Blue-Green Deployment with Helm

This Helm chart is to demonstrate Blue/Green deployment strategy. It creates following primary resources:

- **service-prod**: Serves the production traffic.
- **service-stage**: Serves the test traffic.
- **Ingress**: NGINX ingress controller - Based on HostName routes production traffic to service-prod, and test traffic to service-stage.
- **service-lb**: loadbalances traffic from outside to Ingress.
- **Deployments** [NodeJS Application with Mongo DB backend. It exposes APIs to access database.]
  - Deployment-blue
  - Deployment-green

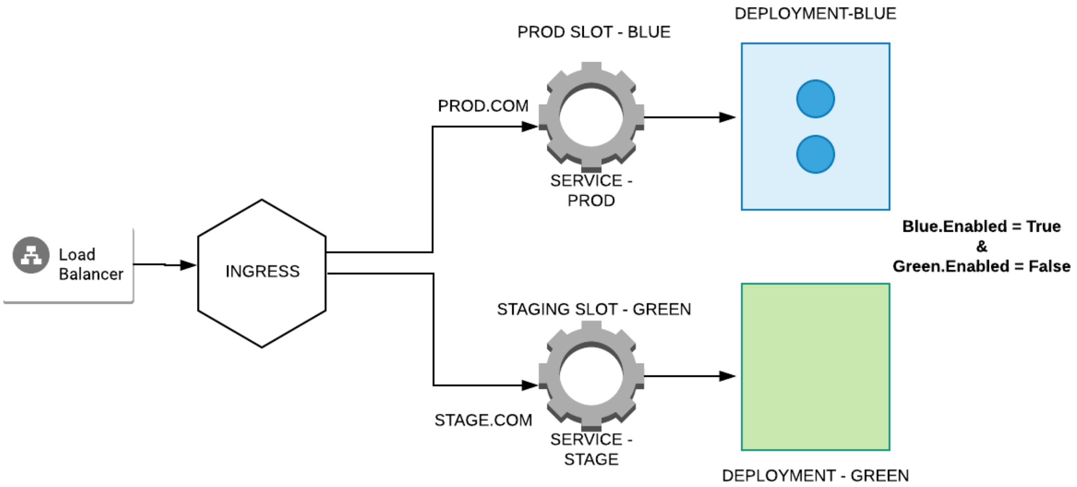
Both deployment yamls are identical: except for they target different slots i.e., blue or green.

Slot is virtual concept achieved by attaching label "**slot=blue|green**" to the pods. Service routes traffic to pods based on the slot value. Which slotes to deploy is controlled by variables **blue.enabled & green.enabled.**

A key variable used in the helm chart is **productionSlot**. Based on its value production service is configured to route the traffic to that slot.

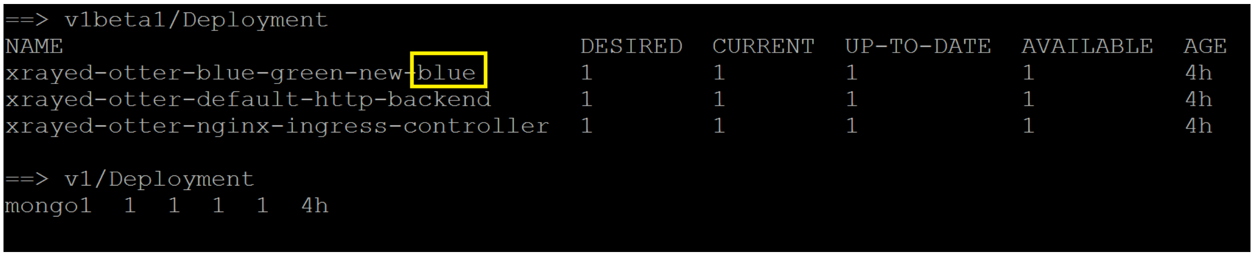
Step 1: **Blue deployed**

```
helm install blue-green-new-0.1.0 ##Helm chart installed
helm upgrade <release-name> /root/test/new/HelmCharts/blue-green-new --set blue.enabled=true --reuse-values ##Blue deployment created. The second argument is the path to the Helm chart dir.
```



