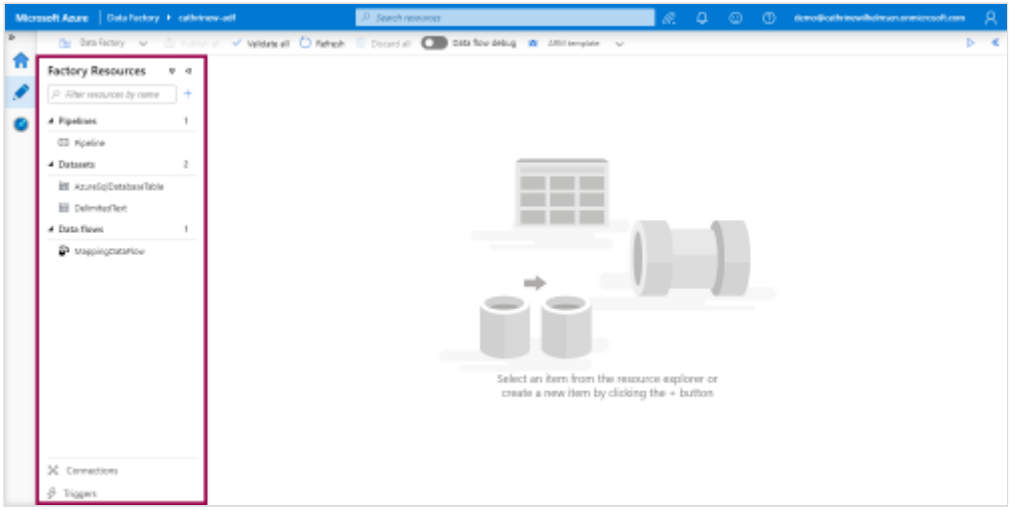
Overview of Azure Data Factory Components

This post is part 3 of 25 in the series Beginner's Guide to Azure Data Factory

In the previous post, we started by creating an Azure Data Factory, then we looked at the user interface and the three main Azure Data Factory pages. In this post, we will go through the *Author* page in more detail. Let's look at the different Azure Data Factory components!

Azure Data Factory Components

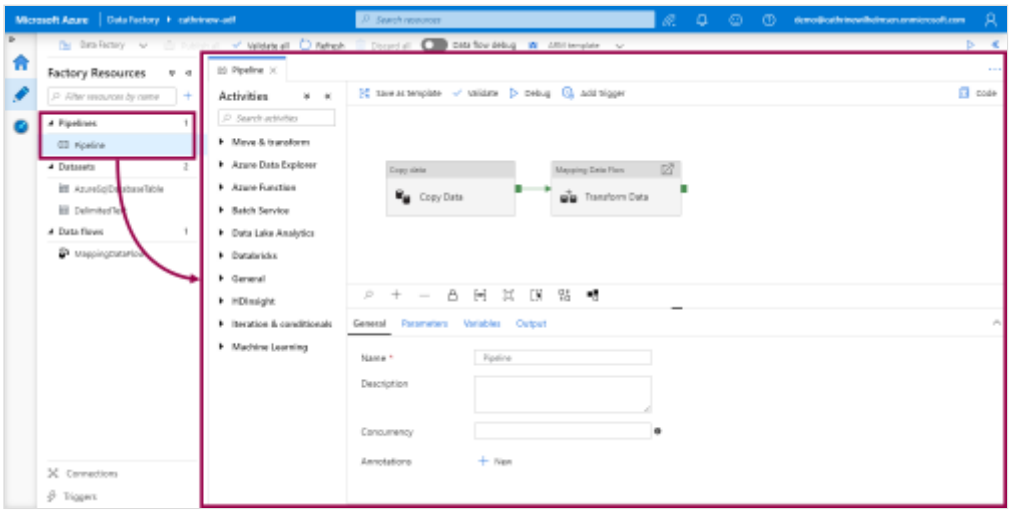
On the left side of the *Author* page, you will see your *factory resources*. In this example, we have already created one *pipeline*, two *datasets*, and one *data flow*:



Let's go through each of these Azure Data Factory components and explain what they are and what they do.

