OpenFN Application Release to Production

The intent of this demo is to familiarize and demonstrate GitOps workflows using ArgoCD. This will show the GitOps repo that is used for application deployment and show how ArgoCD is being used to manage releases into Test and Production environments.

Goals for the Demo:

- Introduce the concept of using GitOps for continuous delivery
- Highlight benefits of in-cluster GitOps for secure delivery practices
- Show the GitOps repo used by the web application
- Introduce ArgoCD & show how it manages deployments
- Show how ArgoCD can monitor drift within your deployed applications

Prerequisites:

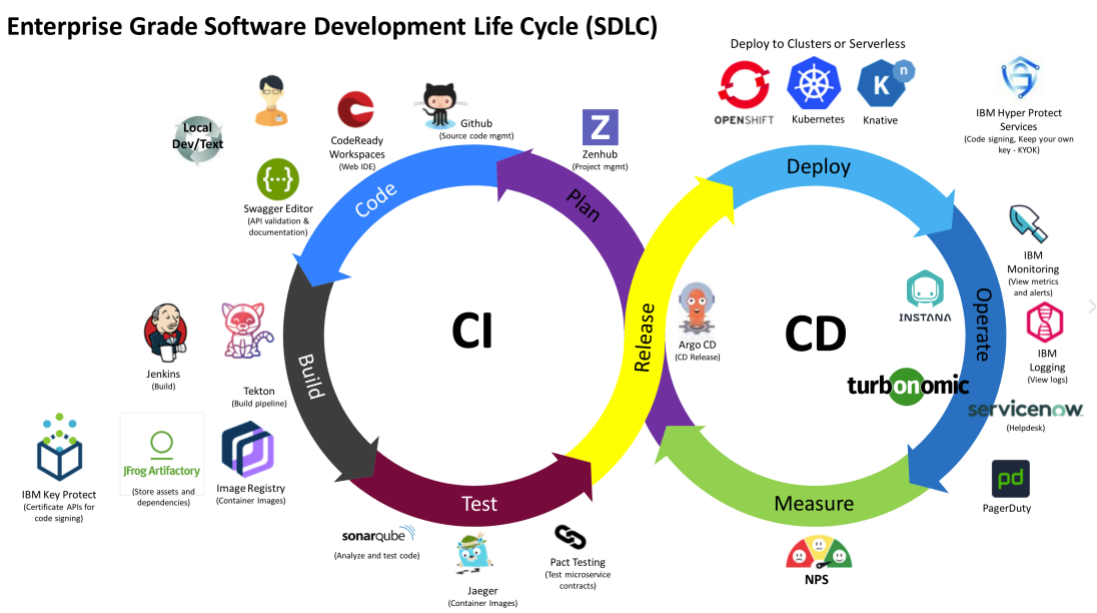
- If you have not already done so, request access to the FS Cloud demo environment at: https://techzone.ibm.com/collection/ibm-cloud-for-financial-services

- Download and install the OpenVPN client
  - Windows https://openvpn.net/community-downloads/
  - MacOS https://openvpn.net/client-connect-vpn-for-mac-os/
  - Linux https://openvpn.net/download-open-vpn/

- Download the techzone.ovpn VPN certificate and add it to the OpenVPN client
  - Link https://techzone-iam-agent.eqtyaj6hk2k.eu-de.codeengine.appdomain.cloud/vpn/download

Demo Steps:

1. Here we will discuss the Continous Delivery aspect of the enterprise software development lifecycle. This covers the delivery container images and artifacts produced by the continuous integration (CI)/DevOps pipelines into production using GitOps with ArgoCD.

