MLOps Maturity Model Architecture with Azure Machine Learning

thumbnail image 10 of blog post titled 
 
 
  
 
 
 
    
  
   
    
      
       MLOps Maturity Model with Azure Machine Learning
       
      
     
   
  
 
   
 
 
 
 
 


*Introduce* Azure ML and MLflow

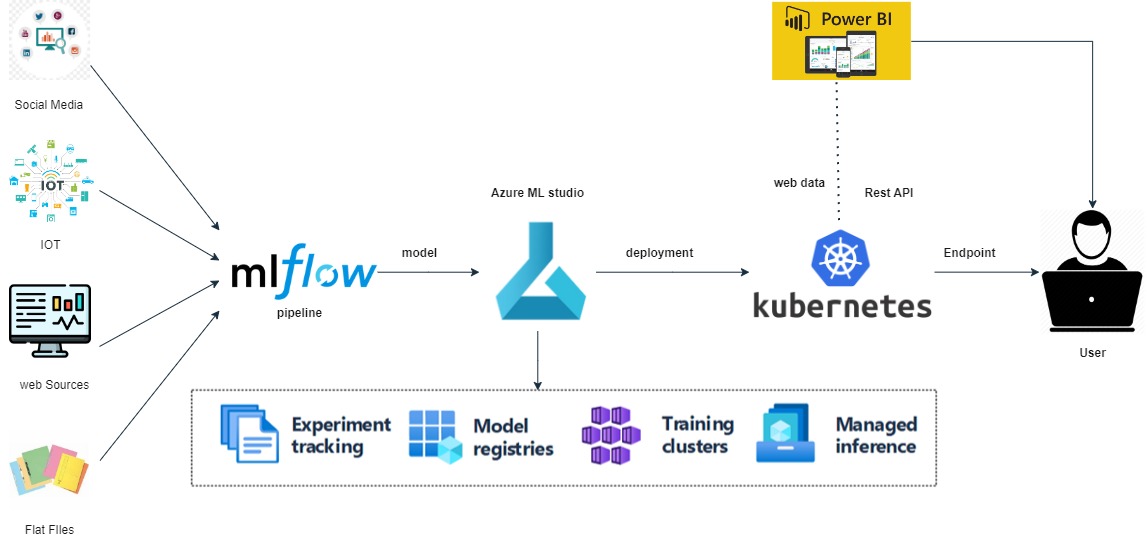
Azure Machine Learning is an open-source friendly, machine learning platform that can be used to implement full machine learning lifecycle and MLOps through integration with GitHub (or Azure DevOps) and Responsible AI technologies which support you to develop, use and govern AI responsibly.

MLflow is an open-source framework, designed to manage the complete machine learning lifecycle. Its ability to train and serve models on different platforms allows users to avoid vendor lock-ins and to move freely from one platform to another one.

*Azure Machine Learning Workspace and assets*

Azure Machine Learning Workspace is the top-level resource of Azure Machine Learning. Azure Machine Learning Workspace leverages Azure Storage Account, Azure Container Registry, Azure Key Vault, Azure Application Insights, and related Azure services depending on your requirements. Here is a list of Azure Machine Learning assets we want you to understand :

* Data
  + connect to Azure storages and manage subset of data.
* Job (Experiments and Runs)
  + handle job requests and execute it.
  + manage metrics and logs in the job.
* Model
  + register and manage trained model with metadata.
* Environment
  + create and manage runtime (Docker Image) for training and inference.
* Component
  + create and manage a self-contained piece of code that does one step in a machine learning pipeline.
* Pipeline
  + create and manage reproducible machine learning workflow.
* Endpoint
  + create and manage inference environments like Managed Endpoint.

Mlflow Architecture  


MLflow

* Tracking, allowing experiments to record and compare parameters, metrics, and results.
* Model’s Registries, supplying a central model store to collaboratively manage the full lifecycle of a model.
* Projects, allowing to package ML code in a reusable, reproducible form to share with other data scientists or transfer to production (on preview on Azure Machine Learning).

