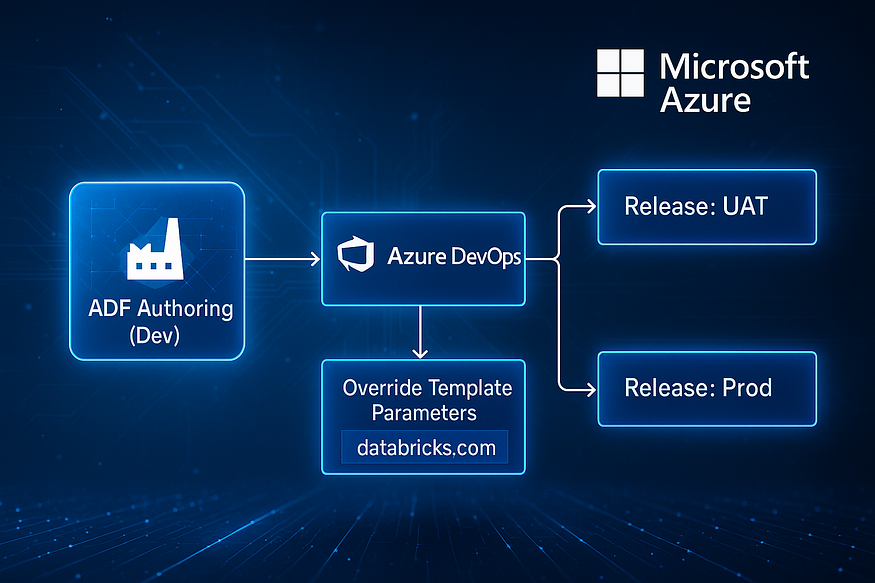
**Azure Data Factory (ADF) CI/CD Using Azure DevOps with Environment-Specific Parameters**

**Introduction**

* Manual deployments are slow, error-prone, and risky.
* CI/CD ensures automated, safe, and environment-specific deployments for ADF pipelines.
* Focus: Use Azure DevOps + ADF + custom parameters (like Databricks URL).



**What is Azure Data Factory (ADF)?**

* ADF = Microsoft’s **cloud-native ETL/ELT service**.
* Key Benefits:
  + Connects to 90+ sources.
  + Works with Databricks, Synapse, Azure Functions, etc.
  + Low-code drag-and-drop UI.
* Ideal for enterprises modernizing data pipelines to the cloud.

**Why CI/CD in ADF?**

* Manual ARM uploads = risky.
* CI/CD Benefits:
  + Automated promotion from Dev → UAT → Prod.
  + Environment-specific configs (endpoints, usernames).
  + Change tracking & rollback via Git.
  + Prevents errors like deploying wrong linked services.

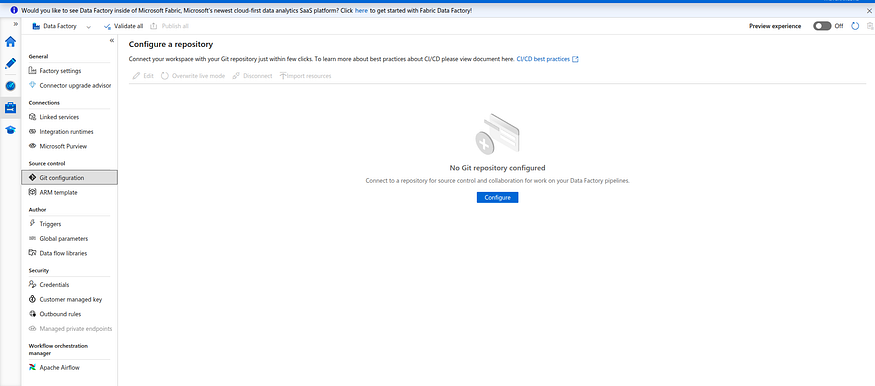
**ADF + Azure DevOps CI/CD Architecture**

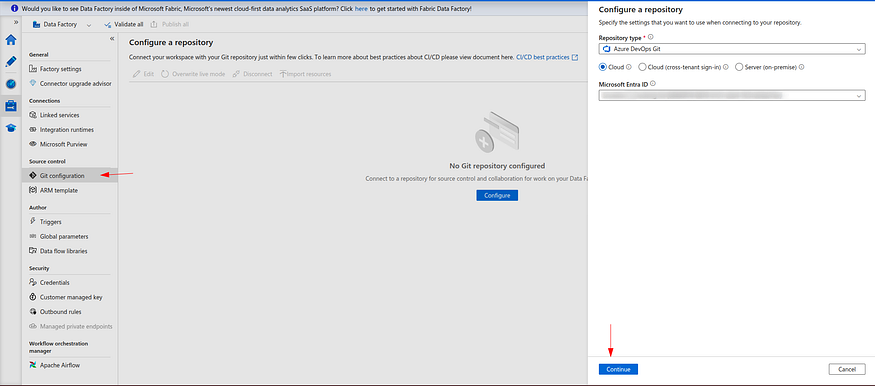
* **Branches:**
  + main → Development branch.
  + adf\_publish → Auto-generated when ADF pipelines are published.

