

Setting Up an Azure Data Factory Pipeline

Introduction

This is a complete documentation that provides a step-by-step guide on how to set up an Azure Data Factory pipeline to copy data from Azure Blob Storage to Azure SQL Database and also Blob Storage to Snowflake. By following this guide, you will be able to build a completely seamless integration between blob data, SQL Database, and Snowflake data warehouse.

Objectives & Requirements

The task focuses on improving how to display the life cycle of a lead. At the moment, each lead's status changes are recorded in a single row. The intention of this task is to migrate to Snowflake's 'LeadEvents' table, where each event in a lead's journey is uniquely recorded. To do this, Azure Data Factory will be used to transfer 100 leads from an Excel data source in SQL to Snowflake. The data will then be transformed by a Python project, offering a detailed, event-by-event snapshot of each lead's evolution. This improved representation attempts to increase the depth and clarity of the analysis.

A tutorial to build and deploy a pipeline into Azure Data Factory. For the complete pipeline, you need to have:

- An **Azure subscription account**, which will help you to build, maintain, and cost optimization.
- A **Blob storage account** is needed for the external data storage and internal transfer procedure.
- And a **Snowflake-privileged** account with all the authorization and authentication.

To complete this whole journey, the following requirements are fulfilled:

1. The pipeline is incremental.

Database

Columns:
10 total

+

Blob

Export data to DbtoBlobSink

Add Source

▼

Source settings

Source options

Projection

Optimize

Inspect

Data preview

Input

☒ Table

☐ Query

☐ Stored procedure

Batch size

Change data capture

☒

Type

☒ Incremental column

☐ SQL Server CDC

Column name

UpdatedDateUtc

Run mode

Full on the first run, then incremental

Isolation level

Read uncommitted

- 2. The pipeline is connected to the [GitHub](#)
- 3. The pipeline is easily extensible and maintainable.
- 4. The pipeline is sending emails on the failure of any activity within the pipeline. Alert:

Your Azure Monitor alert was triggered

Azure monitor alert rule EmailAlert was triggered for vptaskadf at August 6, 2023 23:47 UTC.

Rule ID	/subscriptions/35372e31-9514-48b6-af62-842d21fd1a5e/resourcegroups/DETask/providers/Microsoft.Insights/metricalerts/EmailAlert View Rule >
Resource ID	/subscriptions/35372e31-9514-48b6-af62-842d21fd1a5e/resourcegroups/DETask/providers/Microsoft.DataFactory/factories/vptaskadf View Resource >

Alert Activated Because:

Metric name	PipelineFailedRuns
Metric namespace	factories/vptaskadf
Dimensions	ResourceId = /SUBSCRIPTIONS/35372E31-9514-48B6-AF62-842D21FD1A5E/RESOURCEGROUPS/DETASK/PROVIDERS/MICROSOFT.DATAFACTORY/FACTORIES/VPTASKADF Name = VPTaskPipeline FailureType = UserError
Time Aggregation	Total
Period	Over the last 15 mins
Value	1
Operator	GreaterThan
Threshold	0
Criterion Type	StaticThresholdCriterion

See in the Azure portal >

- 5. The pipelines is using linked services to transfer the data between the storages. Services:

Linked services

Linked service defines the connection information to a data store or compute. [Learn more](#)

New

Filter by name

Annotations : **Any**

Showing 1 - 4 of 4 items

Name	Type	Related	Annotations
BlbSinkIncre	Azure Blob Storage	1	
DBLinkService	Azure SQL Database	5	
vaptaskblob	Azure Blob Storage	2	
vptasksnowflake_connection	Snowflake	1	

- 6. A trigger is attached to the pipelines and will trigger after 30 minutes. Trigger:

Edit trigger

Name *

VPTaskPTTrigger

Description

Type *

ScheduleTrigger

Start date * ⓘ

8/6/2023, 10:13:00 PM

Time zone * ⓘ

Coordinated Universal Time (UTC)

Recurrence * ⓘ

Every

30

Minute(s)



Specify an end date

End On * ⓘ

8/31/2023, 10:13:00 PM

Annotations



New

Status ⓘ



Started



Stopped

OK

Cancel

In summary, the following image will represent the current condition of the pipeline. And to build the same pipeline with the same architecture in the DB end, the documentation will help you to build a complete ADF pipeline.

The screenshot shows the Azure Data Factory Studio interface. At the top, there are tabs for 'main branch', 'Validate all', 'Save all', and 'Publish'. Below this, the 'Activities' pane on the left lists various activity types like 'Move & transform', 'Synapse', 'Azure Data Explorer', etc. The main canvas displays a pipeline named 'VPTaskPipeline' with three activities: 'Copy data' (Blob to DB), 'Data flow' (VPTaskFlow), and 'Copy data' (Bob to Snowflake). The pipeline status is 'Succeeded'.

