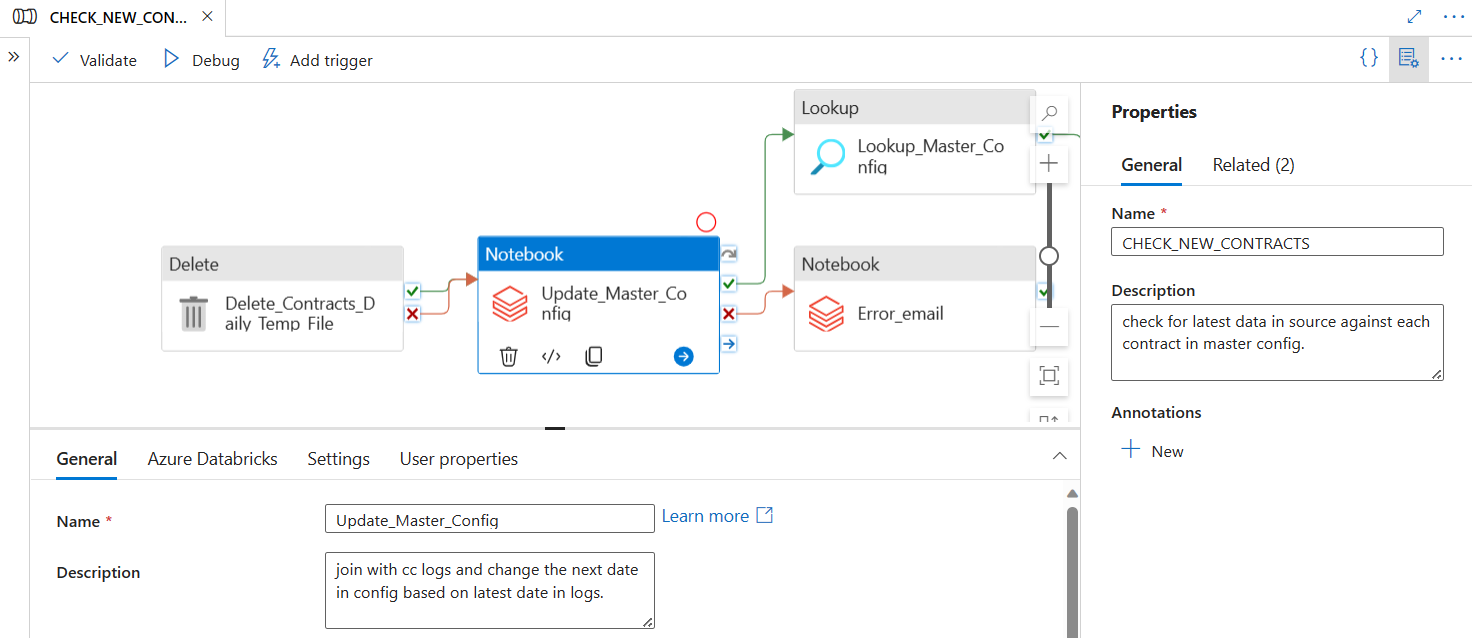
# How to use ADF to make Ops-friendly data pipelines

* Author: Muhammad Muneeb

Azure Data Factory (ADF) is one of the most effective technologies available for creating cloud data pipelines within the Azure ecosystem. To make the most of it, like with anything else, you need to approach it thoughtfully. Although methods differ from project to project, certain practices always apply. Here are some suggestions for best practices to follow, when preparing to use Azure Data Factory.