Pipelines

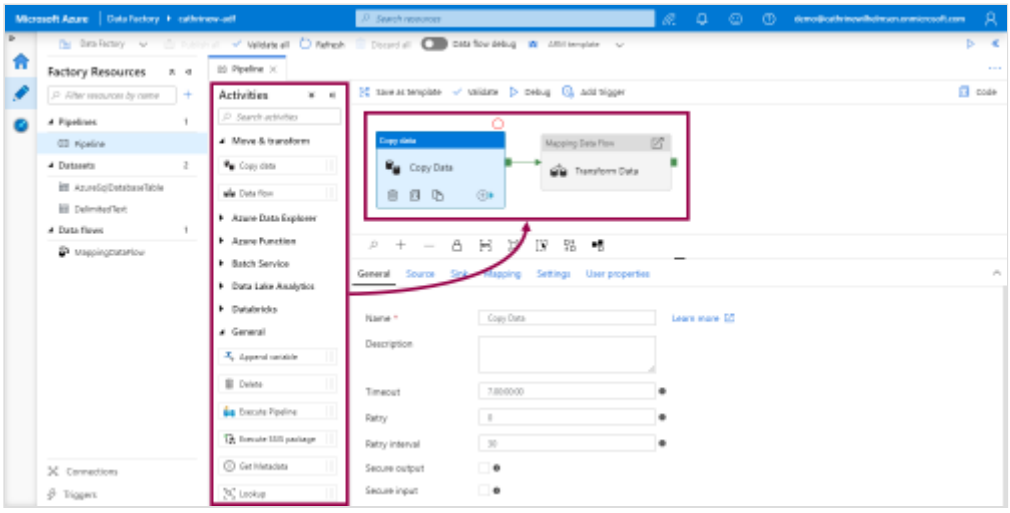
Pipelines are the things you *execute* or *run* in Azure Data Factory, similar to packages in SQL Server Integration Services (SSIS). This is where you define your *workflow*: what you want to do and in which order. For example, a pipeline can first copy data from an on-premises data center to Azure Data Lake Storage, and then transform the data from Azure Data Lake Storage into Azure Synapse Analytics (*previously Azure SQL Data Warehouse*).



When you open a pipeline, you will see the pipeline authoring interface. On the left side, you will see a list of all the *activities* you can add to the pipeline. On the right side, you will see the design canvas with the properties panel underneath it.

Activities

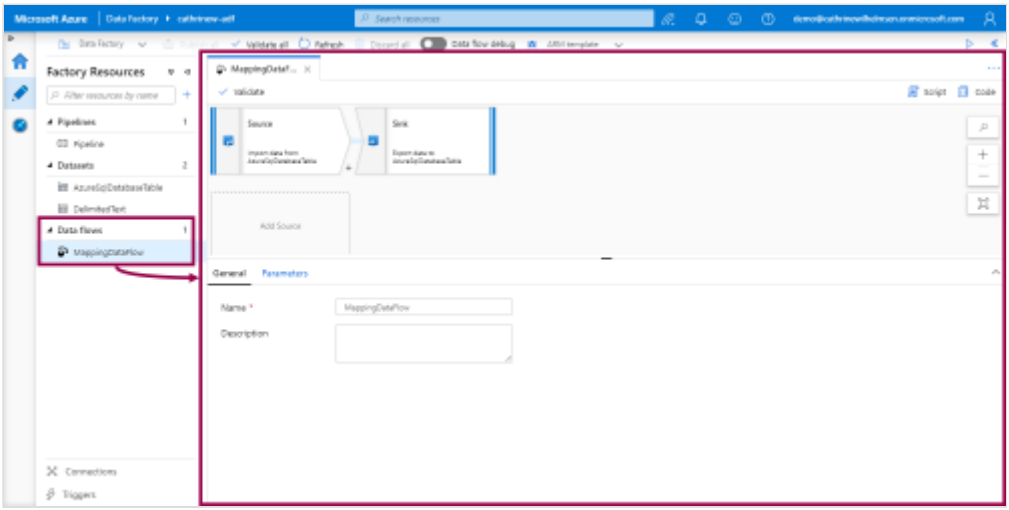
Activities are the individual steps inside a pipeline, where each activity performs a single task. You can chain activities or run them in parallel. Activities can either control the flow inside a pipeline, move or transform data, or perform external tasks using services outside of Azure Data Factory.



You add an activity to a pipeline by dragging it onto the design canvas. When you click on an activity, it will be highlighted, and you will see the activity properties in the properties panel. These properties will be different for each type of activity.