Pipeline run ID: acf6aac2-a161-4e7c-b9a7-293ddda1b0aa

Pipeline status: Succeeded

Export to CSV

Activity name	Activity status	Activity type	Run start	Duration	Log	Integration runtime
Bob to Snowflake	Succeeded	Copy data	8/8/2023, 2:04:06 PM	57s		AutoResolveIntegr.
VPTaskFlow	Succeeded	Data flow	8/8/2023, 2:03:12 PM	53s		debugpool-8Cores
Blob to DB	Succeeded	Copy data	8/8/2023, 2:02:26 PM	45s		AutoResolveIntegr.

The rest of the scripts, python projects, and queries are available in the GitHub repository.

Table of Contents

1. Setting up an Azure Data Factory and a storage account
2. Configure SQL server and Database
3. Configure the Linked services In the ADF Studio
4. Configure Snowflakes linked service and build a connection
5. Build and Deploy the pipeline
6. Add trigger and alert services
7. Monitor

Let's jump into the hands-on experience to build a pipeline from scratch to deployment.

1. Setting up an Azure Data Factory

a. Data factory Creation

- Proceed to the [Azure Data factory](#) portal
- Click on "+ Create a resource" > "Search Services" > enter "Data Factory".
- Click on " Create", it will take you to the data factory to create a resource page
- Fill in the necessary fields (Subscription, Resource Group, Instance details, Git configuration etc.)
- For the Resource Group, Click on "+ Create a resource" > "Search for resource group" > "Give an appropriate name" > "Review + Create" to create the group
- Click "Review + Create", review your settings, and then click "Create".
- Follow the image for the clarification

Create Data Factory

[Basics](#) [Git configuration](#) [Networking](#) [Advanced](#) [Tags](#) [Review + create](#)

One-click to create data factory with sample pipeline and datasets. [Try it](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	<input type="text" value="Azure subscription 1"/>
Resource group *	<input type="text" value="Select your resource group or create new"/>

[Create new](#)

Instance details

Name *	<input type="text" value="Give an appropriate name"/>
Region *	<input type="text" value="East US"/>
Version *	<input type="text" value="V2"/>

[Basics](#) [Git configuration](#) [Networking](#) [Advanced](#) [Tags](#) [Review + create](#)

Azure Data Factory allows you to configure a Git repository with either Azure DevOps or GitHub. Git is a version control system that allows for easier change tracking and collaboration. [Learn more about Git integration in Azure Data Factory](#)

Configure Git later	<input type="checkbox"/> By default it will be enabled, disable it
Repository Type *	<input type="radio"/> Azure DevOps <input checked="" type="radio"/> GitHub
GitHub account *	<input type="text"/>
Repo name *	<input type="text"/>
Branch name *	<input type="text"/>
Root folder *	<input type="text" value="/"/>

[Basics](#) [Git configuration](#) [Networking](#) [Advanced](#) [Tags](#) [Review + create](#)

Managed virtual network

Choose whether you want the default AutoResolveIntegrationRuntime to be provisioned on demand inside an ADF-managed virtual network. If this setting is disabled, after the data factory is created, you can still choose whether to provision explicitly created Azure integration runtime inside an ADF-managed virtual network. [Learn more](#)

Enable Managed Virtual Network on the default AutoResolveIntegrationRuntime	<input type="checkbox"/> You need to enable this
---	--

Self-hosted integration runtime inbound connectivity to Azure Data Factory service

Choose whether to connect your self-hosted integration runtime to Azure Data Factory via public endpoint or private endpoint. This applies to self-hosted integration runtime running either on premises or inside customer managed Azure virtual network. [Learn more](#)

Create Data Factory

[View automation template](#)

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Basics

Subscription	Azure subscription 1
Resource group	DETask
Name	asdasdaw
Region	East US
Version	V2

[Previous](#) [Next](#) [Create](#) [Review and create](#)

b. Setting up a storage account

- Click on "+ Create a resource" > "Search for Storage Account" > "Create".
- Your resource group and subscription will be automatically added.
- Give your storage account an appropriate name (lowercase).
- Depending on your necessity, you can update "Advance", "Networking", "Data Protection", "Encryption", and "tags" or else use the default setup.
- Finally hit "Create".
- Follow along with the given image.

Create a storage account ...

Basics Advanced Networking Data protection Encryption Tags Review

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more about Azure storage accounts](#)

Project details

Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.

