



Helm Registry support within ONAP (OOM-2734)

Vijay Venkatesh Kumar (AT&T)
Krzysztof Kuzmicki (Nokia)

Problem Statement

DCAE Cloudify platform provided local inventory where all MS deployment artifacts (blueprints) were stored; these artifacts were served through rest api for on-demand deployment of DCAE components (CLAMP, Dashboard, Robot etc)

DCAE architecture transformation to Helm, Cloudify and related handler will be deprecated. Hence there is need for internal registry for storing helm chart/packages

OOM deployment steps currently uses local chart repository to be created on RKE node for building charts locally before ONAP can be deployed. Few cons with this approach

- Works okay for manual installation for non-scaled environment

- Does not support access for components within ONAP cluster for on-demand deployment

- Once charts releases are done on nexus3, local build setup is not required

Solution Proposed: Deploy ChartMuseum as internal registry within ONAP cluster as Helm chart repository server hosting all ONAP component charts

Why ChartMuseum?

- ChartMuseum is an open-source [Helm Chart Repository](#) server
- ChartMuseum is a child project under the Helm umbrella
- Used already under ONAP setup currently on RKENode

API

HELM CHART REPOSITORY

- GET /index.yaml - retrieved when you run `helm repo add chartmuseum http://localhost:8080/`
- GET /charts/mychart-0.1.0.tgz - retrieved when you run `helm install chartmuseum/mychart`
- GET /charts/mychart-0.1.0.tgz.prov - retrieved when you run `helm install` with the `--verify` flag

CHART MANIPULATION

- POST /api/charts - upload a new chart version
- POST /api/prov - upload a new provenance file
- DELETE /api/charts/<name>/<version> - delete a chart version (and corresponding provenance file)
- GET /api/charts - list all charts
- GET /api/charts/<name> - list all versions of a chart
- GET /api/charts/<name>/<version> - describe a chart version

Reference : <https://chartmuseum.com/docs/>

Usecases requiring internal HELM registry (Istanbul Release)

➤ [REQ-685](#) - DCAE Transformation to support Helm (Phase2)

[INT-1895](#) Migrate DCAE testsuite to use helm-based component services

Integration have several test specific components and overrides which are deployed post ONAP instantiation using DCAE Cloudify/platform API's. For DCAE services migrated to helm, need placeholder/registry to both push/pull ONAP component charts and supporting components.

[DCAE GEN2-2694](#) Helm charts generation through MOD

MOD will allow MS onboarding and helm packages will be generated dynamically (J release). The charts generated will need to be pushed registry accessible within ONAP for both DCAE and other application to access. Example Scenario: Model imported from Acumos through DCAE-MOD. These components (code/charts) are not required to be formally introduced under OOM.

➤ [REQ-716](#) - Control Loop in TOSCA LCM

[POLICY-3168](#) DCAE interaction for Helm service deployment

Part of TOSCA POC, CLAMP will support service onboarding and deployment via TOSCA, will be referencing either pre-onboarded or new artifact to be loaded into registry, hence require access for both PUSH/PULL to the registry.

Options for ONAP components (static) charts

OPTION – 1

Generate chart packages part of OOM initialization

New charts to deploy chart-museum in target cluster during ONAP deployment

- a) Mirror RKE chartMuseum registry into local registry using [scripts/mirror_k8s_repos.sh](#)
- b) Expose oom/kubernetes into ChartMusuem POD, build and push all component charts (or dcae services alone) into local repo

Istanbul

OPTION – 2

Mirror required component Charts from [ONAP NEXUS3](#) generated part of ONAP CI (including dcae services components)

New charts to deploy chart-museum in target cluster during ONAP deployment and mirror nexus3 repo.

Dependencies:

