Nhóm API

- Được thiết kế để làm cho việc sử dụng cực kỳ đơn giản vừa hiểu vừa mở rộng.
- Nhóm API là một đường dẫn tương thích với REST, đóng vai trò là bộ mô tả kiểu cho đối tượng Kubernetes.
- Được tham chiếu bên trong một đối tượng dưới dạng apiVersion và loại.

Định dạng:

/apis/*<nhóm>*/*<phiên bản>*/*<tài nguyên>*

Ví dụ:

/apis/ứng dụng/v1/triển khai /

apis/lô hàng/v1beta1/cronjob

Phiên bản API

- Ba cấp độ hoàn thiện API.
- Cũng được tham chiếu trong đối tượng apiVersion.

Định dạng:

/apis/*<nhóm>*/*<phiên bản>*/*<tài nguyên>*

Ví dụ:

/apis/ứng dụng/v1/triển khai /

apis/lô hàng/v1beta1/cronjob

- Alpha: Có thể có lỗi và có thể thay đổi. Bị tắt theo mặc định.
- Beta: Đã được thử nghiệm và coi là ổn định. Tuy nhiên, Lược đồ API có thể thay đổi. Được bật theo mặc định.
- Ổn định: Đã phát hành, ổn định và lược đồ API sẽ không thay đổi. Được bật theo mặc định.

Mô hình đối tượng

- Các đối tượng là một “bản ghi ý định” hoặc một thực thể bền vững đại diện cho trạng thái mong muốn của đối tượng trong cụm.
- Tất cả các đối tượng PHẢI có apiVersion, loại và đặt các trường lồng nhau siêu dữ liệu.name, siêu dữ liệu.namespace và siêu dữ liệu.uid.

Yêu cầu về mô hình đối tượng

- apiVersion: Phiên bản API Kubernetes của Đối tượng
- loại: Loại đối tượng Kubernetes
- siêu dữ liệu.name: Tên duy nhất của Đối tượng
- siêu dữ liệu.namespace: Tên môi trường có phạm vi mà đối tượng thuộc về (sẽ mặc định là hiện tại).
- siêu dữ liệu.uid: uid (được tạo) cho một đối tượng.

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ nhóm
    không gian tên:mặc định
    uid:f8798d82-1185-11e8-94ce-080027b3c7a6
```

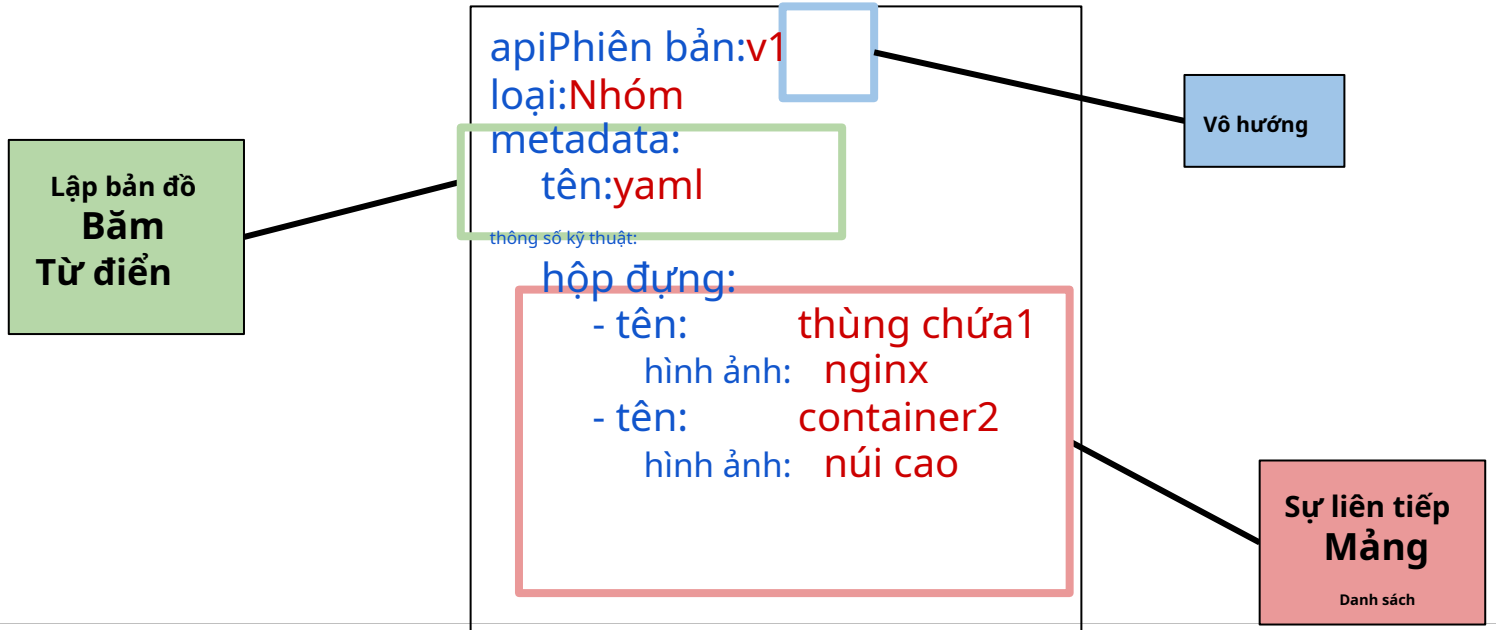
Biểu thức đối tượng - YAML

- Các tệp hoặc cách thể hiện khác của Đối tượng Kubernetes thường được thể hiện trong YAML.
- Tiêu chuẩn tuần tự hóa dữ liệu “Thân thiện với con người”.
- Sử dụng căn chỉnh khoảng trắng (cụ thể là khoảng trắng) để biểu thị quyền sở hữu.
- Ba loại dữ liệu cơ bản:
  - ánh xạ - hàm băm hoặc từ điển,
  - trình tự - mảng hoặc danh sách
  - vô hướng - chuỗi, số, boolean, v.v.

Biểu thức đối tượng - YAML

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:yaml

thông số kỹ thuật:
    hộp đựng:
        - tên:          thùng chứa1
            hình ảnh:   nginx
        - tên:          container2
            hình ảnh:   núi cao
```

Biểu thức đối tượng - YAML



YAML so với JSON

```yaml
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ nhóm

thông số kỹ thuật:
    hộp đựng:
    - tên:          nginx
        hình ảnh:   nginx:ổn định-alpine
        cổng:
        - Cảng container:          80
```

```json
{
  "apiVersion": "v1",
  "loại": "Nhóm",
  "metadata": {
    "tên": "ví dụ nhóm"
  },
  "thông số": {
    "hộp đựng": [
      {
        "tên": "nginx",
        "hình ảnh": "nginx:ổn định-alpine",
        "port": [ { "containerPort":80} ]
      }
    ]
  }
}
```

Mô hình đối tượng - Khối lượng công việc

- Các đối tượng liên quan đến khối lượng công việc trong Kubernetes có thêm thông số và trạng thái của hai trường lồng nhau.
  - spec - Mô tả trạng thái hoặc cấu hình mong muốn của đối tượng được tạo.
  - trạng thái - Được quản lý bởi Kubernetes và mô tả trạng thái thực tế của đối tượng cũng như lịch sử của nó.