Subscription *

Resource group * [Create new](#)

Instance details

Storage account name ⓘ *

Region ⓘ * [Deploy to an edge zone](#)

Performance ⓘ * ☒ **Standard:** Recommended for most scenarios (general-purpose v2 account)
☐ **Premium:** Recommended for scenarios that require low latency.

Redundancy ⓘ *

- You need to set up a container also to hold your data in the storage account.
- Look for "Containers" under "Data Storage" > hit "+Container".
- Give an appropriate name for the container and hit "Create"
- Your container will create shortly.
- Go back to your storage account, select your newly created container, hit "Upload" from the top and upload your data file here

2. Configure SQL server and Database

a. Setting Up SQL Server in Azure

- Click on "+ Create a resource" > "Search for SQL Server" > "Create".
- Subscription: Choose your Azure subscription. By default, one will be selected.
- Resource Group: Select your existing resource group.
- Server Name: Choose a unique name for the server.
- Location: Select the nearest or preferred region.

- In the Authentication, select "Use SQL authentication".
- Server Admin Login & Password: Create an admin username and password for the server. Make sure to remember this; you'll need it to manage the server and databases.
- In the Networking tab, Allow Azure services and resources to access this server set to "Yes".
- You can keep the rest of the tabs on default settings. After that hit "Create" to deploy your server

Create SQL Database Server

Microsoft

...

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource group * ⓘ [Create new](#)

Server details

Enter required settings for this server, including providing a name and location.

Server name * .database.windows.net

Location *

Authentication

Select your preferred authentication methods for accessing this server. Create a server admin login and password to access your server with SQL authentication, select only Azure AD authentication [Learn more](#) using an existing Azure AD user, group, or application as Azure AD admin [Learn more](#), or select both SQL and Azure AD authentication.

Authentication method

☐ Use only Azure Active Directory (Azure AD) authentication
☐ Use both SQL and Azure AD authentication
☒ Use SQL authentication

Server admin login *

Password *

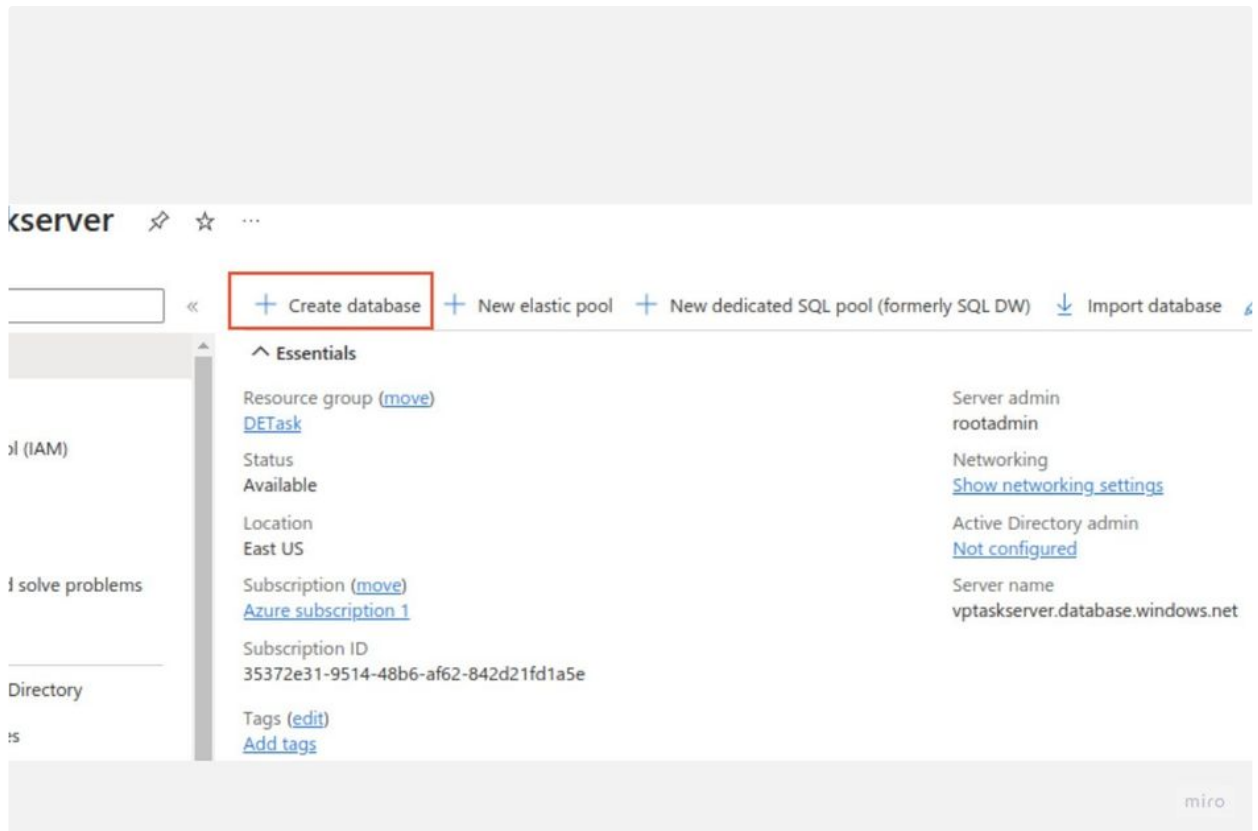
Confirm password *

[Review + create](#)