- OOM build and charts releases under nexus WIP

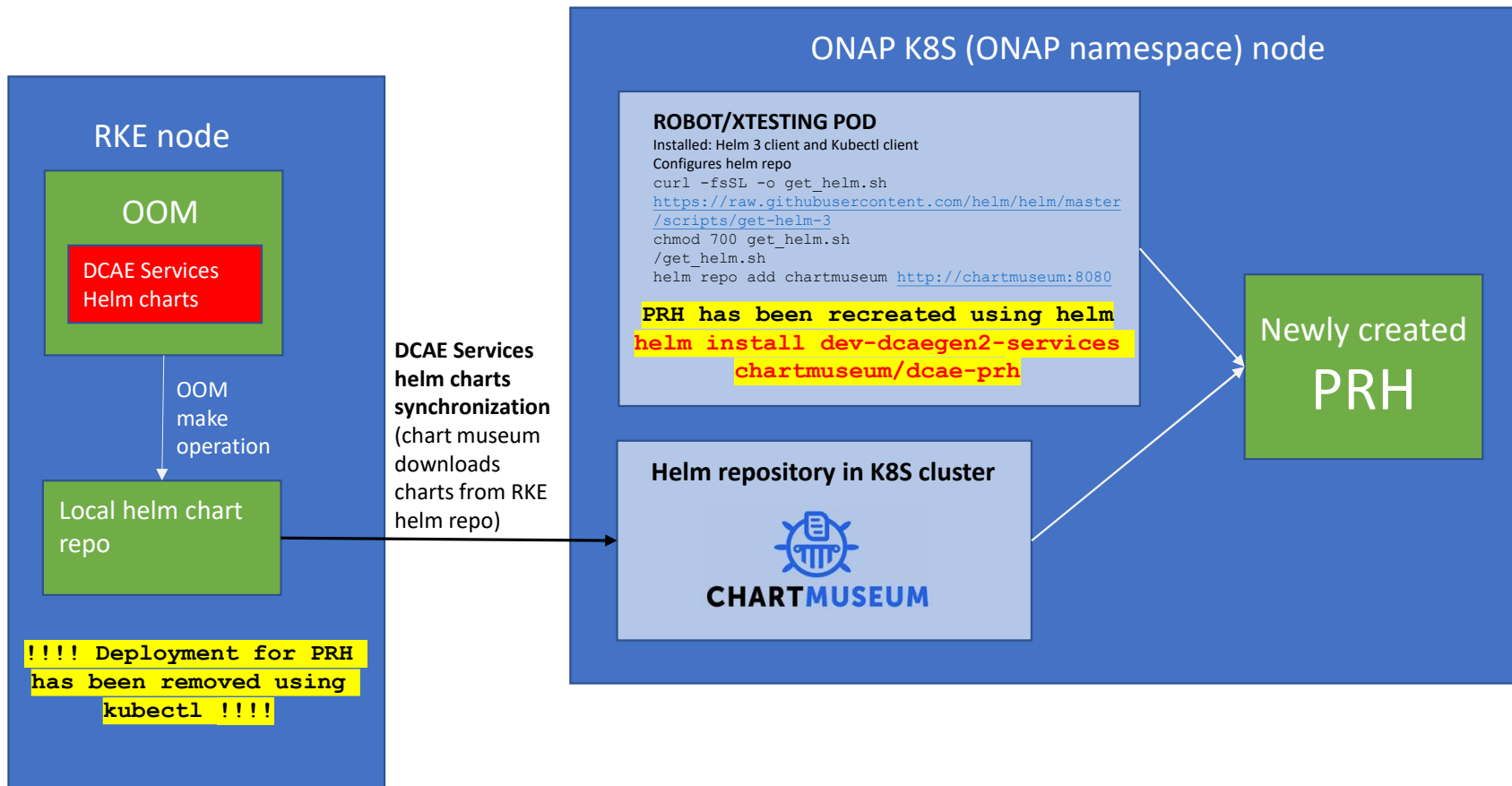
Limitation:

- ONAP offline support issues if ONAP NEXUS3 registry is not accessible

Future Support

Charts:<https://github.com/chartmuseum/charts/tree/main/src/chartmuseum>

Demo Steps



Project impacts for registry setup and integration

OOM

- Add charts for ChartMuseum for deployment from OOM (should be enabled by default)
- Customize init to support source local oom/kubernetes charts (Option-1 B) or mirror RKE Chartmuseum repo (Option-1 A) using [scripts/mirror_k8s_repos.sh](#)

Clients (**Integration/robot** or **POLICY/k8s participant** or **DCAE MOD/RuntimeAPI**)

- Add helm client setup
- Configure/add local chartmuseum repo
- Perform GET/POST to retrieve the charts

Deployment Options

- Wrapper script with helm client to deploy/undeploy
 - Deployment script/container
 - Takes override values.yaml + chart name & version
 - Fetches chart from internal registry (chart-museum)
 - Trigger install/uninstall using helm client
(values.yaml overrides if any to be maintained part of container)
- ➔ Integration/Robot & XTesting
- K8S participant triggered via DMAAP message
- ➔ POLICY/CLAMP

Open questions/dependencies

- SECCOM: Authentication choice between [Basic Auth](#) [Bearer/Token Auth](#) [HTTPS\(with certificates\)](#)
- OOM: Placeholder repo under OOM(oom/kubernetes/common/chartMuseum)
 - Preference of using [charts](#) as-is or custom charts using docker container
- DCAE: Service Application helm charts should be independently installed using “helm install” command
- How Service Applications helm charts will be loaded to chartMuseum:
 - Chart museum will connect to helm repo on RKE/Nexus (depending on installation) and download list of required charts.
 - Extending access outside cluster via Nodeport (for initial load)
 - Any other proposal ?