Comprehensive MLOps Guide – Enhanced

1. What is MLOps?

MLOps (Machine Learning Operations) integrates ML with DevOps to automate and streamline the ML lifecycle. Key pillars include:
Reliability: Reproducible pipelines and environments.
Scalability: Efficient resource management via autoscaling, serverless, and edge deployments.
Collaboration: Unified platforms across data science, engineering, and Ops.
Observability: Monitoring performance, drift, bias, and infra health continuously.

Latest Developments

• **Data-Centric AI:** Focus on improving data quality (Data-centric toolsets).
• **Feature Stores:** Centralized feature versioning and governance (Feast, Tecton).
• **GitOps for ML:** Declarative ML workflow management via Git (Argo CD, Prefect).
• **ML Observability:** Advanced monitoring for drift, bias, and fairness (Evidently AI, WhyLabs).
• **Serverless MLOps:** Cost-efficient model serving via FaaS (AWS Lambda, GCP Cloud Functions).

Core MLOps Practices & Tools

Practice	Purpose	Tools/Examples
Code & Pipelines	Modular, reusable code & workflows	OOP, linting, CI templates
Version Control	Track code/data/model changes	Git, DVC, MLflow
CI/CD Automation	Automated build/test/deploy	GitHub Actions, Jenkins
Containerization	Consistent runtime environments	Docker, Kubernetes
Observability	Track perf. & drift metrics	Prometheus, Grafana, Evidently AI
Feature Management	Consistent feature serving	Feast, Tecton
Infra as Code	Declare infra via code	Terraform, Argo CD
Security & Governance	Ensure compliance	Vault, Seldon Guard

2. The Enhanced MLOps Lifecycle

Data Management: Ingestion, labeling (AutoML), versioning (DVC, Quilt). **Experimentation:** Track experiments & HP sweeps (MLflow, Weights & Biases). **Model Building:** Distributed training & NAS (Ray, Keras Tuner). **Model Deployment:** REST/gRPC, serverless, edge (ONNX, TF Lite). **Observability:** Drift, bias, performance metrics. **Governance:** Model cards, lineage, audit logs. **Retraining:** Automated pipelines (Kubeflow, Prefect).