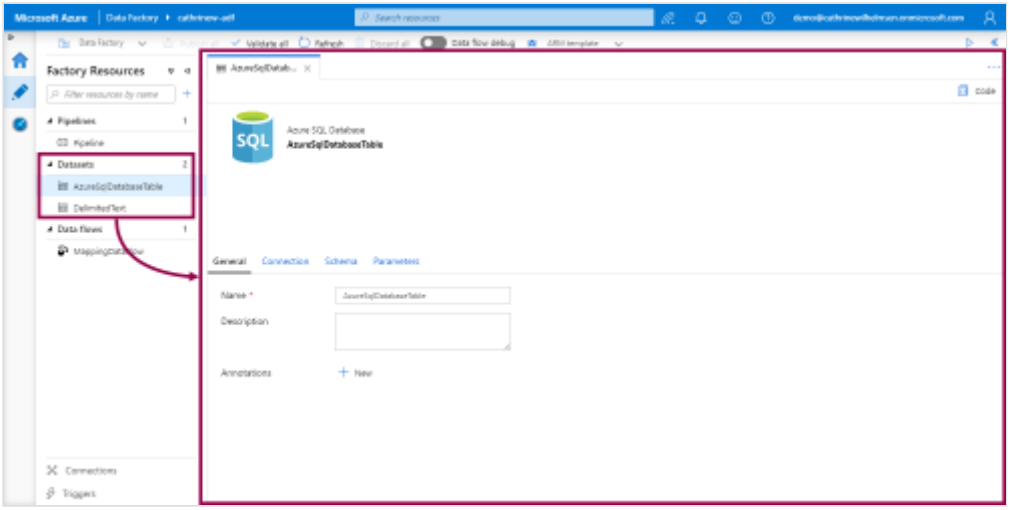
Data Flows

Data Flows are a special type of activity for creating visual data transformations without having to write any code. There are two types of data flows: mapping and wrangling.



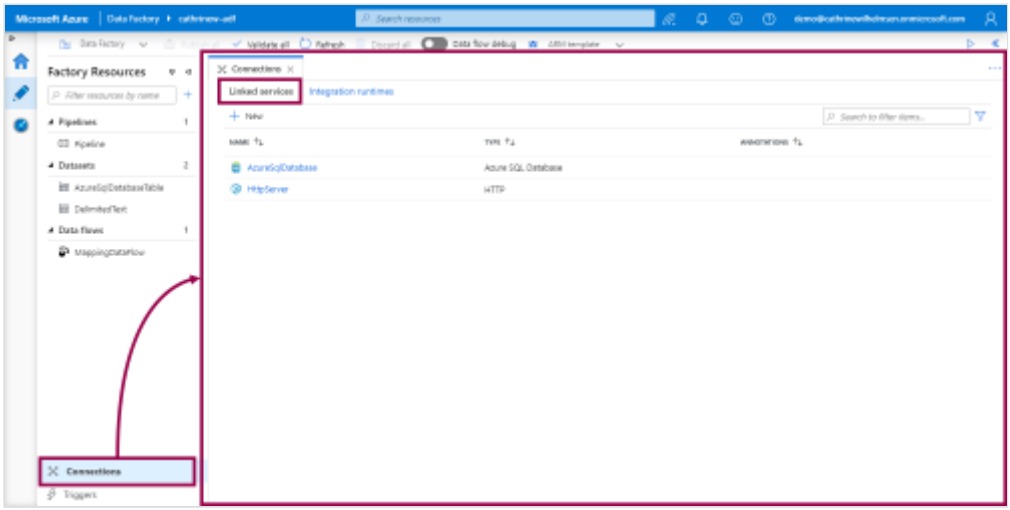
Datasets

If you are moving or transforming data, you need to specify the format and location of the input and output data. Datasets are like named views that represent a database table, a single file, or a folder.



