

# **VilleFutur: Projet d'Entrepôt de Données Météorologiques sur Azure**

**Présenté à : Abderrahim Eloutmadi**

**Realisé par : Faissal Moufla**

# Plan

- Introduction
- Conformité au RGPD
- Configuration des Comptes dans Azure
- Ingestion des Données
- Transformation et Intégration des Données
- Exécution du Pipeline
- Chargement des Données dans Azure Synapse Analytics
- Modélisation des Données et Scripting
- Conclusion

# Introduction

La ville de VilleFutur, une métropole en plein essor, fait face à des défis météorologiques uniques dans ses différentes régions. Afin de mieux comprendre ces variations climatiques et anticiper les besoins futurs en matière d'infrastructures, les autorités municipales ont lancé un projet robuste de gestion des données et d'ETL. Ce projet vise à centraliser les données météorologiques historiques et actuelles, permettant ainsi de prendre des décisions éclairées pour le développement de VilleFutur.

# Conformité au RGPD

Dans le contexte de notre projet, nos données se composent principalement de noms de régions, de dates et d'informations météorologiques. Même si nos données ne contiennent pas d'informations sensibles, nous accordons une grande importance à la conformité RGPD (Règlement général sur la protection des données). C'est pourquoi nous documentons scrupuleusement les types de données collectées, leurs sources et leurs finalités. Cette démarche assure une totale transparence quant à la manière dont nous traitons les données, garantissant ainsi la responsabilité de nos actions. Notre engagement envers la conformité RGPD reflète notre souci constant de protéger la vie privée des individus, même dans le cas de données non sensibles. Parallèlement, nous mettons en place des procédures de maintenance régulières, y compris la suppression des données obsolètes, pour préserver l'intégrité de nos données et garantir leur qualité continue.

# **Configuration des Comptes dans Azure**

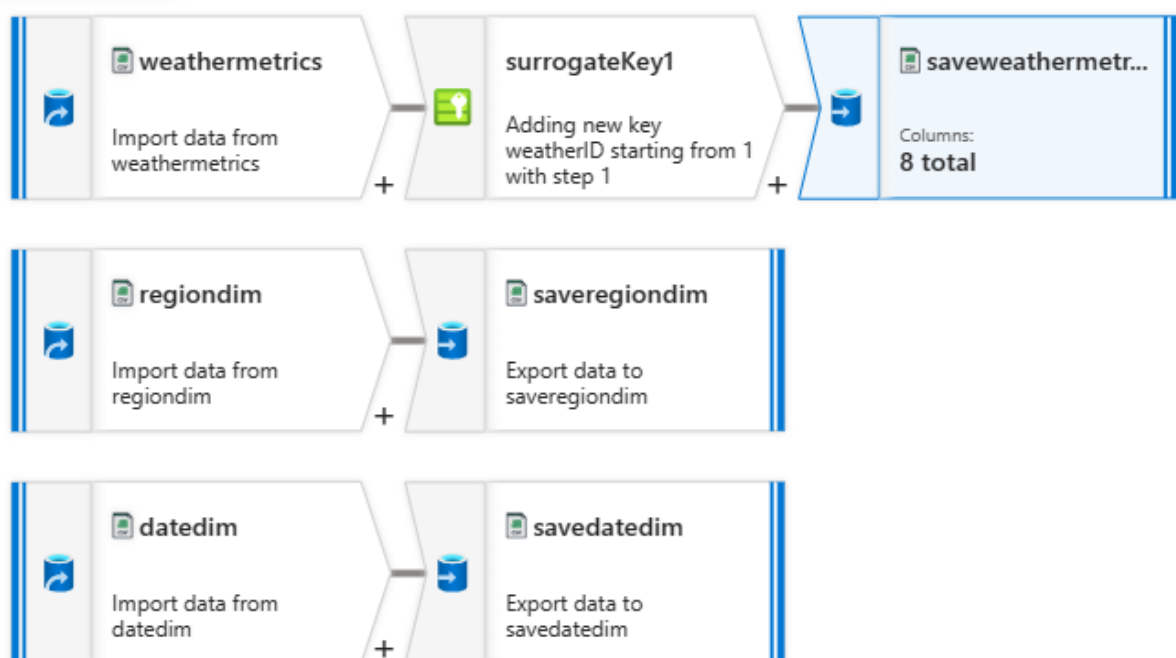
Dans la phase initiale du projet, des comptes Azure essentiels ont été établis. Cela comprenait Azure Storage pour le stockage des données, Azure Data Factory pour le traitement des données, Azure Synapse Analytics pour le data warehousing, et la création d'un pool dédié au sein de Synapse Analytics.