The blue deployment is set as production environment using the below command.

helm upgrade *<release-name>* /root/test/new/HelmCharts/blue-green-new --set productionSlot=blue --reuse-values

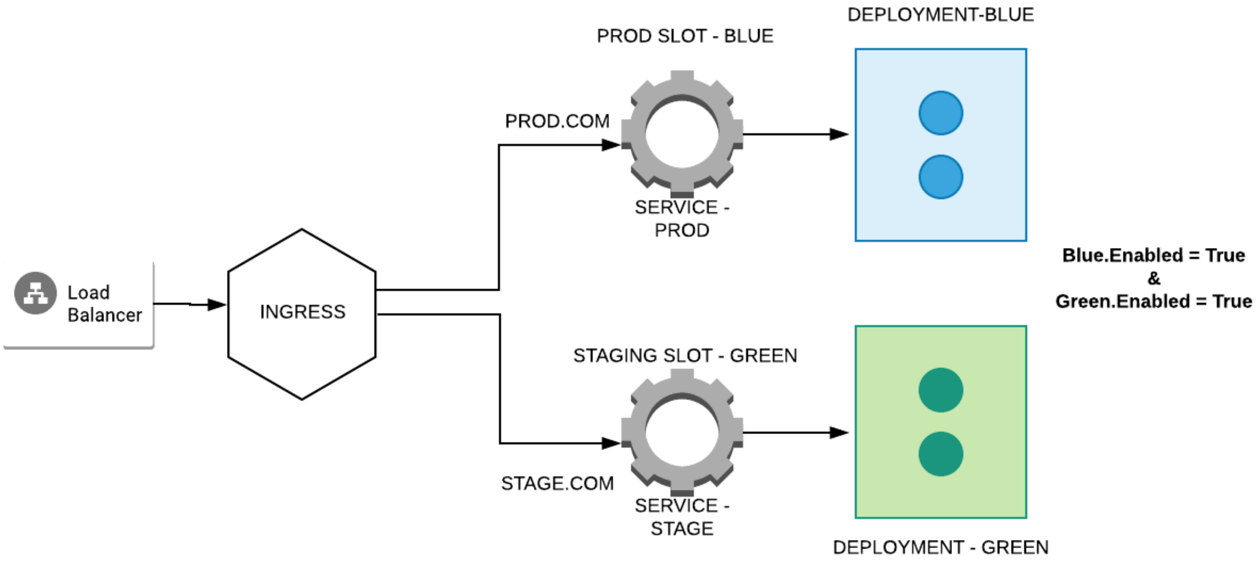


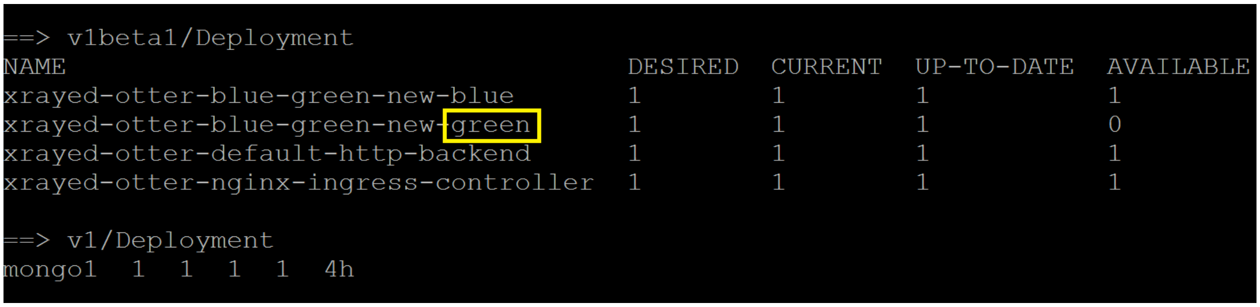
```
==> v1beta1/Deployment
NAME                                   DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
xrayed-otter-blue-green-new-blue       1        1        1           1          4h
xrayed-otter-default-http-backend      1        1        1           1          4h
xrayed-otter-nginx-ingress-controller  1        1        1           1          4h

==> v1/Deployment
mongo1  1  1  1  1  4h
```

