Sri Lanka Institute of Information Technology

Data warehousing and Business Intelligence

Assignment 1

Student Registration – IT20133054
Student Name – M.D.Ernst

Step 1: Data selection

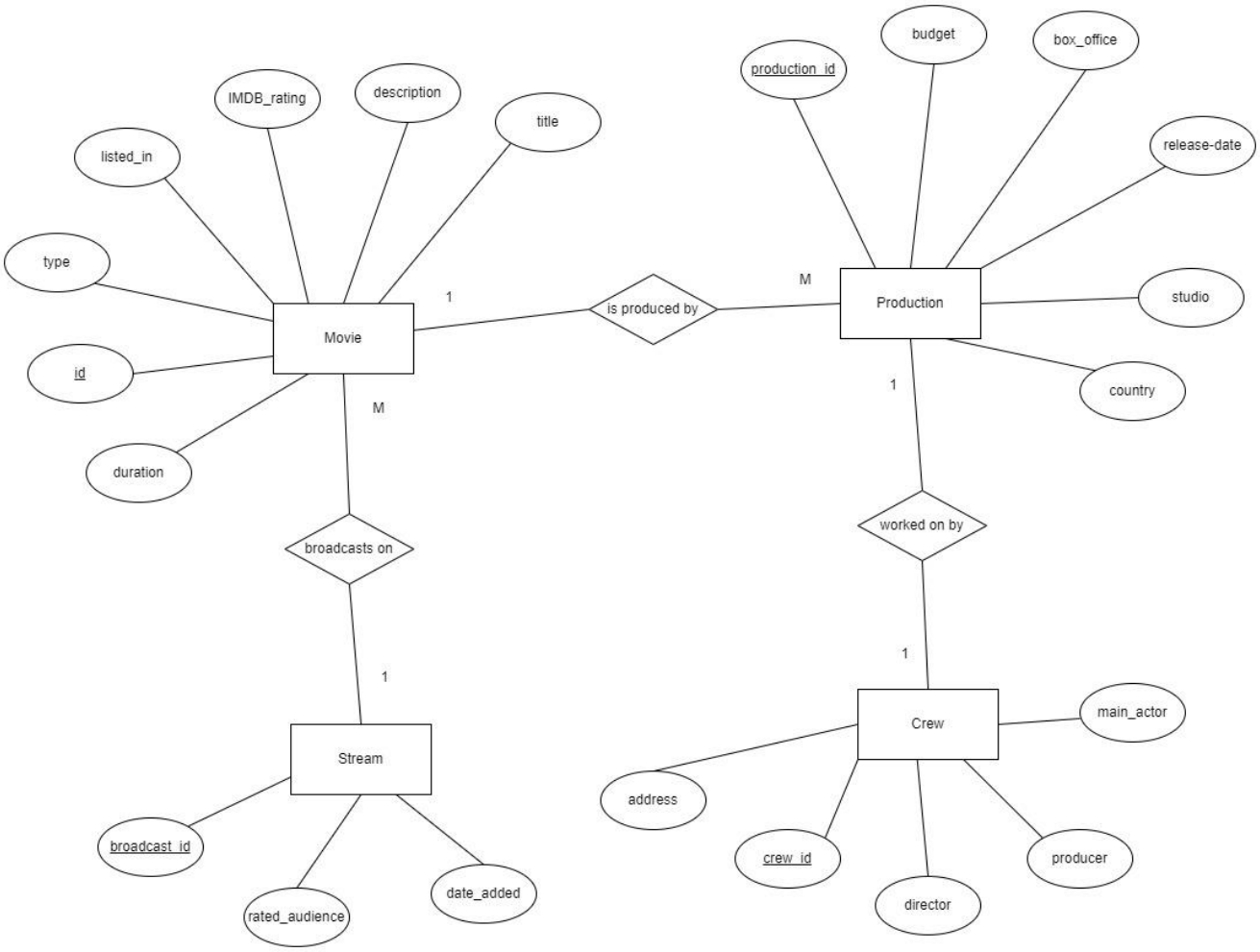
About this Dataset: Amazon Prime is another one of the most popular media and video streaming platforms. They have close to 10000 movies or tv shows available on their platform, as of mid-2021, they have over 200M Subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Amazon Prime, along with details such as - cast, directors, ratings, release year, duration, etc.*

This dataset contains Amazon prime show details,

- Movie/show details
- Production details
- Broadcast info
- Crew and cast info

Additional hypothetical data were also added to this database.

ER diagram



Step 2: Preparation of Data Sources

The whole of data was in ‘csv’ file type, and they were separated into the following data sources, Text, and csv. And they were used to create the following

