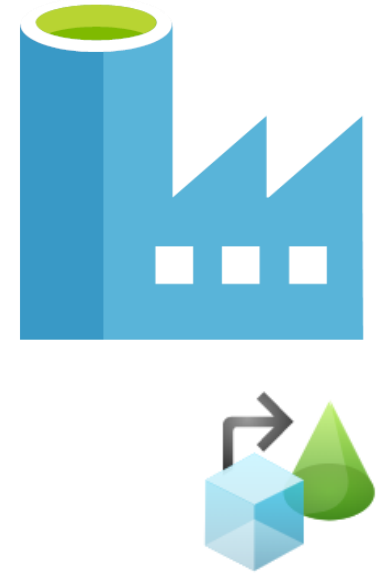


Azure Data Factory v2 with Mapping Data Flow (first blood)



Kamil Nowiński

Principal Microsoft Consultant

Kamil Nowiński



Microsoft Data Platform **MVP**
Speaker, blogger, data enthusiast
Principal Microsoft Consultant at Altius
(www.altiusdata.com)

15+ yrs experience as DEV/BI/(DBA)

Member of the Data Community PL

Project member of „SCD Merge Wizard”

„azure.datafactory.tools” & „azure.datafactory.devops”

Founder of blog SQLPlayer (www.SQLplayer.net)

SQL Server Certificates:

MCITP, MCP, MCTS, MCSA, MCSE Data Platform,

MCSE Data Management & Analytics

Moreover: Bicycle, Running, Digital photography

@NowinskiK, @SQLPlayer

BLOG & Interviews

- Technical posts
- Various skill level
- Cheet sheets
- Recommended books
- Many useful other links
- YouTube Channel ←
- Interviews (Podcast)



SQL Player

Play with data & have fun!

www.SQLPlayer.net



Scan me

Popular uploads ▶ PLAY ALL



How to automate deployment of Microsoft SQL database...

4.9K views • 8 months ago

Azure Data Factory | Continuous Integration &...

2.2K views • 5 months ago

Azure Data Factory | How to securely store passwords...

1.1K views • 6 months ago

Azure Data Factory | Deployment from master...

761 views • 4 months ago

Azure Data Factory | Publish from code with one task in...

357 views • 1 month ago

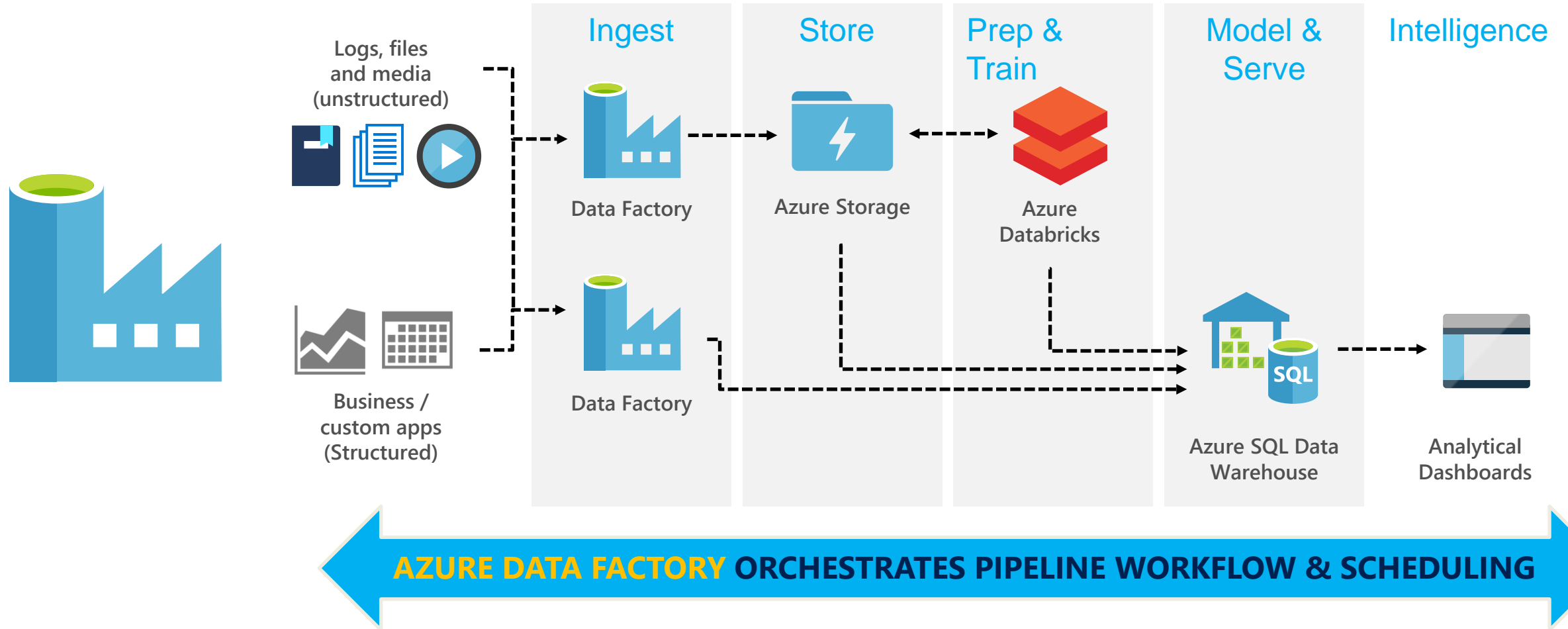
PODCAST – interviews with...



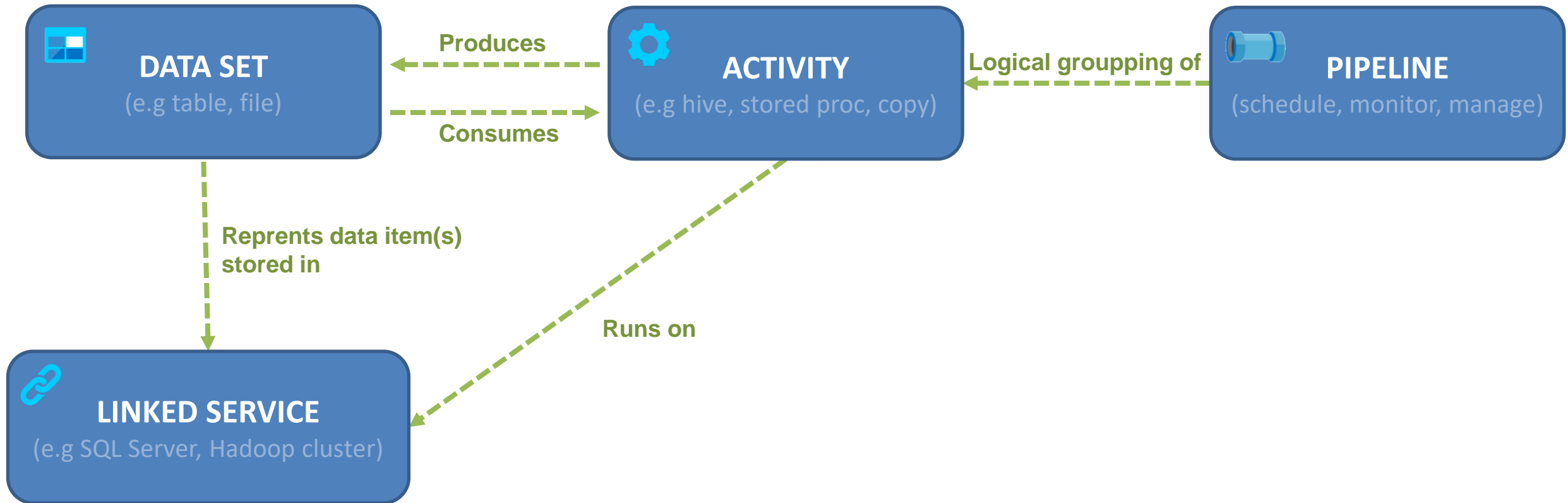
Scan me



What the Azure Data Factory is?



