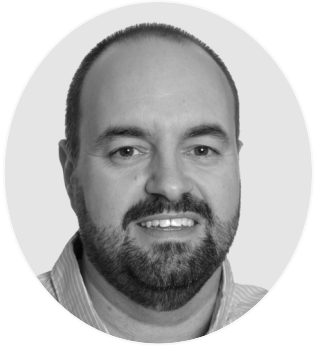
INTRODUCTION

Create Data Pipelines with Azure Data Factory



Marcelo Pastorino
SOFTWARE DEVELOPER / SOLUTIONS ARCHITECT

@evangeloper softwaredeveloper.io/marcelo

Azure Data Factory

IoT sensors generating thousands of events per day

The data they generate lacks context

We go from vague JSON files to a meaningful data set

Azure Data Factory

We need to transform and enrich these events

We also have to store it into an Azure SQL Server database

We need to create a pipeline using Azure Data Factory and Azure Databricks

Integrate Azure LogicApps into in the pipeline to send transactional emails

CLIP 1

Pipeline Parameters

We can parameterize Pipelines, Datasets and Linked Services

By passing dynamic values

Parameters = Reusability

Pipeline Parameters

We create parameters to define the name of Azure Blob Storage containers

CLIP 2
NO SLIDES

CLIP 3
NO SLIDES

CLIP 4

Creating a Reusable Dataset

We can use parameters to create reusable components

By parameterizing this Dataset, we can access files in different Blob Storage containers

CLIP 5

ForEach Activity

We need to move files between 2 storage containers

Sensor Sink RAW -> Sensor Sink Staging

ForEach Activity

A staging container is used to guarantee that we process files only once

We cannot delete all files at once from Sensor Sink RAW container

This container may receive new files as we are running the pipeline

Get a snapshot of files using the Get Metadata activity

Then move files one by one and preserve new ones

ForEach Activity

Defines a repeating control flow in a pipeline

Similar to ForEach statement found in programming languages

Used to iterate over a collection and to execute activities in a loop

Moving Blob Storage Container Files in ADF

No native way to move files

Move = Copy + Delete

Workaround ForEach + Copy Data + Delete

Iterates thru files then copy and delete one at a time

ADF has a predefined template

ForEach Activity

Executing sub-activities inside a ForEach activity results in a higher number of running activities

This increases the total cost of running the pipeline

Exact number of activities directly related to the number of items in the collection

Familiarize yourself with the Azure Data Factory pricing model

Use the architectural design that best fits your needs

CLIP 6

Azure Databricks

An IoT sensor reading contains 4 properties that do not have much context

We want to transform and enrich it during the pipeline execution

The company wants to determine air pollution levels in a given country but want to work with a friendlier format

We need to use Azure Databricks to transform data

Azure Databricks

Azure Databricks provides a managed Spark-based analytics service in the Azure platform

We can transform, enrich, and process massive amounts of data

We use Azure Databricks in our pipeline, to transform, enrich, and store our IoT sensor events

Learn More About Azure Databricks

https://azure.microsoft.com/en-us/services/databricks/

CLIP 7

Databricks Notebooks

A notebook is a document that contains runnable code, visualizations, and text

It's a way of interacting with Databricks infrastructure

Download the Sample Databricks Notebook

https://github.com/evangeloper/pluralsight-integrating-data-azure/blob/master/databricks/TransformEnrichPollutionData.dbc?raw=true

Transform and Enrich Sensor Events Data

- sensor-sink-stage contains unprocessed IoT sensor events
- One event per file in JSON format
- Combine data from JSON files into a big data set
- Transform and enrich events with reference data stored in Azure SQL
- Store processed events in an Azure SQL database called SensorReadings

Learn More About Azure Databricks

https://azure.microsoft.com/en-in/services/databricks

CLIP 8

Send Transactional Emails from Data Factory

The company wants to receive an email notification when the pipeline runs

We need to integrate a service to our pipeline

The solution is Azure LogicApps

Azure LogicApps

A service that helps us automate and orchestrate workflows

Workflows start with a trigger, which fires when a specific event happens

It runs actions as a response to the trigger

Learn More About Azure LogicApps

https://azure.microsoft.com/en-us/services/logic-apps

CLIP 9
NO SLIDES

CLIP 10
NO SLIDES

CLIP 11
NO SLIDES

CLIP 12

Data Factory Pipelines

Extract and load in Data Factory

Transformations in Azure Databricks notebook

New Azure Data Factory Features

Mapping Data Flows
(Public Preview)

Wrangling Data Flows
(Limited Private Preview)

Mapping Data Flows

Provide a visual experience to develop data transformation logic in the cloud without writing any code

A natural progression to SSIS that exists within Azure Data Factory

Mapping Data Flows

Perform native data transformations

- Data cleaning
- Aggregation

Automatically translates processes to code that runs on Azure Databricks clusters

Mapping Data Flows

Currently offers over 10 different data set manipulation operations

- Branch
- Join
- Conditional splits

Wrangling Data Flows

Visual tool that allows us to examine and model datasets

Helps make them more suitable for a variety of downstream purposes

Mapping Data Flows or Wrangling Data Flows

Wrangling Data Flows is about data preparation

Mapping Data Flows is about data transformation

Learn More About Mapping Data Flows

https://docs.microsoft.com/en-us/azure/data-factory/concepts-data-flow-overview

CLIP 13

Creating Azure Data Factory Resources Programmatically

Powershell

Python

C#

Azure REST API

ARM templates

Creating Resources Programmatically in Powershell

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-powershell

Creating Resources Programmatically in .NET

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-dot-net

Creating Resources Programmatically in Python

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-python

Creating Resources Programmatically with the Azure REST API

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-rest-api

Creating Resources Programmatically with ARM Templates

https://docs.microsoft.com/en-us/azure/data-factory/quickstart-create-data-factory-resource-manager-template

CLIP 14

Azure Data Factory

- Over 80 prebuilt connectors
- Azure services
- Databases
- No-SQL data stores
- File servers
- Other types of services and apps

Azure Connectors

- Azure Cosmos DB
- Azure Data lake, gen 1 and 2
- Azure Database for Maria DB
- MySQL and Postgre
- Azure table storage

Database Connectors

- Amazon Redshift
- DB2
- Google BigQuery
- HBase
- Hive
- Oracle
- SAP databases

NO-SQL Connectors

Cassandra

Couchbase

MongoDB

File and Other Storage Systems

- Local files systems
- FTP servers
- Google Cloud Storage
- HDFS

Protocols

HTTP

OData

ODBC

REST

Apps and Services

Inside and outside the Azure ecosystem

Dynamics 365

Google AdWords

Jira

Magento

Office 365

SAP services

Azure Data Factory Supported Data Stores

https://docs.microsoft.com/en-us/azure/data-factory/copy-activity-overview#supported-data-stores-and-formats

SUMMARY

Summary

We transformed and enriched data

Learned about new pipeline activities

Created reusable components

Used Azure Databricks

Integrated Data Factory with Azure LogicApps

Triggered our pipeline on a schedule

Discovered Mapping and Wrangling Data Flows