1. **Text (.txt)**

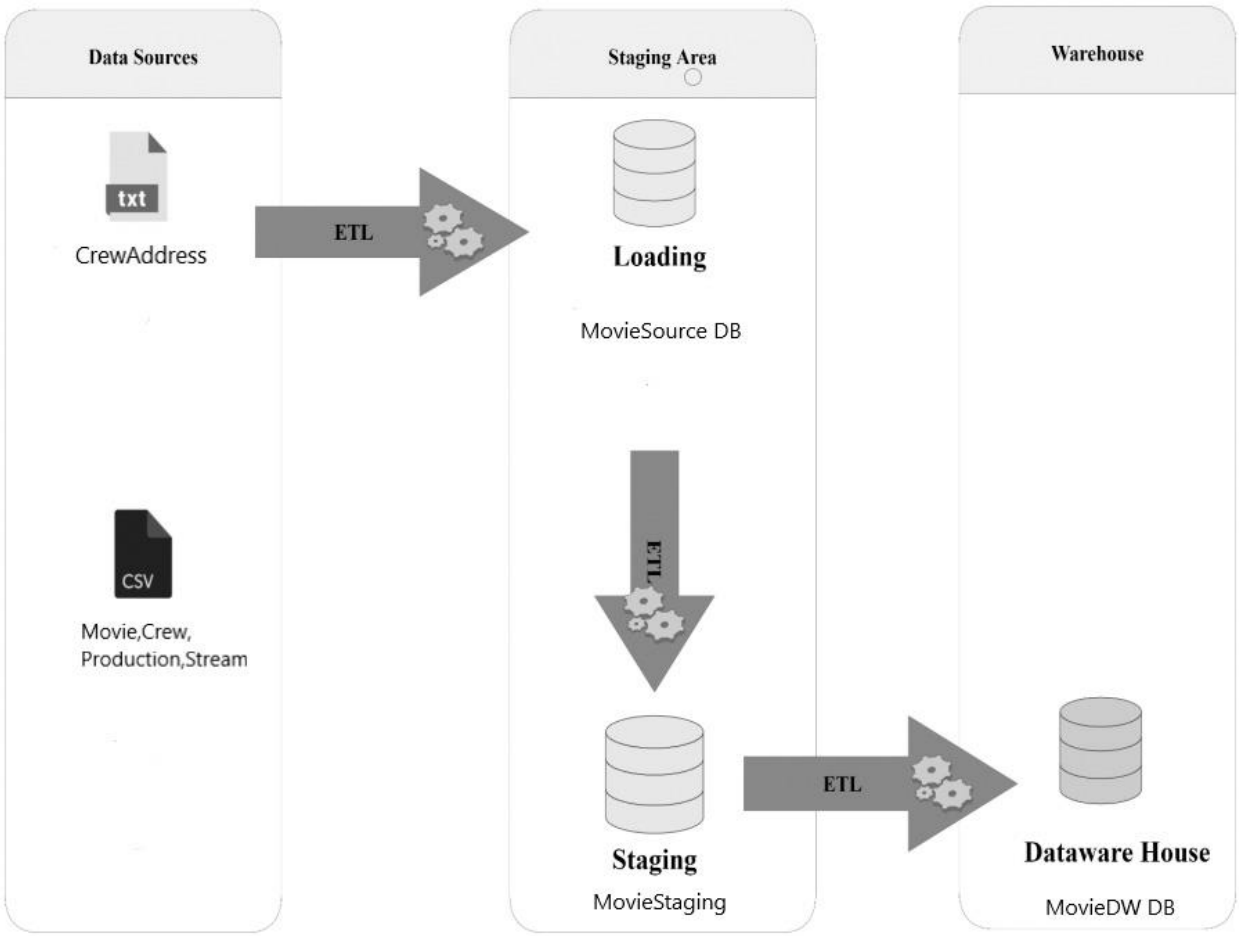
   CrewAddress.txt was used directly

2. **CSV (.csv)**

   Movie.csv
   Production.csv
   Stream.csv
   Crew.csv

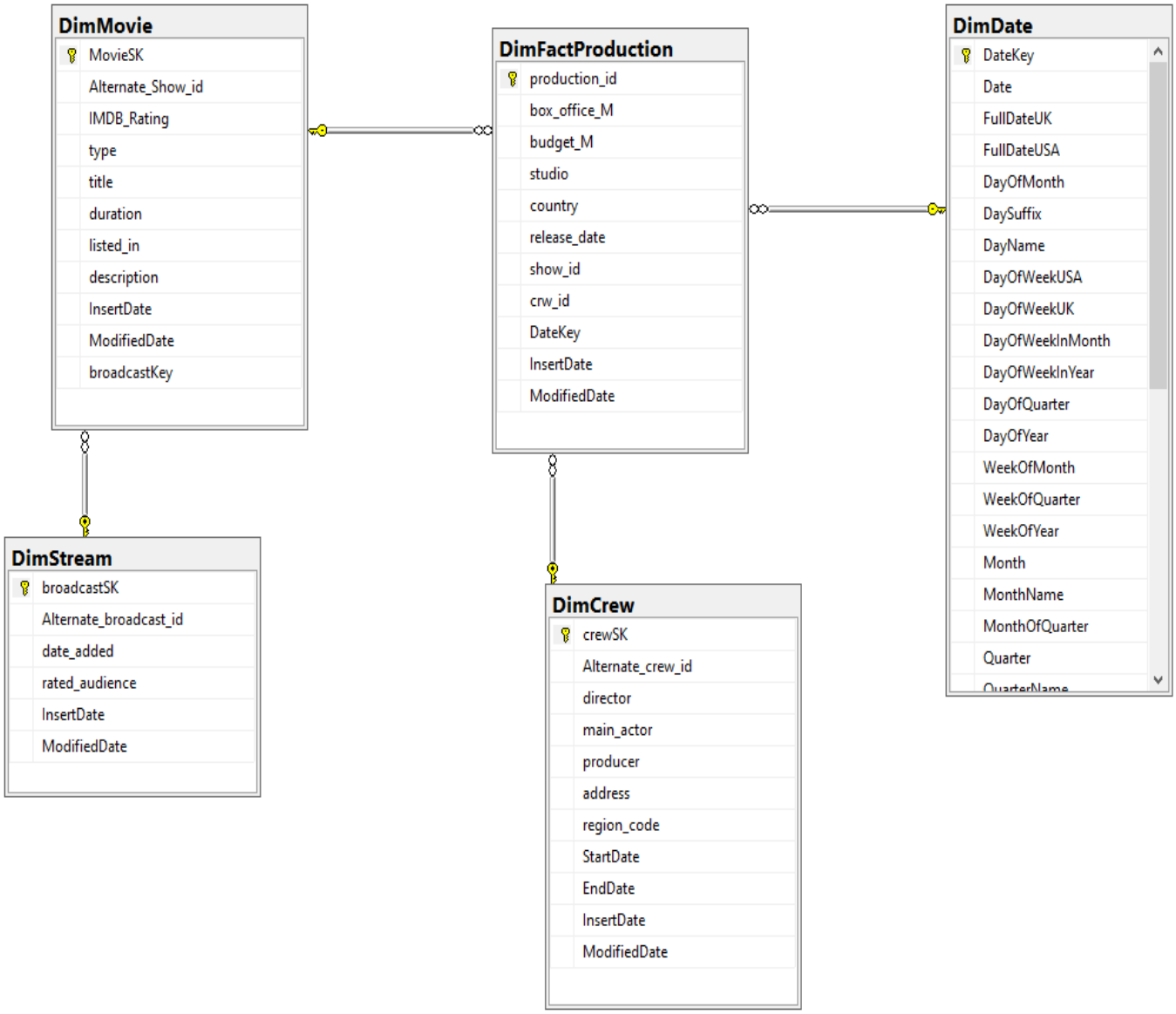
Data Source Type	Source Name	Column Name	Data Type
csv	Crew.csv	[crew_id]	Varchar(50)
		[director]	Varchar(50)
		[main_actor]	Varchar(50)
		[producer]	Varchar(50)
	Movie.csv	[Show_id]	nvarchar(50)
		[type]	nvarchar(50)
		[title]	nvarchar(50)
		[IMDB_rating]	float
		[duration]	nvarchar(50)
		[listed_in]	nvarchar(50)
		[description]	nvarchar(50)
		[broadcast_id]	nvarchar(50)
	Production.csv	[production_id]	nvarchar(50)
		[box_office_M]	money
		[budget_M]	money
		[studio]	nvarchar(50)
		[country]	nvarchar(50)
		[release_date]	date
		[show_id]	nvarchar(50)
		[crw_id]	nvarchar(50)
	Stream.csv	[broadcast_id]	nvarchar(50)
		[date_added]	date
		[rated_audience]	nvarchar(50)
text	Crewaddress.txt	[crew_id]	Varchar(50)
		[address]	Varchar(50)
		[region_code]	Varchar(50)

Step 3: Solution architecture



Step 4: Data Warehouse Design & Development

Following figure will show how the fact table and dimension tables was combined in a rational manner



Schema Type

For this scenario, snowflake schema type was used.

Dimension Types

- Hierarchical Dimension
- Slowly Changing Dimension
- Fact Table

Assumptions

Crew Address dimension was used as a slowly changing dimension.