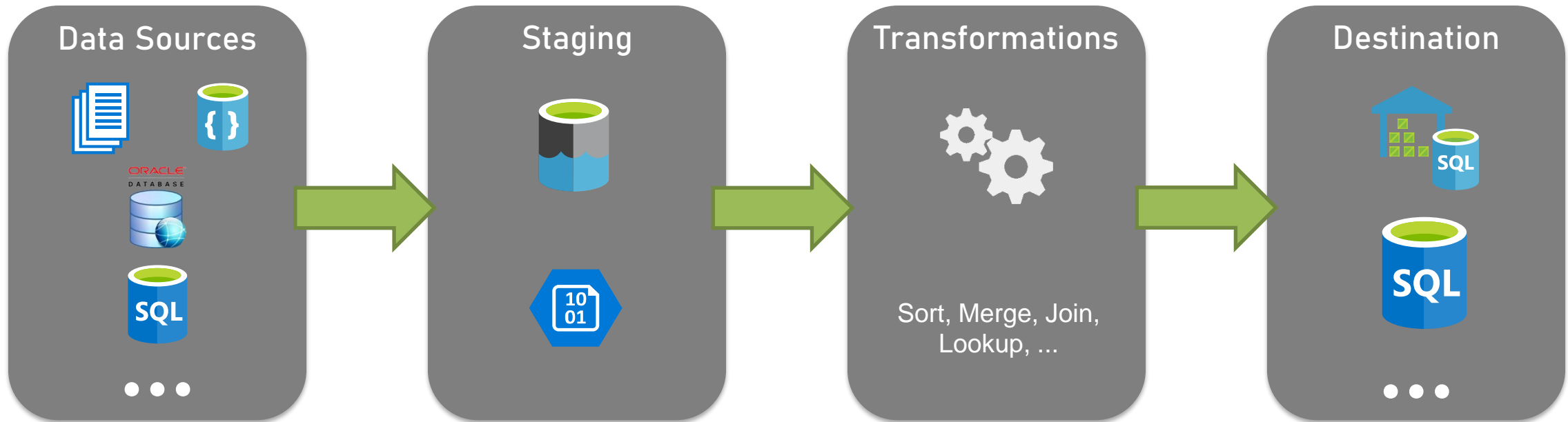
ADF Key Concepts



Visual Data Transformations with

MAPPING DATA FLOW

What the hell (Mapping) Data Flows are?



- Explicit user action
- User places data source(s) on design surface, from toolbox
- Select explicit sources

- Implicit/Explicit
- Data Lake staging area as default
- User does not need to configure this manually
- Advanced feature to set staging area options
- File formats/types: (Parquet, JSON, txt, CSV, ...)

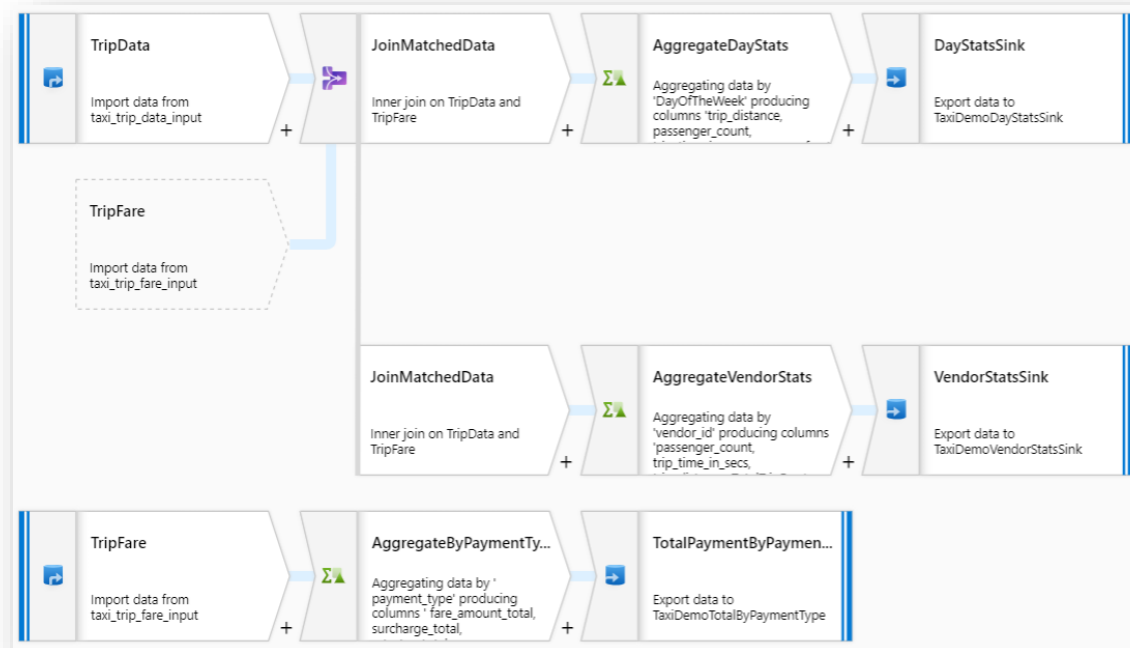
- Explicit user action
- User places transformations on design surface, from toolbox
- User must set properties for transformation steps and step connectors

- Explicit user action
- User chooses destination connector(s)
- User sets connector property options

src: (Microsoft) ADF Data Flow Private Preview Overview

Code-free Data Transformation at Scale

- Does not require understanding of Spark, Big Data Execution Engines, Clusters, Scala ...
- Focus on building business logic and data transformation
 - Data cleansing
 - Aggregation
 - Data conversions
 - Data prep
 - Data exploration
 - ETL Data Loading into DW



