Labels&Annotations

Table of Contents

- Labels and Selectors
- Annotations

1 Labels and Selectors

Labels

- Labels are key/value pairs that are attached to objects, such as pods.
- Labels can be attached to objects at creation time and subsequently added and modified at any time.
- Each object can have a set of key/value labels defined.
- Example labels:
  - **"environment" : "dev", "environment" : "qa", "environment" : "production"**
  - **"tier" : "frontend", "tier" : "backend", "tier" : "cache"**

Labels Selectors

- Unlike **names and UIDs**, labels do not provide uniqueness. In general, we expect many objects to carry the same label(s).
- The Selector matches the label. Labels and selectors are required to make connections between some objects.
- The API currently supports two types of selectors:
  - ○ **equality-based**
  - ○ **set-based**.

Equality-based requirement

- Three kinds of operators are admitted **=,==,!=**.
- The first two represent **equality (and are synonyms)**, while the latter represents i**nequality**.
- For example:
  - ○ environment = production
  - ○ tier != frontend

Equality-based requirement

- A label selector can be made of multiple requirements which are **comma-separated**.
- In the case of multiple requirements, all must be satisfied so the comma separator acts as a **logical AND (&&)** operator.

Equality-based requirement

env=prod,tier!=frontend

pod	**labels**
pod1	env=dev, tier=frontend
pod2	env=dev, tier=backend
pod3	env=qa, tier=frontend
pod4	env=qa, tier=backend
pod5	env=prod, tier=frontend
pod6	env=prod, tier=backend

Equality-based requirement

env=qa,tier=backend

pod	labels
pod1	env=dev, tier=frontend
pod2	env=dev, tier=backend
pod3	env=qa, tier=frontend
pod4	env=qa, tier=backend
pod5	env=prod, tier=frontend
pod6	env=prod, tier=backend

Set-based requirement

- Set-based label requirements allow filtering keys according to a set of values.
- Three kinds of operators are supported:
  - **in,notin and exists (only the key identifier)**.
- For example:
  - environment in (production, qa)
  - tier notin (frontend, backend)
  - partition
  - !partition

'env in (prod, dev)'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

'tier notin (frontend)'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

'tier notin (frontend, backend)'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

'team'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

'!team'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

team=alfa,'env in (dev)'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

'team in (alfa, beta)'

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

team=alfa,team=beta

pod	labels
pod1	team=alfa, env=dev, tier=frontend
pod2	team=alfa, env=dev, tier=backend
pod3	team=alfa, env=prod, tier=frontend
pod4	team=alfa, env=prod, tier=backend
pod5	team=beta, env=dev, tier=frontend
pod6	team=beta, env=dev, tier=backend
pod7	team=beta, env=prod, tier=frontend
pod8	team=beta, env=prod, tier=backend
pod 9	env=qa, tier=frontend
pod10	env=qa, tier=backend

8 Annotations

Annotations

- You can use Kubernetes annotations to attach arbitrary non-identifying metadata to objects.
- Clients such as tools and libraries can retrieve this metadata.
- You can use either labels or annotations to attach metadata to Kubernetes objects.
- Labels can be used to select objects and to find collections of objects that satisfy certain conditions. In contrast, annotations are not used to identify and select objects.

Annotations

storageclass.kubernetes.io/is-default-class

Example: **storageclass.kubernetes.io/is-default-class: "true"**

Used on: StorageClass

When a single StorageClass resource has this annotation set to "true", new PersistentVolumeClaim resource without a class specified will be assigned this default class.

.

THANKS!

Any questions?

You can find me at:

- James@clarusway.com