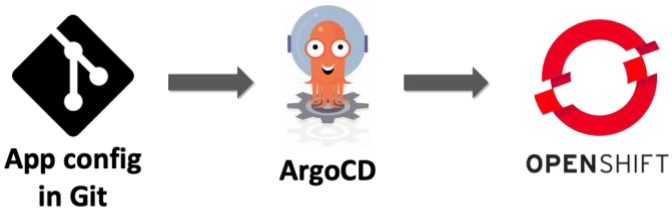


2. Introduce the concept of GitOps

   a. **GitOps** is a set of practices to manage infrastructure and application configurations using Git. GitOps works by using Git as a single source of truth to manage your deployments.

      ArgoCD is a declarative, GitOps continuous delivery tool for Kubernetes. This is what we are using, and is the foundation of OpenShift GitOps.

      Since this is declarative in nature, you define the end state in code, and rely on the tools to achieve and maintain that state.



3. Explain that we are using ArgoCD running inside of the OpenShift clusters to manage deployments inside of those clusters.

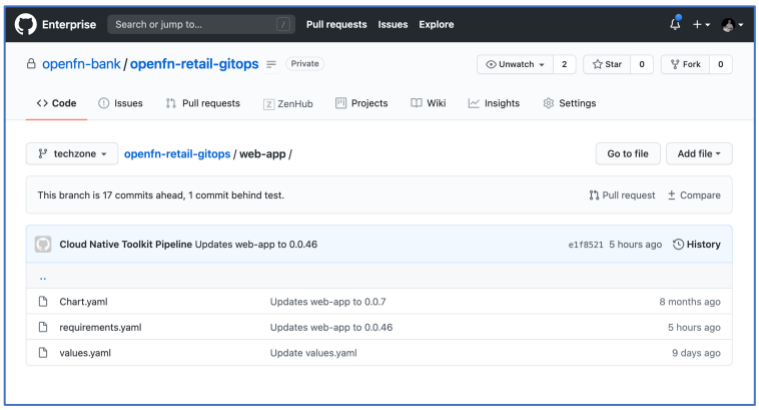
    a. There is an ArgoCD instance on the management cluster that deploys to the **test** namespace/instance.

    b. There is an ArgoCD instance on the workload cluster that deploys to the **production** namespace/instance.

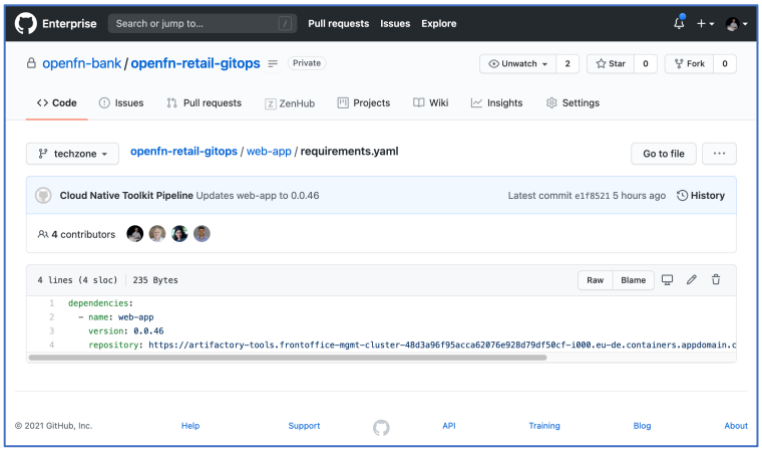
    c. Using ArgoCD/GitOps **inside** of the cluster is inherently more secure than using a push model from an external pipeline. In this case, Argo will pull resources that are needed (helm charts, container images, etc…) based on what is defined in the deployment. No additional networking configurations or inbound ports or mappings are needed in order to deploy the application.

4. Navigate to https://github.ibm.com/openfn-bank/openfn-retail-gitops/tree/techzone/web-app

    This is the helm chart used by ArgoCD to deploy the OpenFN retail banking application.