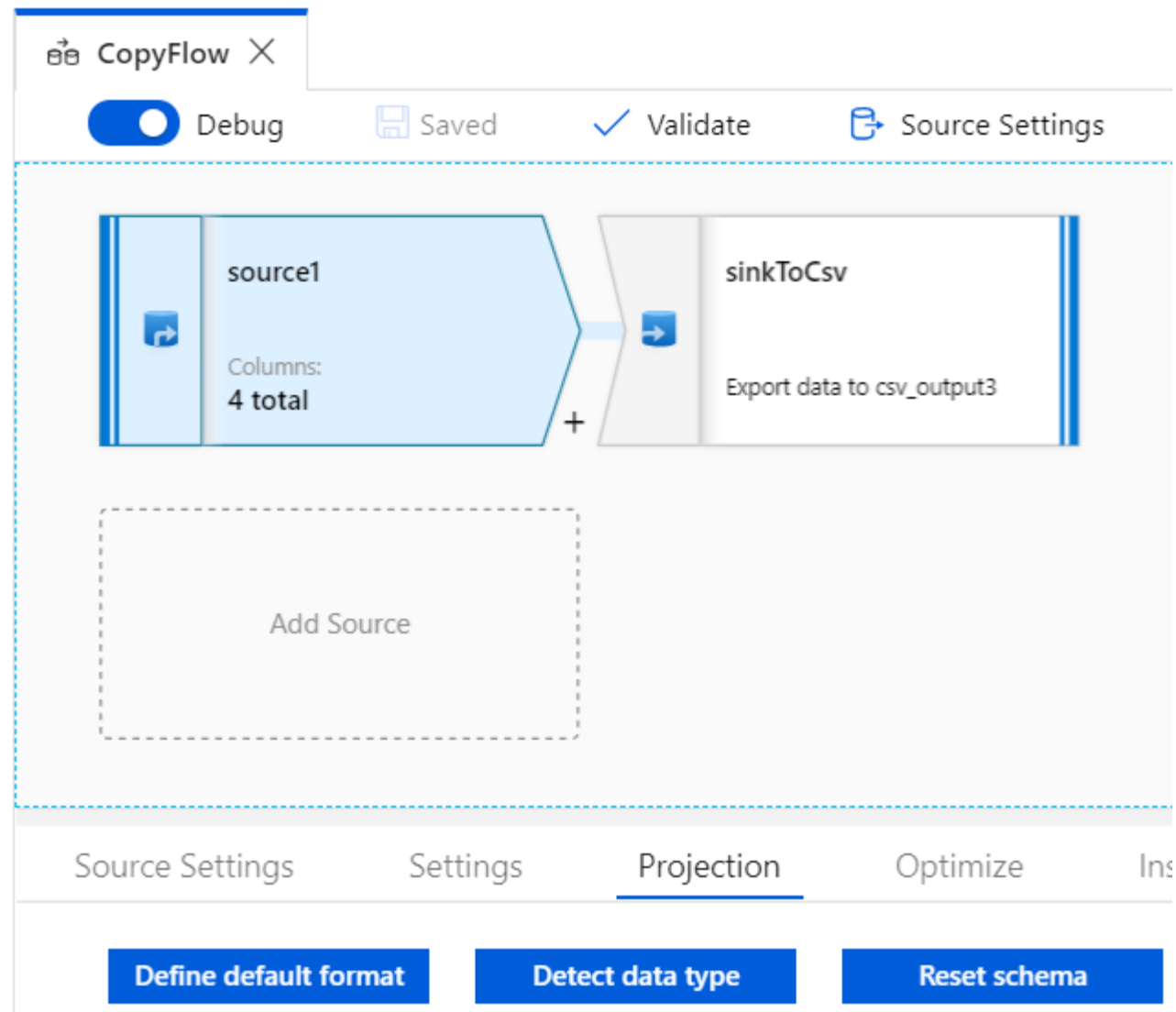
... not

```
1
2
3 HDI Cluster Details:
4 AdfHdi.azurehdinsight.net
5 Admin
6 Adf@123456
7
8 Storage:
9 adfhdistorage
10 /anyPw661j7f81lBwmiSo/YGdJyGt4d+5lJAr+5nS7b3g954706gK0K0ksZ19U0ut40z28x104wdMwQ==
11
12 Cluster Remote Login Details:
13 Adf
14 India@1234
15
16 HiveQuery:
17 DROP TABLE IF EXISTS MovieRatings;
18 CREATE EXTERNAL TABLE MovieRatings
19 (
20   UserID int,
21   MovieID int,
22   Rating int,
23   TimeStamp string
24 ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n' STORED AS TEXTFILE LOCATION '${hiveconf:MovieRatings}';
25
26 DROP TABLE IF EXISTS MovieTitles;
27 CREATE EXTERNAL TABLE MovieTitles
28 (
29   MovieID int,
30   MovieName string
31 ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n' STORED AS TEXTFILE LOCATION '${hiveconf:MovieTitles}';
```

Authoring of Azure Data Factory (v2) – what's new?







The screenshot shows the Microsoft Azure Data Factory portal interface for a workspace named 'SQLPlayerDemo2'. The top navigation bar includes 'Microsoft Azure', 'Data Factory', and the workspace name, along with a search bar labeled 'Search resources'. Below the navigation bar, there are action buttons: 'Data Factory' (with a dropdown arrow), 'Publish All', 'Validate All' (with a checkmark), 'Refresh', and 'Discard All'. The left sidebar, titled 'Factory Resources', contains a search bar 'Filter resources by name' and a list of resource types: 'Pipelines' (2), 'Datasets' (12), and 'Data Flows (Preview)' (5). The 'Data Flows (Preview)' item is highlighted with an orange box, and an orange arrow points from this box to the main content area. The main content area shows a list of data flows: 'CopyFlow', 'users', and 'dstUsersBlob', each with a close button. Below the list, there are buttons for 'Debug' (with a toggle switch), 'Validate' (with a checkmark), and 'Source Settings'.

Simple Copy Flow










Mapping Data Flow: Components = Actions *




Multiple inputs/outputs

-  New branch
-  Join
-  Conditional Split
-  Exists
-  Union
-  Lookup


Schema modifier

-  Derived Column
-  Select
-  Aggregate
-  Surrogate Key
-  Pivot
-  Unpivot
-  Window