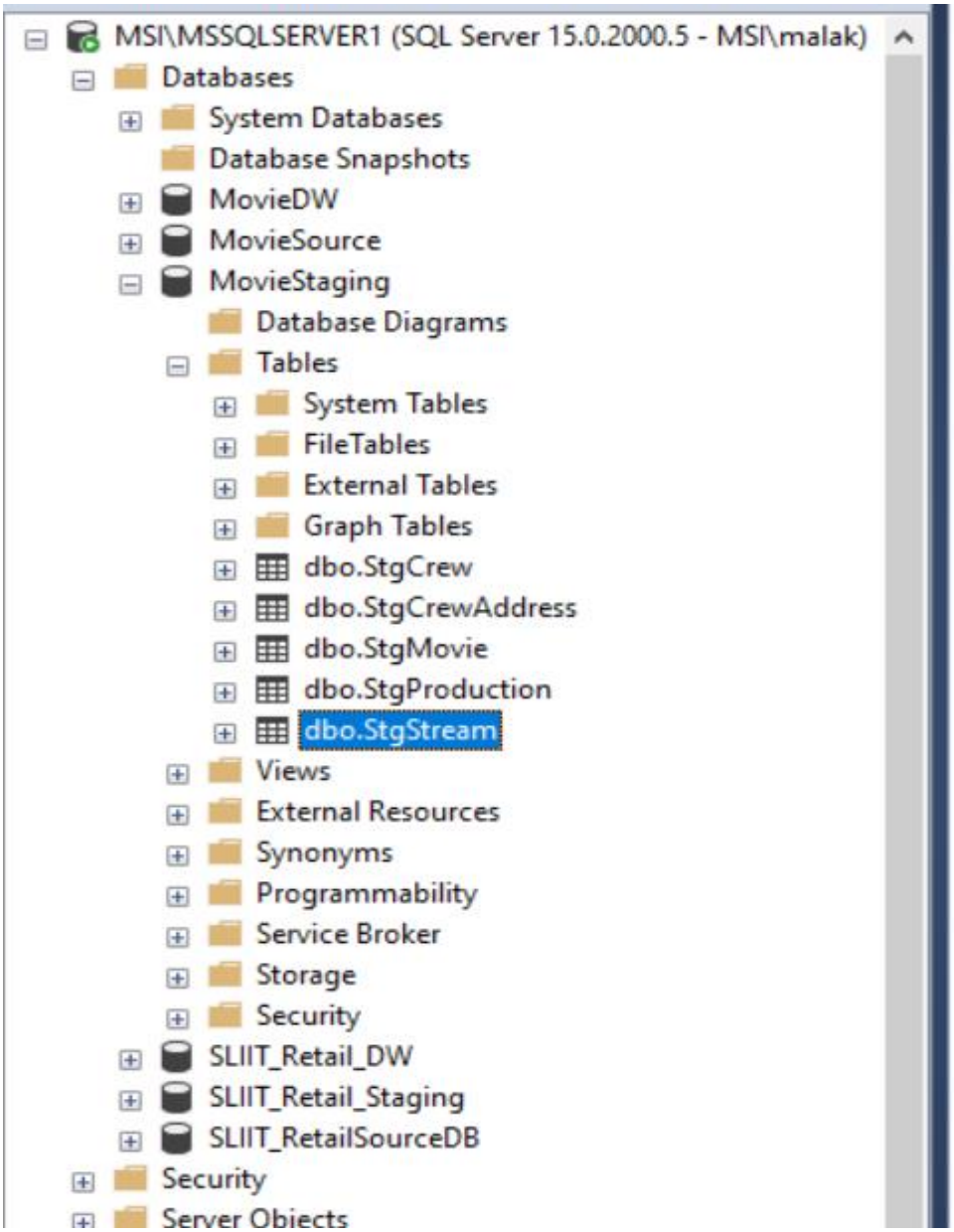
Step 5: ETL Development

1. Extract

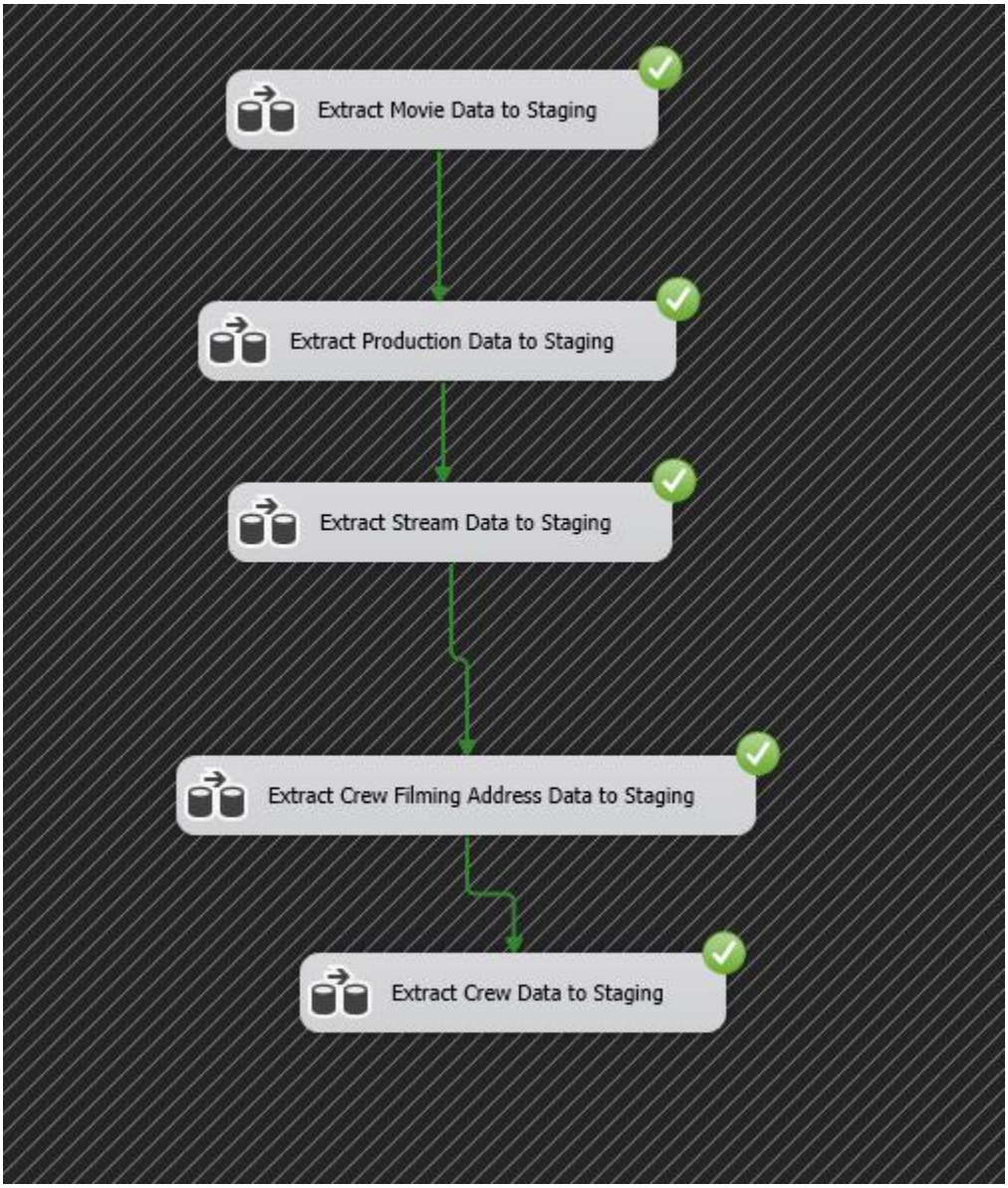
In this step, All the data sources were imported to the staging tables by using the relevant Data
connection.
Flat file connection was used for text files and csv files. All those tables were imported to the MovieStaging DB which contains the below tables.

