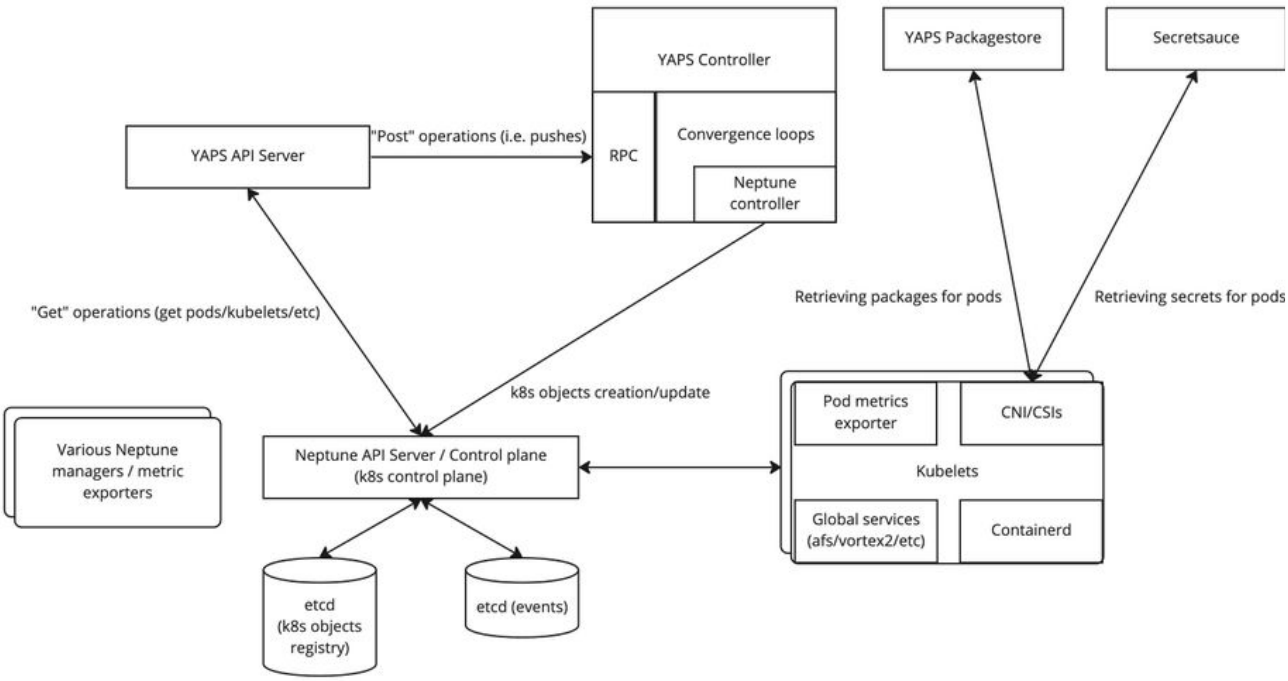
Neptune Overview

Components

Note: familiarity with YAPS architecture is suggested. Also, it's recommended to read about general k8s components to gain more complete context.

The (simplified) diagram of Neptune internals:



In addition to already known YAPS components covered here, Neptune-specific parts are:

- **Neptune Control plane**: more or less k8s counterpart of YAPS API Sever and YAPS Controller combined - the component responsible for converging requested k8s state, registering Kubelets and exposing API service for outside requests. Consists of:
  - **k8s API Server** - provides REST API for managing k8s cluster. All other components (both k8s internal as kubelets and external as yaps controller) communicate with cluster via API calls. Applications discover apiservers using Cezanne - internal dns-to-service-discovery service.
  - **k8s Scheduler** - control plane process which assigns Pods to Kubelets.
  - **k8s Controller Manager** - core control loops.
  - Various sidecars (i.e. service discovery register, secret annotation service, scheduler metrics exporter, etc)
- **Kubelets**: (also commonly referred as "physical nodes") k8s worker nodes which host and manage Pods. Also on Kubelet nodes we have:
  - Containerd - CRI object, responsible for running Pod containers using CGroupsV2.
  - Global services - programs which are deployed on each Kubelet node and help Pods to access global "features", for example reporting metrics, access to AFS and more.
  - CNI/CSIs - see Storage
  - Pod metrics exporter - program which is deployed on each Kubelet node, responsible for collecting Pod stats (CPU/mem usage, networking, etc).
- Other managers/exporters which are vital for successful Neptune operation.

Topology

Neptune topology source of truth is located here. All current clusters and used hardware types for Kubelet nodes are listed here. Some special cases include:

- **Stage clusters (dfw-stage/bwi-stage)**: stage clusters used for Neptune feature testing. Their stability is not guaranteed, so it's heavily discouraged to run services which are even remotely critical there.
- **Changes clusters (global-changes-stage/global-changes)**: Changes-specific clusters, which host majority of Changes Jobs. Not for general use.
- **Team-specific clusters (dfw-mp/ttd-ml-stage/ttd-ml)**: Clusters created for specific needs of specific teams - usually not even being part of regular YAPS ecosystem. Not for general use.
- **Global**: cluster which combines four AWS remote metros (lgw/hhn/hnd/ool) under the single control plane management. Available for general use, in case if remote region deployments are needed.

YAPS → k8s config translation

YAPS Config Service is responsible for generating k8s configs. Currently, it calls yaps_to_neptune binary to perform the translation.