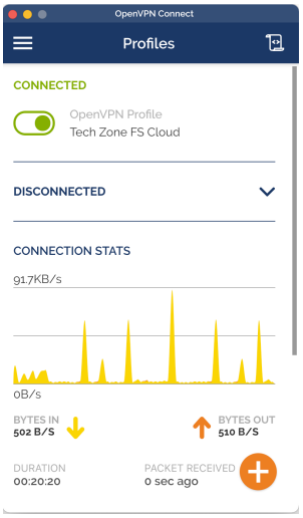
5. Click on “requirements.yaml”



Call out the version and repository values.

a. The version is the last version published from the DevOps pipeline
b. The repository is a url to the versioned application helm chart in Artifactory, which was published by the DevOps pipeline.

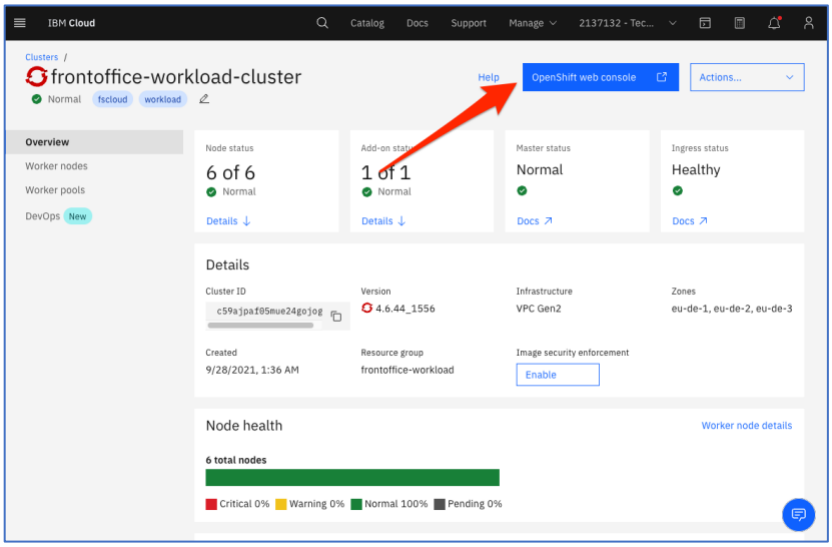
6. Now, we need to view the workload cluster. First connect the OpenVPN Client with the Tech Zone demo profile



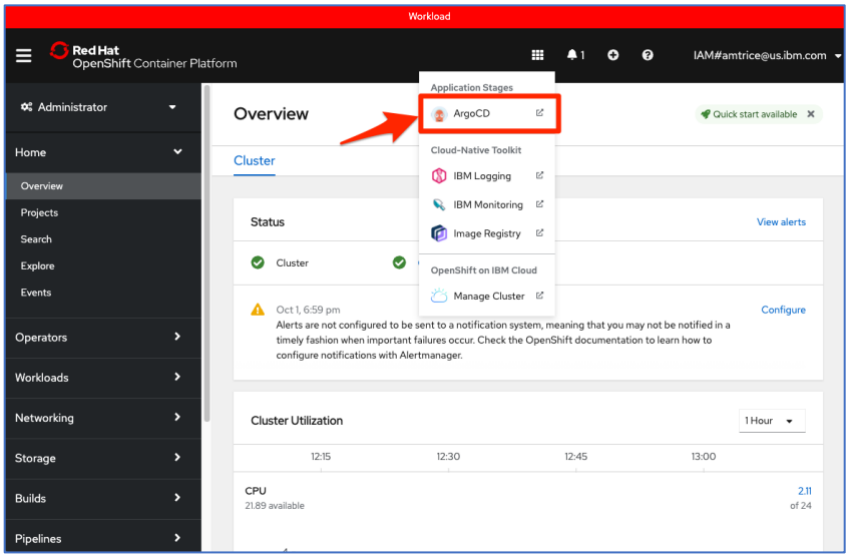
7. Navigate to https://cloud.ibm.com/kubernetes/clusters