- [dbo].[StgCrew]
- [dbo].[StgCrewAddress]
- [dbo].[StgMovie]
- [dbo].[StgProduction]
- [dbo].[StgStream]

Snapshot of SSMS Staging Database

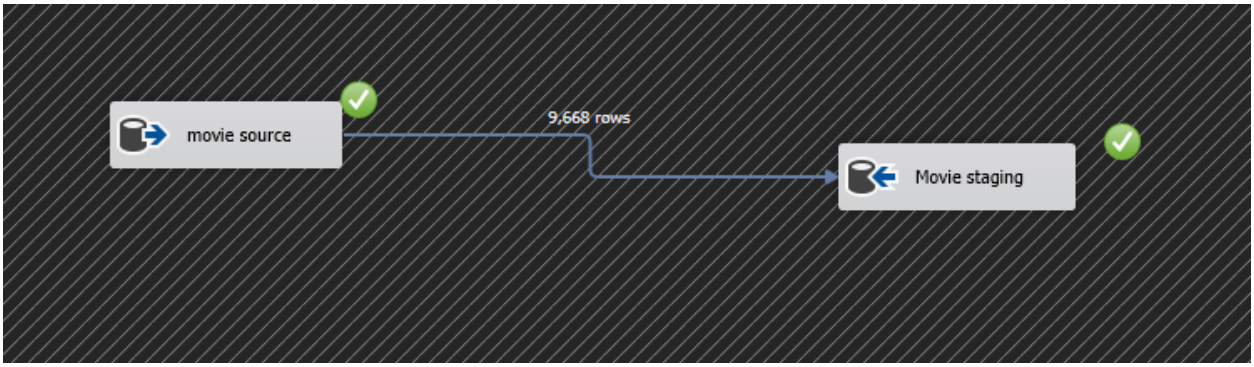


Snapshot of Visual Studio Control Flow of Extract

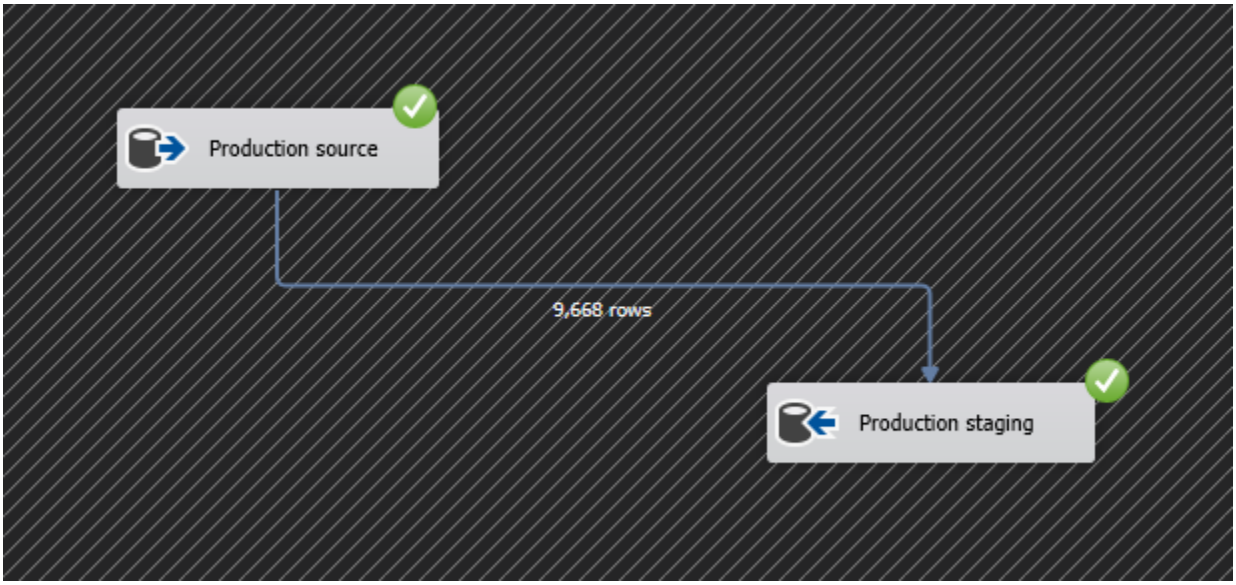


Snapshots of several data types of Data Flows

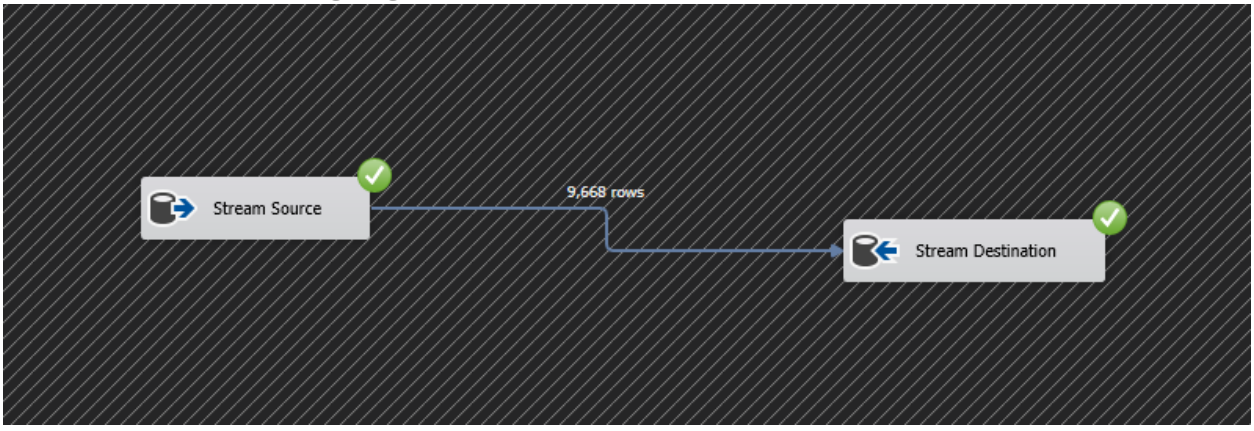
Movie data to staging



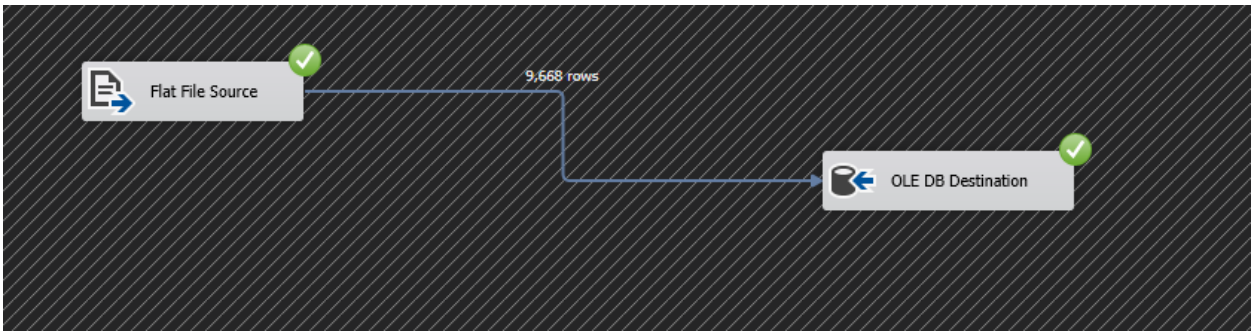
production data to staging



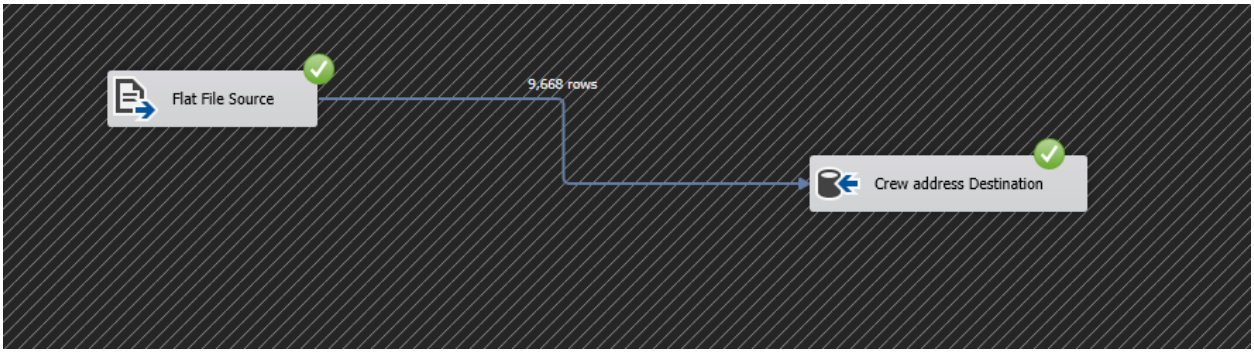
stream data to staging



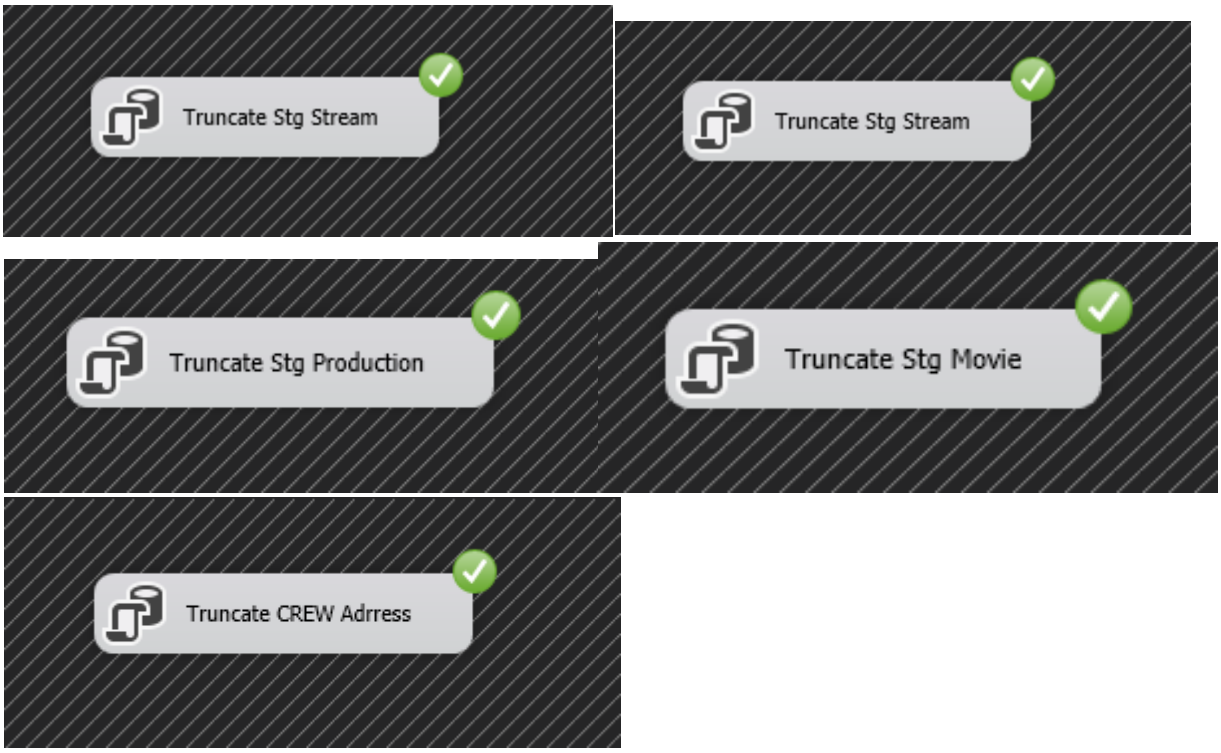
crew data to staging



crew address data to staging



Event Handling (Truncate Staging Data)



3.Transform and Load

In this step, both the 'Transform' and 'Load' are done. Firstly, The Dimension tables in the Datawarehouse DB data were created. Then, using the relevant components, data from the staging tables was loaded into the warehouse tables, MovieDW, which contains the below tables,

1. [dbo].[DimCrew]
2. [dbo].[DimFactProduction]
3. [dbo].[DimDate]
4. [dbo].[DimStream]
5. [dbo].[DimMovie]

Used Transformation Tasks

1. Lookups