Ví dụ về đối tượng khối lượng công việc

Đối tượng mẫu

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ nhóm
thông số kỹ thuật:
    hộp đựng:
    - tên:          nginx
      hình ảnh:    nginx:ổn định-alpine
      cổng:
      - Cảng container:          80
```

Đoạn trạng thái ví dụ

```
trạng thái:
    điều kiện:
    - thời gian thăm dò cuối cùng:vô giá trị
      lần chuyển tiếp cuối cùng:2018-02-14T14:15:52Z trạng
      thái:"ĐÚNG VẬY"
      kiểu:Sẵn sàng
    - thời gian thăm dò cuối cùng:vô giá trị
      lần chuyển tiếp cuối cùng:2018-02-14T14:15:49Z trạng
      thái:"ĐÚNG VẬY"
      kiểu:Đã khởi tạo
    - thời gian thăm dò cuối cùng:vô giá trị
      lần chuyển tiếp cuối cùng:2018-02-14T14:15:49Z trạng
      thái:"ĐÚNG VẬY"
      kiểu:PodĐã lên lịch
```

Sử dụng API

(hay còn gọi là sử dụng CLI)

Phòng thí nghiệm - github.com/mrbobbytables/k8s-intro-tutorials/blob/master/cli

Cốt lõi

Các đối tượng

- Không gian tên
- Vỏ
- Nhãn
- Bộ chọn
- Dịch vụ

Khái niệm và tài nguyên

Khái niệm cốt lõi

- Kubernetes có một số khối xây dựng cốt lõi tạo nên nền tảng cho các thành phần cấp cao hơn của chúng.

Không gian tên

Vỏ

Dịch vụ

Nhãn

Bộ chọn

Không gian tên

- Không gian tên là một cụm hoặc môi trường logic và là phương pháp chính để phân vùng một cụm hoặc truy cập phạm vi.

```
apiPhiên bản:v1
loại:Không gian tên
metadata:
    tên:sản phẩm
    nhãn:
        ứng dụng:MyBigWebApp
```

```
$ kubectl nhận ns --show-labels TÊN
                  TRẠNG THÁI   TUỔI   NHÃN
mặc định          Tích cực     11h    <không có>
kube-public       Tích cực     11h    <không có>
hệ thống kube     Tích cực     11h    <không có>
sản phẩm          Tích cực     6 giây app=MyBigWebApp
```

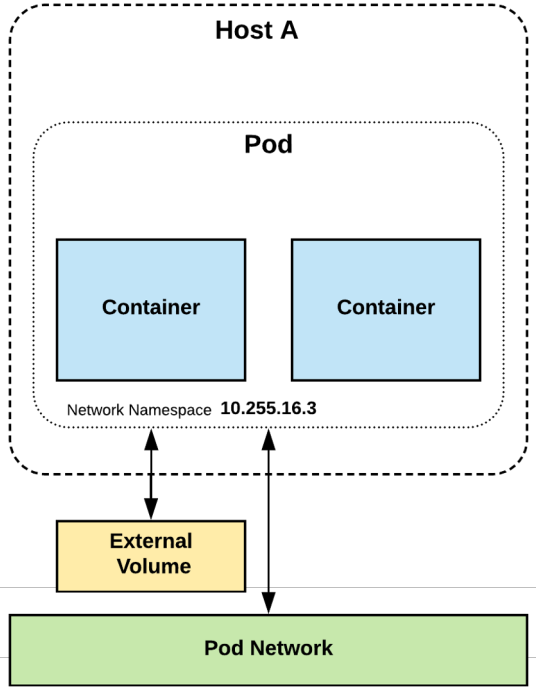
Không gian tên mặc định

- mặc định: Không gian tên mặc định cho bất kỳ đối tượng nào không có không gian tên.
- kube-system: Đóng vai trò là ngôi nhà cho các đối tượng và tài nguyên do chính Kubernetes tạo ra.
- kube-public: Một không gian tên đặc biệt; tất cả người dùng đều có thể đọc được, dành riêng cho việc khởi động và cấu hình cụm.

```
$ kubectl nhận ns --show-labels TÊN
                  TRẠNG THÁI     TUỔI     NHÃN
mặc định          Tích cực       11h      <không có>
kube-public       Tích cực       11h      <không có>
hệ thống kube     Tích cực       11h      <không có>
```

Nhóm

- Đơn vị nguyên tử hay "đơn vị công việc" nhỏ nhất của Kubernetes.
- Khối xây dựng nền tảng của Khối lượng công việc Kubernetes.
- Nhóm là một hoặc nhiều vùng chứa chia sẻ các khối, không gian tên mạng và là một phần của một ngữ cảnh.



Ví dụ về nhóm

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ nhóm

thông số kỹ thuật:
    hộp đựng:
    - tên:          nginx
        hình ảnh:   nginx:ổn định-alpine
        cổng:
        - Cảng container:          80
```

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ về nhiều container thông số kỹ
    thuật:
    hộp đựng:
    - tên:nginx
        hình ảnh:nginx:ổn định-alpine
        cổng:
        - Cảng container:80
        số lượng:
        - tên:html
            mountPath:/usr/share/nginx/html
    - tên:nội dung
        hình ảnh:núi cao:mới nhất
        yêu cầu:["/bin/sh", "-c"]
        lập luận:
            -   trong khi đúng; LÀM
                    ngày >> /html/index.html;
                    ngủ 5;
                xong
        số lượng:
        - tên:html
            mountPath:/html
    khối lượng:
    - tên:html
        trốngDir:{}
```

Thuộc tính vùng chứa nhóm chính

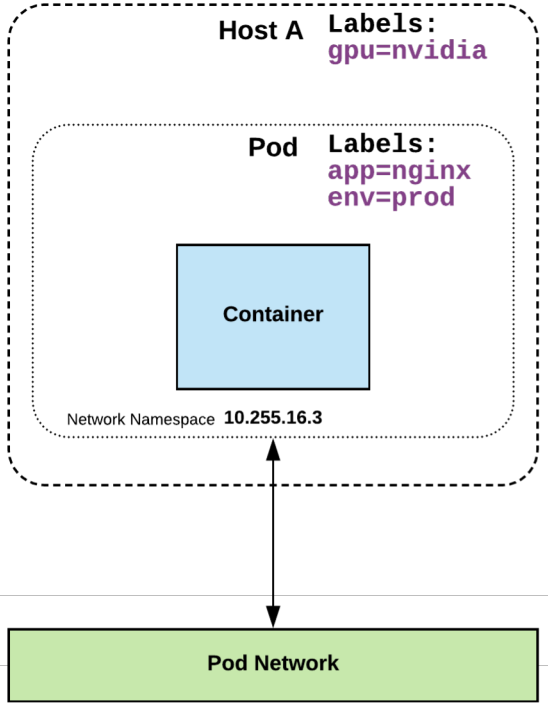
- tên - Tên thùng chứa
- hình ảnh - Hình ảnh thùng chứa
- cổng - mảng các cổng để lộ.
  Có thể được cấp một tên thân thiện và giao thức có thể được chỉ định
- env-mảng biến môi trường
- yêu cầu-Mảng điểm vào (tương đương với DockerĐIỂM VÀO)
- lập luận-Các đối số để truyền vào lệnh (tương đương với DockerCMD)

Thùng đựng hàng