8. Select the "**frontoffice-workload-cluster**" instance to view the cluster details

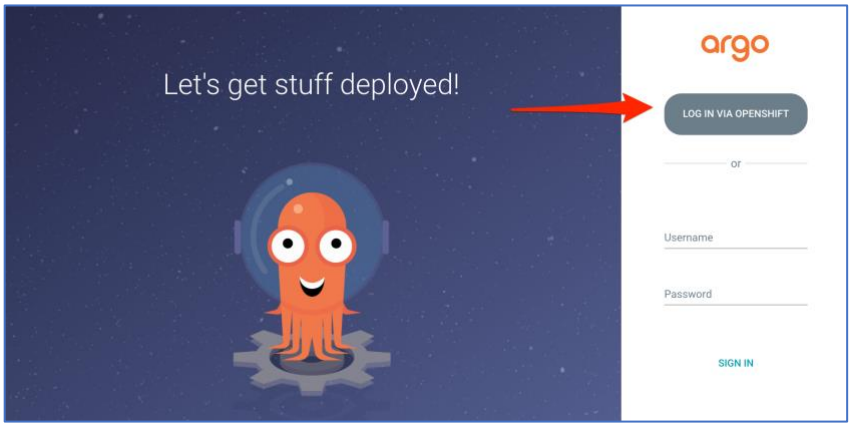
9. Click the “OpenShift web console” button to bring up the OpenShift Dashboard



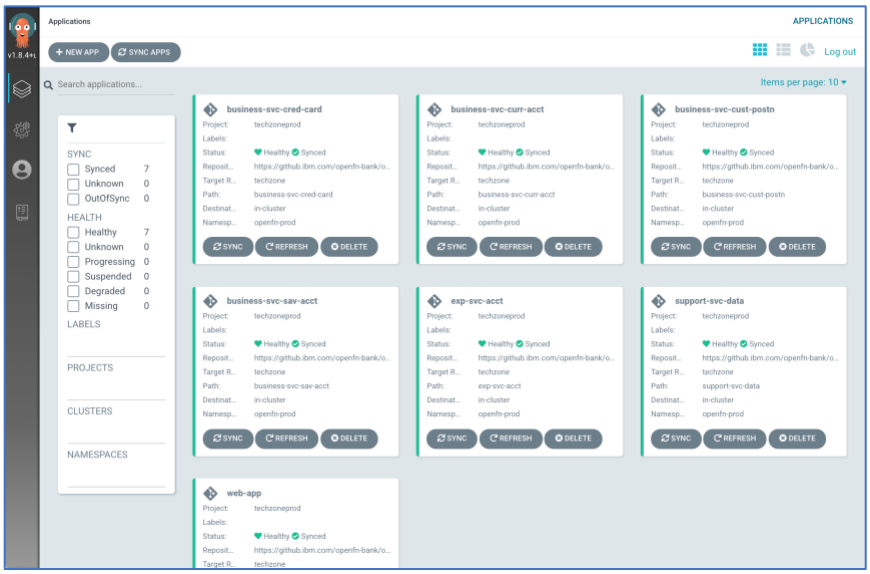
10. In the OpenShift Dashboard, open the Application Launcher menu and select the ArgoCD option.