   - Date lookup
   - Crew lookup
   - Movie lookup

2. Derived Columns

Replace NULL director values in DimCrew Table

3. Union

Union is used in the Extract step to combine and get all the data from both crew and

crew_Address data csv files.

Update Functions used

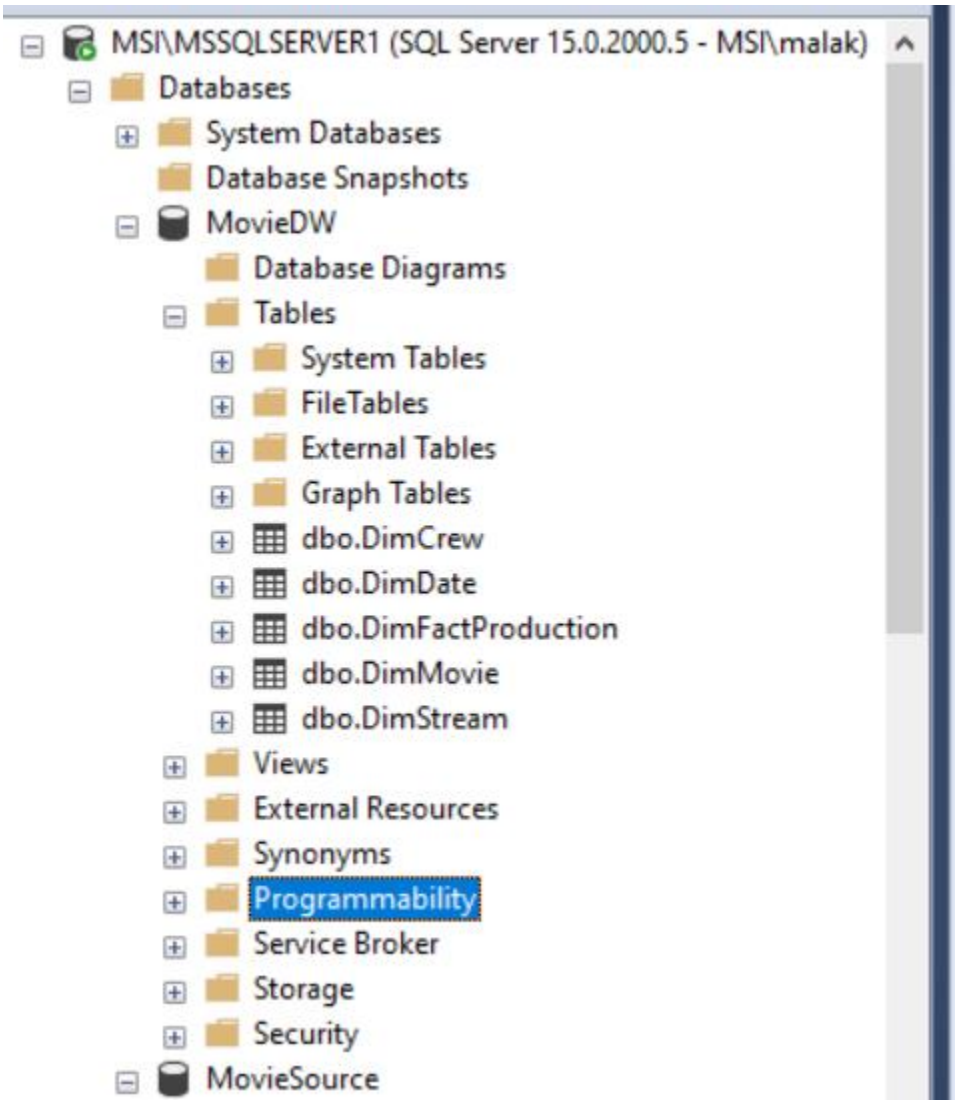
```
USE [MovieDW]
GO
/****** Object:  StoredProcedure [dbo].[UpdateDimCrew]    Script Date: 5/21/2022 3:14:47 AM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[UpdateDimCrew]
@crew_id varchar(50),
@director varchar(50),
@main_actor varchar(50),
@producer varchar(50)


AS
BEGIN
if not exists (select CrewSK
from dbo.DimCrew
where Alternate_crew_id = @crew_id)
BEGIN
insert into dbo.DimCrew (Alternate_crew_id, director, main_actor, producer, InsertDate, ModifiedDate)
values (@crew_id, @director, @main_actor, @producer, GETDATE(), GETDATE())
END;
if exists (select CrewSK
from dbo.DimCrew
where Alternate_crew_id = @crew_id)
BEGIN
update dbo.DimCrew
set director = @director, main_actor = @main_actor, producer = @producer, ModifiedDate = GETDATE()
where Alternate_crew_id = @crew_id
END;
END;
```