[Next: Networking >](#)

b. Configure your SQL Database

- Go to your newly created SQL Server.
- Click on "+ Create database".



■ Fill out the following:

- Database Name: Provide a name for your database.
- Server: Select the SQL server you created in the previous steps.
- Compute + storage: Choose the appropriate configuration based on your needs. For testing purposes, you can select "Standard." Although make sure to change the size of the Data max size depending on your need. Depending on your size, you will be billed monthly.
- Networking, Security, Additional settings, and Tags tabs can be kept as default
- Finally hit create and wait to finish the process.

Create SQL Database ...

Microsoft

Database details

Enter required settings for this database, including picking a logical server and configuring the compute and storage resources

Database name *

Server *

[Create new](#)

Want to use SQL elastic pool? ☐ Yes ☒ No

Workload environment ☐ Development ☒ Production

i Default settings provided for Production workloads. Configurations can be modified as needed.

Compute + storage *

General Purpose

Standard-series (Gen5), 2 vCores, 3 GB storage, zone redundant disabled

[Configure database](#)

Backup storage redundancy

Choose how your PITR and LTR backups are replicated. Geo restore or ability to recover from regional outage is only available when geo-redundant storage is selected.

Backup storage redundancy ☒ Locally-redundant backup storage ☐ Zone-redundant backup storage ☐ Geo-redundant backup storage

[Review + create](#)

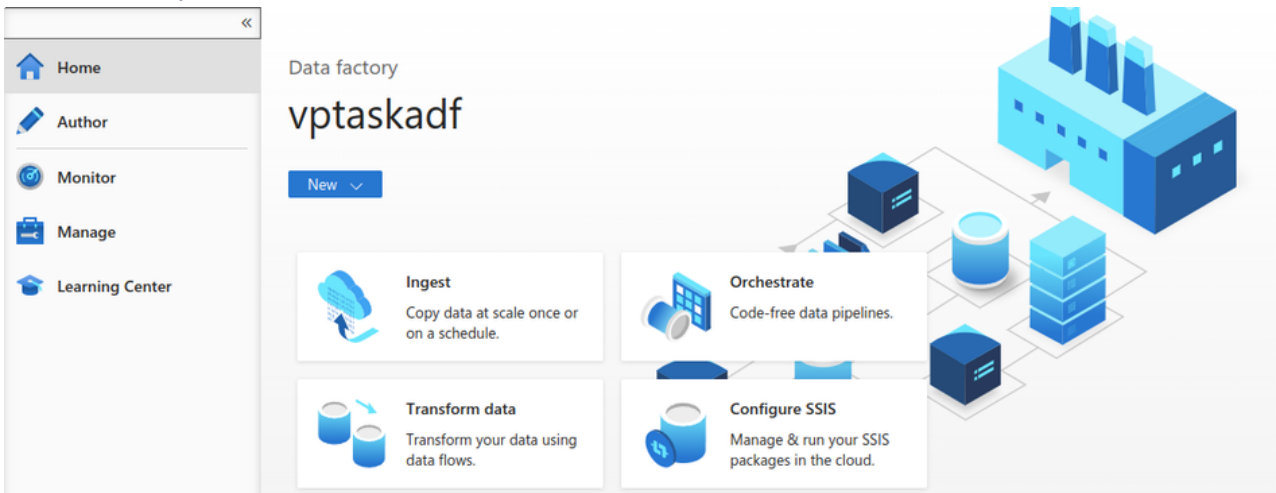
[Next: Networking >](#)

- **Additional Tips about Firewall: add it later**

Since we have configured most of the necessary configuration in the Azure portal, we are good to go to build our first pipeline. Go to your created Data factory and click **“Launch Studio”**