# Ingestion des Données

La collecte de données a débuté avec l'importation des données météorologiques dans le compte Azure Storage, dans le dossier "datasource". Ce dossier contenait trois fichiers CSV, chacun représentant des attributs de données spécifiques.

# Transformation et Intégration des Données

Azure Data Factory était l'outil principal pour la transformation et l'intégration des données. Un Data Flow a été créé pour traiter et nettoyer les données. Trois sources ont été configurées pour importer les fichiers CSV, avec des transformations de type de données effectuées dans les paramètres de Projection.



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

Define default format Detect data type Import projection Reset schema

Column name	Type	Format
DatelID	integer	Specify format
Date	date	yyyy-MM-dd

Une colonne ID unique a été générée pour le fichier WeatherMetrics CSV.

## Introduction

Save Validate Data flow debug Debug Settings

```
graph LR; A[weathermetrics  
Import data from weathermetrics] --> B[surrogateKey1  
Columns: 8 total]; B --> C[Reference: 0  
Columns: 8 total];
```

**Surrogate key settings** Optimize Inspect Data preview ●

Output stream name \*

surrogateKey1

Learn more

Description

Adding new key weatherID starting from 1 with step 1

Reset

Incoming stream \*

weathermetrics

Key column \*

weatherID

Start value \*

1

Step value

1



Une transformation Alter Row a été mise en place pour appliquer une Dimension de Changement Lent (SCD de Type 1) à la table RegDim, facilitant le suivi des données historiques. Les données nettoyées ont été enregistrées dans un dossier "cleaneddata" dans Azure Storage à l'aide d'un Sink.

The screenshot displays the Azure Data Factory (ADF) interface. On the left, the 'Factory Resources' pane shows a tree view with 'Pipelines' (1), 'Change Data Capture (preview)' (0), 'Datasets' (7), 'Data flows' (1), 'Transformations' (selected), 'Power Query' (0), and 'Templates' (0). The main canvas shows a data flow named 'VilleFutur' with three parallel paths: 1) 'weathermetrics' source connected to 'surrogateKey1' transformation and then to 'saveweatherme...' sink; 2) 'regiondim' source connected to '2 Columns' transformation and then to 'saveregiondim' sink; 3) 'datedim' source connected to 'savedatedim' sink. Below the canvas, the 'Alter row settings' tab is active for the 'Alter Row' transformation. The configuration includes: 'Output stream name' set to 'SCD' with a 'Learn more' link; 'Description' field with placeholder text 'Add expressions to alter rows' and a 'Reset' button; 'Incoming stream' set to 'regiondim'; and 'Alter row conditions' set to 'Upsert if' with the condition '1 == 1'.

Factory Resources

- Pipelines 1
  - VilleFutur
- Change Data Capture (preview) 0
- Datasets 7
- Data flows 1
- Transformations
- Power Query 0
- Templates 0

VilleFutur Transformations

Save Validate Data flow debug

weathermetrics surrogateKey1 saveweatherme...

regiondim 2 Columns saveregiondim

datedim savedatedim

Add Source

Alter row settings Optimize Inspect Data preview

Output stream name \* SCD [Learn more](#)

Description Add expressions to alter rows [Reset](#)