<u>**Step 2**</u>**: Enable Green deployment in parallel to blue. Both deployments co-exist with blue being production.**

helm upgrade *<release-name>* /root/test/new/HelmCharts/blue-green-new --set green.enabled=true --reuse-values *##GREEN deployment created. The second argument is the path to the Helm chart dir.*

Below diagram and screen shot shows green deployment activated with the latest version of charts, but in staging. Both blue and green deployments are now running.

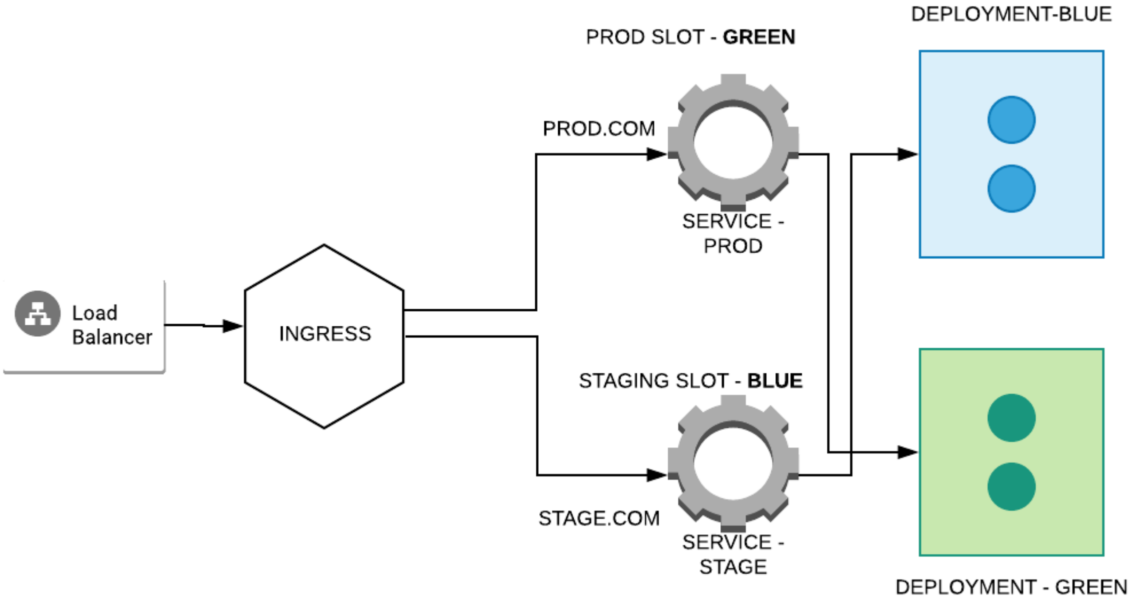




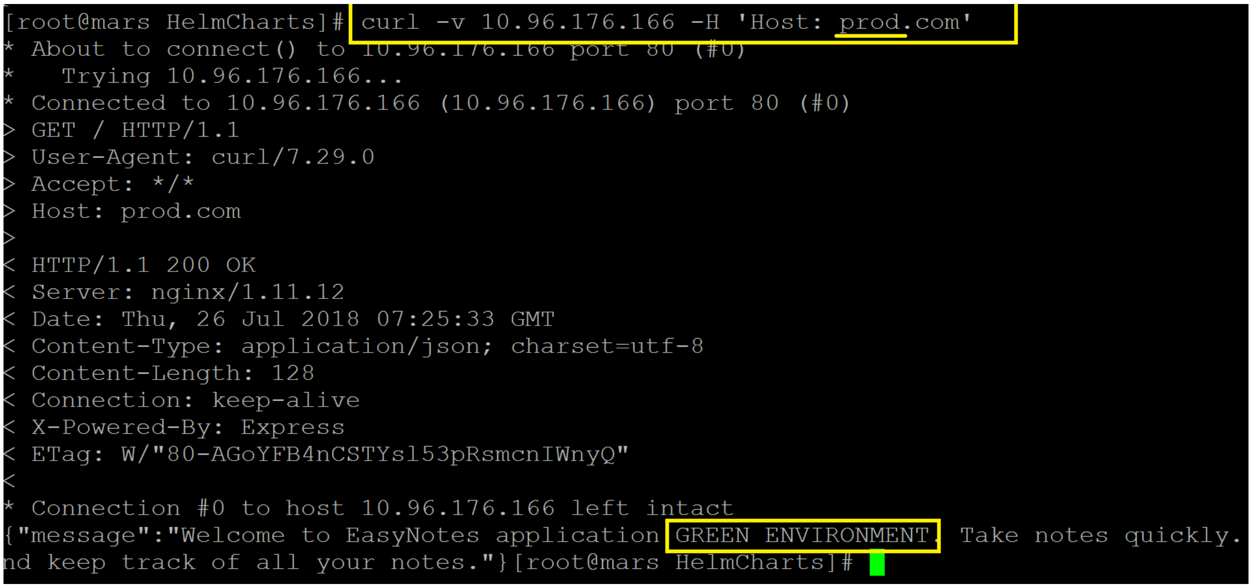
```
==> v1beta1/Deployment
NAME                                DESIRED   CURRENT   UP-TO-DATE   AVAILABLE
xrayed-otter-blue-green-new-blue    1         1         1            1
xrayed-otter-blue-green-new-green   1         1         1            0