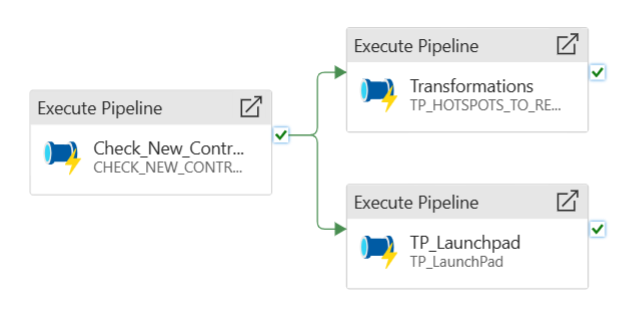
## Naming Convention

Adherence to standard naming conventions of activities and pipelines really helps with debugging, event logging, job monitoring, and CI/CD. In larger development teams, where multiple people may contribute to the same data factory, consistent naming standards create a common language that makes it simple to understand the role and goal of each activity and pipeline. So always use standard and clear naming conventions for both activities and pipelines, and also try to include a short description if needed for other developers and stakeholders.



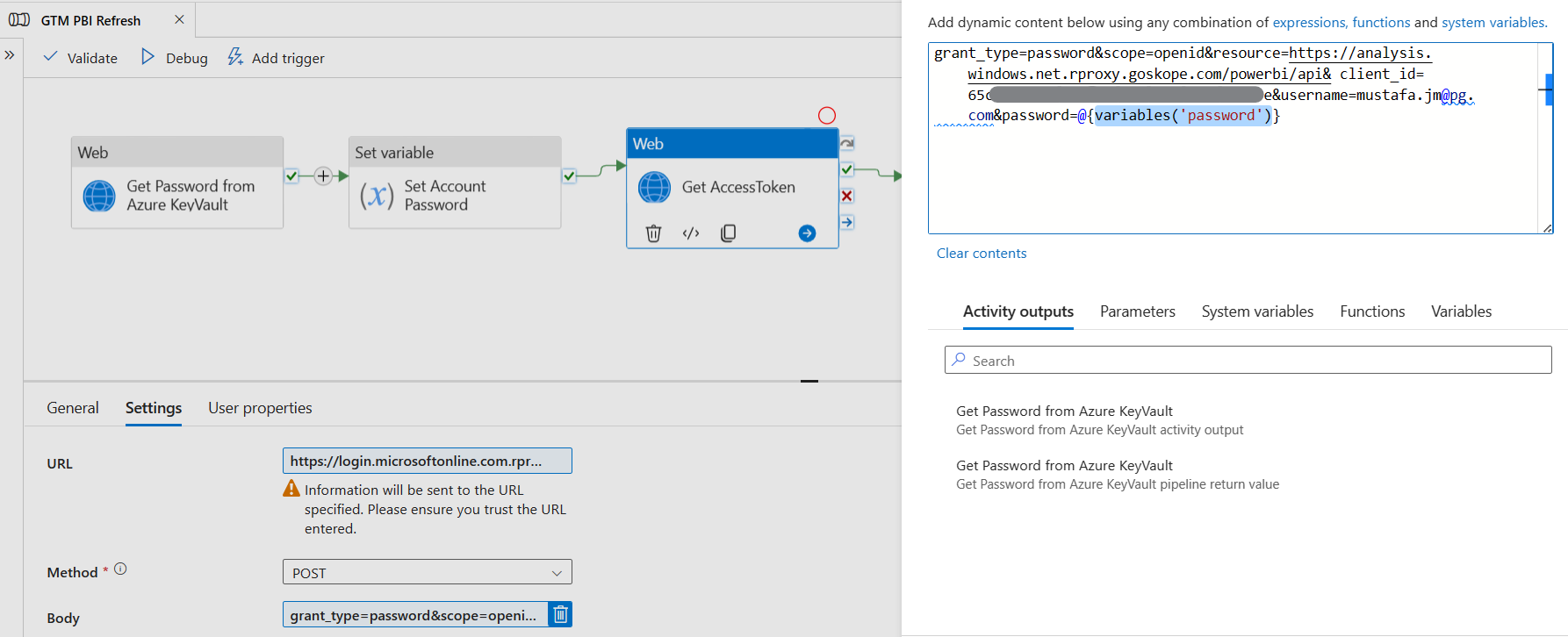
## Build Smaller / Reusable Components

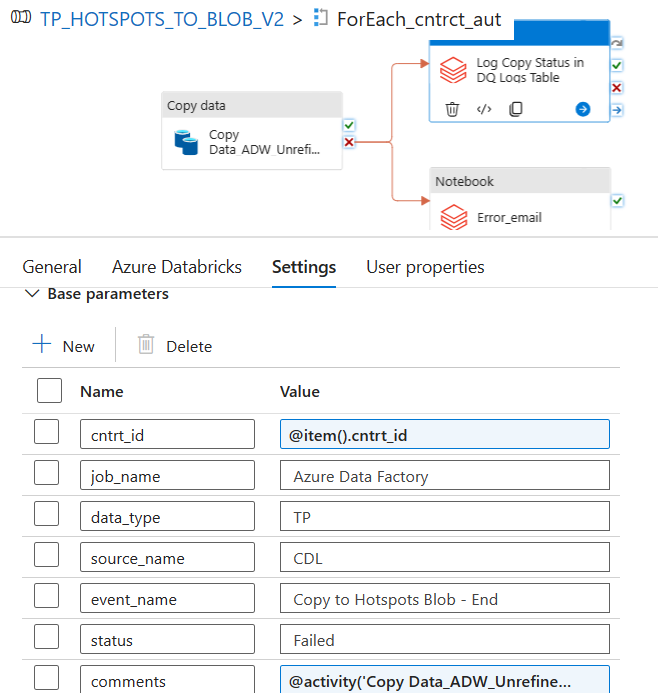
While using Azure Data Factory (ADF), always try to break lengthy flows into reusable and smaller components, which results in better scalability, maintainability, and efficiency. By breaking down complex processes into smaller, more manageable components, you can improve the readability of the architecture to facilitate troubleshooting and debugging issues when they arise. Smaller components can also be reused in other pipelines, which encourages the same consistency across the system and eliminates the need for duplicating components. This reusability guarantees that updates or changes may be deployed consistently throughout the system and speeds up development cycles. Smaller components also make collaborative development easier as more individuals can work on various components at the same time.



## Use Parameters and Variables

For data workflows to be more flexible, reusable, and maintainable, parameters and variables in Azure Data Factory (ADF) must be used. With parameters, pipeline components can be dynamically configured, making it more flexible and simple to change settings without altering the pipeline itself. On the other hand, variables make it easier to temporarily store data during runtime, which allows for conditional logic and dynamic transformations in pipelines. In order to reduce code redundancy and streamline maintenance, parameters and variables are used to build more flexible and reusable pipelines.