3. Configure the Linked services In the ADF Studio

a. Azure Data Factory Studio will look like this,

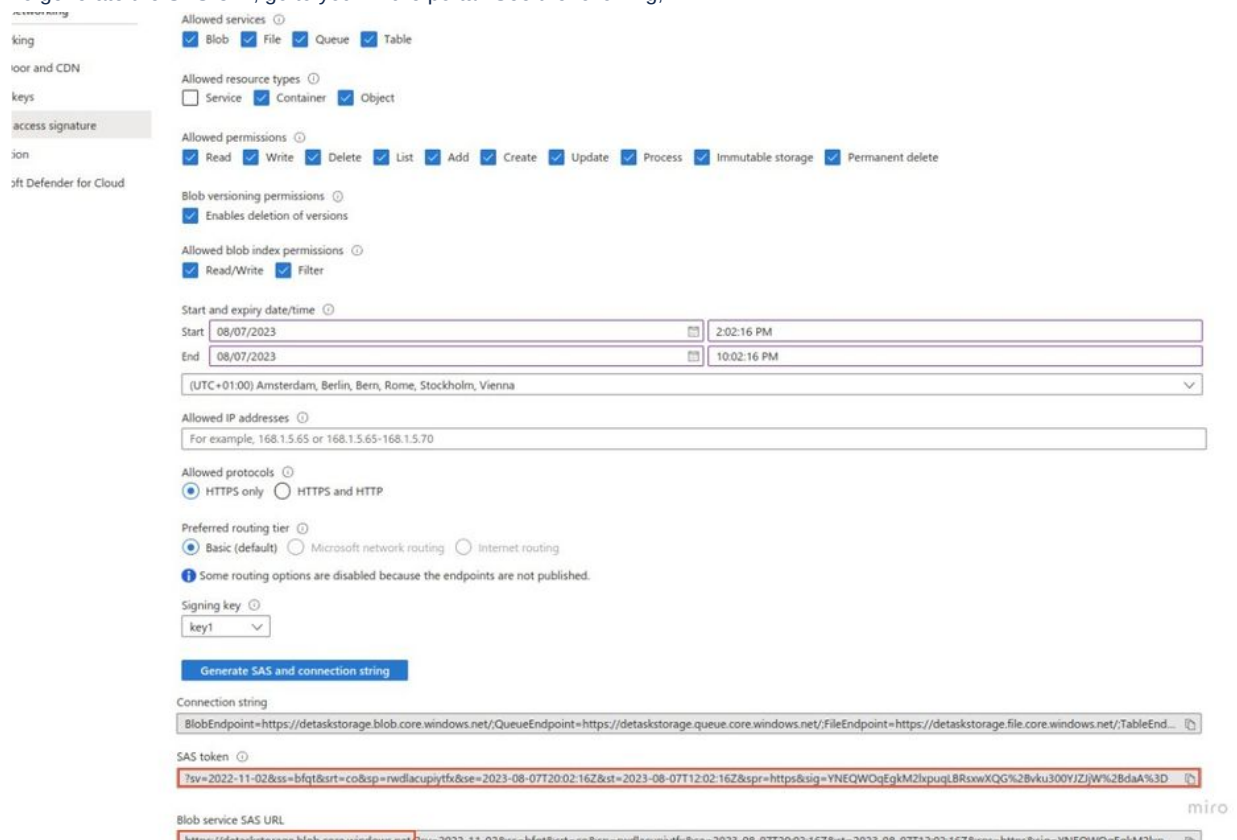


b. From the left side panel, click "Manage" > "Linked Services" > "+New" from the top. You need multiple linked services for the pipeline.

For the exemplified pipeline I have created 3 linked services, one for Blob Storage, one for SQL database, and one for Snowflakes

c. To configure Blob Storage linked service,

- Click "+New" > Search "Azure Blob Storage" > "Continue".
- Add a suitable name and description.
- In the Authentication Type, select "SAS URI". You have to provide "SAS URL" AND "SAS Token".
- To generate the SAS URI, go to your Azure portal. See the following,



The screenshot shows the 'New storage account' configuration page in the Azure portal. The left sidebar lists navigation options: Overview, Networking, Blob, File, Queue, Table, Storage and CDN, Keys, Access signature, and Blob Defender for Cloud. The 'Access signature' option is selected.

The main configuration area includes the following sections:

- Allowed services:** Blob, File, Queue, Table (all checked).
- Allowed resource types:** Service, Container, Object (all checked).
- Allowed permissions:** Read, Write, Delete, List, Add, Create, Update, Process, Immutable storage, Permanent delete (all checked).
- Blob versioning permissions:** Enables deletion of versions (checked).
- Allowed blob index permissions:** Read/Write, Filter (both checked).
- Start and expiry date/time:**
 - Start: 08/07/2023 2:02:16 PM
 - End: 08/07/2023 10:02:16 PM
 - Location: (UTC+01:00) Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna
- Allowed IP addresses:** For example, 168.1.5.65 or 168.1.5.65-168.1.5.70
- Allowed protocols:** HTTPS only (selected), HTTPS and HTTP
- Preferred routing tier:** Basic (default) (selected), Microsoft network routing, Internet routing. A note states: "Some routing options are disabled because the endpoints are not published."
- Signing key:** key1

A blue button labeled "Generate SAS and connection string" is located below the configuration options.