Row modifier

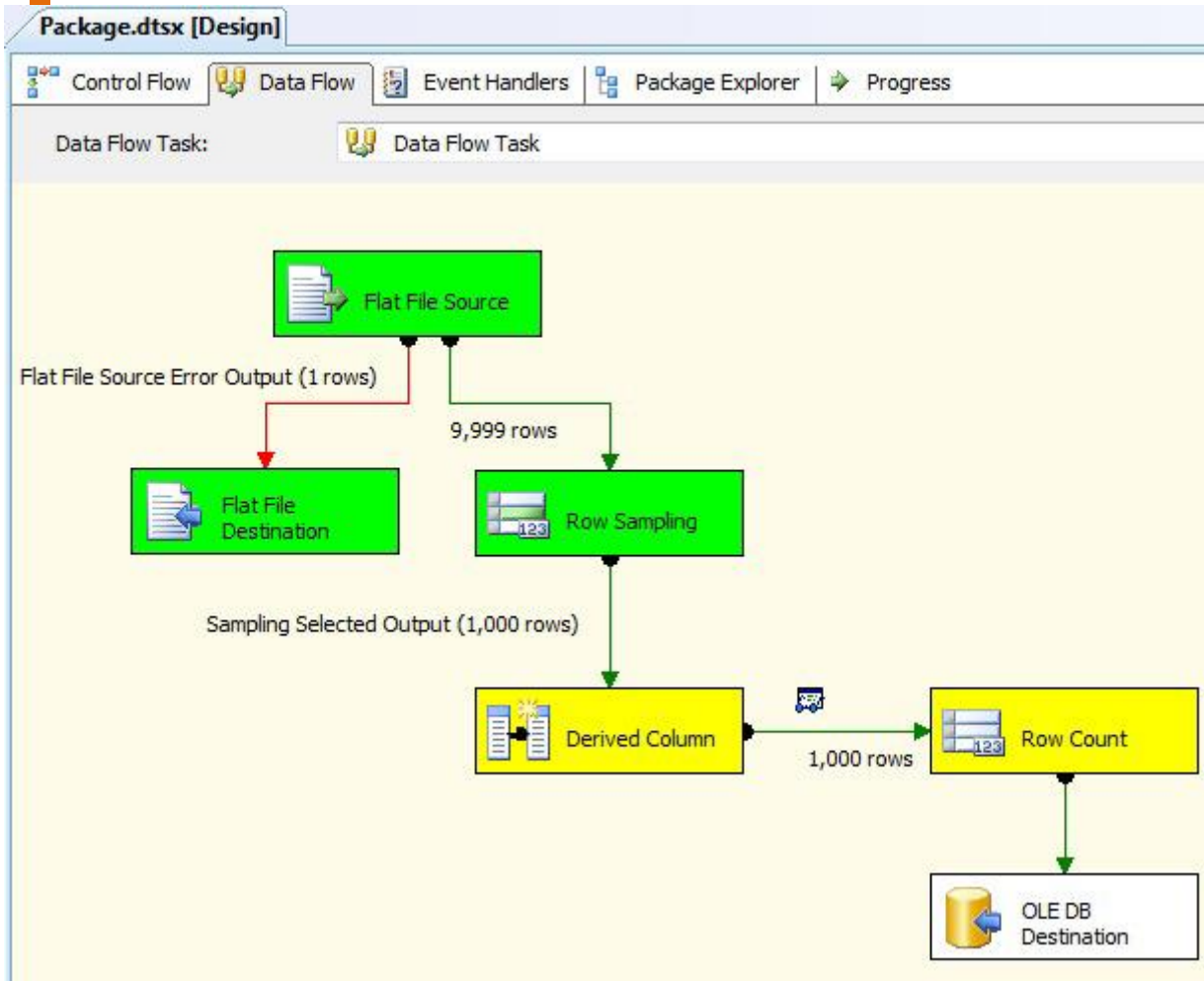
-  Filter
-  Sort
-  Alter Row

Destination

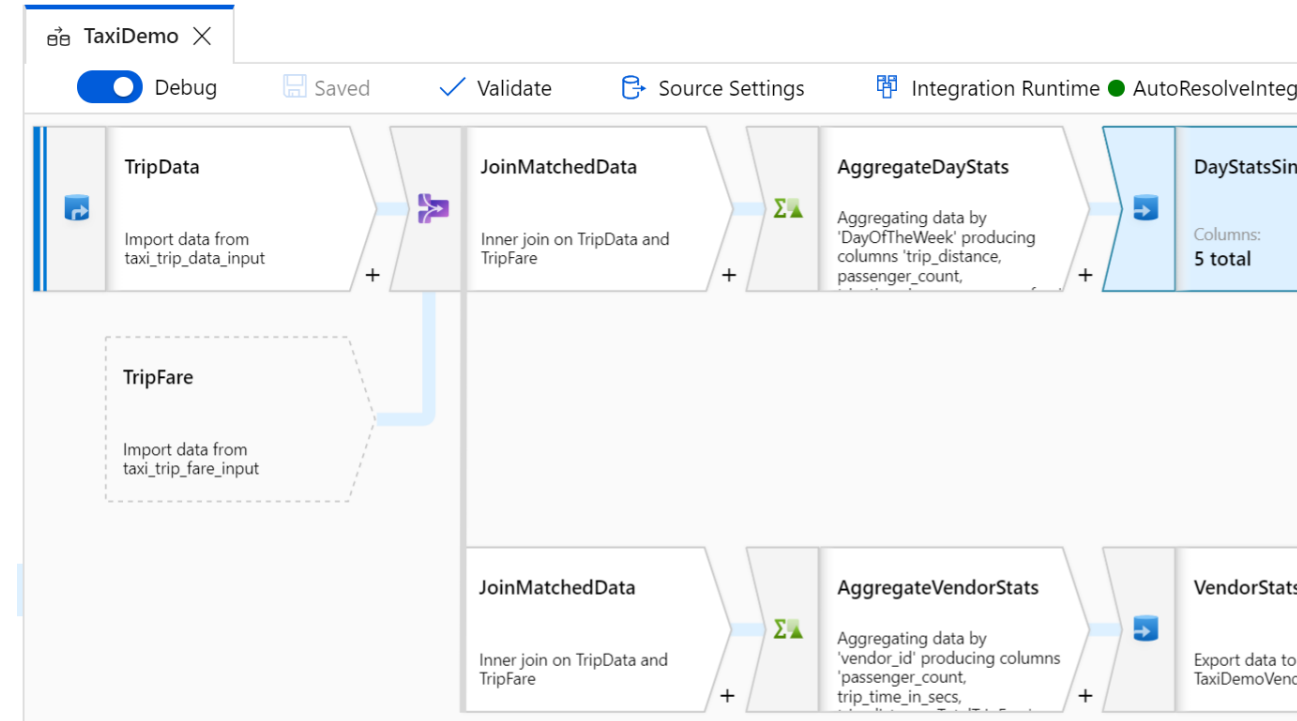
-  Sink

* With some small exceptions

SSIS Data Flow VS ADF Mapping Data Flow



<https://www.red-gate.com/simple-talk/sql/ssis/debugging-data-flow-in-sql-server-integration-services/>



Authoring of Azure Data Factory (v2)

Microsoft Azure

Search resources

BigPlayer Data Factory Publish All Validate All Refresh Discard All ARM Template

Factory Resources Filter Resources

- Pipelines 2
- Datasets 9
 - Badges
 - BadgesBlob
 - BadgesBlobWithHeader
 - BadgesStatsByName
 - BadgesStatsByNameBlob
 - Crimes_BlobCsv
 - Src_Users
 - Users_BlobCsv
 - UsersTest
- Data Flows 3
 - StackOverflow
 - badgesGroupByName
 - badgesGroupByName2
 - users

users

Debug Validate

sourceUsers Import data from Users_BlobCsv

Select1 Renaming sourceUsers to Select1 with columns 'DisplayName', 'DownVotes', 'LastAccessDate', 'Location', 'Reputation'

FilterByReputation Filtering rows using expressions on columns 'Reputation'

GroupByLocation Aggregating data by 'Location' producing columns 'SumOfReputation', 'SumOfViews', 'Count'

SortByLocation Sorting rows on columns 'Location'

Wrong Conditionally distributing the data in 2 groups, based on columns 'Location', 'Location', 'Location', 'Location', 'Location'

AllRight Conditionally distributing the data in 2 groups, based on columns 'Location', 'Location', 'Location', 'Location', 'Location'

