





















Azure Data Factory – Mapping Data Flows’ Components

| Activity | Description | SSIS equivalent | SQL Server equivalent |
|--------------------------------|---|--|---|
| Multiple inputs/outputs | | | |
| 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> |
| Exists | Check the existence of data in another stream | Lookup / Merge Join | <code>SELECT * FROM Table WHERE EXISTS (SELECT ...) JOIN NOT EXISTS</code> |
| Union | Collect data from multiple streams | Union All | <code>SELECT colla UNION (ALL) SELECT collb</code> |
| Lookup | Lookup additional data from another stream | Lookup | <code>LEFT RIGHT JOIN</code> |
| Schema modifier | | | |
| Derived Column | Compute new columns based on the existing once | Derived Column | <code>SELECT Column1 * 1.09 as NewColumn</code> |
| Select | Choose columns to flow to the next stream | OUTPUT in components, mapping columns | <code>SELECT Column1, Column4 FROM Table</code> |
| Aggregate | Calculate aggregation on the stream | Aggregate | <code>SELECT Year(DateOfBirth) as Year, MIN(), MAX(), AVG() GROUP BY Year(DateOfBirth)</code> |
| Surrogate Key | Add a surrogate key column to output stream from a specific value | Script Component | <code>SELECT ROW_NUMBER() OVER(ORDER BY n ASC) AS R#, n FROM sys.databases + Incremental Primary Key (with limited capabilities)</code> |
| Pivot | Pivots row values into columns, groups columns and aggregates data | Pivot | <code>SELECT rowCol, c1, c2 FROM (SELECT sourceCols FROM Table) PIVOT (SUM(sumCol) FOR col IN (...))</code> |
| Unpivot | Unpivots columns into row values and ungroups columns | Unpivot | <code>SELECT rowCol, col, X FROM (SELECT rowCol, c1, c2 FROM pvt) UNPIVOT (X FOR col FROM (c1, c2)) AS unpvt</code> |
| Window | Aggregates data based on a window and joins with original data | [Sort] + Custom Script | <code>SELECT fun() OVER(PARTITION BY pc ORDER BY oc) newc, pc, oc, otherCols FROM Table</code> |
| Rank | Generates an ordered ranking based upon sort conditions specified by the user | [Sort] + Custom Script | <code>SELECT ROW_NUMBER() OVER (PARTITION BY pc ORDER BY oc) newc, pc, oc, otherCols FROM Table</code> |
| External Call | Enables custom function by calling out to external REST endpoints row-by-row | Script Component | Not available out of the box. Some tricks are possible: OLE Automation, CLR function, although not recommended. |



Azure Data Factory – Mapping Data Flows' Components

| Activity | Description | SSIS equivalent | SQL Server equivalent |
|--|--|--|---|
| Formatters | | | |
|  Flatten | Takes array values from hierarchical structures such as JSON (denormalization) |  Script Component or 3 rd Party Component (ZappySys SSIS JSON Parser Transform) | <pre>SELECT * FROM OPENJSON(json) WITH (id INT 'strict \$.id', Name NVARCHAR(50) '\$.info.name');</pre> |
|  Parse | Parses columns in document form data (JSON, XML, delimited text) |  Script Component or 3 rd Party Component (ZappySys SSIS JSON Parser Transform) | <pre>SELECT * FROM OPENJSON ... SELECT * FROM OPENXML ... SELECT value FROM STRING_SPLIT('clothing,road', ',');</pre> |
|  Stringify | Turns complex data types into strings |  Script Component | |
| Row modifier | | | |
|  Filter | Filter rows in the stream based on a condition |  Conditional Split | <pre>SELECT * FROM Table WHERE [Column] LIKE '%pattern%'</pre> |
|  Sort | Order data in the stream based on column(s) |  Sort | <pre>SELECT * FROM Table ORDER BY [Column] ASC</pre> |
|  Alter Row | Set action policy on rows when database is sink |  Conditional Split + <i>n</i> Destinations | <pre>MERGE INSERT, UPDATE, DELETE IF WHERE</pre> |
|  Assert | Enables you to build custom rules for data quality and data validation |  Conditional Split or DQS Cleansing | <pre>WHERE, REPLACE, SUBSTRING, IF, RAISERROR, THROW</pre> |
| Source / Destination | | | |
|  Source | Source for your data flow. Obligatory first element of every Data Flow in ADF |  OLE DB Source and more ... | <pre>SELECT * FROM SourceTable</pre> |
|  Sink | Destination for your data flow stream |  OLE DB Destination and more... | <pre>INSERT INTO TargetTable</pre> |

Version: 2.0 (Jan 2022)