****

**Deployments**

A Deployment controller provides declarative updates for Pods and ReplicaSets.

Deployments are upgraded and higher version of replication controller. They manage the deployment of replica sets which is also an upgraded version of the replication controller. They have the capability to update the replica set and are also **capable of rolling back** to the previous version.

Pod--Application----v1

Pod---Application-v2

A deployment is a supervisor for pods and replica sets, giving you fine-grained control over how and when a new pod version is rolled out as well as rolled back to a previous state.

## Use Case

The following are typical use cases for Deployments:

* Create a Deployment to rollout a ReplicaSet. The ReplicaSet creates Pods in the background. Check the status of the rollout to see if it succeeds or not.

They provide many updated features of **matchLabels** and **selectors**. We have got a new controller in the Kubernetes master called the deployment controller which makes it happen. It has the capability to change the deployment midway.

## Changing the Deployment

**Updating** − The user can update the ongoing deployment before it is completed. In this, the existing deployment will be settled and new deployment will be created.

**Deleting** − The user can pause/cancel the deployment by deleting it before it is completed. Recreating the same deployment will resume it.

**Rollback** − We can roll back the deployment or the deployment in progress. The user can create or update the deployment by using **DeploymentSpec.PodTemplateSpec = oldRC.PodTemplateSpec.**

## Deployment Strategies

Deployment strategies help in defining how the new RC should replace the existing RC.

**Recreate** − This feature will kill all the existing RC and then bring up the new ones. This results in quick deployment however it will result in downtime when the old pods are down and the new pods have not come up.

**Rolling Update** − This feature gradually brings down the old RC and brings up the new one. This results in slow deployment, however there is no deployment. At all times, few old pods and few new pods are available in this process.

**Note:** matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is “key”, the operator is “In”, and the values array contains only “value”. The requirements are ANDed.