General External dependencies

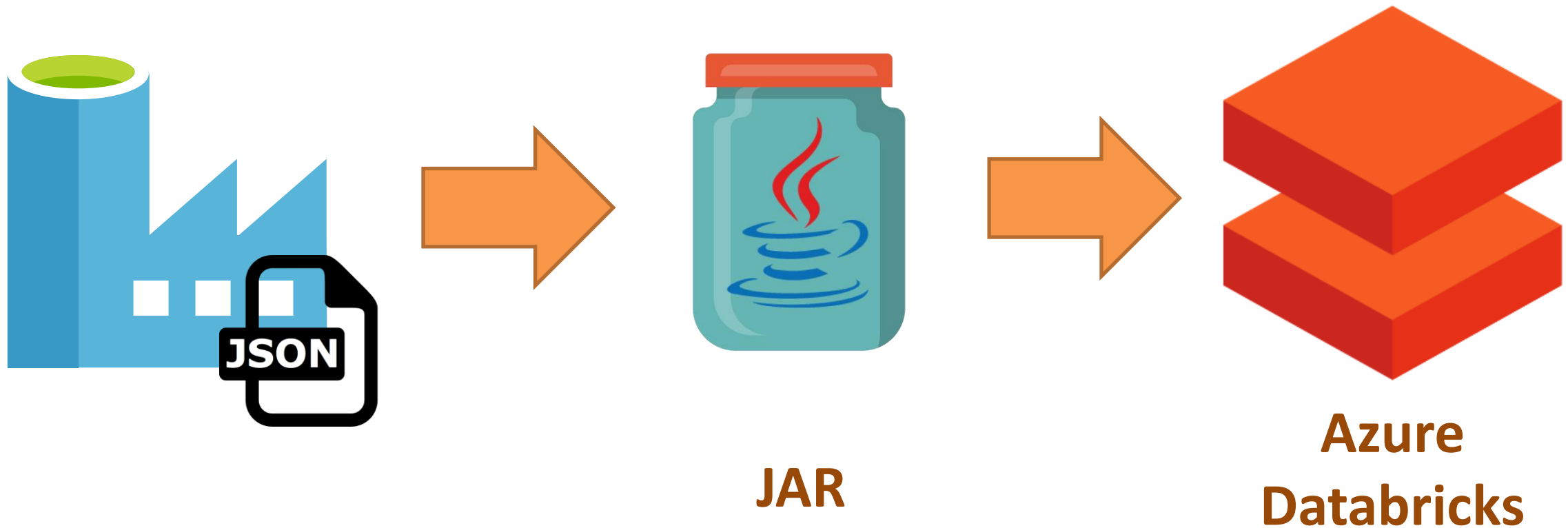
Name * users

Description

Guided experience to build data flows

The screenshot displays the Microsoft Azure Data Factory interface for a workspace named 'SQLPlayerDemo'. The top navigation bar includes options like 'Publish All', 'Validate All', 'Refresh', 'Discard All', and 'ARM Template'. On the left, the 'Factory Resources' pane lists 'Pipelines' (5), 'Datasets' (16), and 'Data Flows (Preview)' (6). The 'usersql' data flow is selected, showing a pipeline with steps: 'source1' (13 total columns), 'Select1' (renaming source1 to Select1 with columns 'Id, Age, DisplayName, DownVotes'), 'FilterByReputation' (filtering rows using expressions on columns 'Reputation'), 'GroupByLocation' (aggregating data by 'Location' producing columns 'Reputation, DownVotes, Views'), 'Sort1' (sorting rows on columns 'Location'), 'Filter1' (filtering rows using expressions on columns 'Location, Location'), and 'sink1' (Export data to AzureBlob2). A context menu is open over the 'source1' step, listing options under 'Multiple inputs/outputs' (New Branch, Join, Conditional Split, Union, Lookup), 'Schema modifier' (Derived Column, Aggregate, Surrogate Key, Pivot, Unpivot, Window), and 'Row modifier'. The bottom pane shows the 'Source Settings' for 'source1', including 'Source Dataset' (stack_users), 'Options' (Allow schema drift checked), 'Sampling' (Enable selected), and 'Rows limit' (1000).

What is going on behind the scenes?



DEMO TIME

Data Preview in Debug mode

Currency Conv... X Currency Conv... X MovieDemoPi... X TaxiDemo X MovieDemo X TaxiDemo X

Debug Saved Validate Source Settings Integration Runtime AutoResolveIntegrationRuntime

TripData
Import data from taxi_trip_data_input

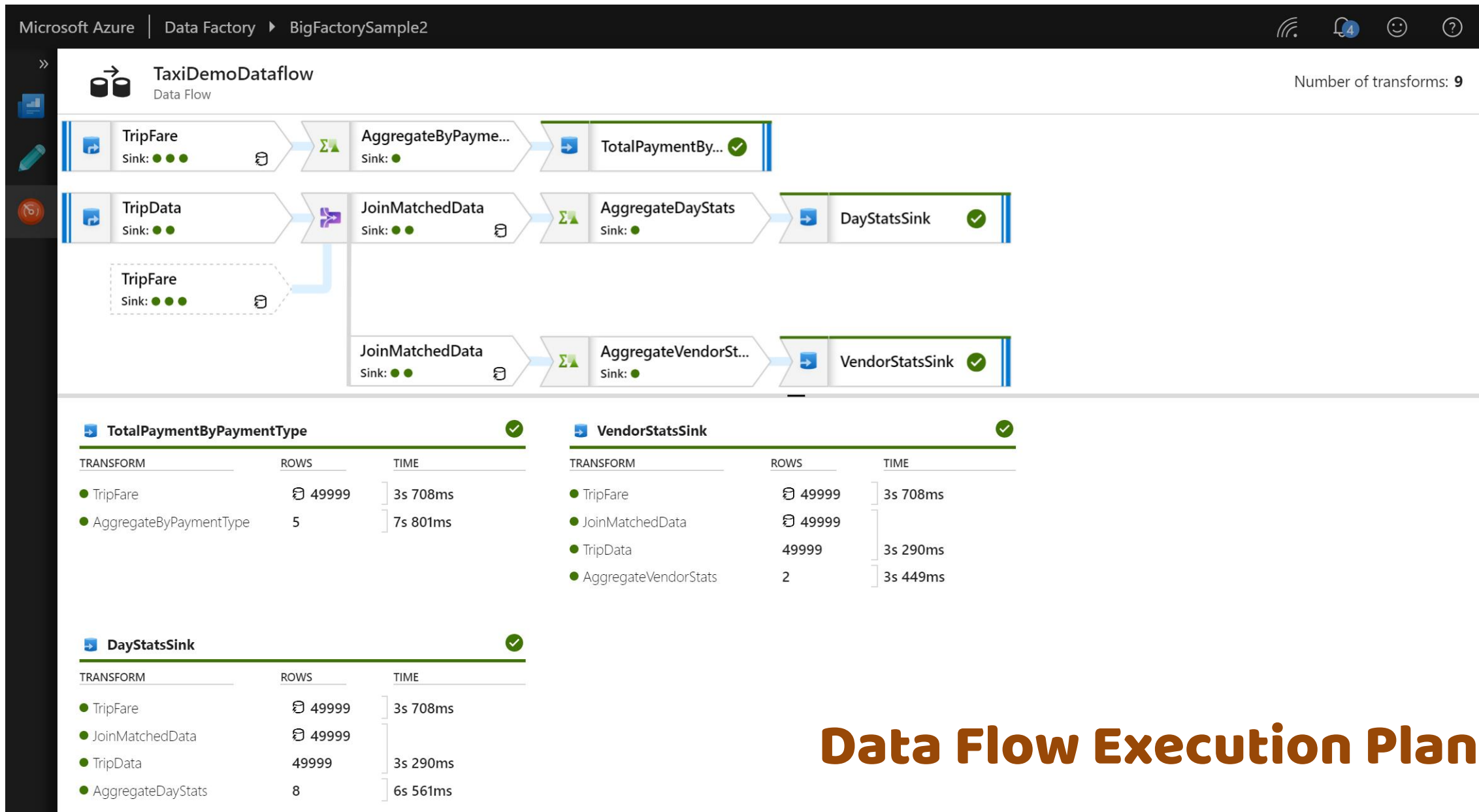
JoinMatchedData
Inner join on TripData and TripFare

AggregateDayStats
Aggregating data by 'DayOfTheWeek' producing columns 'trip_distance', 'passenger_count', 'trip_time_in_secs', 'average_fare'