```
tên:        nginx
hình ảnh:   nginx:ổn định-alpine
cổng:
    - Cảng container:          80
        tên:http
        giao thức:      TCP
env:
    - tên:          MYVAR
        giá trị:      nó thật tuyệt
yêu cầu:["/bin/sh",                  "-c"]
lập luận:["tiếng vang ${MYVAR}"]
```

Nhãn

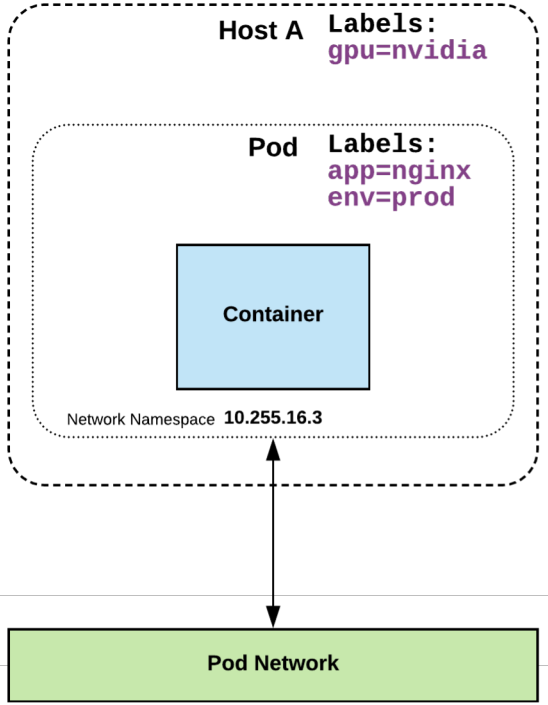
- các cặp khóa-giá trị được sử dụng để xác định, mô tả và nhóm các tập hợp đối tượng hoặc tài nguyên có liên quan với nhau.
- KHÔNG phải là đặc tính duy nhất.
- Có cú pháp chặt chẽ với bộ ký tự hơi hạn chế*.



* https://kubernetes.io/docs/concepts/overview/working-with-objects/labels/#syntax-and-character-set

Ví dụ về nhãn

```
apiPhiên bản:    v1
loại:Nhóm
metadata:
    tên:      ví dụ về nhãn-pod
    nhãn:
        ứng dụng:  nginx
        env:       sản phẩm
thông số kỹ thuật:
    hộp đựng:
    - tên:         nginx
        hình ảnh:  nginx:ổn định-alpine
        cổng:
        - Cảng container:        80
```



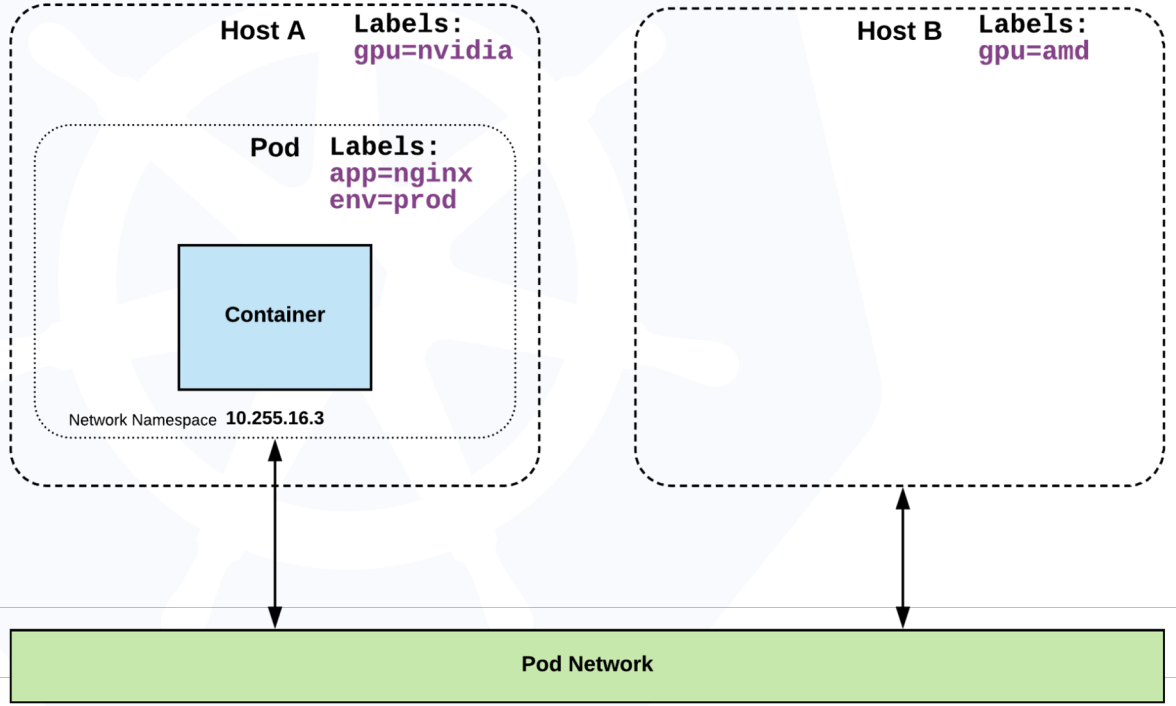
Bộ chọn

Bộ chọn sử dụng nhãn để lọc hoặc chọn đối tượng và được sử dụng xuyên suốt Kubernetes.

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ về nhãn-pod nhãn:

        ứng dụng:nginx
        env:sản phẩm

thông số kỹ thuật:
    hộp đựng:
    - tên:nginx
        hình ảnh:nginx:ổn định-alpine
        cổng:
        - Cảng container:80
    nútChọn:
        gpu:nvidia
```

Ví dụ về bộ chọn

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ về nhãn-pod nhãn:

        ứng dụng:nginx
        env:sản phẩm

thông số kỹ thuật:
    hộp đựng:
    - tên:nginx
        hình ảnh:nginx:ổn định-alpine
        cổng:
        - Cảng container:80
    nútChọn:
        gpu:nvidia
```



Các loại bộ chọn

Dựa trên sự bình đẳngbộ chọn cho phép lọc đơn giản (=,== hoặc !=).

```
bộ chọn:
    matchLabels:
        gpu:nvidia
```

Dựa trên tập hợpbộ chọn được hỗ trợ trên một tập hợp con giới hạn của các đối tượng. Tuy nhiên, chúng cung cấp phương pháp lọc trên một tập hợp các giá trị và hỗ trợ nhiều toán tử bao gồm: TRONG,không phải, Vàhiện hữu.

```
bộ chọn:
    biểu thức khớp:
        - chìa khóa:gpu
          nhà điều hành:TRONG
          giá trị:[“nvidia”]
```

Dịch vụ

- Phương pháp thống nhất để truy cập khối lượng công việc hiện có của Pod.
- Tài nguyên bền vững (không giống Pod)
  - IP tĩnh duy nhất của cụm
  - tên DNS được đặt tên tĩnh

<tên dịch vụ>.<không gian tên>.svc.cluster.local

Dịch vụ

- Nhóm mục tiêu sử dụng bộ chọn dựa trên sự bình đẳng.
- Sử dụng kube-proxy để cung cấp khả năng cân bằng tải đơn giản.
- kube-proxy hoạt động như một daemon tạo các mục cục bộ trong iptables của máy chủ cho mọi dịch vụ.

Loại dịch vụ

- Có 4 loại dịch vụ chính:
  - ClusterIP (mặc định)
  - Cổng nút
  - Cân bằng tải
  - Tên bên ngoài

Dịch vụ IP cụm

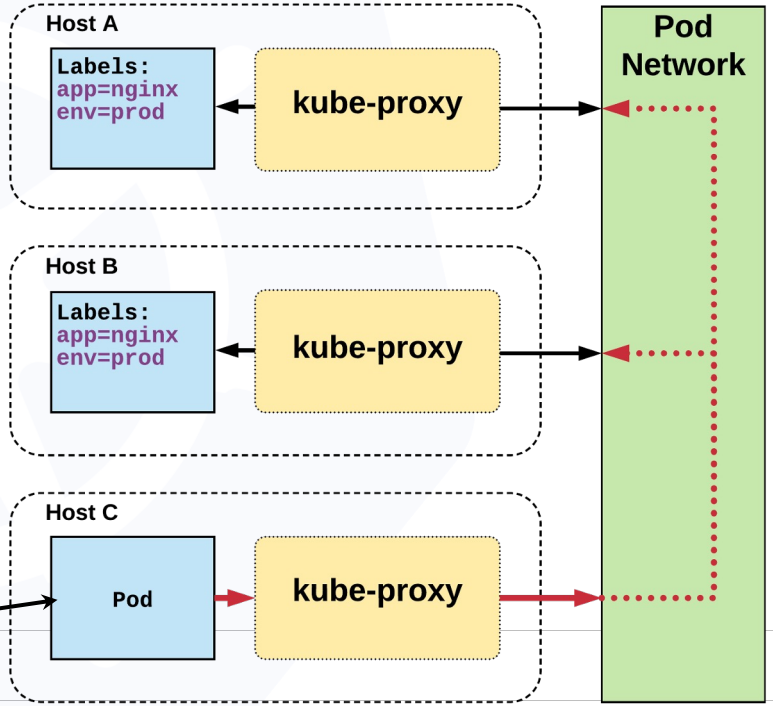
cụm IPdịch vụ

hiển thị một dịch vụ trên một IP ảo nội bộ theo cụm nghiêm ngặt.