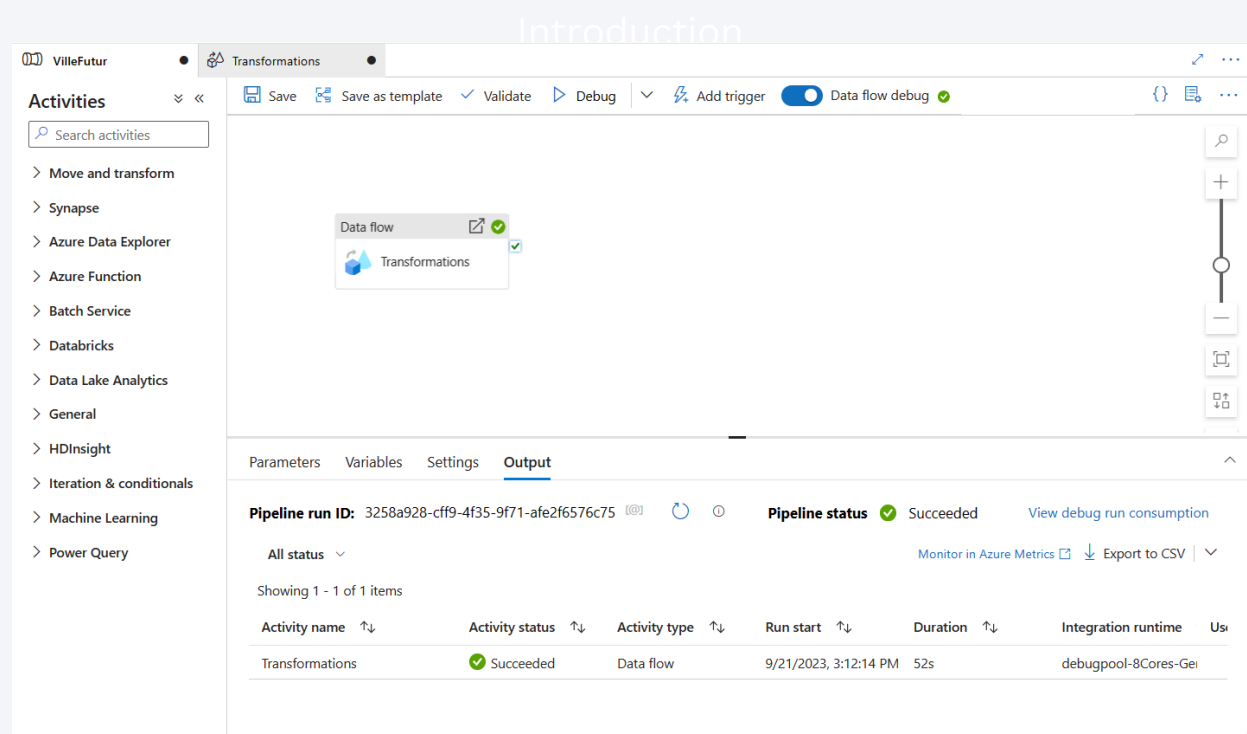
Incoming stream \* regiondim

Alter row conditions \* Upsert if 1 == 1

# Exécution du Pipeline

Un pipeline a été créé dans Azure Data Factory pour exécuter le Data Flow, permettant ainsi une exécution sans faille du processus de transformation des données.

Introduction



The screenshot displays the Azure Data Factory 'Transformations' tab. On the left, the 'Activities' pane lists various activity types. The main canvas shows a 'Data flow' activity named 'Transformations' with a green checkmark. Below the canvas, the 'Output' tab is active, showing the 'Pipeline run ID' as 3258a928-cff9-4f35-9f71-afe2f6576c75 and the 'Pipeline status' as 'Succeeded'. A table below lists the activity details:

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	Us
Transformations	✓ Succeeded	Data flow	9/21/2023, 3:12:14 PM	52s	debugpool-8Cores-Gei	

# **Chargement des Données dans Azure Synapse Analytics**

Azure Synapse Analytics, équipé d'un pool dédié, a été utilisé pour le data warehousing. L'activité "Copy Data" a été employée pour importer les données nettoyées depuis Azure Storage. Cette étape impliquait la configuration du pool dédié Synapse comme destination pour le stockage des données.

Synapse live Validate all Publish all

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Pipeline 1

Activities

Search activities

- Synapse
- Move and transform
- Azure Data Explorer
- Azure Function
- Batch Service
- Databricks
- Data Lake Analytics
- General
- HDInsight
- Iteration & conditionals
- Machine Learning

Copy data

Load weathermetrics

Copy data

Load regiondim

Copy data

Load datedim

Parameters Variables Settings Output

Showing 1 - 3 of 3 items

Activity name	Activity status	Activity type	Run start	Duration	Log
Load datedim	✓ Succeeded	Copy data	9/21/2023, 3:23:02 PM	15s	
Load regiondim	✓ Succeeded	Copy data	9/21/2023, 3:22:49 PM	13s	
Load weathermetrics	✓ Succeeded	Copy data	9/21/2023, 3:22:33 PM	15s	

## Introduction

data-factory branch Validate all Commit all Publish

Data

Workspace Linked

Filter resources by name

SQL database 1

- faissalmouflapool (SQL)
  - Tables
    - dbo.DateDim
    - dbo.RegionDim
    - dbo.WeathermetricsFact
  - External tables
  - External resources
  - Views
  - Programmability
  - Schemas
  - Security

SQL script 3

Run Undo Commit Query plan Connect to faissalmouflapool

```
1 select * from WeathermetricsFact
```