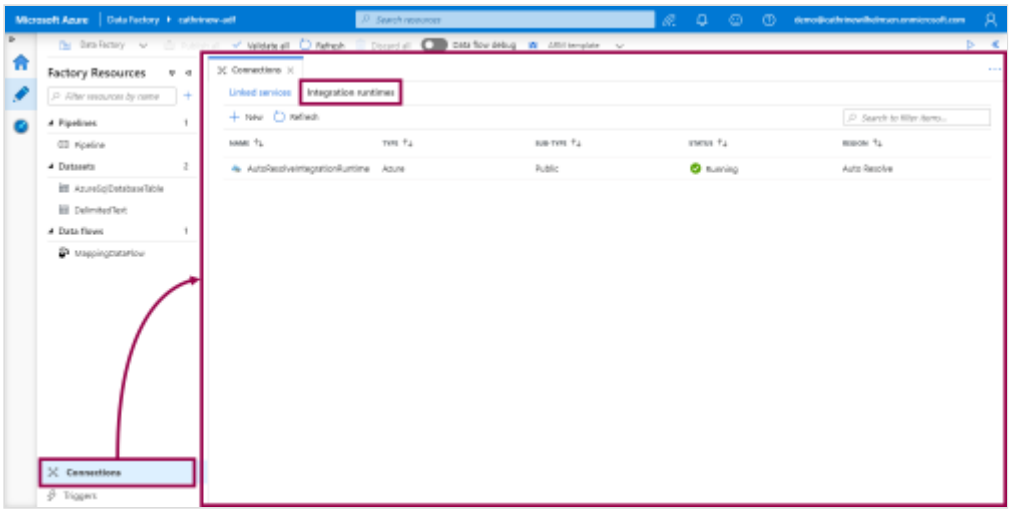
Linked Services

Linked Services are like connection strings. They define the connection information for data sources and services, as well as how to authenticate to them.



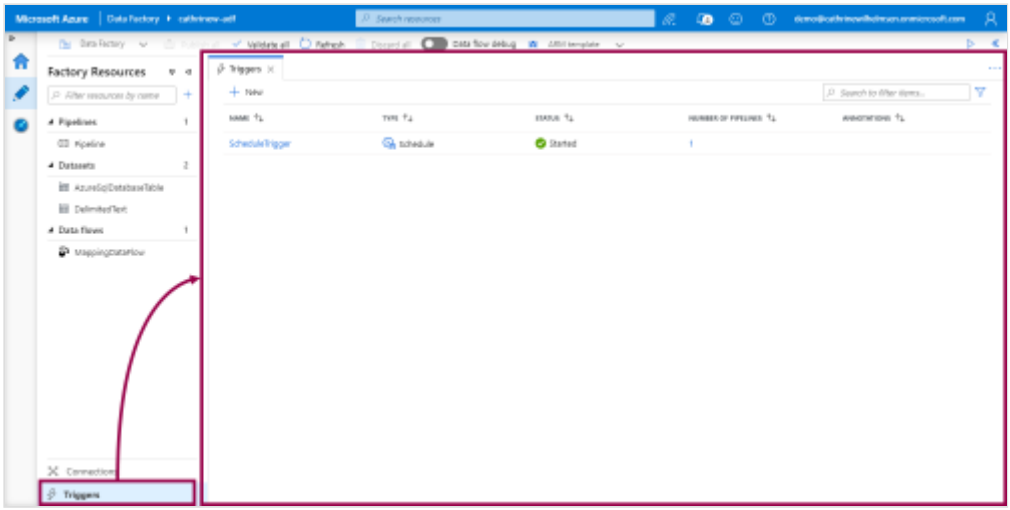
Integration Runtimes

Integration runtimes specify the infrastructure to run activities on. You can create three types of integration runtimes: *Azure*, *Self-Hosted*, and *Azure-SSIS*. Azure integration runtimes use infrastructure and hardware managed by Microsoft. Self-Hosted integration runtimes use hardware and infrastructure managed by you, so you can execute activities on your local servers and data centers. Azure-SSIS integration runtimes are clusters of Azure virtual machines running the SQL Server Integration (SSIS) engine, used for executing SSIS packages in Azure Data Factory.



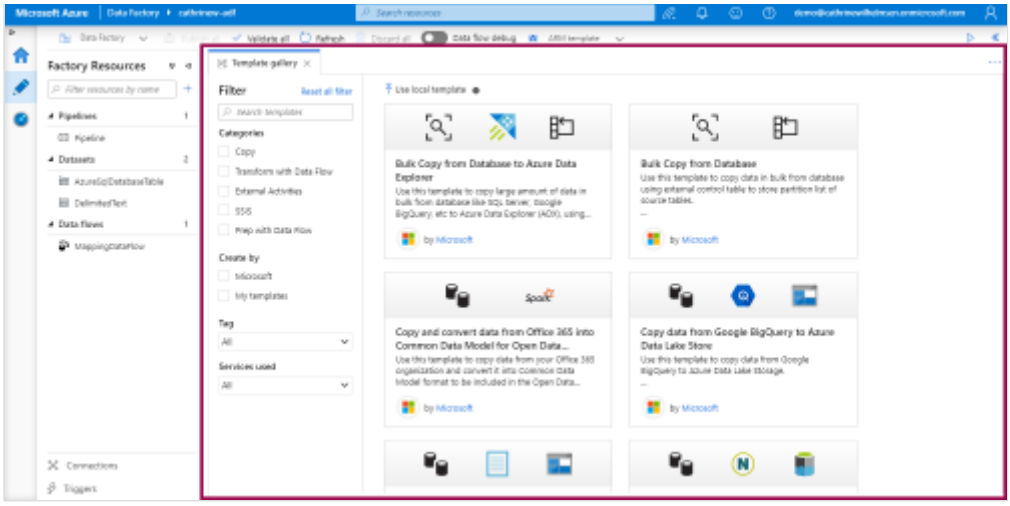
Triggers

Triggers determine when to execute a pipeline. You can execute a pipeline on a wall-clock schedule, in a periodic interval, or when an event happens.