```
apiPhiên bản:v1
loại:Dịch vụ
metadata:
    tên:ví dụ-prod

thông số kỹ thuật:
    bộ chọn:
        ứng dụng:  nginx
        env:  sản phẩm
    cổng:
    - giao thức:  TCP
        Hải cảng:80
        cổng đích:  80
```

Dịch vụ IP cụm

Tên:	ví dụ-prod
Bộ chọn:	ứng dụng=nginx,env=prod
Kiểu:	cụm IP
IP:	10.96.28.176
Hải cảng:	<chưa đặt> 80/TCP
Cổng đích:	80/TCP
Điểm cuối:	10.255.16.3:80, 10.255.16.4:80



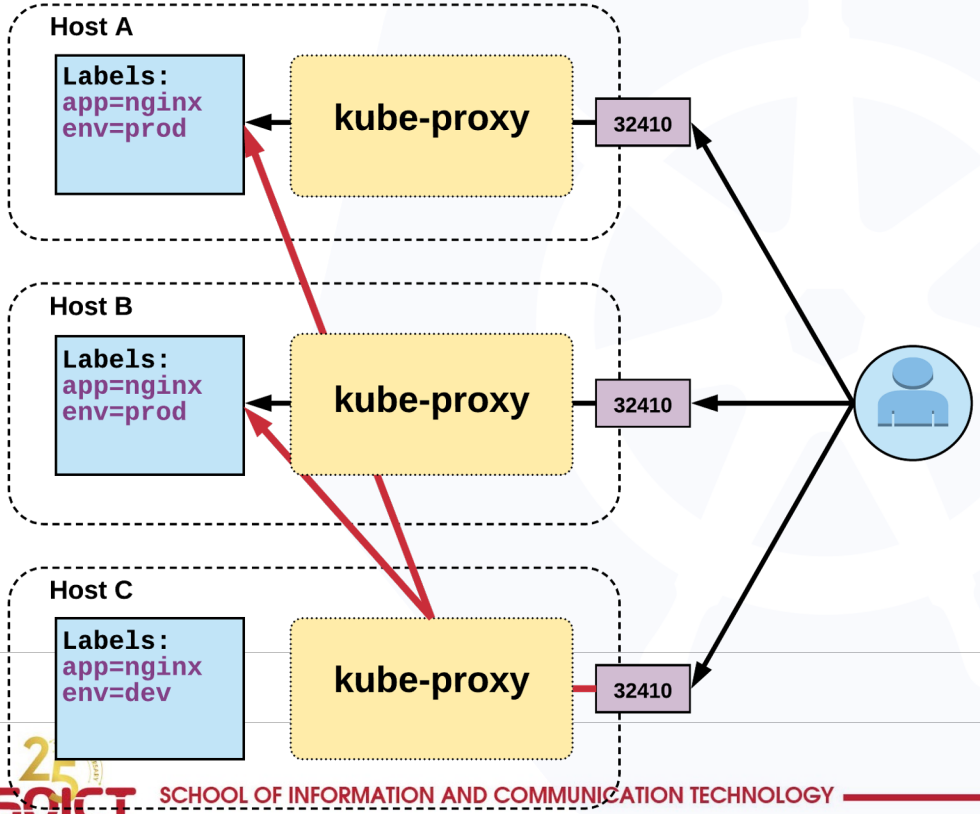

```
/ # nslookup example-prod.default.svc.cluster.local

Tên:        ví dụ-prod.default.svc.cluster.local
Địa chỉ 1: 10.96.28.176 example-prod.default.svc.cluster.local
```

Dịch vụ NodePort

- Dịch vụ NodePort mở rộng dịch vụ ClusterIP.
- Hiển thị một cổng trên IP của mỗi nút.
- Cổng có thể được xác định tĩnh hoặc được lấy động trong phạm vi từ 30000-32767.

```
apiPhiên bản:v1
loại:Dịch vụ
metadata:
    tên:      ví dụ-prod
thông số kỹ thuật:
    kiểu:     Cổng nút
    bộ chọn:
        ứng dụng:  nginx
        env:    sản phẩm
    cổng:
    - nútPort:          32410
        giao thức:     TCP
        Hải cảng:80
        cổng đích:        80
```

Dịch vụ NodePort



Tên:	ví dụ-prod
Bộ chọn:	ứng dụng=nginx,env=prod
Kiểu:	Cổng nút
IP:	10.96.28.176
Hải cảng:	<không đặt> 80/TCP
Cổng đích:	80/TCP
NútPort:	<không đặt> 32410/TCP
Điểm cuối:	10.255.16.3:80, 10.255.16.4:80

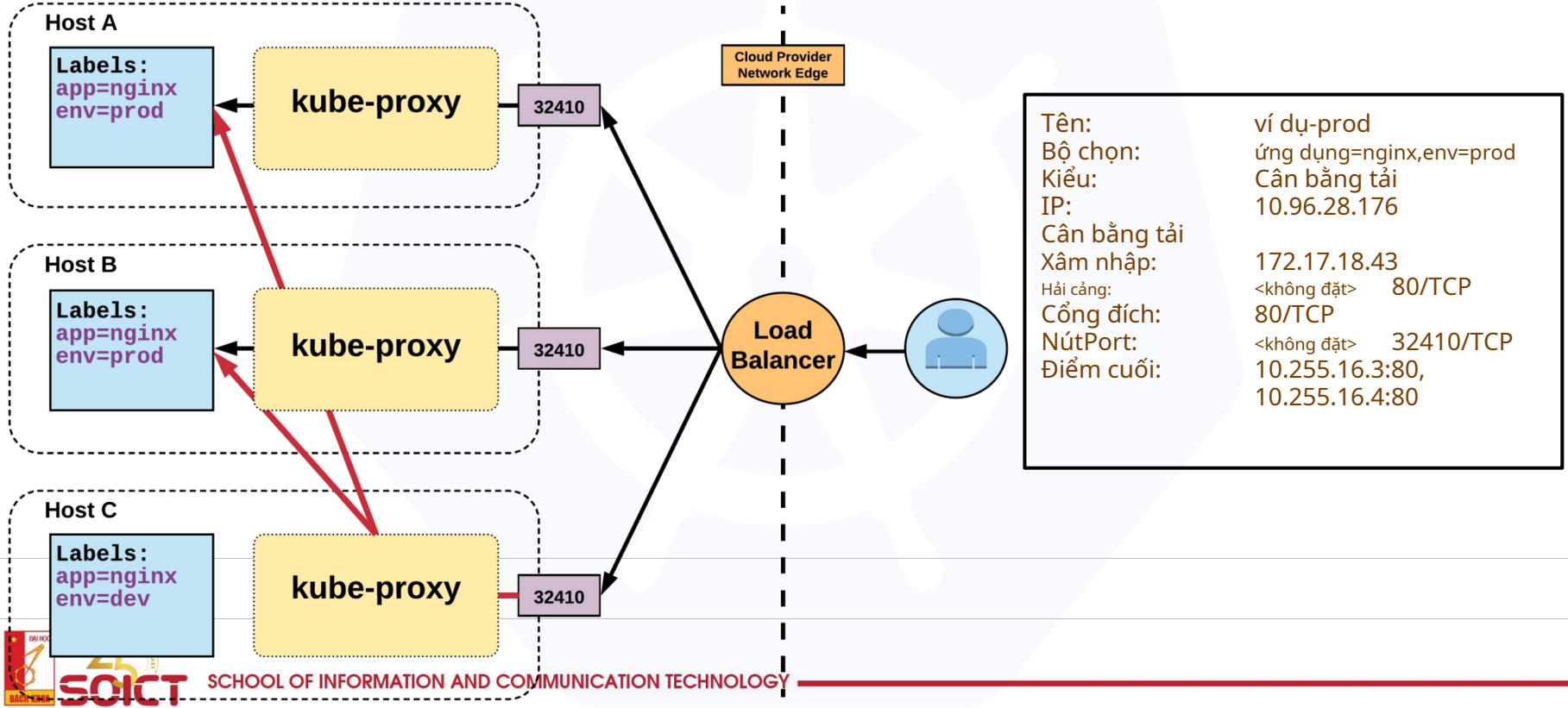
Dịch vụ cân bằng tải