**Step-by-Step Setup in Azure DevOps**

**Step 1: Git Integration**

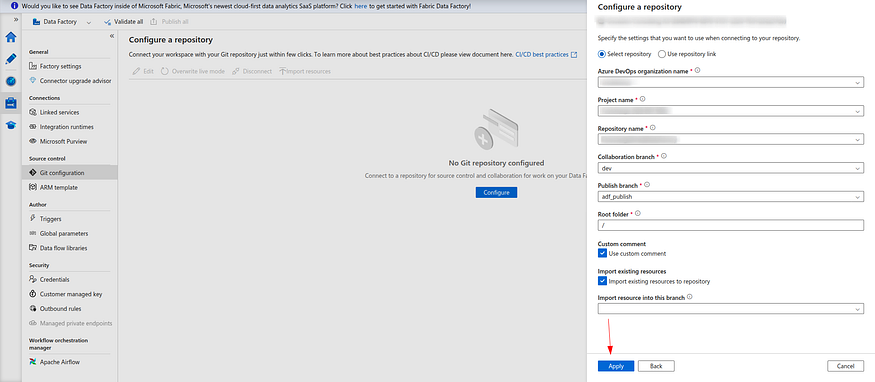
* Configure Git in ADF Studio.
* Use Azure DevOps Git repo.
* Collaboration branch: main, Publish branch: adf\_publish.
* Publishes ARM templates automatically into Git.

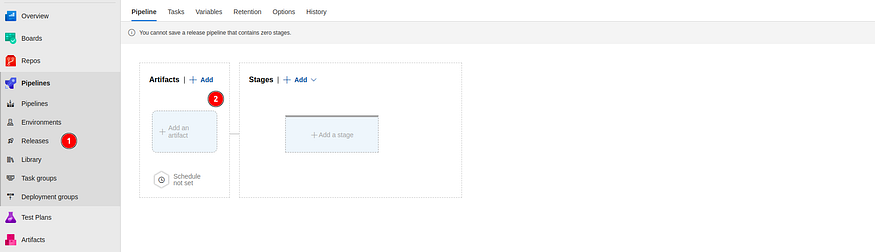


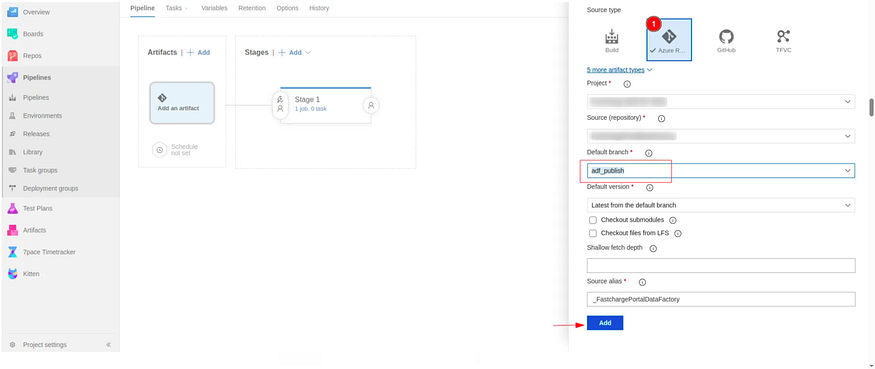


**Step 2: Release Pipeline + Approvals**

* Create release pipeline in Azure DevOps.
* Add ARM Template Deployment task.
* Configure deployment scope (Resource Group).
* Use ARMTemplateForFactory.json and ARMTemplateParametersForFactory.json.
* Override environment-specific parameters (DB server, endpoints, etc.).
* Deployment mode: Incremental.

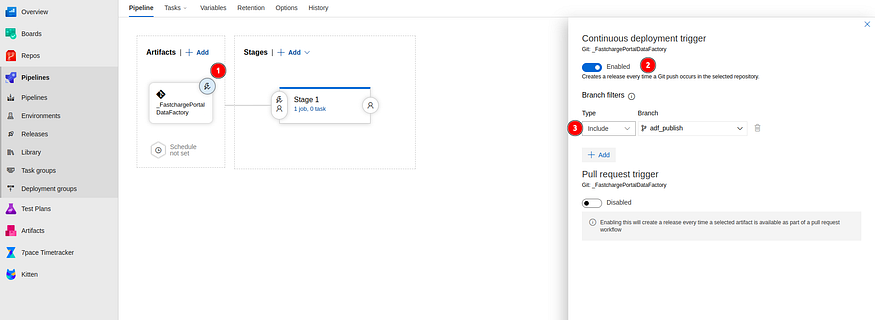




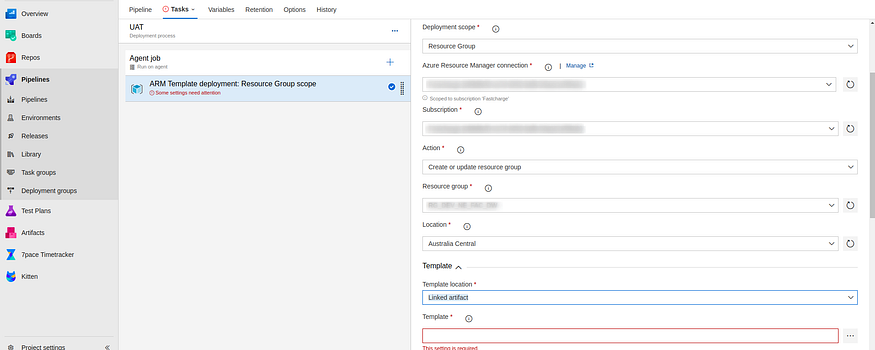


**Step 3: Parameter Overrides**

* Parameterize Linked Services (e.g., Databricks URL).
* Modify ARM template JSON to allow environment-specific overrides.
* Dev → Publish → Pipeline injects correct URL for UAT/Prod.



Press enter or click to view image in full size



Got To Azure DevOps → Release Pipeline → Stage → Azure Resource Group Deployment → Override Template Parameters