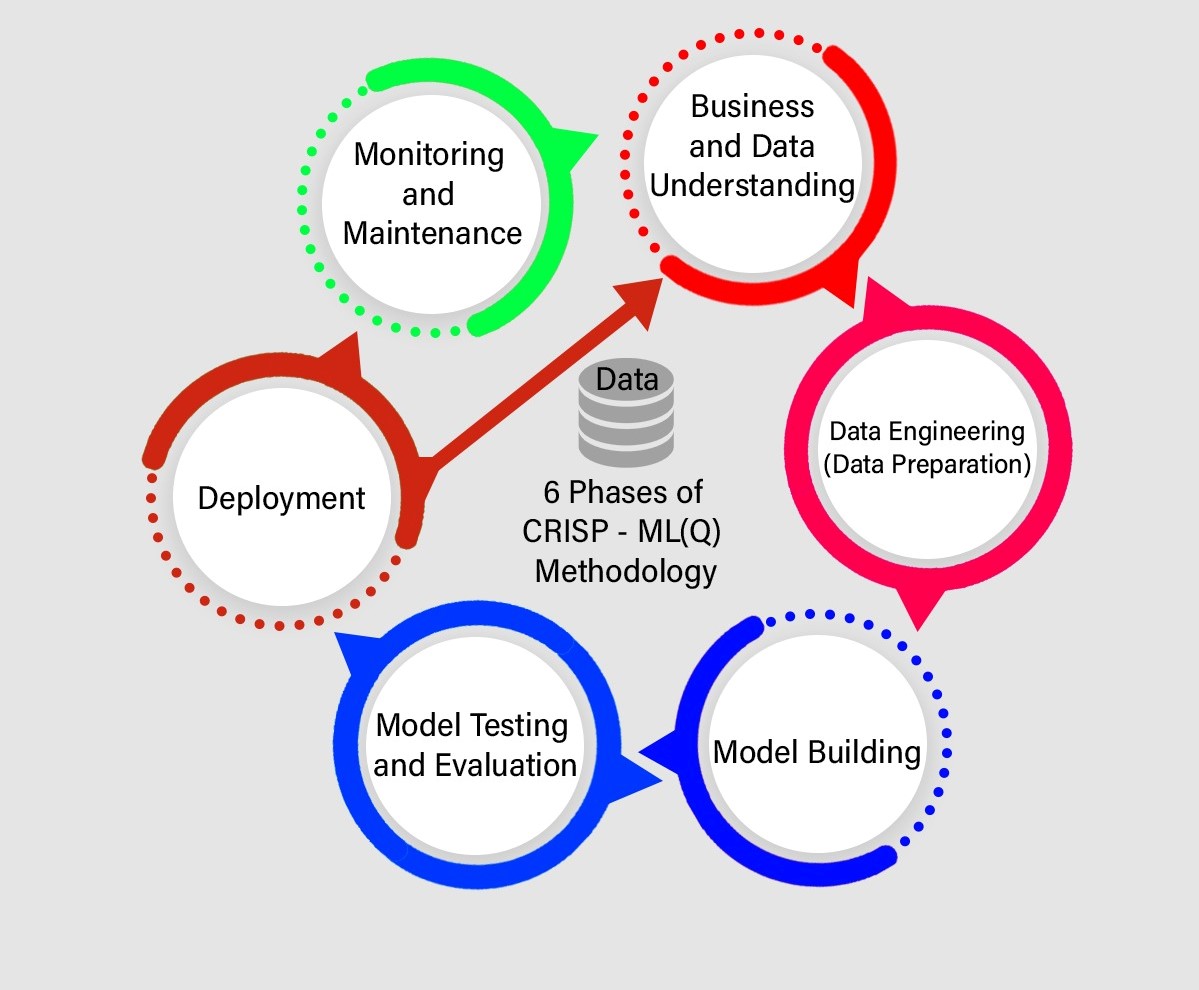
4 levels of Mlops maturity level

Why do we need MLOps

ML model deployment challenges

Crispml(Q)

CRISPML(Q)



**ML model deployment challenges**

ne example of a model deployment challenge highlighted in the paper is the issue of versioning. In traditional software engineering, it's common to use version control systems to manage changes to code and other artifacts over time. However, in machine learning systems, it's not always clear what should be versioned or how to version it.

For example, a model can be thought of as the output of a machine learning training process, which itself is a complex, multi-step process involving data processing, feature engineering, model selection, and hyperparameter tuning. If any of these steps change, the resulting model may be significantly different. Moreover, models are often trained on evolving data sets, so changes to the data can also impact the resulting model.

All of this makes it difficult to manage the versions of models and related artifacts (e.g., training data, feature pipelines) in a systematic way. Without proper versioning, it can be hard to reproduce and debug issues, or to track the impact of changes over time. This can lead to technical debt and create maintenance headaches for machine learning systems.

**Neccessity of MLOps (Maturity Model)**

LEVEL 0 to LEVEL4

**LEVEL 4 Architecture Diagram**

**AKS**

POWER BI REPORTS ( Visualizing Machine Learning Predictions with Power BI )