```
apiPhiên bản:v1
loại:Dịch vụ
metadata:
    tên:      ví dụ-prod
thông số kỹ thuật:
    kiểu:     Cân bằng tải
    bộ chọn:
        ứng dụng:  nginx
        env:    sản phẩm
    cổng:
        giao thức:     TCP
        Hải cảng:80
        cổng đích:        80
```

- **Cân bằng tải**dịch vụ mở rộng**NútPort.**
- Hoạt động cùng với hệ thống bên ngoài để ánh xạ cụm

  IP bên ngoài đến dịch vụ được hiển thị.

Dịch vụ cân bằng tải



Dịch vụ tên bên ngoài

```
apiPhiên bản:v1
loại:Dịch vụ
metadata:
    tên:        ví dụ-prod
thông số kỹ thuật:
    kiểu:       Tên bên ngoài
thông số kỹ thuật:
    Tên bên ngoài:        ví dụ.com
```

- **Tên bên ngoài**được sử dụng để tham chiếu các điểm cuối **NGOÀI**cụm.
- Tạo nội bộ **CNAME**Mục nhập DNS có bí danh khác.

Khám phá cốt lõi

Phòng thí nghiệm - github.com/mrbobbytables/k8s-intro-tutorials/blob/master/core

Khối lượng công việc

- Bộ bản sao
- Triển khai
- Bộ Daemon
- Bộ trạng thái
- Công việc
- CronJob

Khái niệm và tài nguyên

Phòng thí nghiệm - github.com/mrbobbytables/k8s-intro-tutorials/blob/master/workloads

Sử dụng khối lượng công việc

Khối lượng công việc

- Khối lượng công việc trong Kubernetes là các đối tượng cấp cao hơn quản lý Pod hoặc các đối tượng cấp cao hơn khác.

- Trong TẤT CẢ CÁC TRƯỜNG HỢP, Mẫu Pod được bao gồm và đóng vai trò là cấp quản lý cơ bản.

Mẫu nhóm

- Bộ điều khiển khối lượng công việc quản lý các phiên bản của Pod dựa trên mẫu được cung cấp.
- Mẫu Pod là thông số kỹ thuật của Pod với siêu dữ liệu hạn chế.
- Sử dụng bộ điều khiển

  Mẫu nhóm để

  tạo ra các nhóm thực tế.

```
apiPhiên bản:v1
loại:Nhóm
metadata:
    tên:ví dụ nhóm
    nhãn:
        ứng dụng:nginx

thông số kỹ thuật:
    hộp đựng:
    - tên:nginx
        hình ảnh:nginx
```

```
bản mẫu:
    metadata:
        nhãn:
            ứng dụng:nginx

    thông số kỹ thuật:
        hộp đựng:
        - tên:nginx
            hình ảnh:nginx
```

Bộ bản sao

- Phương pháp chính để quản lý bản sao nhóm và vòng đời của chúng.
- Bao gồm việc lập kế hoạch, chia tỷ lệ và xóa.
- Công việc của họ rất đơn giản: Luôn đảm bảo số lượng nhóm mong muốn đang chạy.

Bộ bản sao

- bản sao:Các số lượng mong muốn phiên bản của Pod.
- bộ chọn:Bộ chọn nhãn cho **Bộ bản sao**sẽ quản lý**TẤT CẢ**Nhóm các trường hợp mà nó nhắm tới; dù đó là mong muốn hay không.

```
apiPhiên bản:ứng dụng/v1
loại:Bộ bản sao
metadata:
    tên:ví dụ rs

thông số kỹ thuật:
    bản sao:3
    bộ chọn:
        matchLabels:
            ứng dụng:nginx
            env:sản phẩm
    bản mẫu:
        <mẫu nhóm>
```

Bộ bản sao

```
apiPhiên bản:ứng dụng/v1
loại:Bộ bản sao
metadata:
    tên:ví dụ rs

thông số kỹ thuật:
    bản sao:3
    bộ chọn:
        matchLabels:
            ứng dụng: nginx
            env: sản phẩm
    bản mẫu:
        metadata:
            nhãn:
                ứng dụng: nginx
                env: sản phẩm

            thông số kỹ thuật:
                hộp đựng:
                - tên: nginx
                  hình ảnh: nginx:ổn định-alpine
                  cổng:
                  - Cảng container: 80
```

```
$ kubectl nhận nhóm
TÊN                 SẴN SÀNG   TRẠNG THÁI   KHỞI ĐỘNG LẠI   TUỔI
rs-example-9l4dt    1/1        Đang chạy    0               1 giờ
rs-example-b7bcg    1/1        Đang chạy    0               1 giờ
rs-example-mkll2    1/1        Đang chạy    0               1 giờ
```

