

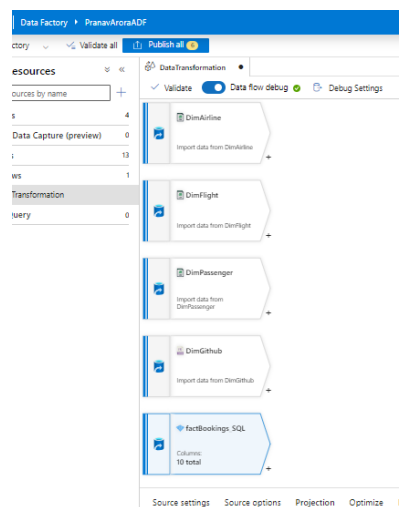
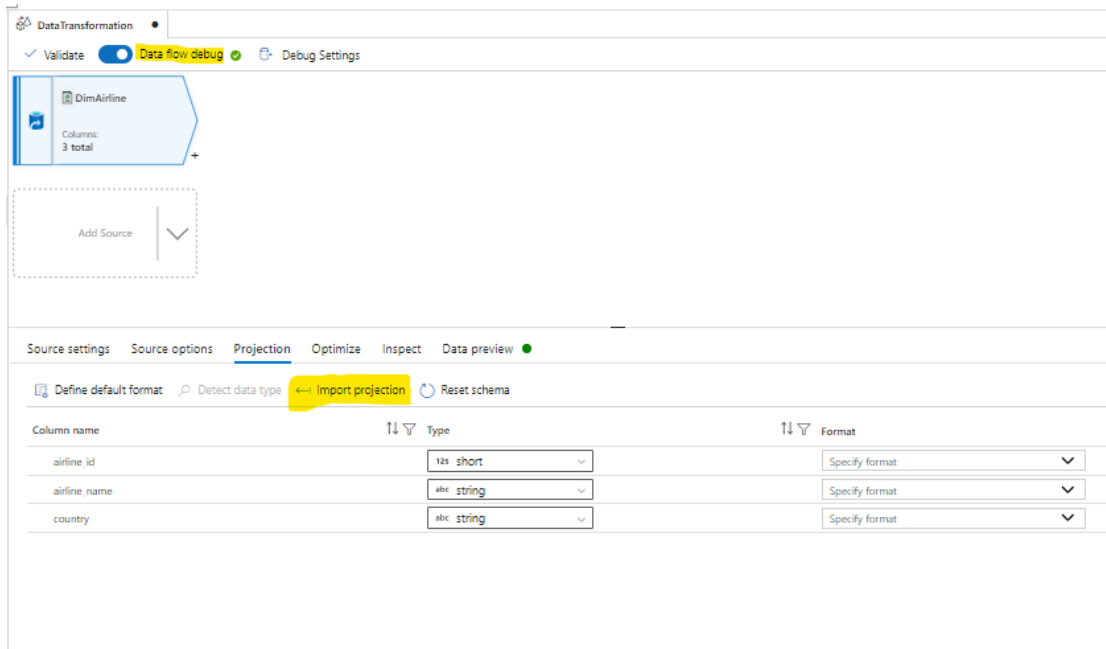
Project Documentation – Data Flow Transformations (Silver Layer Load)

Steps Followed:

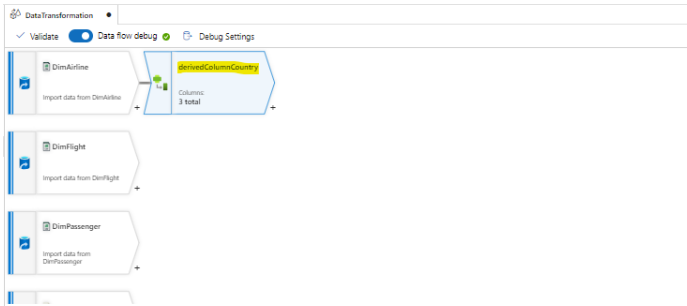
1. Now we have our data in bronze layer from all the Sources so we will populate data in our silver and gold layer, and we will perform some transformations on the data. So, we use data flow for this.

We will add our bronze data from container in the sources using existing/by creating new datasets to read the data files.

Also, turn on Data flow debug to see your data schema, perform transformations, etc. Also, whenever we perform any transformations on our source to see the output, we must Import Projections on our source to get the schema and data.