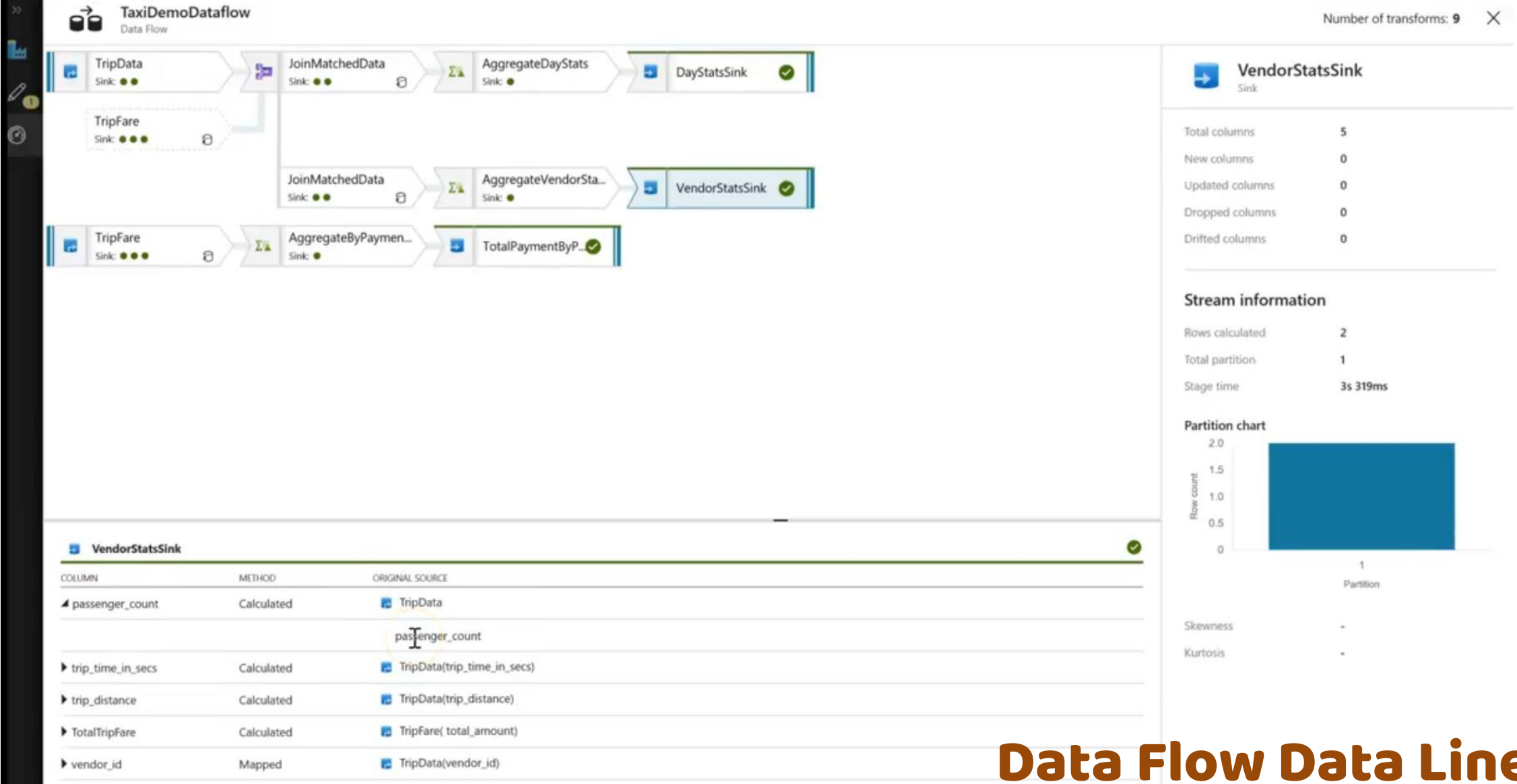
DayStatsSink
Columns: 5 total

Sink Settings Mapping Optimize Inspect Data Preview

	Updated*	New+	Unchanged	Total
Number of rows	N/A	N/A	N/A	8
DayOfTheWeek 123	trip_distance 1.2	passenger_count 1.2	trip_time_in_secs 1.2	average_fare 1.2
NULL	2.58	2.16	10.21	12.82
1	3.03	2.17	10.12	13.89
6	1.75	1.43	9.98	10.84
3	2.26	1.41	9.83	11.02
5	2.71	1.24	10.9	15.59



Data Flow Execution Plan



Data Flow Data Lineage

ADF Template Gallery

Template gallery X

Filter

Reset all filter

Search templates

Categories

☐ Copy

☐ Data Flow

☐ SSIS

☐ Transform

Create by

☐ Microsoft

☐ My templates

Tag

All

Services used

All

Import template

Bulk Copy from Database to Azure Data Explorer

Use this template to copy large amount of data in bulk from database like SQL Server, Google BigQuery, etc to Azure Data Explorer (ADX), using...

by Microsoft

Bulk Copy from Database

Use this template to copy data in bulk from database using external control table to store partition list of source tables.

...

by Microsoft

Copy data from Google BigQuery to Azure Data Lake Store

Use this template to copy data from Google BigQuery to Azure Data Lake Storage.

...

by Microsoft

Copy data from HDFS to Azure Data Lake Store

Use this template to copy data from HDFS (Hadoop Distributed File System) to Azure Data Lake Storage.

...

by Microsoft

Copy data from Netezza to Azure Data Lake Store

Use this template to copy data from Netezza server to Azure Data Lake Storage.

...

by Microsoft

Copy data from on premise SQL Server to SQL Azure

Use this template to copy data from on premise SQL Server to SQL Azure.

...

by Microsoft

Copy data from on premise SQL Server to SQL Data Warehouse

Use this template to copy data from on premise SQL Server to SQL Data Warehouse.

...

by Microsoft

Copy data from Oracle to SQL Data Warehouse

Use this template to copy data from Oracle server to SQL Data Warehouse.

...

by Microsoft

Copy delta data from AWS S3 to Azure Data Lake Storage Gen2

Use this template to copy delta data from petabytes

...

Copy multiple files containers between File Stores

Use this template to copy all files from AWS S3 to

...

Copy new files only by LastModifiedDate

Use this template to copy new and changed files only by using LastModifiedDate.

...

Data Flow Search Log Analytics

This is a sample that takes the U-SQL SearchLog analytics example and turns it into an ADF Data Flow.

...

Azure DevOps GIT


Factory Resources

Filter resources by name


+

Pipeline

Pipeline from template

 **SQL Player**
Play with data & have fun!

Azure Data Factory with Mapping Data Flow (first blood)

altius  @NowinskiK

Mapping Data Flow – Source & Sink



Azure Blob Storage



Azure Cosmos DB (SQL API)



Azure Data Lake Storage Gen1



Azure Data Lake Storage Gen2



Azure SQL Database



Azure Synapse Analytics (formerly SQL DW)

Mapping Data Flow – Source & Sink capabilities



- New capabilities for Source transformations:
 - wildcards, file sets,
 - move file / Delete file,
 - auto-detect types,
 - schema validation
 - query statement



- New capabilities for Sink transformations:
 - output to single file,
 - clear folder,
 - truncate table / recreate table,
 - naming patterns