The generated output is displayed in a text area:

Connection string: BlobEndpoint=https://detaskstorage.blob.core.windows.net/;QueueEndpoint=https://detaskstorage.queue.core.windows.net/;FileEndpoint=https://detaskstorage.file.core.windows.net/;TableEnd...


SAS token: ?sv=2022-11-02&ss=bfqt&srt=co&sp=rwdlacupiyth&se=2023-08-07T20:02:16Z&st=2023-08-07T12:02:16Z&spr=https&sig=YNEQWQgEgkM2lxpuqlBRxswXQG%2Bvku300Y/ZjW%2BdaA%3D

Blob service SAS URL: https://detaskstorage.blob.core.windows.net/?sv=2022-11-02&ss=bfqt&srt=co&sp=rwdlacupiyth&se=2023-08-07T20:02:16Z&st=2023-08-07T12:02:16Z&spr=https&sig=YNEQWQgEgkM2lxpuqlBRxswXQG%2Bvku300Y/ZjW%2BdaA%3D

- Your SAS URL will be like this: <https://example.blob.core.windows.net/>
- Copy your token and save it somewhere. Before applying the changes, make sure to test your connection.

Edit linked service


 Azure Blob Storage [Learn more](#)

 To avoid publishing immediately to Data Factory, please use Azure Key Vault to retrieve secrets securely. [Learn more](#)

Name *

vaptaskblob

Description

Connect via integration runtime * 

AutoResolveIntegrationRuntime

Authentication type

SAS URI

SAS URI

[Azure Key Vault](#)

SAS URL * 

<https://detaskstorage.blob.core.windows.net/>

SAS token

[Azure Key Vault](#)

SAS token 

Test connection 

☒ To linked service ☐ To file path

Annotations


[+ New](#)

[Parameters](#)

[Advanced](#) 

Apply

Cancel

 Test connection

- If the connection is successful, click apply. Linked services will be created.

d. Configure your SQL Database linked service

- Click "+New" > Search "Azure SQL Database" > "Continue".
- Fill up the Name, and add a description if needed.
- Select "Enter manually" from the Account selection method, and give your created Database name, and your admin user name and password (you set it up when you configured your SQL server and setup your SQL authentication)
- Test your connection and apply the changes.

Description

Connect via integration runtime * ?
AutoResolveIntegrationRuntime

Connection string **Azure Key Vault**

Account selection method ?
☐ From Azure subscription ☒ Enter manually

Fully qualified domain name *
vptaskserver.database.windows.net

Database name *
CompaneLeadsDB

Authentication type *
SQL authentication

User name *
rootadmin

Password **Azure Key Vault**
Password *

Always encrypted ? ☐

Additional connection properties
+ New

Annotations

4. Configure Snowflakes linked service and build a connection

- Create an account in the [Snowflake](#) portal.
- Create a warehouse, database, and table to hold the data. You can execute these queries in the worksheet and your warehouse, database, and table will be created. Note down your Snowflake user id, user name, and password. You need this for later use

```

1  CREATE WAREHOUSE IF NOT EXISTS YourWareHouseName
2    WAREHOUSE_SIZE = 'XSMALL'
3    AUTO_SUSPEND = 600
4    AUTO_RESUME = TRUE;
5
6  CREATE DATABASE IF NOT EXISTS YourDbName;
7
8  CREATE TABLE YourDbName.PUBLIC.YourTableName (
9    Id INT,
10   Name STRING,
11   LikeThis VARCHAR(20)
12 );

```

- Once you have created everything, go back to your ADF Studio > Manage > Linked services
- Click "+New" > Search "Snowflake" > "Continue".
- Follow the image for the later steps,

Edit linked service


 Snowflake [Learn more](#) 