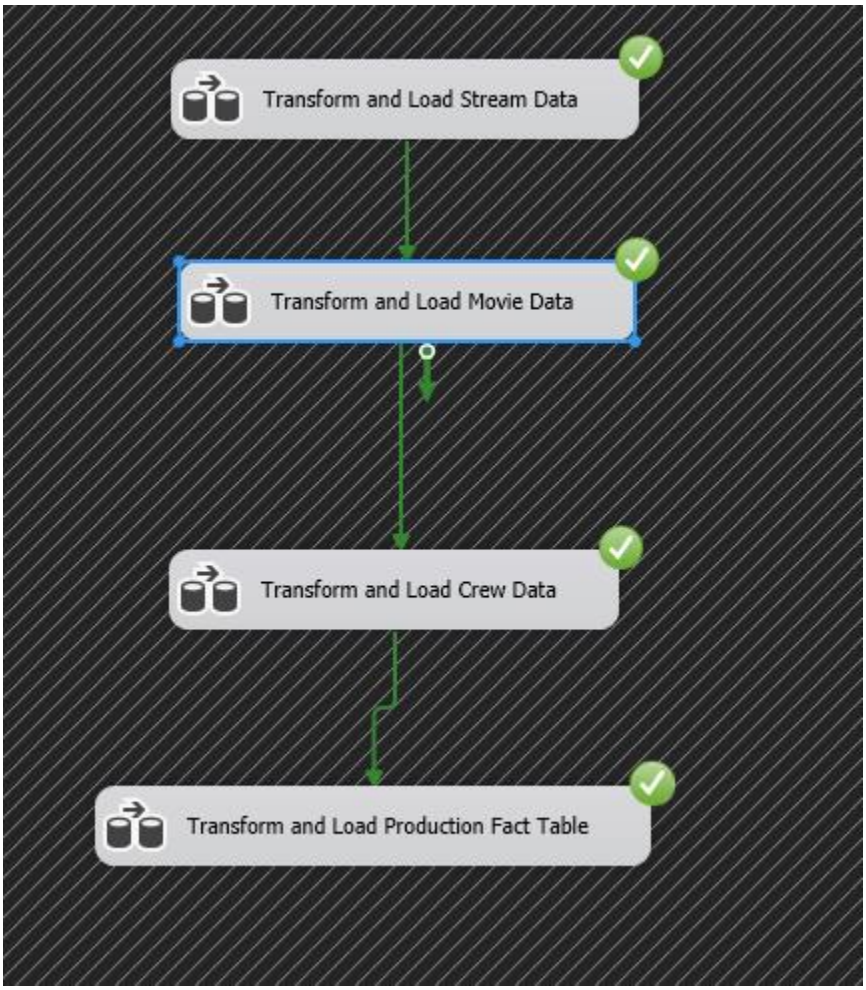
```sql
USE [MovieDW]
GO
/****** Object:  StoredProcedure [dbo].[UpdateDimMovie]    Script Date: 5/17/2022 10:44:45 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[UpdateDimMovie]
@Show_id nvarchar(50),
@IMDB_Rating float,
@type nvarchar(50),
@title nvarchar(50),
@duration nvarchar(50),
@listed_in nvarchar(50),
@description nvarchar(50),
@broadcastKey int
AS
BEGIN
if not exists (select MovieSK
from dbo.DimMovie
where Alternate_Show_id = @Show_id)
BEGIN
insert into dbo.DimMovie (Alternate_Show_id, broadcastKey, IMDB_Rating, type, title, duration, listed_in, description,
values (@Show_id, @broadcastKey, @IMDB_Rating, @type, @title, @duration, @listed_in, @description, GETDATE(), GETDATE(
END;
if exists (select MovieSK
from dbo.DimMovie
where Alternate_Show_id = @Show_id)
BEGIN
update dbo.DimMovie
set broadcastKey = @broadcastKey, IMDB_Rating = @IMDB_Rating, type = @type, title = @title, duration = @duration, list
where Alternate_Show_id = @Show_id
END;
END;
```

```sql
USE [MovieDW]
GO
/****** Object:  StoredProcedure [dbo].[UpdateDimStream]    Script Date: 5/17/2022 10:54:52 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[UpdateDimStream]
@broadcast_id nvarchar(50),
@date_added date,
@rated_audience nvarchar(50)
AS BEGIN
if not exists (select broadcastSK
from dbo.DimStream
where Alternate_broadcast_id = @broadcast_id)
BEGIN
insert into dbo.DimStream (Alternate_broadcast_id, date_added, rated_audience, InsertDate, ModifiedDate)
values (@broadcast_id, @date_added, @rated_audience, GETDATE(), GETDATE())
END;
if exists (select broadcastSK
from dbo.DimStream
where Alternate_broadcast_id = @broadcast_id)
BEGIN
update dbo.DimStream
set date_added = @date_added, rated_audience = @rated_audience, ModifiedDate = GETDATE()
where Alternate_broadcast_id = @broadcast_id
END;
END
```