## Avoid Scalable Transformations

Scalable transformations in ADF should always be avoided because this tool isn't best suited for complex and scalable workflows. While Azure Data Factory and Azure Databricks are both strong tools within the Azure ecosystem, it is very crucial to know their respective functions and strengths for processing and analyzing data. The particular requirements of your analytics and data processing workflows will determine which one of Azure Data Factory and Azure Databricks to use. Consider factors like cost, complexity, and workload demands when evaluating the scalability requirements of your data workflows.

Azure Databricks is frequently picked for scenarios where advanced analytics, distributed computing, and scalability are critical requirements. On the other hand, Azure Data Factory is a data integration service that focuses on orchestrating data workflows, automating data movement and scheduling tasks rather than advanced analytics and complex data processing, making it a valuable component within a comprehensive data solution when used in conjunction with other services like Azure Databricks. It is common to see both services used together in data pipelines, with Azure Data Factory orchestrating the workflow and Azure Databricks performing the advanced analytics and processing tasks.

## Event Logging and Monitoring

As essential components of the Azure Data Factory, event logging and job monitoring offer vital insights into the performance and execution of data workflows. Data engineers and administrators can improve the overall effectiveness, performance, and health of their data workflows in Azure Data Factory by utilizing both event logging and job monitoring. Azure Data Factory makes use of Azure Monitor and Azure Log Analytics to enable users to create alerts that are triggered by particular events or performance thresholds, ensuring early problem detection and resolution.

TBA : (Azure Monitor)