**Argo CD**

[Argo CD](https://github.com/argoproj/argo-cd) is a GitOps tool for Continuous Deployment. Argo CD will compare the Kubernetes manifests in a git repository to the manifests it reads in your Kubernetes cluster, can synchronize those manifests from git into Kubernetes, and helps you monitor the state of your services in Kubernetes.

**Argo CD Chart**

A Helm chart for ArgoCD, a declarative, GitOps continuous delivery tool for Kubernetes.

Source code can be found [here](https://argoproj.github.io/argo-cd/)

**Additional Information**

This chart installs [argo-cd](https://argoproj.github.io/argo-cd/), a declarative, GitOps continuous delivery tool for Kubernetes.

The default installation is intended to be similar to the provided ArgoCD [releases](https://github.com/argoproj/argo-cd/releases).

Requirements

* Installed [kubectl](https://kubernetes.io/docs/tasks/tools/install-kubectl/) command-line tool
* Installed [Helm](https://helm.sh/docs/intro/install/) locally.

**Install Argo CD Helm chart**

***kubectl create namespace argocd***

***git clone*** [***https://github.ual.com/bag-manager/bm-helm-charts.git***](https://github.ual.com/bag-manager/bm-helm-charts.git)

***cd bm-helm-charts***

***helm install argocd ./argocd -n argocd***

This will create all ArgoCD deployments, services, serviceaccounts, roles, rolebindings in namespace – argocd.

**ArgoCD Application of Applications**

Argo CD applications, projects and settings can be defined declaratively using Kubernetes manifests.

The Application CRD is the Kubernetes resource object representing a deployed application instance in an environment. It is defined by two key pieces of information:

* source reference to the desired state in Git (repository, revision, path, environment)
* destination reference to the target cluster and namespace.

A minimal Application spec in Helm chart is as follows:

*apiVersion: argoproj.io/v1alpha1*

*kind: Application*

*metadata:*

*name: bm-bag-data-svc*

*namespace: {{ .Values.metadata.namespace }}*

*spec:*

*project: bag-manager*

*source:*

*helm:*

*valueFiles:*

*- {{ .Values.spec.helm.appvalues }}*

*repoURL: {{ .Values.spec.source.repoURL }}*

*targetRevision: HEAD*

*path: bm-bag-data-svc*

*destination:*

*server: {{ .Values.spec.destination.server }}*

*namespace: bag-manager*

*syncPolicy:*

*automated:*

*prune: {{ .Values.spec.sync.prune }}*

*selfHeal: {{ .Values.spec.sync.selfheal }}*

*validate: {{ .Values.spec.sync.validate }}*

And the Values.yaml for the helm chart is as follows:

*spec:*

*destination:*

*server: https://kubernetes.default.svc*

*source:*

*repoURL: https://github.ual.com/bag-manager/bm-helm-charts.git*

*sync:*

*prune: true*

*selfheal: true*

*validate: true*

*helm:*

*appvalues: values.yaml*

*metadata:*

*namespace: argocd*

**Installing ArgoCD-App**

***kubectl create namespace bag-manager***

***git clone*** [***https://github.ual.com/bag-manager/bm-helm-charts.git***](https://github.ual.com/bag-manager/bm-helm-charts.git)

***cd bm-helm-charts***

***helm install argocd-app ./argocd-app -n argocd***

This will create all the Kubernetes resources in the bag-manager namespace.

Here we have multiple Values.yaml files for specific Environments (Dev, QA, Prod)

We will be running following commands specific to environment.

Dev: ***helm install argocd-app ./argocd-app --values=./argocd-app/values-dev.yaml -n argocd***

QA: ***helm install argocd-app ./argocd-app --values=./argocd-app/values-qa.yaml -n argocd***

Prod: ***helm install argocd-app ./argocd-app --values=./argocd-app/values.yaml -n argocd***

ArgoCD-App Chart Layout:

A screenshot of a cell phone

Description automatically generated

After the command

***helm install argocd-app ./argocd-app --values=./argocd-app/values-dev.yaml -n argocd***

We can see the deployments like this:

A screenshot of a cell phone

Description automatically generated