Mapping Data Flow – DataSet File Formats



Avro



DelimitedText



Excel



Binary



ORC



Json



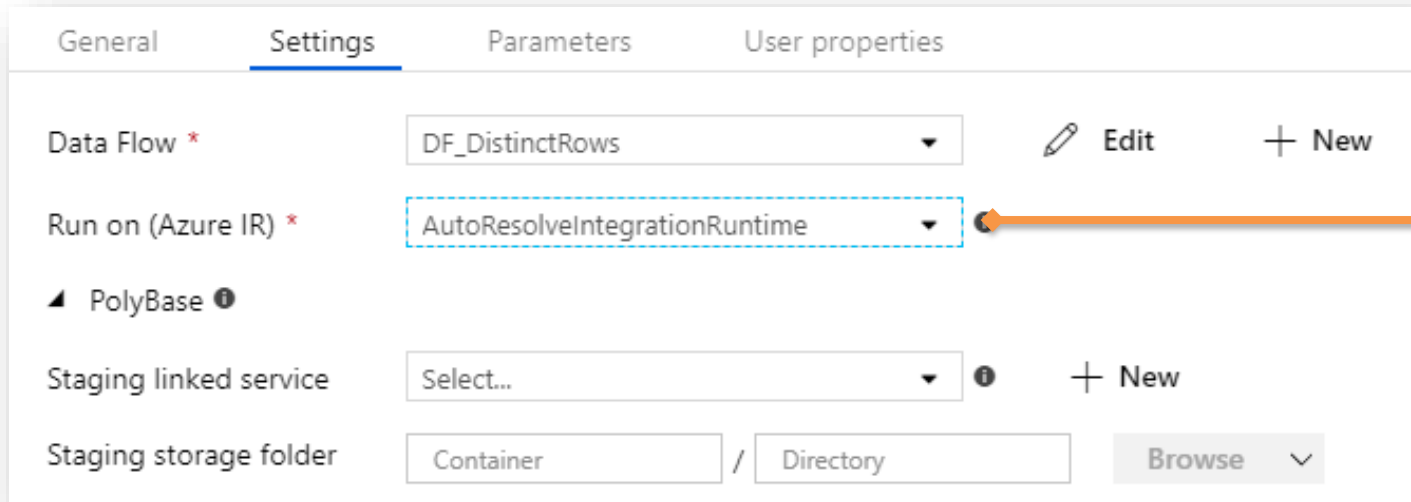
Parquet



XML

Mapping Data Flow – Execution Settings

- The Execute Data Flow transformation:
 - Support **parameterized datasets**
 - Control **size of cluster** for specific Azure IR
 - Define **TTL (Time-To-Live)** to Azure IR to reduce data flow activity time



General Settings Parameters User properties

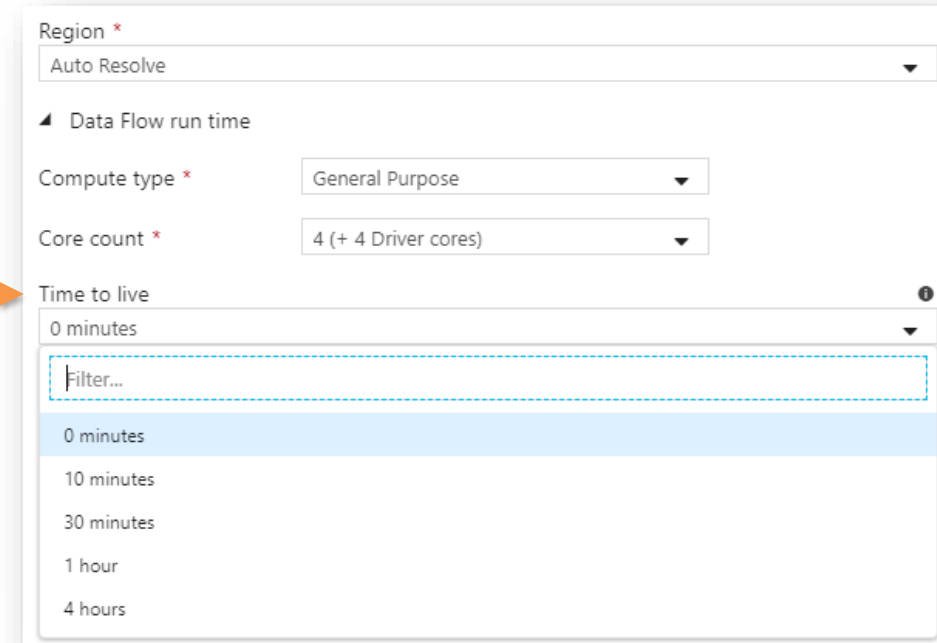
Data Flow * DF_DistinctRows Edit + New

Run on (Azure IR) * AutoResolveIntegrationRuntime

PolyBase ⓘ

Staging linked service Select... ⓘ + New

Staging storage folder Container / Directory Browse ▾



Region * Auto Resolve ▾

Data Flow run time

Compute type * General Purpose ▾

Core count * 4 (+ 4 Driver cores) ▾

Time to live ⓘ

0 minutes ▾

Filter...

0 minutes

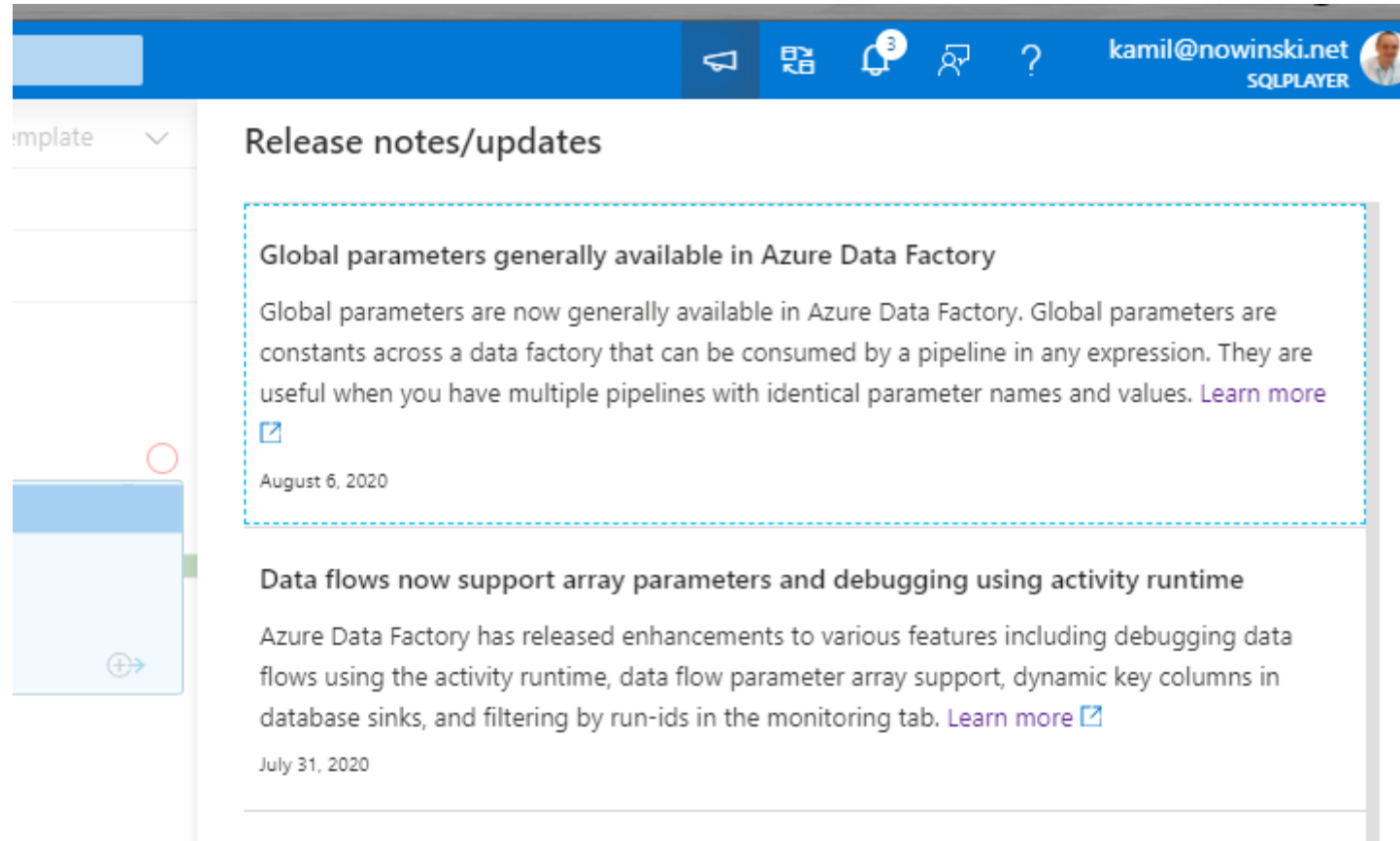
10 minutes

30 minutes

1 hour

4 hours

Latest updates?