```
$ kubectl mô tả rs rs-example Tên:
                ví dụ rs
Không gian tên:     mặc định
Bộ chọn:            ứng dụng=nginx,env=prod
Nhãn:               ứng dụng=nginx
                    env=prod
Chú thích:          <không có>
Bản sao:            3 hiện tại / 3 mong muốn
Trạng thái nhóm:    3 Đang chạy / 0 Chờ / 0 Thành công / 0 Thất bại
Mẫu nhóm:
  Nhãn:             ứng dụng=nginx
                    env=prod
  Hộp đựng:
    nginx:
      Hình ảnh:     nginx:ổn định-alpine
      Hải cảng:     80/TCP
      Môi trường:   <không có>
      Gắn kết:      <không có>
  Tập:              <không có>
Sự kiện:
  Kiểu      Lý do       Tuổi   Từ                      Tin nhắn
  ----      ------      ----   ----                     -------
  Bình thường  Thành côngTạo  16 tuổi  bộ điều khiển bản sao  Tạo  nhóm: rs-example-mkll2
  Bình thường  Thành côngTạo  16 tuổi  bộ điều khiển bản sao  Tạo  nhóm: rs-example-b7bcg
  Bình thường  Thành côngTạo  16 tuổi  bộ điều khiển bản sao  Tạo  nhóm: rs-example-9l4dt
```

Triển khai

- Phương thức khai báo quản lý Pod thông qua **Bản sao.**
- Cung cấp chức năng khôi phục và kiểm soát cập nhật.
- Các bản cập nhật được quản lý thông qua**pod-templatehash**nhãn.
- Mỗi lần lặp lại tạo ra một nhãn duy nhất được gán cho cả**Bộ bản sao**và các Pod tiếp theo.

Triển khai

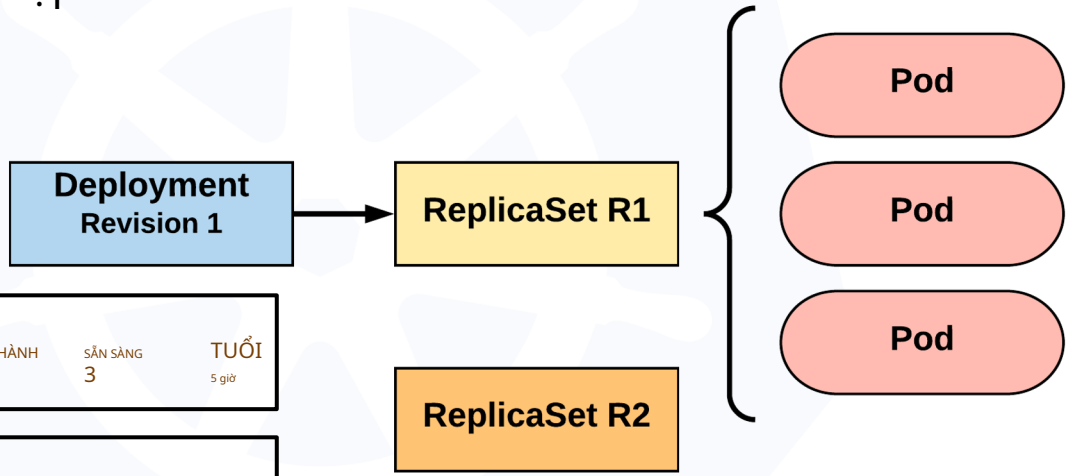
- sửa đổiHistoryLimit:Số lần lặp lại quá trình Triển khai trước đó cần giữ lại.
- chiến lược:Mô tả phương pháp cập nhật Pod dựa trênkiểu. Các tùy chọn hợp lệ là Tạo lạihoặc Cập nhật lăn.
  - Tạo lại:Tất cả các Pod hiện có đều bị hủy trước khi các Pod mới được tạo.
  - Cập nhật lăn:Chu kỳ cập nhật Pod theo các tham số:maxSurgeVà maxKhông có sẵn.