Name *

vptasksnowflake_connection

Description

This is a linked services that holds connection strings for snowflakes

Connect via integration runtime * 


AutoResolveIntegrationRuntime

Connection string

Azure Key Vault

Account name * 

jt14297.north-europe.azure


User name * 

shazadulalam

Password

Azure Key Vault

Password * 

Database * 

COMPANYLEADSDB


Warehouse * 


CompanyLeadsWarehouse

Role 

ACCOUNTADMIN

Additional connection properties

 Connection successful

 Test connection

Apply

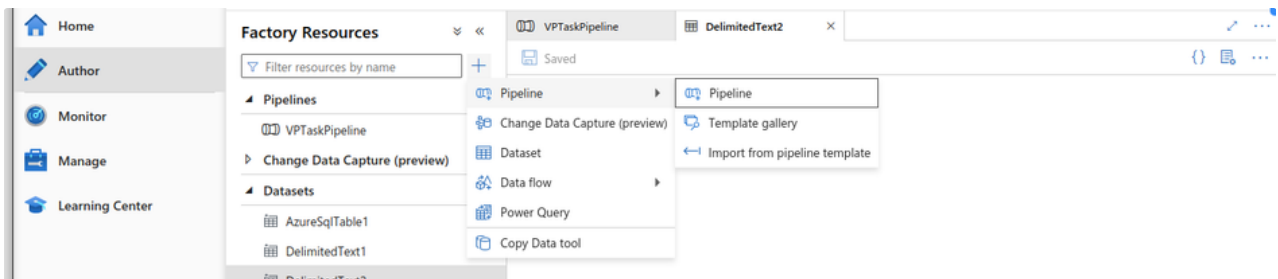
Cancel

- Your account name will be like this: {user_id}.north-europe.azure

Now the most exciting part of this tutorial documentation. You are good to go since you have configured all the necessary services.

5. Build and Deploy the pipeline

- Click Author from the left side panel of the studio, and click "+" > "Pipeline" > "Pipeline".



- Give a suitable name and click on it.
- You will see a panel like this,

Activities

Search activities

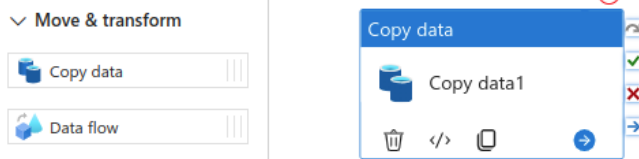
Move & transform

Copy data

Data flow

- > Synapse
- > Azure Data Explorer
- > Azure Function
- > Batch Service
- > Databricks
- > Data Lake Analytics
- > General
- > HDInsight
- > Iteration & conditionals
- > Machine Learning
- > Power Query

d. Drag “Copy data” to the right-side edit panel.



e. Click on it, give an appropriate name, and select your service on the source side and sink side.


General Source Sink Mapping Settings User properties

f. Map your data and save it

g. Add an additional DataFlow in the panel and double-click it. It will be like this:



The source and sink setup will be the same as copy data, the only difference will be on the source side. In the source option add these changes:



Source settings **Source options** Projection Optimize Inspect Data preview ●

Input ☒ Table ☐ Query ☐ Stored procedure

Batch size ⓘ

Change data capture ⓘ ☒

Type ☒ Incremental column ☐ SQL Server CDC

Column name ⓘ ▼

Run mode ▼

Isolation level ⓘ ▼

h. Now go back to the main panel. Add another copy of data for Snowflake

i. Debug your pipeline, if there is no error, publish it. Some error might occur due to the firewall, which addressed in the constraints and solutions

6. Add trigger and alert services

a. After publishing your pipeline add a trigger. To add a trigger click “Trigger” > “New or Edit” > add name and time (when should the trigger will be triggered)

Edit trigger

Name *

VPTaskPTTrigger

Description

Type *

ScheduleTrigger

Start date * ⓘ

8/6/2023, 10:13:00 PM

Time zone * ⓘ

Coordinated Universal Time (UTC)

Recurrence * ⓘ

Every

30

Minute(s)

☒ Specify an end date

End On * ⓘ

8/31/2023, 10:13:00 PM

Annotations

+ New


Status ⓘ

☒ Started ☐ Stopped

OK

Cancel

7. Monitor

- From the left side panel, click on the Monitor tab. Here you can monitor if your pipeline is running successfully or not. If there is any failure you can see from here. You can also add alerts and matrices from here.
- To add an email alert on failure, click " Alerts & metrics" > "New Alert Rule".

New alert rule

Alert rule name *

Description

Severity *

Target criteria * ⓘ

+ Add criteria

i There will be a monthly rate for the configured criteria. [Learn more about Pricing](#) 

Configure Email/SMS/Push/Voice notification * ⓘ

+ Configure notification

Enable rule upon creation



c. Fill up the details and click > “+Configure Notification”

Configure notification

Notify your team via email and text messages or automate actions using webhooks, runbooks, functions logic apps or integrating with external ITSM solutions.

☒ Create new ☐ Use existing

Action group name *

Short name *

Notifications *

+ Add notification

d. Click “+Add notification” > select “Email” > add your email address. Ther alert service will be created.

Add notification

Learn more about [Pricing](#) and [Privacy statement](#).

