# Understanding the GitOps Flow with ArgoCD

**What is GitOps?**

**GitOps** is a modern way to **manage Kubernetes deployments** where:

* **Git is the source of truth** for your infrastructure and application configurations.
* **Changes are made through Git commits** (like updating a deployment YAML).
* A GitOps tool (like **ArgoCD**) automatically detects those changes and **syncs** them to your Kubernetes cluster.

**🔄 GitOps Workflow Steps**

**1. Store Kubernetes Manifests in GitHub**

You write your app's Kubernetes YAML files:

* deployment.yaml
* service.yaml
* And an ArgoCD app.yaml

And push them to a GitHub repo.

**2. ArgoCD Watches the GitHub Repo**

ArgoCD is installed in your Kubernetes cluster. It:

* Connects to your GitHub repo
* Monitors the specific folder and branch you configure

**3. ArgoCD Syncs to Kubernetes**

When ArgoCD detects a new commit:

* It automatically **syncs** the GitHub manifest changes
* It **applies them to the Kubernetes cluster**
* Example: If you update replicas: 1 to replicas: 2, it will scale the app automatically

This ensures your cluster always matches what’s in Git.

**4. View & Manage in ArgoCD UI**

You can:

* See your application status (Healthy / OutOfSync)
* Watch the deployment tree visually
* Trigger sync manually if auto-sync is disabled
* Roll back to older versions using Git commit history

**✅ Benefits of GitOps**

* **Version Control:** Git tracks every change (who, what, when, why)
* **Automation:** ArgoCD deploys automatically from Git
* **Consistency:** What’s in Git is what runs in production
* **Easy Rollbacks:** Revert to previous commit → ArgoCD redeploys

**🧠 Real-Life Analogy**

Think of GitOps like **Google Drive auto-sync**:

* You edit a file on your laptop → it's instantly updated in the cloud
* Similarly, you edit a YAML in Git → ArgoCD auto-updates the cluster