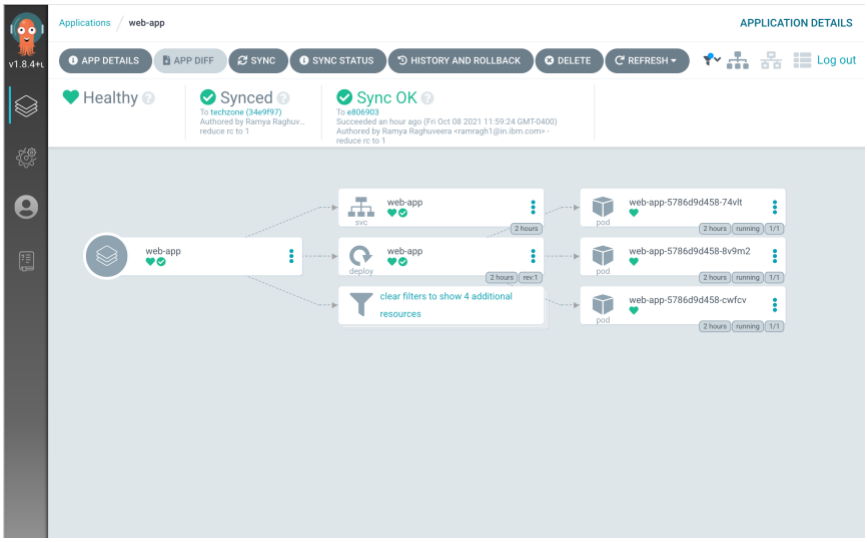
11. This will navigate to the ArgoCD login screen. Click on the “Log in via OpenShift” button.



12. Once logged in, you can see all of the microservices that are being managed by this instance of ArgoCD and are running in the workload cluster.



13. Click on the “web-app” instance to view details about the web app deployment.



Here we can see that the web application has been deployed and is healthy. There's also a visual representation of the deployment, and we can see there are three pods in the deployment. This represents the configuration defined for the deployment.

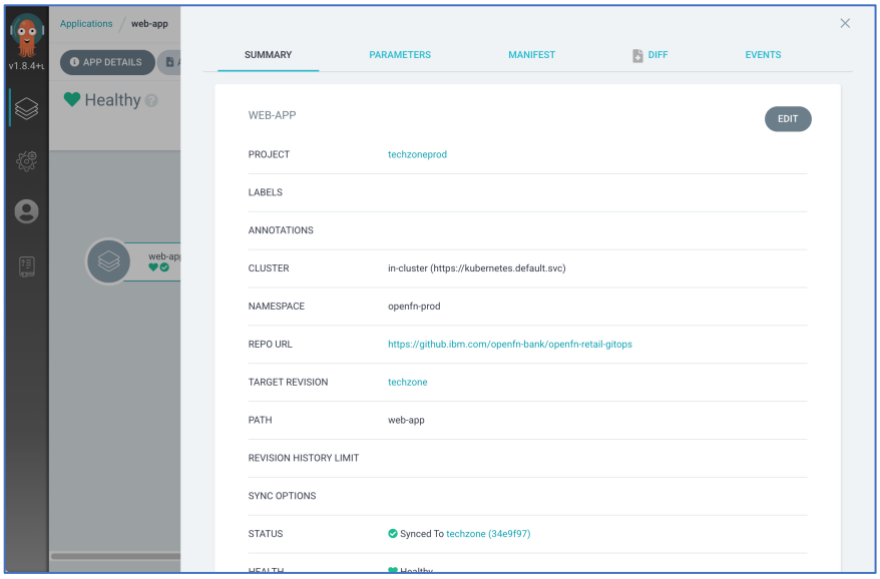
If there is a change in the GitOps repository that has not yet been synched to the deployment within the cluster, the sync status will show "Out of Sync", and you will be able to drill into details to see differences between what has been deployed, and the definition in the GitOps repository.

This will also show drift in the deployment. Meaning, if a user manually updates the deployment within the cluster (for example, changing the number of pods in the deployment), then ArgoCD will also show the "Out of Sync" status, highlighting that the deployment and the declarative definition are not in sync.

