



Helm Chart / Kubernetes Support (Beta)

This documentation outlines how to use HEAVY.AI's Helm Chart within a Kubernetes environment. It assumes the user is a network administrator within your organization and is an experienced Kubernetes administrator. This is not a beginner guide and does not instruct on Kubernetes installation or administration. It is quite possible you will require additional manifest files for your environment.

Overview

The HEAVY.AI Helm Chart is a template of how to configure deployment of the HEAVY.AI platform. The following files need to be updated/created to reflect the customer's deployment environment.

- `values.yml`
- `<customer_created>-pv.yml`
- `<customer_created>-pvc.yml`

Once the files are updated/created, follow the installation instructions below to install the Helm Chart into your Kubernetes environment.

Where to get the Helm Chart?

The Helm Chart is located in the HEAVY.AI github repository. It can be found here:
<https://releases.heavy.ai/ee/helm/heavyai-1.0.0.tgz>

What's included?

```
Helm-workspace
├── heavyai
│   ├── Chart.yml
│   ├── values.yml
│   ├── templates
│   │   ├── README.txt
│   │   └── deployment.yml
│   └── misc
│       ├── example-heavyai-pv.yml
│       └── example-heavyai-pvc.yml
```

File Name	Description
-----------	-------------

Chart.yml	HEAVY.AI Helm Chart. Contains version and contact information.
values.yml	Copy this file and edit values specific to your HEAVY.AI deployment. This is where to note the PVC name. This file is annotated to identify typical customizations and is pre-populated with default values.
README.txt	Instructions on how to use.
deployment.yml	HEAVY.AI platform deployment template. DO NOT EDIT
example-heavyai-pv.yml	Example of PV file.
example-heavyai-pvc.yml	Example of PVC file.

How to install?

1. Before installing, create a PV/PVC that the deployment will use. Save these files in the regular PVC/PV location used in the customer's environment. Reference the `README.txt` file found in the Helm Chart under templates and the example PV/PVC manifests in the `misc` folder in the helm chart. The PVC name is then provided to the `helm install` command.
2. In your current directory; copy the `values.yml` file from the HEAVY.AI Helm Chart and customize for your needs.
3. Run the `helm install` command with the desired deployment name and Helm Chart.
 - a. When using a `values.yml` file:

```
$ helm install heavyai --values values.yml
heavyaihelmchart-0.0.1.tgz
```

- b. When not using a `values.yml` file:

If you only need to change a value or two from the default `values.yml` file you can use `--set` instead of a custom `values.yml` file.

For example:

```
$ helm install heavyai --set pvcName=MyPVCName
heavyaihelmchart-0.0.1.tgz
```

How to uninstall?

To uninstall the helm installed HEAVY.AI instance:

```
$ helm uninstall heavyai
```

NOTE: The PVC and PV space defined for the HEAVY.AI instance is not removed. The retained space must be manually deleted.

Example: values.yml

```
# Default values for heavyai.
# This is a YAML-formatted file.
# Declare variables to be passed into your templates.
#
# Version of heavyai to install in the format 'v7.0.0' or 'latest' for the latest
version released.
version: v7.0.0
# Persistent volume claim name to use with heavyai.
pvcName: heavyai-pvc
# Namespace to install heavyai in.
namespace: heavyai
# Number of GPU's to assign to heavyai or 0 to run the CPU version of heavyai.
gpuNumber: 1
# NodeName to install heavyai on, if you wish to let Kubernetes schedule a host, leave
it blank.
nodeName: heavyai-node
# Immerse port redirect of 6273.
hostPortImmerse: 9273
# TCP port redirect of 6274.
hostPortTCP: 9274
# HTTP port redirect of 6278.
hostPortHTTP: 9278
```

Example: example-heavyai-pvc.yml

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: heavyai-pvc
  namespace: heavyai
```

```
spec:
  volumeMode: Filesystem
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 100Gi
  storageClassName: heavyai
```

Example: example-heavyai-pv.yml

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: heavyai-pv
spec:
  capacity:
    storage: 100Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteOnce
  persistentVolumeReclaimPolicy: Retain
  storageClassName: heavyai
  mountOptions:
    - hard
    - nfsvers=4.1
  nfs:
    path: {your nfs path goes here }
    server: { your nfs server name goes here }
```