```
apiPhiên bản:ứng dụng/v1
loại:Triển khai
metadata:
    tên:ví dụ triển khai
thông số kỹ thuật:
    bản sao:3
    sửa đổiHistoryLimit:              3
    bộ chọn:
        matchLabels:
            ứng dụng:  nginx
            env:  sản phẩm
    chiến lược:
        kiểu:Cập nhật lăn
        lănCập nhật:
            maxSurge:1
            maxKhông có sẵn:0
    bản mẫu:
        <mẫu nhóm>
```

Triển khai RollingUpdate

Cập nhật mẫu nhóm sẽ tạo ra một mẫu mới**Bộ bản sao**ôn tập.

R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d



$ kubectl lấy bản sao TÊN

	MONG MUỐN	HIỆN HÀNH	SẴN SÀNG	TUỔI
mydep-6766777fff	3	3	3	5 giờ

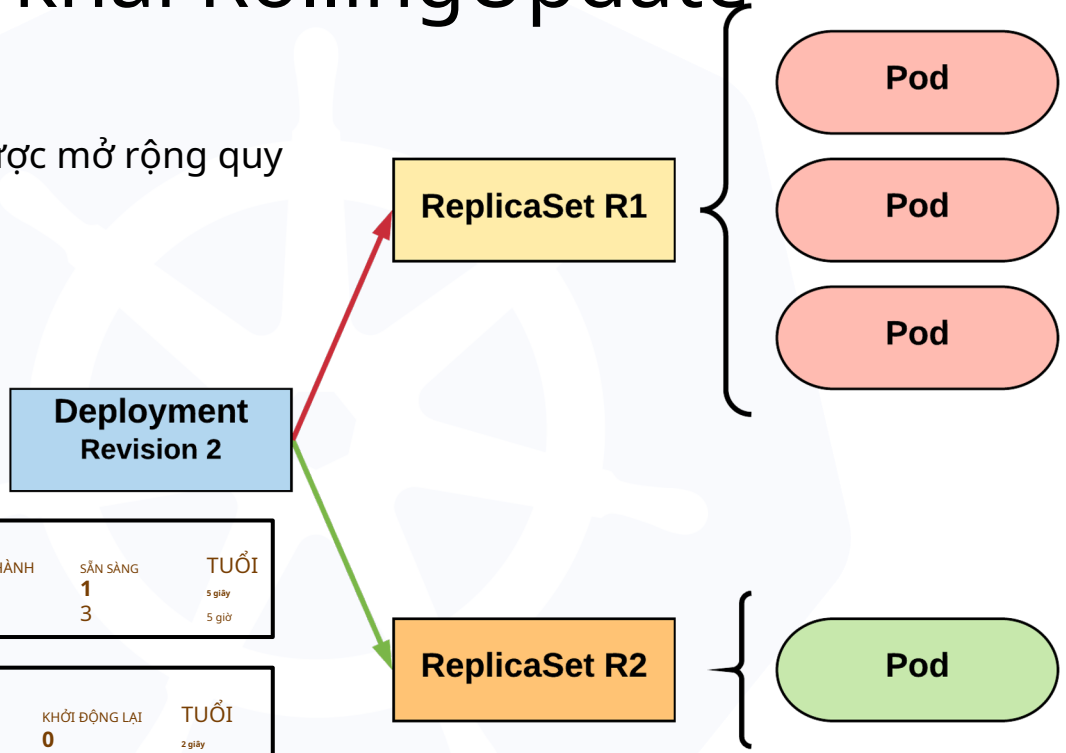
$ kubectl nhận nhóm

TÊN	SẴN SÀNG	TRẠNG THÁI	KHỞI ĐỘNG LẠI	TUỔI
mydep-6766777fff-9r2zn	1/1	Đang chạy	0	5 giờ
mydep-6766777fff-hsfz9	1/1	Đang chạy	0	5 giờ
mydep-6766777fff-sjxhf	1/1	Đang chạy	0	5 giờ

Triển khai RollingUpdate

Mới**Bộ bản sao**ban đầu được mở rộng quy mô dựa trênmaxSurge.

R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d



$ kubectl lấy bản sao TÊN

	MONG MUỐN	HIỆN HÀNH	SẴN SÀNG	TUỔI
mydep-54f7ff7d6d	**1**	**1**	**1**	5 giây
mydep-6766777fff	2	3	3	5 giờ

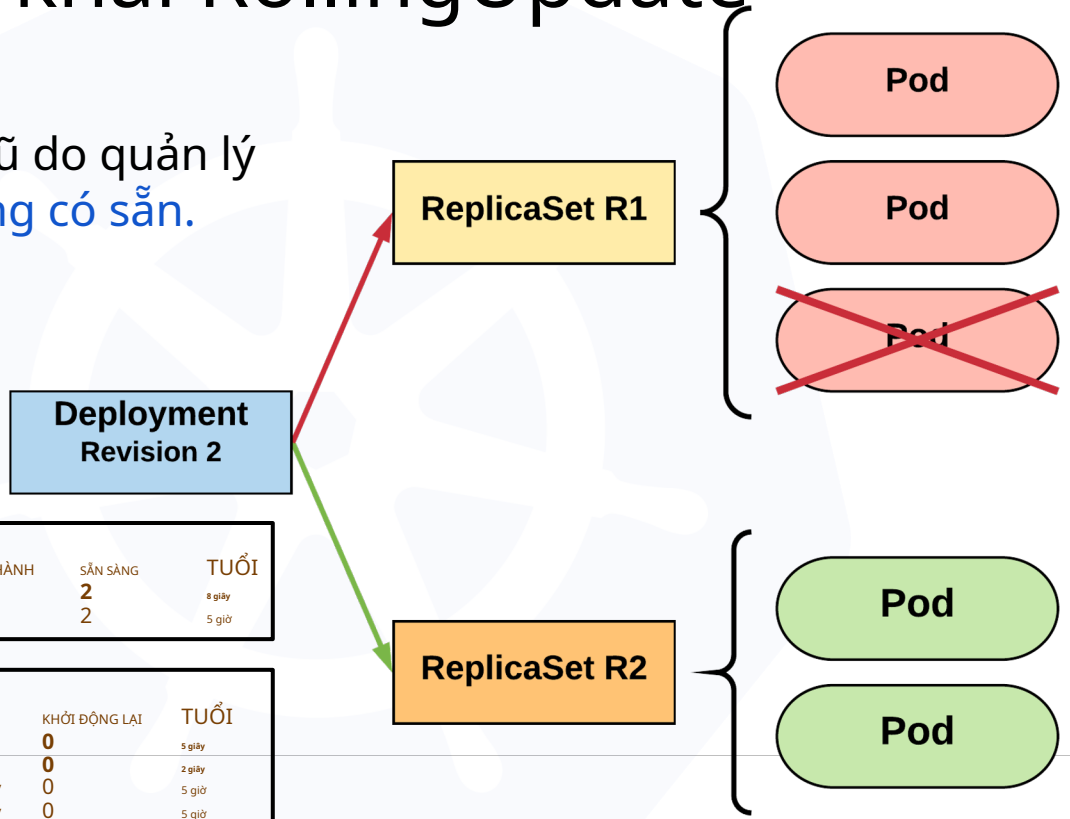
$ kubectl nhận nhóm

TÊN	SẴN SÀNG	TRẠNG THÁI	KHỞI ĐỘNG LẠI	TUỔI
mydep-54f7ff7d6d-9gvll	**1/1**	**Đang chạy**	**0**	2 giây
mydep-6766777fff-9r2zn	1/1	Đang chạy	0	5 giờ
mydep-6766777fff-hsfz9	1/1	Đang chạy	0	5 giờ
mydep-6766777fff-sjxhf	1/1	Đang chạy	0	5 giờ

Triển khai RollingUpdate

Loại bỏ dần các Pod cũ do quản lý maxSurgeVàmaxKhông có sẵn.

R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d



$ kubectl lấy bản sao TÊN

	MONG MUỐN	HIỆN HÀNH	SẴN SÀNG	TUỔI
mydep-54f7ff7d6d	**2**	**2**	**2**	8 giây
mydep-6766777fff	2	2	2	5 giờ

$ kubectl nhận nhóm

TÊN	SẴN SÀNG	TRẠNG THÁI	KHỞI ĐỘNG LẠI	TUỔI
mydep-54f7ff7d6d-9gvll	**1/1**	**Đang chạy**	**0**	5 giây
mydep-54f7ff7d6d-cqvlq	**1/1**	**Đang chạy**	**0**	2 giây
mydep-6766777fff-9r2zn	1/1	Đang chạy	0	5 giờ
mydep-6766777fff-hsfz9	1/1	Đang chạy	0	5 giờ

Rolli

Loại bỏ Pod cũ
maxSurgeVàtối đaU

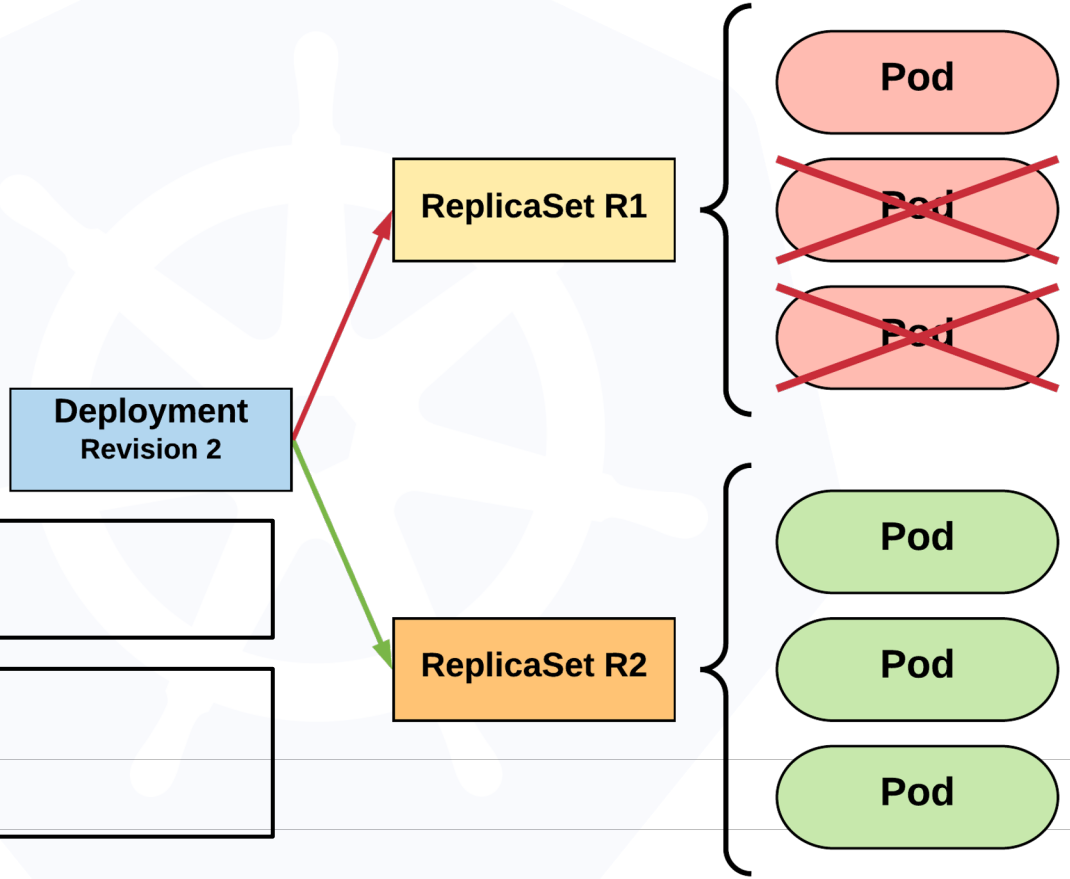
R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d

$ kubectl lấy bản sao TÊN

	MONG MUỐN	C
mydep-54f7ff7d6d	**3**	**3**