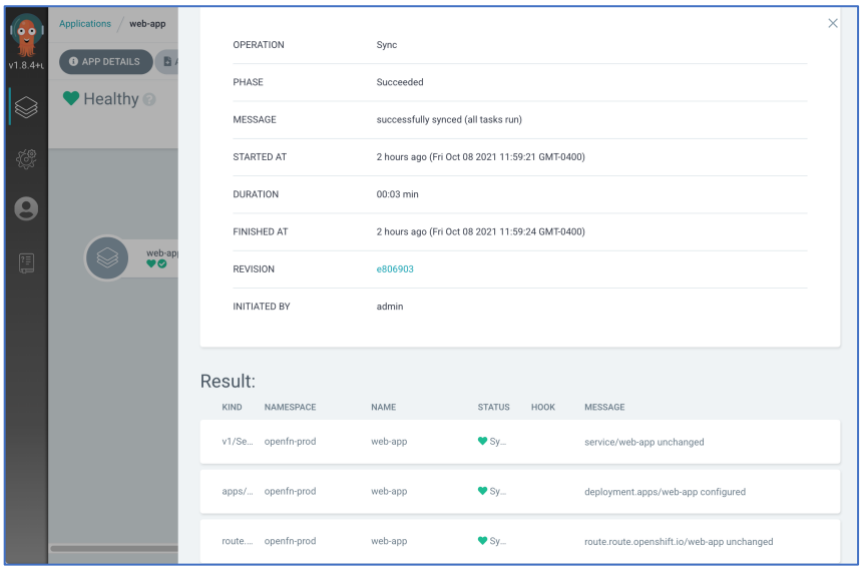
ArgoCD can be configured for automatic synchronization or manual synchronization. This application is currently configured for manual synchronization, so a user has to manually click the "Synchronize" button to synch the deployed application with the definition in the GitOps repository. If this application was configured for automatic synchronization, then the configuration would be synchronized automatically any time that the GitOps repository is changed, or if the deployment drifts from the configuration specified in the GitOps repo.

Notice that there is also a "History and Rollback" button, that enables the administrator to easily roll back to an earlier version.

14. Click on the "App Details" button to view details about the web app configuration.

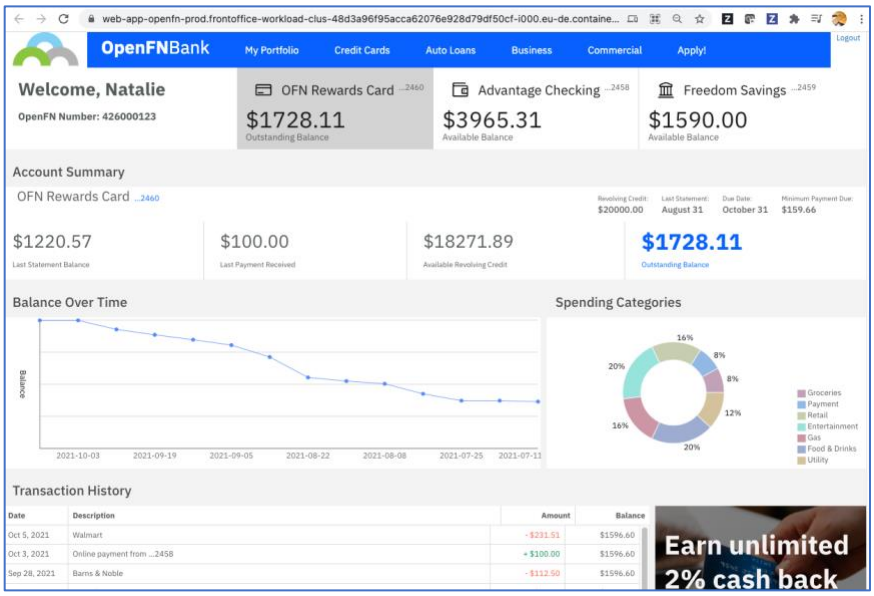


15. Close the App Details modal, and click on the "Sync OK" status or "Sync Status" button to view details about the most recent synchronization.



16. Navigate to https://web-app-openfn-prod.frontoffice-workload-clus-48d3a96f95acca62076e928d79df50cf-i000.eu-de.containers.appdomain.cloud and you can see the application running in the production environment.

    Login with the username techzone-demo@ibm.com with the password "techzone" and click "SignIn"



THIS CONCLUDES THE DEMO STEPS