ONE PAGE SUMMARY

IT & Software › Kubernetes › GitOps on Kubernetes

Introduction to GitOps on Kubernetes

Streamline Kubernetes management with GitOps principles.

DEFINITION

GitOps is a modern approach to managing Kubernetes applications using Git as the single source of truth. It automates deployment and operations through version control practices.

KEY CONCEPTS

Git as Source of Truth

In GitOps, all configurations and application states are stored in a Git repository.

Continuous Deployment

Changes in the Git repository automatically trigger deployments to Kubernetes clusters.

Declarative Infrastructure

Kubernetes resources are defined declaratively, allowing for easy tracking and rollback of changes.

Automated Rollbacks

If a deployment fails, GitOps tools can revert to the last known good state automatically.

Observability

Monitoring tools are integrated to provide visibility into the state of deployments and infrastructure.

EXAMPLES

→ Using ArgoCD to synchronize Git repository changes with Kubernetes.

→ Deploying an application by merging a pull request in Git.

→ Rolling back to a previous version by reverting a commit in Git.

MEMORY TIPS

★ Think 'Git is King' to remember Git's central role in GitOps.

★ Use 'Deploy from Git' to recall the deployment process triggered by Git changes.

★ Remember 'Rollback with ease' for the automated rollback feature.

COMMON MISTAKES

✗ Neglecting to keep Git repositories updated with the latest configurations.

✗ Overlooking security practices when managing Git access.

✗ Failing to monitor deployments, leading to unnoticed issues.

QUICK RECAP

GitOps simplifies Kubernetes management by using Git for version control and automation. It emphasizes declarative configurations and enables easy rollbacks, enhancing deployment reliability and observability.