mydep-6766777fff	0	1

$ kubectl nhận nhóm

TÊN	SẴN SÀNG	THỐNG KÊ
mydep-54f7ff7d6d-9gvll	**1/1**	**Chạy**
mydep-54f7ff7d6d-cqvlq	**1/1**	**Chạy**
mydep-54f7ff7d6d-gccr6	**1/1**	**Chạy**
mydep-6766777fff-9r2zn	1/1	Chạy



Rolli

Loại bỏ Pod cũ
maxSurgeVàtối đaU

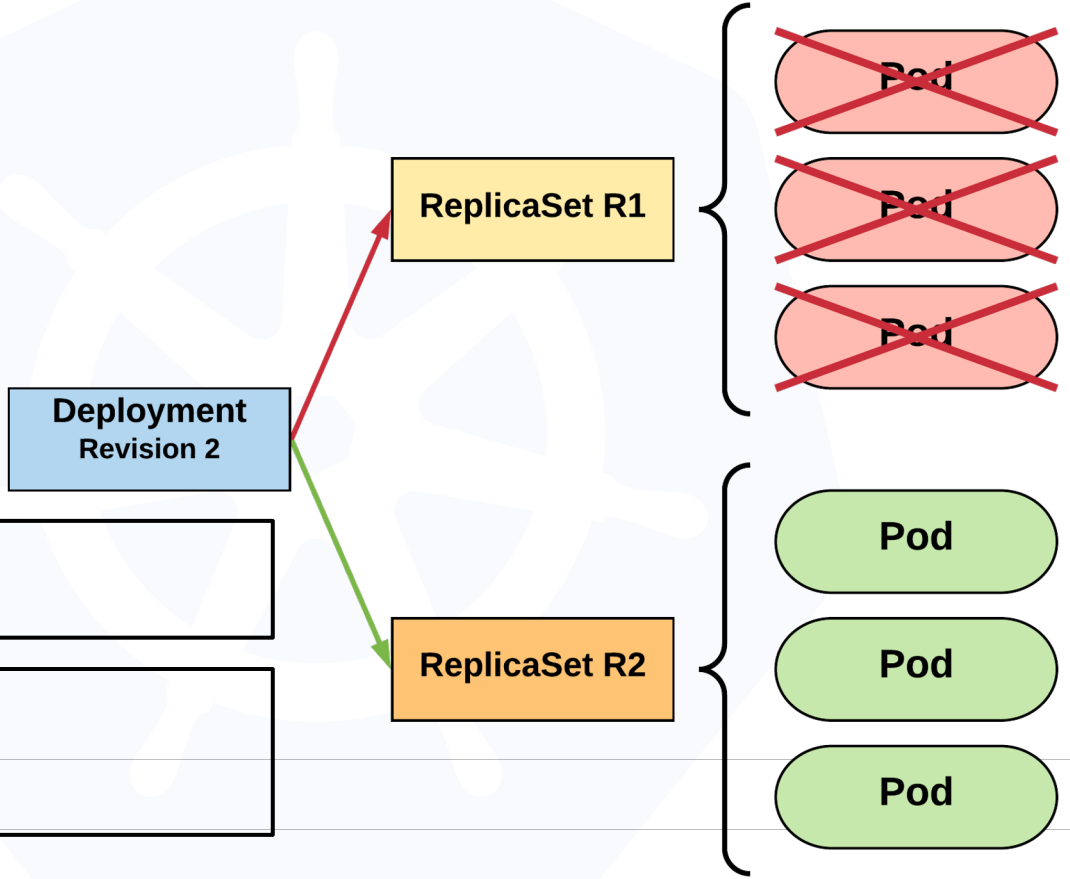
R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d

$ kubectl lấy bản sao TÊN

	MONG MUỐN	C
mydep-54f7ff7d6d	**3**	**3**
mydep-6766777fff	0	0

$ kubectl nhận nhóm

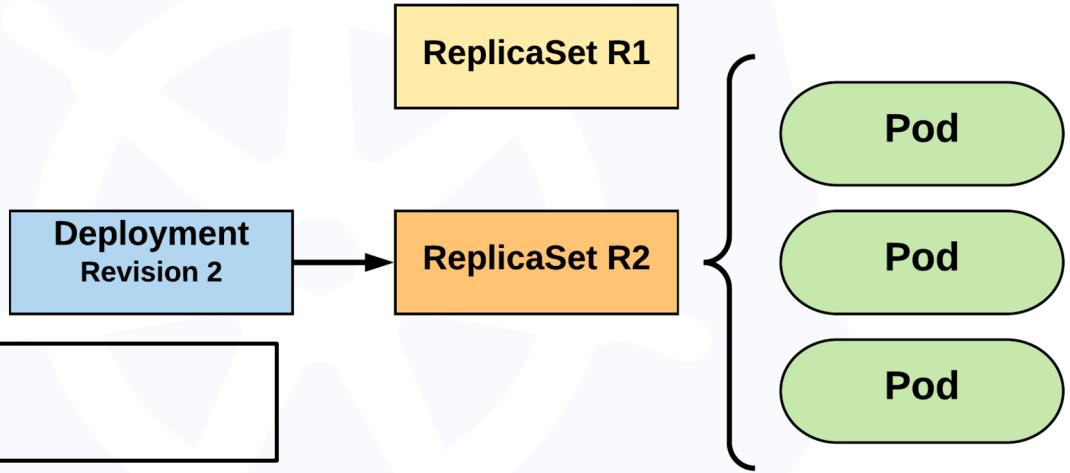
TÊN	SẴN SÀNG	THỐNG KÊ
mydep-54f7ff7d6d-9gvll	**1/1**	**Chạy**
mydep-54f7ff7d6d-cqvlq	**1/1**	**Chạy**
mydep-54f7ff7d6d-gccr6	**1/1**	**Chạy**



Triển khai RollingUpdate

Cập nhật lên dep mới hoàn thành.

R1 pod-template-băm:
676677fff
R2 pod-template-băm:
54f7ff7d6d



$ kubectl lấy bản sao TÊN

	MONG MUỐN	C
mydep-54f7ff7d6d	**3**	**3**
mydep-6766777fff	0	0

$ kubectl nhận nhóm

TÊN	SẴN SÀNG	TRẠNG THÁI	KHỞI ĐỘNG LẠI	TUỔI
mydep-54f7ff7d6d-9gvll	**1/1**	**Đang chạy**	**0**	12 giây
mydep-54f7ff7d6d-cqvlq	**1/1**	**Đang chạy**	**0**	10 giây
mydep-54f7ff7d6d-gccr6	**1/1**	**Đang chạy**	**0**	7 giây

DaemonSet

- Đảm bảo rằng tất cả các nút phù hợp với tiêu chí nhất định sẽ chạy một phiên bản của Pod được cung cấp.
- Họ**đường vòng**cơ chế lập kế hoạch mặc định.
- Lý tưởng cho các dịch vụ trên toàn cụm như chuyển tiếp nhật ký hoặc theo dõi tình trạng.
- Các bản sửa đổi được quản lý thông qua một**bộ điều khiển sửa đổi**nhãn.

DaemonSet

- sửa đổiHistoryLimit:Số lần lặp lại trước đó của DaemonSet cần giữ lại.
- cập nhậtChiến lược:Mô tả phương pháp cập nhật Pod dựa trênkiểu. Các tùy chọn hợp lệ là Cập nhật lănhoặcBậtXóa.
  - Cập nhật lăn:Xoay vòng thông qua việc cập nhật các Pod theo giá trị của maxKhông có sẵn.
  - BậtXóa:Phiên bản mới của Pod được triển khai**CHỈ MỘT**sau khi phiên bản hiện tại bị xóa.

```
apiPhiên bản:ứng dụng/v1
loại:DaemonSet
metadata:
    tên:ds-ví dụ
thông số kỹ thuật:
    sửa đổiHistoryLimit:3 bộ chọn:

        matchLabels:
            ứng dụng:nginx
    cập nhậtChiến lược:
        kiểu:Cập nhật lăn
        lănCập nhật:
            maxKhông có sẵn:1
    bản mẫu:
        thông số kỹ thuật:
            nútChọn:
                loại nút:bờ rìa
            <mẫu nhóm>
```