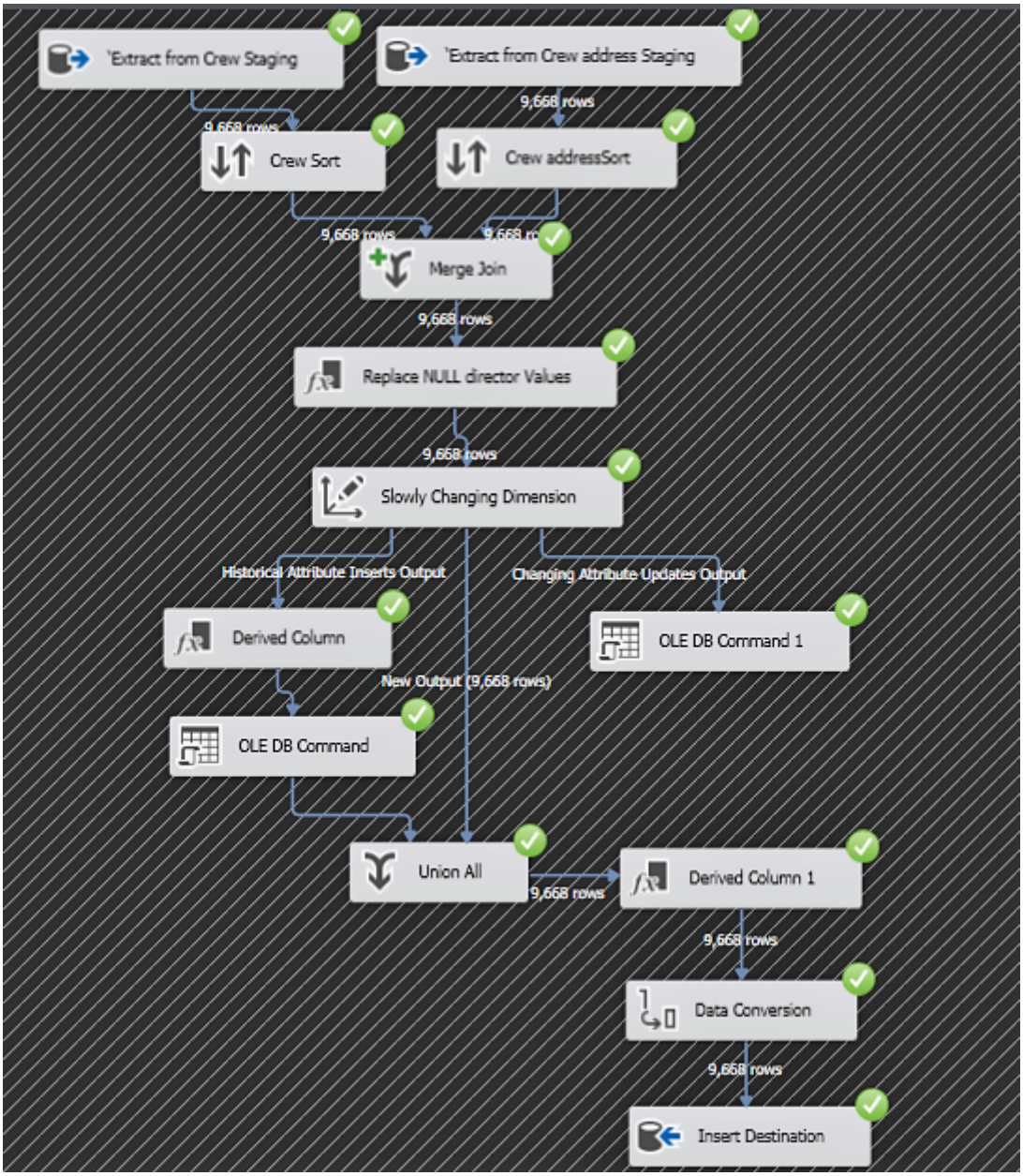
Snapshot of SQL server Data warehouse Database



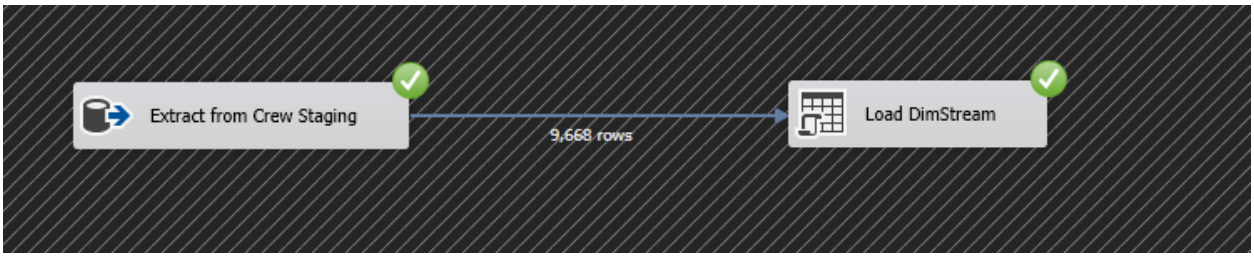
Snapshots of Visual Studio Control Flow of Extraction



