**Blue-green deployments**

Blue-green deployments involve running two versions of an application at the same time and moving traffic from the in-production version (the green version) to the newer version (the blue version). You can use a Rolling strategy or switch services in a route.

Because many applications depend on persistent data, you must have an application that supports *N-1 compatibility*, which means it shares data and implements live migration between the database, store, or disk by creating two copies of the data layer.

Consider the data used in testing the new version. If it is the production data, a bug in the new version can break the production version.

**Setting up a blue-green deployment**

Blue-green deployments use two DeploymentConfigs. Both are running, and the one in production depends on the service the route specifies, with each DeploymentConfig exposed to a different service.

|  |  |
| --- | --- |
|  | Routes are intended for web (HTTP and HTTPS) traffic, so this technique is best suited for web applications. |

You can create a new route to the new version and test it. When ready, change the service in the production route to point to the new service and the new (blue) version is live.

If necessary, you can roll back to the older (green) version by switching the service back to the previous version.

Procedure

1. Create two copies of the example application:
2. $ oc new-app openshift/deployment-example:v1 --name=example-green

$ oc new-app openshift/deployment-example:v2 --name=example-blue

This creates two independent application components: one running the v1 image under the example-green service, and one using the v2 image under the example-blue service.

1. Create a route that points to the old service:

$ oc expose svc/example-green --name=bluegreen-example

1. Browse to the application at example-green.<project>.<router\_domain> to verify you see the v1 image.
2. Edit the route and change the service name to example-blue:

$ oc patch route/bluegreen-example -p '{"spec":{"to":{"name":"example-blue"}}}'

1. To verify that the route has changed, refresh the browser until you see the v2 image.