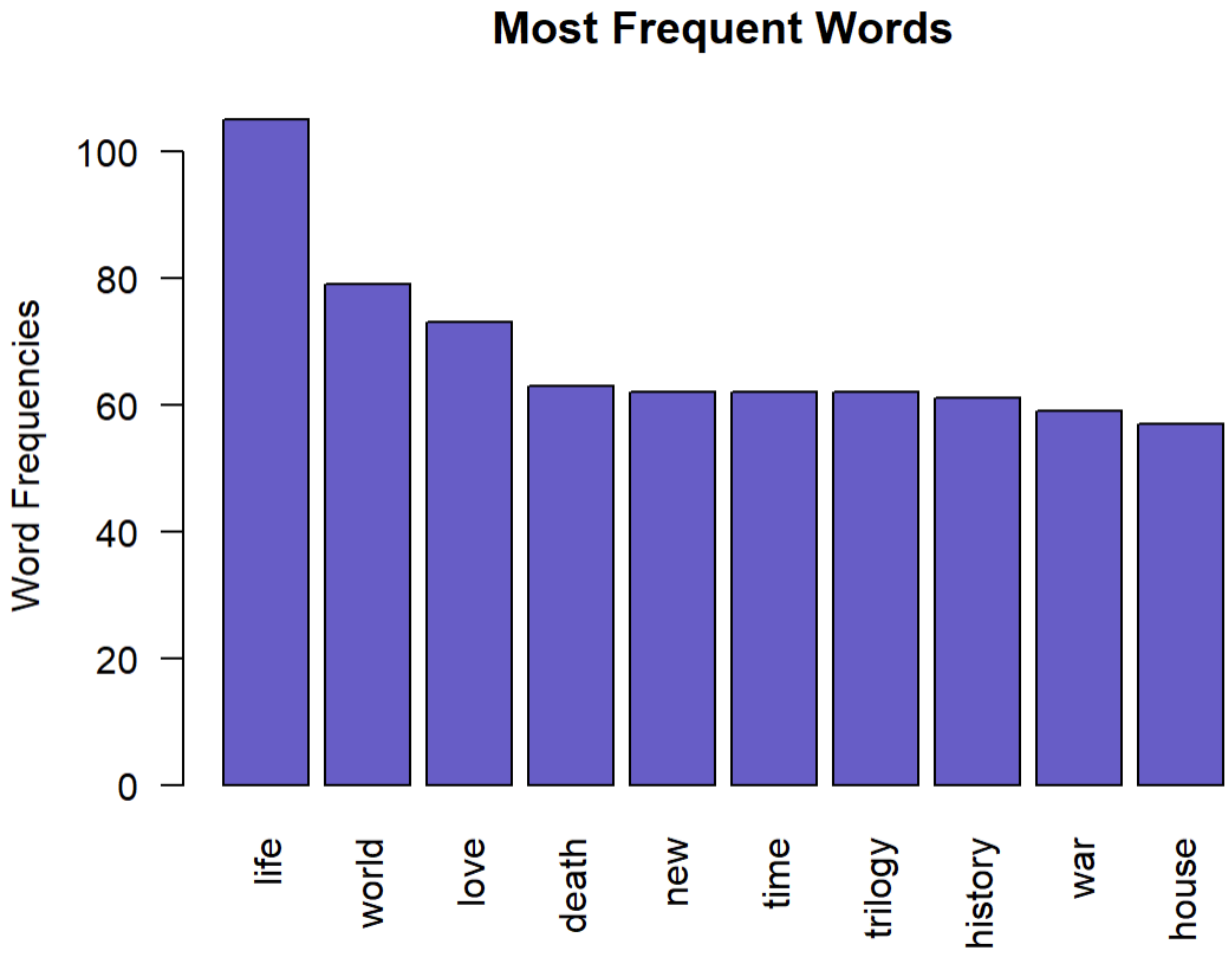
6. Publishers with most books published



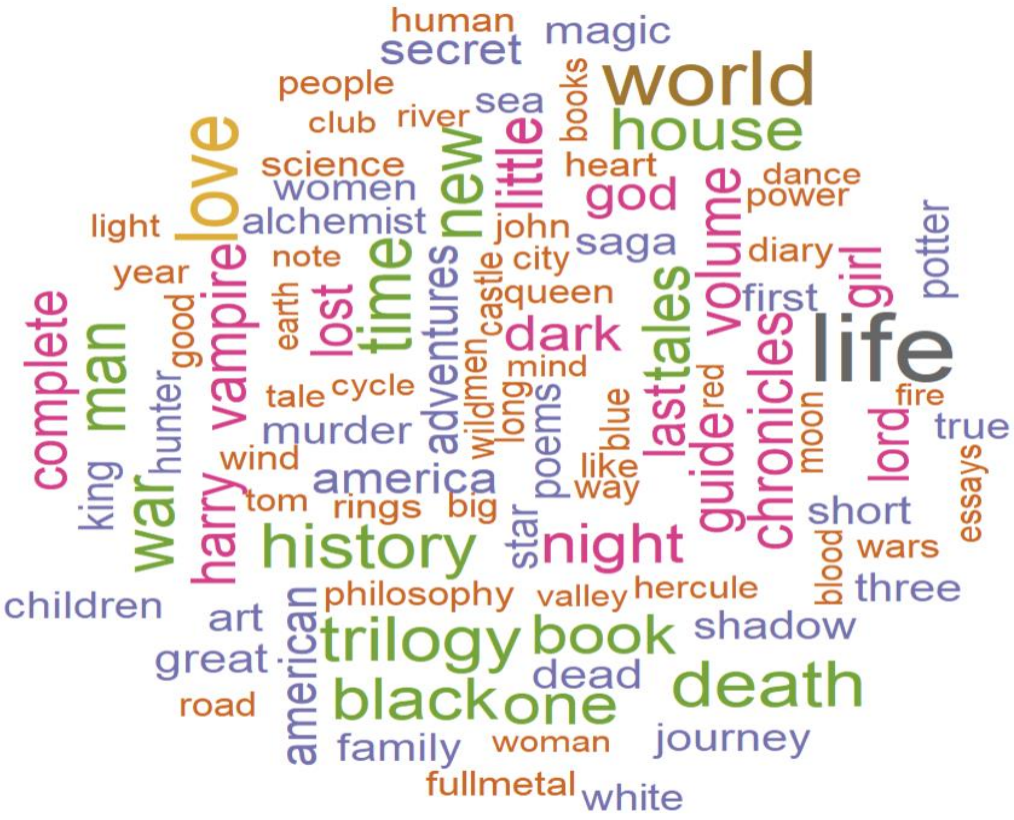
7. Publishers with longest books (with respect to number of pages)



8. Most frequently occurring words in the books

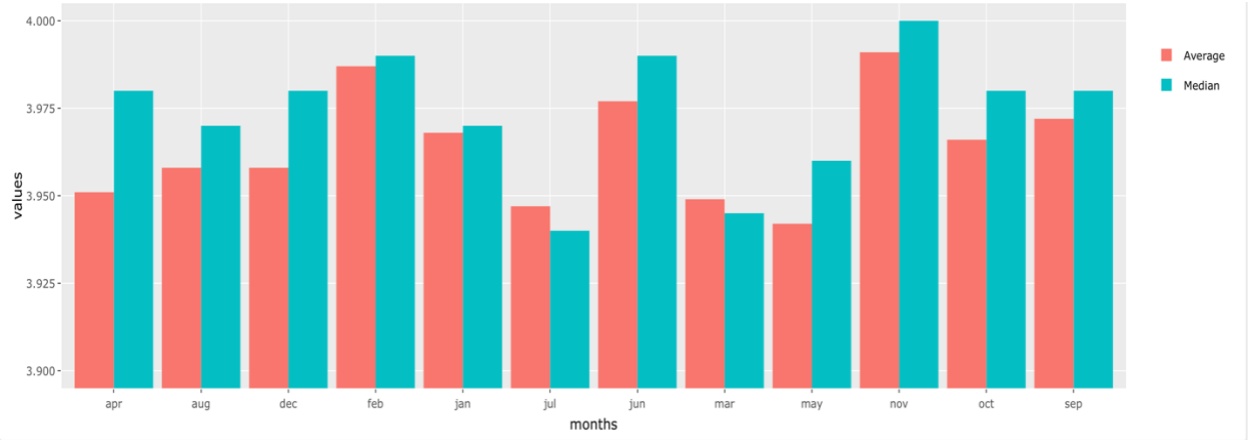


9. Word Cloud for the Books Description



10. Rating v/s Publication month

Observation: Publication month has minimal effect on Average Rating or Median Rating.



Conclusion

We collected three groups of datasets: meta-data of the books, user-book interactions (users' public shelves) and users' detailed book reviews. We merged these datasets together by matching book/user/review ids. The dataset contained information about Books, Shelves they are kept in and Book reviews. Also, the books were categorized based on Genre such as Children, Comic and Graphic, Fantasy and Paranormal, History and Biography, Mystery, Thriller and Crime, Poetry, Romance, Young Adult. We used Star schema model for the data warehouse, with the fact tables at the center and the dimension tables branching out of it. With the Fact tables containing measures regarding Ratings and its aggregates, the book counts, number of pages and shelf count in it. We have collected number of files containing data from the Goodreads website, which includes data regarding the authors, books, series, popular shelves, ratings, roles of authors, genre of book, etc. We load the data from these files into the staging area tables, which are all varchar datatype, to ensure ease of loading. We faced some Challenges while loading the dataset because of the vast genre of books. The major challenge was for the genre attributes that had to be combined into a single attribute called Genre for the DIM_Genre dimension table. This challenge was solved using LOOKUP which would check for the type of Genre in staging data and add it to Genre attribute in DIM_Genre. Tableau and R studio was used for data visualizations. The reason for using Tableau was that using the server connect, it can be easily connected to Microsoft SQL server and extract the data for visualization. Also, in the connect pain, one can view the table through join and preview the outcome.

Future Scope

- To improve the data model and selection of appropriate dimensions Data profiling could be done using tools like Talend.
- We can move flat file loading completely to cloud to implement all cloud-based infrastructure.