xrayed-otter-default-http-backend   1         1         1            1
xrayed-otter-nginx-ingress-controller 1       1         1            1

==> v1/Deployment
mongo1   1   1   1   1   4h
```

Step 3: Once testing is successful, we can swap green to be production and blue to staging

```
helm upgrade <release-name> /root/test/new/HelmCharts/blue-green-new  --set
productionSlot=green --reuse-values ## The second argument is the path to the Helm chart dir
```



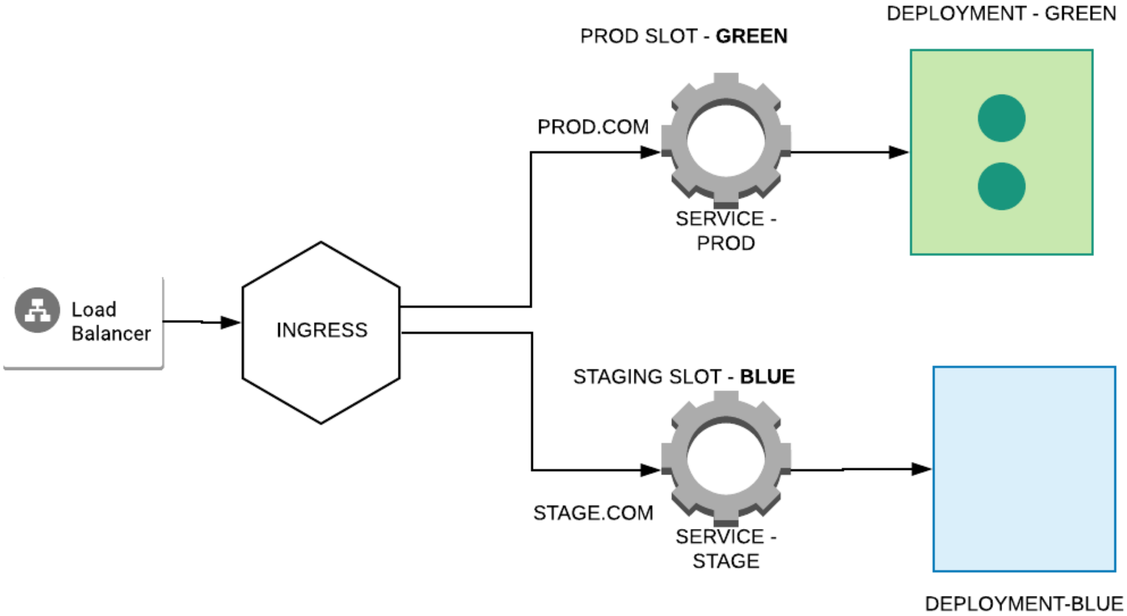
Below screen shot shows the response for CURL command that hits the service load balancer with argument -H having value "prod.com" as defined in values.yaml. The Ingress directs the request to the Green deployment as the value for ProductionSlot variable is set as Green.