2. Now we can add our transformations using Dataflow, for example:



Derived column's settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Columns *

Column	Expression
<input type="checkbox"/> country	<input type="text" value="upper(country)"/> <input type="button" value="Alt"/> <input type="button" value="+"/>



Select settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

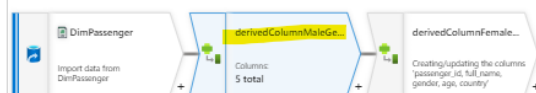
Options

☒ Skip duplicate input columns

☒ Skip duplicate output columns

Input columns * ☐ Auto mapping [Reset](#) [Add mapping](#) 4 mappings

DimFlight's column	Name as
<input type="checkbox"/> flight_id	<input type="text" value="flight_id"/>
<input type="checkbox"/> flight_number	<input type="text" value="flight_number"/>
<input type="checkbox"/> departure_timestamp	<input type="text" value="departure_timestamp"/>
<input type="checkbox"/> arrival_timestamp	<input type="text" value="arrival_timestamp"/>



Derived column's settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Columns *

Column	Expression
<input type="checkbox"/> gender	<input type="text" value="regexReplace(gender, 'M', 'Male')"/> <input type="button" value="Alt"/> <input type="button" value="+"/>

Diagram showing a data pipeline with three components: DimPassenger, derivedColumnMaleGender, and derivedColumnFemaleGender. The derivedColumnFemaleGender component is highlighted in yellow.

Derived column's settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Columns * [+ Add](#) [Clone](#) [Delete](#) [Open expression builder](#)

Column	Expression
<input type="checkbox"/> gender	<input type="text" value="regexReplace(gender, 'F', 'Female')"/> abc + 🗑

Diagram showing a data pipeline with four components: DimPassenger, derivedColumnMaleGender, derivedColumnFemaleGender, and FilterGreater25. The FilterGreater25 component is highlighted in yellow.

Filter settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Filter on *

Diagram showing a data pipeline with five components: DimPassenger, derivedColumnMaleGender, derivedColumnFemaleGender, FilterGreater25, and derivedColumnFirstName. The derivedColumnFirstName component is highlighted in yellow.

Derived column's settings

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Columns * [+ Add](#) [Clone](#) [Delete](#) [Open expression builder](#)

Column	Expression
<input type="checkbox"/> first name	<input type="text" value="split(full_name, ' ')[0]"/> abc + 🗑

factBookings_SQL

castCost

Columns: 10 total

Cast settings | Optimize | Inspect | Data preview

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Columns *

Column name	Type	Format
ticket_cost	us_integer	000

Assert type check ☒

- Now we want to write our data to the silver layer, and we want to maintain Upsert logic every time we load data to silver using our Data flow. To upsert we need to add an alter activity before our sink activity.

DataTransformation

Validate Data flow debug Debug Settings

DimAirline

Import data from DimAirline

Reference: 1

Columns: 3 total

alterRowForAirline

Columns: 3 total

DimFlight

Import data from DimFlight

SelectToRenameColumns

Renaming DimFlight to SelectToRenameColumns with columns 'flight_id, flight_number,

Alter row settings | Optimize | Inspect | Data preview

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Alter row conditions *

Under the Alter row condition, we chose Upsert, and we need to give a condition usually basis which our upsert will happen (ex: Upsert should happen for all records where Airline <> 'Air India', but usually we do all filtering before the upsert, so we will put a pseudo condition like $1>0$, which will always be true and hence upsert will happen on all of our data.

- Now we will create a Sink Activity, and in this Sink for Sink type we will Choose Inline (Since Inline support delta), and Inline Dataset type will be "Delta" as our choice. Also in our Sink Settings, we will add the PK column used for Upsert and select Update Method as "Allow Upsert"

DataTransformation

✓ Validate Data flow debug Debug Settings

Sink Settings Errors Mapping Optimize Inspect Data preview

Output stream name * [Learn more](#)

Description [Reset](#)

Incoming stream *

Sink type * ☐ Dataset ☒ **Inline** ☐ Cache

Inline dataset type *

Linked service * [Test connection](#) [Edit](#) [New](#)

Options ☒ Allow schema drift ☐ Validate schema

Sink Settings Errors Mapping Optimize Inspect Data preview

Folder path * / [Browse](#)

Compression type

Vacuum

Table action ☒ None ☐ Overwrite ☐ Truncate

Update method ☐ Allow insert ☐ Allow delete ☒ **Allow upsert** ☐ Allow update

Key columns * ☒ List of columns ☐ Custom expression

[Add dynamic content \(Alt+Shift+D\)](#)

- Once our data flow is ready, we go to Pipelines and create a new Pipeline. In this pipeline we will add a dataflow activity and then configure our dataflow and its ready for Silver load.

DataTransformation

SilverDataFlowLoad

Validate

Debug

Add trigger

Data flow debug

data

Move and transform

Data flow

Azure Data Explorer

Azure Data Explorer C...

Data Lake Analytics

U-SQL

Data flow

SilverDataFlow

General

Settings

Parameters

User properties

Data flow *

DataTransformation

Open

New

Run on (Azure IR) *

AutoResolveIntegrationRuntime

Compute size *

Small

Advanced

Logging level *

Verbose

Basic

None

Sink properties

Staging