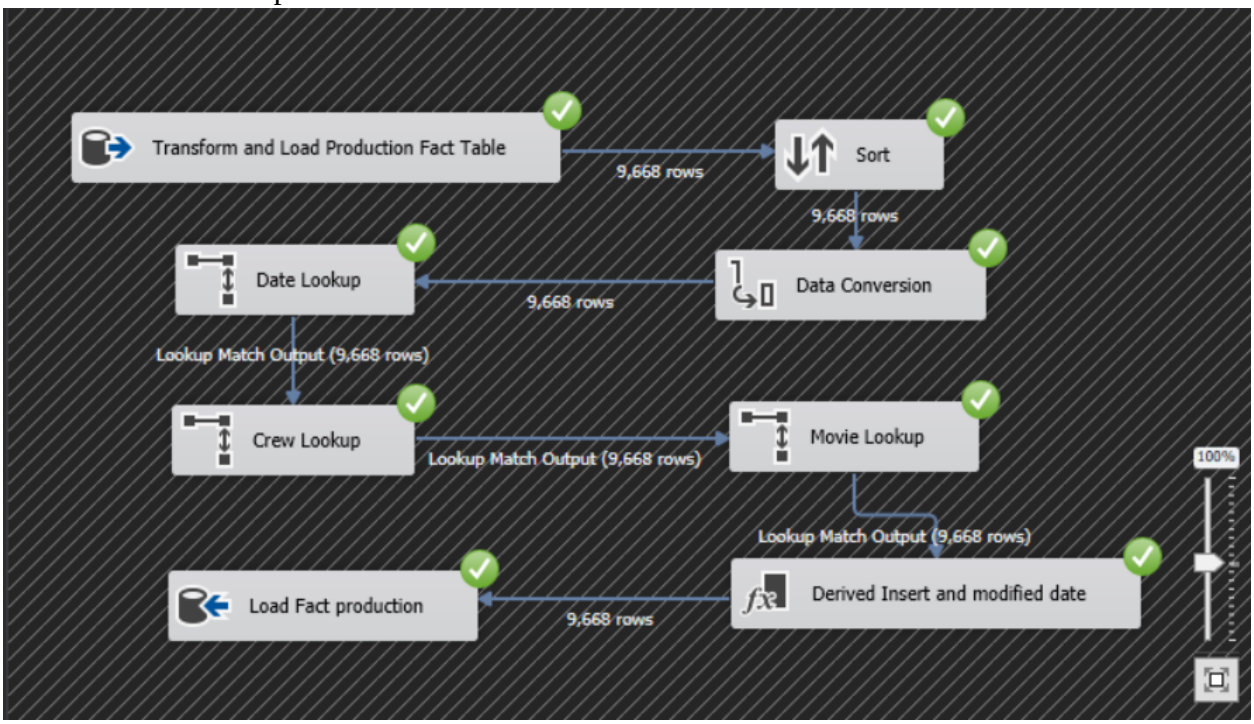
Transform and Load crew



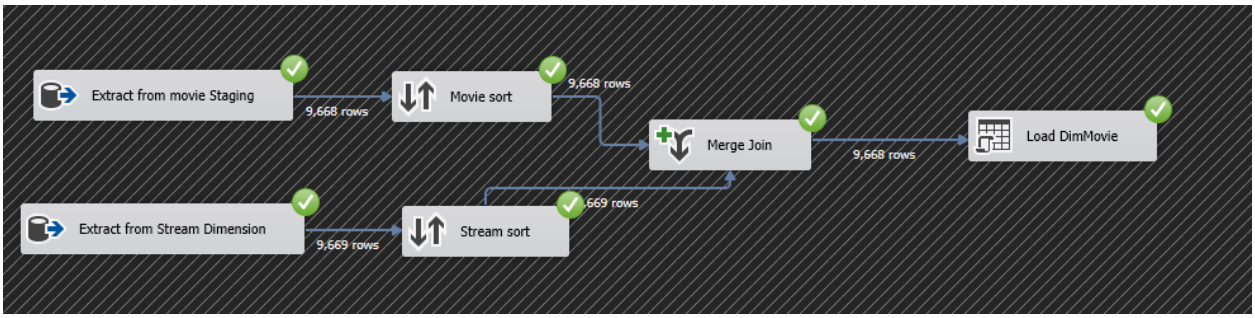
Transform and Load Stream



Transform and Load production

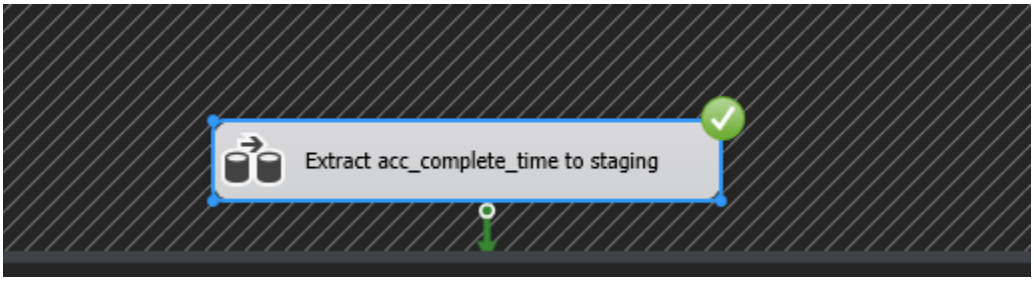


Transform and Load movie

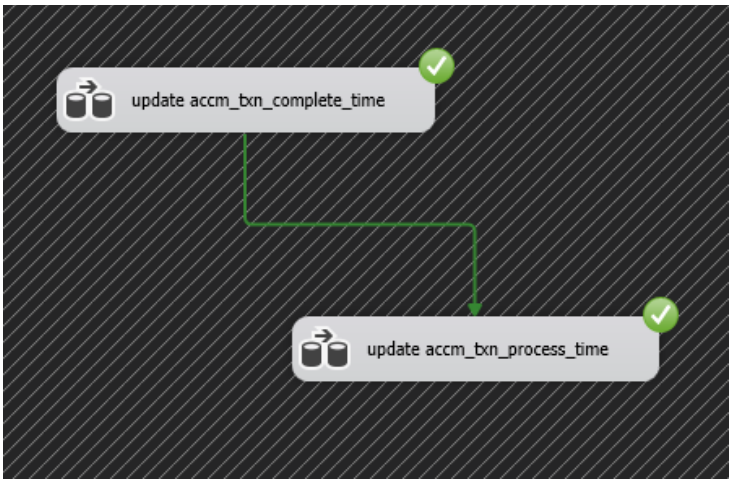


Step 6: ETL Development – Accumulating Fact table

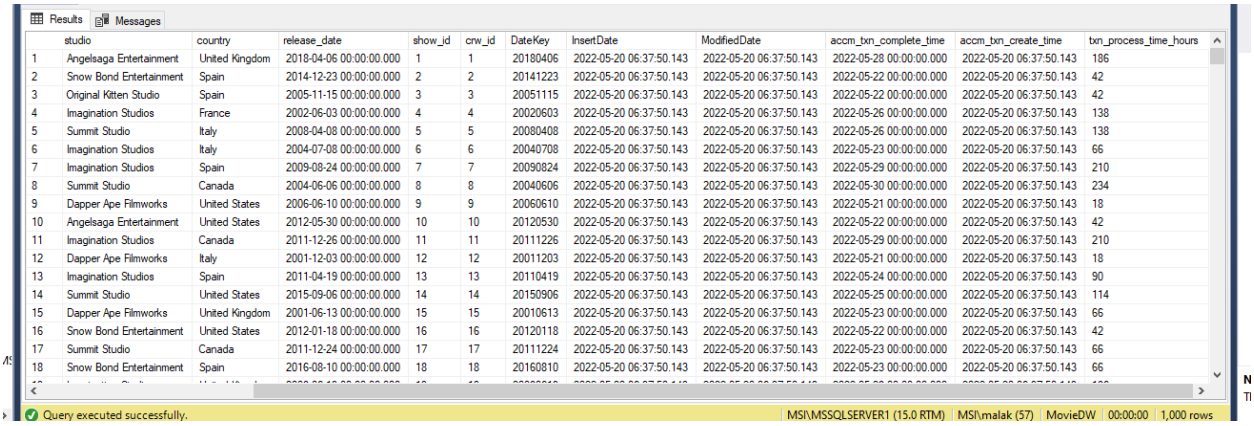
Complete time



Process time



After adding and processing



Results | Messages

	studio	country	release_date	show_id	crw_id	DateKey	InsertDate	ModifiedDate	accm_txn_complete_time	accm_txn_create_time	txn_process_time_hours
1	Angelsaga Entertainment	United Kingdom	2018-04-06 00:00:00.000	1	1	20180406	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-28 00:00:00.000	2022-05-20 06:37:50.143	186
2	Snow Bond Entertainment	Spain	2014-12-23 00:00:00.000	2	2	20141223	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-22 00:00:00.000	2022-05-20 06:37:50.143	42
3	Original Kitten Studio	Spain	2005-11-15 00:00:00.000	3	3	20051115	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-22 00:00:00.000	2022-05-20 06:37:50.143	42
4	Imagination Studios	France	2002-06-03 00:00:00.000	4	4	20020603	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-26 00:00:00.000	2022-05-20 06:37:50.143	138
5	Summit Studio	Italy	2008-04-08 00:00:00.000	5	5	20080408	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-26 00:00:00.000	2022-05-20 06:37:50.143	138
6	Imagination Studios	Italy	2004-07-08 00:00:00.000	6	6	20040708	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-23 00:00:00.000	2022-05-20 06:37:50.143	66
7	Imagination Studios	Spain	2009-08-24 00:00:00.000	7	7	20090824	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-29 00:00:00.000	2022-05-20 06:37:50.143	210
8	Summit Studio	Canada	2004-06-06 00:00:00.000	8	8	20040606	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-30 00:00:00.000	2022-05-20 06:37:50.143	234
9	Dapper Ape Filmworks	United States	2006-06-10 00:00:00.000	9	9	20060610	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-21 00:00:00.000	2022-05-20 06:37:50.143	18
10	Angelsaga Entertainment	United States	2012-05-30 00:00:00.000	10	10	20120530	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-22 00:00:00.000	2022-05-20 06:37:50.143	42
11	Imagination Studios	Canada	2011-12-26 00:00:00.000	11	11	20111226	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-29 00:00:00.000	2022-05-20 06:37:50.143	210
12	Dapper Ape Filmworks	Italy	2001-12-03 00:00:00.000	12	12	20011203	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-21 00:00:00.000	2022-05-20 06:37:50.143	18
13	Imagination Studios	Spain	2011-04-19 00:00:00.000	13	13	20110419	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-24 00:00:00.000	2022-05-20 06:37:50.143	90
14	Summit Studio	United States	2015-09-06 00:00:00.000	14	14	20150906	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-25 00:00:00.000	2022-05-20 06:37:50.143	114
15	Dapper Ape Filmworks	United Kingdom	2001-06-13 00:00:00.000	15	15	20010613	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-23 00:00:00.000	2022-05-20 06:37:50.143	66
16	Snow Bond Entertainment	United States	2012-01-18 00:00:00.000	16	16	20120118	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-22 00:00:00.000	2022-05-20 06:37:50.143	42
17	Summit Studio	Canada	2011-12-24 00:00:00.000	17	17	20111224	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-23 00:00:00.000	2022-05-20 06:37:50.143	66
18	Snow Bond Entertainment	Spain	2016-08-10 00:00:00.000	18	18	20160810	2022-05-20 06:37:50.143	2022-05-20 06:37:50.143	2022-05-23 00:00:00.000	2022-05-20 06:37:50.143	66

Query executed successfully. | MSI\MSSQLSERVER1 (15.0 RTM) | MSI\malak (57) | MovieDW | 00:00:00 | 1,000 rows