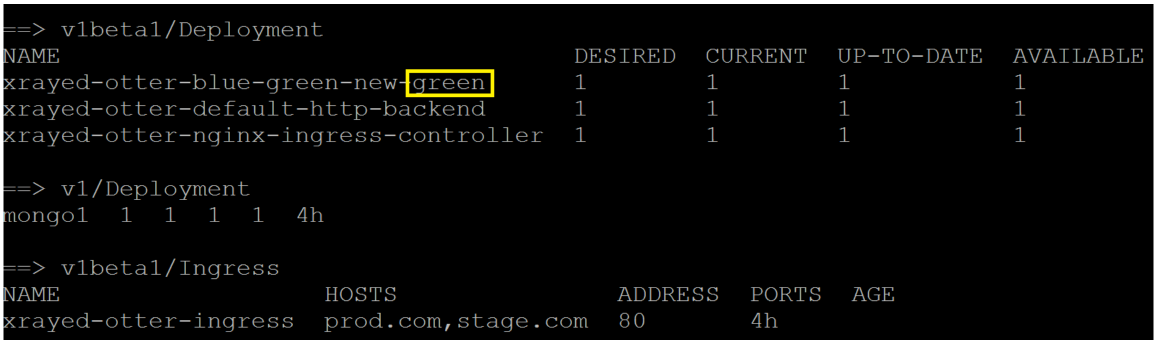


```
[root@mars HelmCharts]# curl -v 10.96.176.166 -H 'Host: prod.com'
* About to connect() to 10.96.176.166 port 80 (#0)
*   Trying 10.96.176.166...
* Connected to 10.96.176.166 (10.96.176.166) port 80 (#0)
> GET / HTTP/1.1
> User-Agent: curl/7.29.0
> Accept: */*
> Host: prod.com
>
< HTTP/1.1 200 OK
< Server: nginx/1.11.12
< Date: Thu, 26 Jul 2018 07:25:33 GMT
< Content-Type: application/json; charset=utf-8
< Content-Length: 128
< Connection: keep-alive
< X-Powered-By: Express
< ETag: W/"80-AGoYFB4nCSTYsl53pRsmcnIWnyQ"
<
* Connection #0 to host 10.96.176.166 left intact
{"message":"Welcome to EasyNotes application GREEN ENVIRONMENT. Take notes quickly.
nd keep track of all your notes."}[root@mars HelmCharts]#
```

Step 4: Now, if required, blue deployment can be deleted:

```
helm upgrade <release-name> /root/test/new/HelmCharts/blue-green-new  --set blue.enabled=false --reuse-values ##Blue deployment created. The second argument is the path to the Helm chart dir
```





```
==> v1beta1/Deployment
NAME                                   DESIRED  CURRENT  UP-TO-DATE  AVAILABLE
xrayed-otter-blue-green-new-green      1        1        1           1
xrayed-otter-default-http-backend      1        1        1           1
xrayed-otter-nginx-ingress-controller  1        1        1           1

==> v1/Deployment
mongo1  1  1  1  1  4h

==> v1beta1/Ingress
NAME                  HOSTS               ADDRESS  PORTS  AGE
xrayed-otter-ingress  prod.com,stage.com  80       4h
```

The kubectl shows the latest running green deployment only.