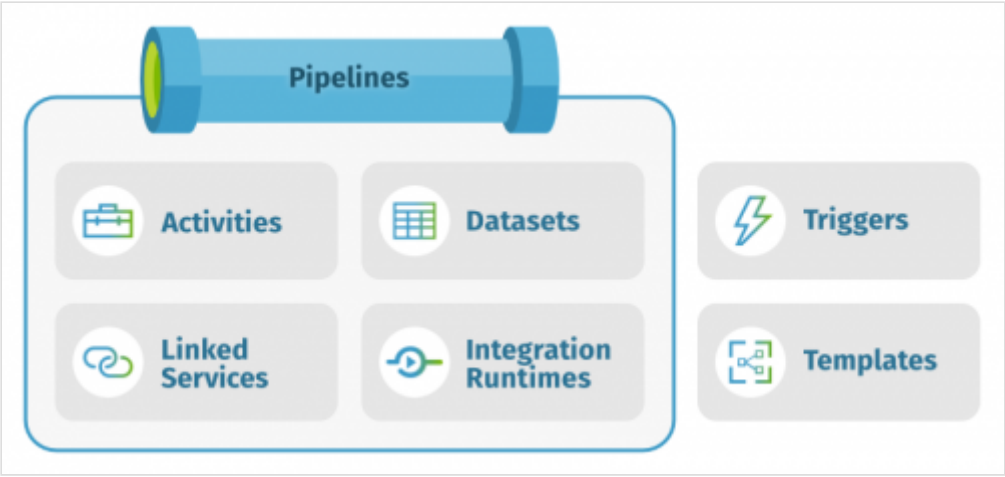
Templates

Finally, if you don't want to create all your pipelines from scratch, you can use the pre-



Summary

In this post, we went through the *Author* page in more detail and looked at the different Azure Data Factory components. I like to illustrate and summarize these in a slightly different way:



You create **pipelines** to execute one or more **activities**. If an activity moves or transforms data, you define the input and output format in **datasets**. Then, you connect to the data sources or services through **linked services**. You can specify the infrastructure and location where you want to execute the activities by creating **integration runtimes**. After you have created a pipeline, you can add **triggers** to automatically execute it at specific times or based on events. Finally, if you don't want to create your pipelines from scratch, you can start from pre-defined or custom **templates**.

Alrighty! Enough theory. Are you ready to *make things happen*? I am! Let's copy some data using the Copy Data Wizard :)

← Creating an Azure Data Factory

Copy Data Wizard in Azure Data Factory →

Related

Creating an Azure Data Factory

Copy Data Wizard in Azure Data Factory

Pipelines in Azure Data Factory

Dec 3, 2019 Data Platform Azure Data Factory

About the Author

Cathrine Wilhelmsen is a Microsoft Data Platform MVP, BimlHero Certified Expert, Microsoft Certified Solutions Expert, international speaker, author, blogger, and chronic volunteer who loves

teaching and sharing knowledge. She works as a Senior Business Intelligence Consultant at Inmeta, focusing on Azure Data and the Microsoft Data Platform. She loves sci-fi, chocolate, coffee, craft beers, ciders, cat gifs and smilies :)

Find me!

Subscribe to new posts?

E-mail

Yes, subscribe me!

Recent Posts

Sneaking back in… (2020 edition)

Keyboard shortcuts for moving text lines and windows (T-SQL Tuesday #123)

Speaking at NIC 2020

Azure Data Factory Training Day at SQLBits 2020

Personal Highlights from 2019

Popular Posts

Table Partitioning in SQL Server - The Basics

Parameters in Azure Data Factory

Preparing for and Taking Microsoft Exam DP-200 (Implementing an Azure Data Solution)

Table Partitioning in SQL Server - Partition Switching

Custom Power BI Themes: Page Background Images

Top Tags

Azure Data Factory Biml Certifications Don't Repeat Yourself Microsoft Ignite Notepad++ PASS Summit Personal Precon Speaking SQLBits SQLFamily SQLHangout SQLSatOslo SQLSaturday SQL Server SSIS T-SQL Tuesday Volunteering Webinar

All Categories

Select Category

Full Archive

Select Month