The screenshot shows the SQL Player application interface. At the top, there is a blue header bar with navigation icons (home, search, notifications, help) and a user profile section for 'kamil@nowinski.net' with a profile picture and the text 'SQLPLAYER'. Below the header, the main content area is titled 'Release notes/updates'. It contains two update entries, each enclosed in a dashed blue border. The first entry is titled 'Global parameters generally available in Azure Data Factory' and describes how global parameters are now available as constants across a data factory. The second entry is titled 'Data flows now support array parameters and debugging using activity runtime' and describes enhancements to debugging data flows. Both entries include a 'Learn more' link and a date.

Release notes/updates

Global parameters generally available in Azure Data Factory

Global parameters are now generally available in Azure Data Factory. Global parameters are constants across a data factory that can be consumed by a pipeline in any expression. They are useful when you have multiple pipelines with identical parameter names and values. [Learn more](#)

August 6, 2020

Data flows now support array parameters and debugging using activity runtime

Azure Data Factory has released enhancements to various features including debugging data flows using the activity runtime, data flow parameter array support, dynamic key columns in database sinks, and filtering by run-ids in the monitoring tab. [Learn more](#)

July 31, 2020

Latest updates? Go Twitter!



Daniel Perlovsky @big_data_da

Did a big update and reorg of the to understand the ADF's team re

Disagree with anything? Put it in

docs.microsoft.com/en-us/azure/

#Azure #DataFactory #MappingD



Mapping data
Learn about
mapping data
[docs.micr](https://docs.microsoft.com/en-us/azure/)



22



Mark Kromer Retweeted

Azure Data Factory

@AzDataFactory

Global parameters are now gener
There are constants at the factory
consumed by a pipeline in any ex
our blog!

[techcommunity.microsoft.com/t5,](https://techcommunity.microsoft.com/t5/Azure-Data-Factory/Global-Parameters-are-now-general-constants-at-the-factory-consumed-by-a-pipeline-in-any-ex-our-blog/ba-p/1611111)

#Azure #DataFactory #Parameteri



Global Parameters generally av
Global parameters are now ge
Factory. Global parameters are
[techcommunity.microsoft.cc](https://techcommunity.microsoft.com/t5/Azure-Data-Factory/Global-Parameters-are-now-general-constants-at-the-factory-consumed-by-a-pipeline-in-any-ex-our-blog/ba-p/1611111)

8:00 pm · 6 Aug 2020 · [Twitter Web App](#)



Mark Kromer @KromerBigData · 18 Aug

When designing #Azure #DataFactory data flows, take into consideration the debug setting row limits to understand why Joins can produce different results at runtime vs. design time. #ADF .@AzDataFactory

General Parameters

Data flow debug IR: dataflowduster

source1

☒ Source dataset ☐ Sample file

Row limit

1000

source2

☒ Source dataset ☐ Sample table

Row limit

1000

ADF Data Flows: Why Joins sometimes fail while Debugging

Azure Data Factory's data flows are designed to provide cloud-scale ETL and big data analytics with an east-to-use UI that can scale automaticall...

kromerbigdata.com



SQL Player
Play with data & have fun!















Azure Data Factory with Mapping Data Flow (first blood)

altius



@NowinskiK

SSIS vs ADF activities vs T-SQL

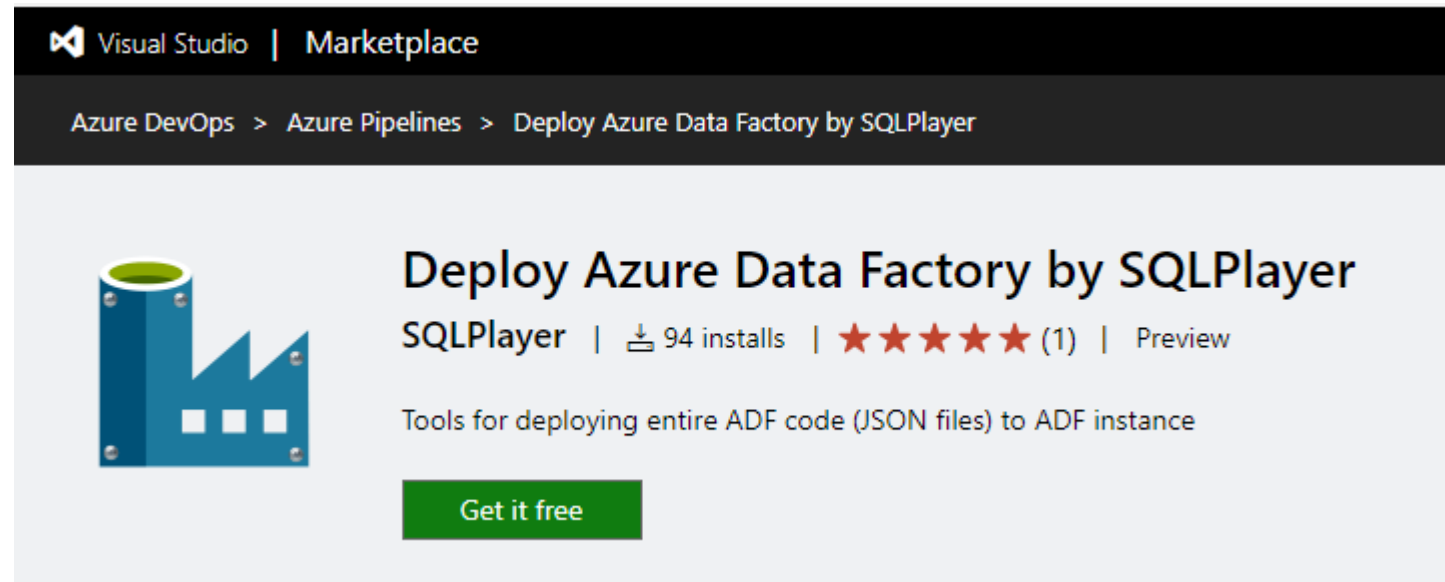
Activity	Description	SSIS equivalent	SQL Server equivalent
 New branch	Create a new flow branch with the same data	 Multicast (+icon)	<code>SELECT INTO SELECT OUTPUT</code>
 Join	Join data from two streams based on a condition	 Merge join	<code>INNER LEFT RIGHT JOIN, CROSS FULL OUTER JOIN</code>
 Conditional Split	Route data into different streams based on conditions	 Conditional Split	<code>SELECT INTO WHERE condition1 SELECT INTO WHERE condition2 CASE ... WHEN</code>
 Union	Collect data from multiple streams	 Union All	<code>SELECT col1a UNION (ALL) SELECT col1b</code>
 Lookup	Lookup additional data from another stream	 Lookup	<code>LEFT RIGHT JOIN</code>
 Derived Column	Compute new columns based on the existing once	 Derived Column	<code>SELECT Column1 * 1.09 as NewColumn</code>
 Aggregate	Calculate aggregation on the stream	 Aggregate	<code>SELECT Year(DateOfBirth) as Year, MIN(), MAX(), AVG() GROUP BY Year(DateOfBirth)</code>

<http://bit.ly/ADFDFvsSSIS>

<http://bit.ly/ADFDF-CheatSheet>

ADF Deployment: TOOLS!

 <https://github.com/SQLPlayer/azure.datafactory.tools>
<https://github.com/SQLPlayer/azure.datafactory.devops>



Resources



<http://sqlplayer.net/ADF>

Q&A



Thank you!

teşekkür ederim!



kamil@nowinski.net



@NowinskiK

@SQLPlayer



SQLPlayer.net



<https://github.com/NowinskiK/CommunityEvents>



Kamil Nowinski

Microsoft Data Platform MVP

MCSE Data Platform & MCSE Data Management and Analytics