Action name *

NewAction

Select which notifications you'd like to receive

☐

Email

email@example.com

☐

SMS

Country code

1

Phone number *

1234567890

Carrier charges may apply.

☐

Azure app push notifications

Enter your email used to log into your Azure account. [Learn about connecting to your Azure resources using the Azure app.](#)

email@example.com

☐

Voice

Country code

1

Phone number *

1234567890

Add notification

Cancel

ⓘ You need SAS URI to connect from Blob to Snowflake. The default selected authentication type "Account key" will work only when you are copying data from Blob storage to SQL Database.

Constraints & Solutions

One of the biggest constraints will be when you will try to copy your data from Blob storage to your Snowflake database. You need to configure **Snowflake Integration Object**. To configure it, execute this query

```
1 CREATE STORAGE INTEGRATION azure_blob_integration
2 TYPE = EXTERNAL_STAGE
3 STORAGE_PROVIDER = AZURE
4 ENABLED = TRUE
5 AZURE_TENANT_ID = '<Azure_Active_Directory_Tenant_ID>'
6 AZURE_CLIENT_ID = '<Azure_Application_ID>'
7 AZURE_CLIENT_SECRET = '<Azure_Application_Secret>'
8 STORAGE_ALLOWED_LOCATIONS = ('azure://<your_storage_account>.blob.core.windows.net/<your_container>');
```

Replace the placeholder "< >" with your Blob storage information. You need Azure tenant_id, client_id, client_secret, storage_allowed_locations. These steps will provide you with the necessary information:

- Azure Active Directory (Azure AD) Application Registration

- Navigate to the [Azure Portal](#).
- In the left-hand navigation pane, click on "Azure Active Directory".
- Then, click on "App registrations" and then on "New registration".
- Fill in the name for the application, and under "Redirect URI", you can select "Web" and put a placeholder URL (like `http://localhost`), unless you know the specific URI you need.
- Click "Register".
- Within the application you just registered, on the "Overview" page, you can find:
 - **Application (client) ID:** This is your `Client ID`.
 - **Directory (tenant) ID:** This is your `Tenant ID`.
- In the application you registered, click on "Certificates & Secrets" in the left-hand menu.
- Click on "New client secret".
- Add a description for the client secret (e.g., "Snowflake Integration").
- Choose an expiration that fits your needs (e.g., 1 year, 2 years, or never).
- Click "Add".
- And for the storage-allowed locations, it will be your Blob SAS URL, like `<your_storage_account>.blob.core.windows.net/<your_container>/`, you don't need to use `https:` in here.
- After adding everything executes the query.
- You also need to create a stage that will refer to Azure Blob Storage in the Snowflake integration. Edit and run this query:

```
1 CREATE STAGE my_stage
2 URL = 'azure://<your_storage_account>.blob.core.windows.net/<your_container>/'
3 STORAGE_INTEGRATION = azure_blob_integration;
```

After you might stumble upon another error issue related to your client's IP address. Errors will look like this:

❌ Cannot connect to SQL Database. Please contact SQL server team for further support. Server: '`vptaskserver.database.windows.net`', Database: 'CompaneLeadsDB', User: 'rootadmin'. Check the linked service configuration is correct, and make sure the SQL Database firewall allows the integration runtime to access. Cannot open server 'vptaskserver' requested by the login. Client with IP address '20.42.3.136' is not allowed to access the server. To enable access, use the Azure Management Portal or run `sp_set_firewall_rule` on the master database to create a firewall rule for this IP address or address range. It may take up to five minutes for this change to take effect., SqlErrorNumber=40615,Class=14,State=1, Activity ID: af71423e-7e57-4403-be0b-3b1c42852bf9.

To avoid this issue, you need to add a range of IP addresses. For that use Azure CLI and execute this:

```
az sql server firewall-rule create --resource-group {Your Resource Group Name} --server {Your SQL Server name} --name AllowAzureDataFactoryRange --start-ip-address 20.42.2.100 --end-ip-address 20.42.2.200
```

Note that, the IP Address range should start from the IP address shown in the error message. Upon execution, you are good to go for the pipeline deployment.

Another way to add IP range is to add it from your **Azure Portal Firewall Rules**. This was the initial approach, where you add a rule directly via the Azure Portal interface. Ensure you're adding the rule at the SQL server level and not just the database level. Go to the SQL server's settings (the server that houses your `DATABASE`) in the Azure portal. Find the "Firewalls and virtual networks" option there. You should be able to add a rule for an IP address by giving the start IP and end IP.

Other documents

- In details task descriptions are given here: [PDF Data Engineer Task \(VP\).pdf](#)
- And this is the dataset: [Excel Data Enginner Task Dummy Data.xlsx](#)