Results Messages

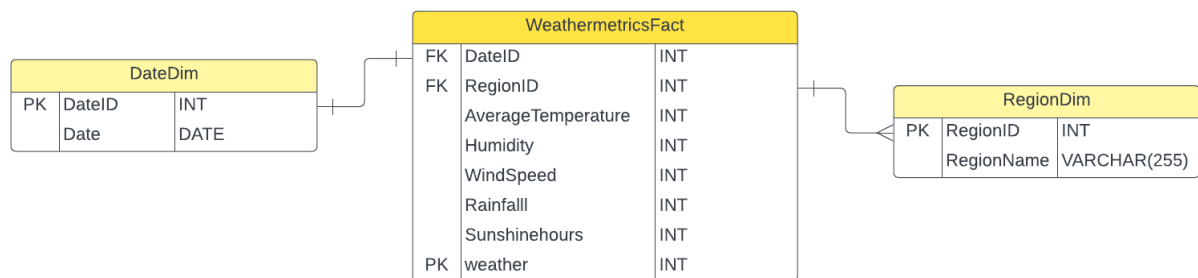
View Table Chart Export results

Search

DateID	RegionID	AverageTempe...	Humidity	WindSpeed	Rainfall	SunshineHours
1	1	21	98	5	2	0
43	1	0	92	0	2	2
84	4	21	1	17	0	15

00:00:00 Query executed successfully.

# Modélisation des Données et Scripting



## Introduction

Des scripts SQL ont été élaborés pour adapter la structure du data warehouse. Cela comprenait la définition des attributs et la mise en œuvre d'un schéma en étoile. De plus, des scripts ont été générés pour attribuer des clés primaires aux ID uniques au sein de la base de données.

data-factory branch   Validate all   Commit all   Publish

**Data**   Workspace   Linked

Filter resources by name

- Columns
  - DateID (nvarchar(max)...
  - RegionID (nvarchar(m...
  - AverageTemperature (...
  - Humidity (nvarchar(m...
  - WindSpeed (nvarchar(...
  - Rainfall (nvarchar(max)...
  - SunshineHours (nvarc...
  - weatherID (nvarchar(m...
- External tables
- External resources
- Views

**SQL script 1**

Run   Undo   Commit   Query plan   Connect to   faissalmoufflapool

```
1 -- Change the data types
2 ALTER TABLE DateDim
3 ALTER COLUMN DateID INT;
4
5 ALTER TABLE DateDim
6 ALTER COLUMN Date DATE;
7
8 ALTER TABLE RegionDim
9 ALTER COLUMN RegionID INT;
10
11 ALTER TABLE WeathermetricsFact
12 ALTER COLUMN DateID INT;
13 ALTER TABLE WeathermetricsFact
14 ALTER COLUMN RegionID INT;
15 ALTER TABLE WeathermetricsFact
16 ALTER COLUMN AverageTemperature INT;
```

**Properties**

General   Related (0)

Name \*  
SQL script 1

Description

Type  
.sql script

Size  
0 bytes

Results settings per query ⓘ  
☒ First 5000 rows (default)

Results   Messages

00:00:03 Query executed successfully.

## Introduction

We use optional cookies to provide a better experience. [Learn more](#)   Accept   Reject   More options

data-factory branch   Validate all   Commit all   Publish

**Data**   Workspace   Linked

Filter resources by name

- SQL database
  - faissalmoufflapool (SQL)

**SQL script 2**

Run   Undo   Commit   Query plan   Connect to   faissalmoufflapool

```
1 -- Change the 'DateID' column to non-nullable
2 ALTER TABLE DateDim
3 ALTER COLUMN DateID INT NOT NULL;
4
5 -- Add the primary key constraint
6 ALTER TABLE DateDim
7 ADD CONSTRAINT PK_DateDim PRIMARY KEY NONCLUSTERED (DateID) NOT EN
8
9 -- Change the 'RegionID' column to non-nullable
10 ALTER TABLE RegionDim
11 ALTER COLUMN RegionID INT NOT NULL;
12
13 -- Add the primary key constraint
14 ALTER TABLE RegionDim
15 ADD CONSTRAINT PK_RegionDim PRIMARY KEY NONCLUSTERED (RegionID) N
16
17 -- Change the 'weatherID' column to non-nullable
18 ALTER TABLE WeatherMetricsFact
19 ALTER COLUMN weatherID INT NOT NULL;
```

**Properties**

General   Related (0)

Name \*  
SQL script 2

Description

Type  
.sql script

Size  
0 bytes

Results settings per query ⓘ  
☒ First 5000 rows (default)  
☐ All rows

Results   Messages

00:00:01 Query executed successfully.

# Conclusion

En conclusion, ce projet complet de gestion des données et d'ETL a jeté les bases d'une meilleure compréhension des données météorologiques de VilleFutur. L'établissement des ressources Azure, le traitement méticuleux des données, l'exécution transparente du pipeline et le data warehousing structuré garantissent que les décideurs, chercheurs et urbanistes de VilleFutur ont accès à des données de haute qualité pour une planification future éclairée.