



云原生边缘计算公开课

Cloud Native Edge Computing

张逸飞

浙江大学SEL实验室
谐云科技有限公司



云原生边缘计算节点管理及安全接入

Cloud Native Edge Computing

- 云原生边缘计算公开课 -



目录

Contents

边缘应用跨地域部署场景及问题

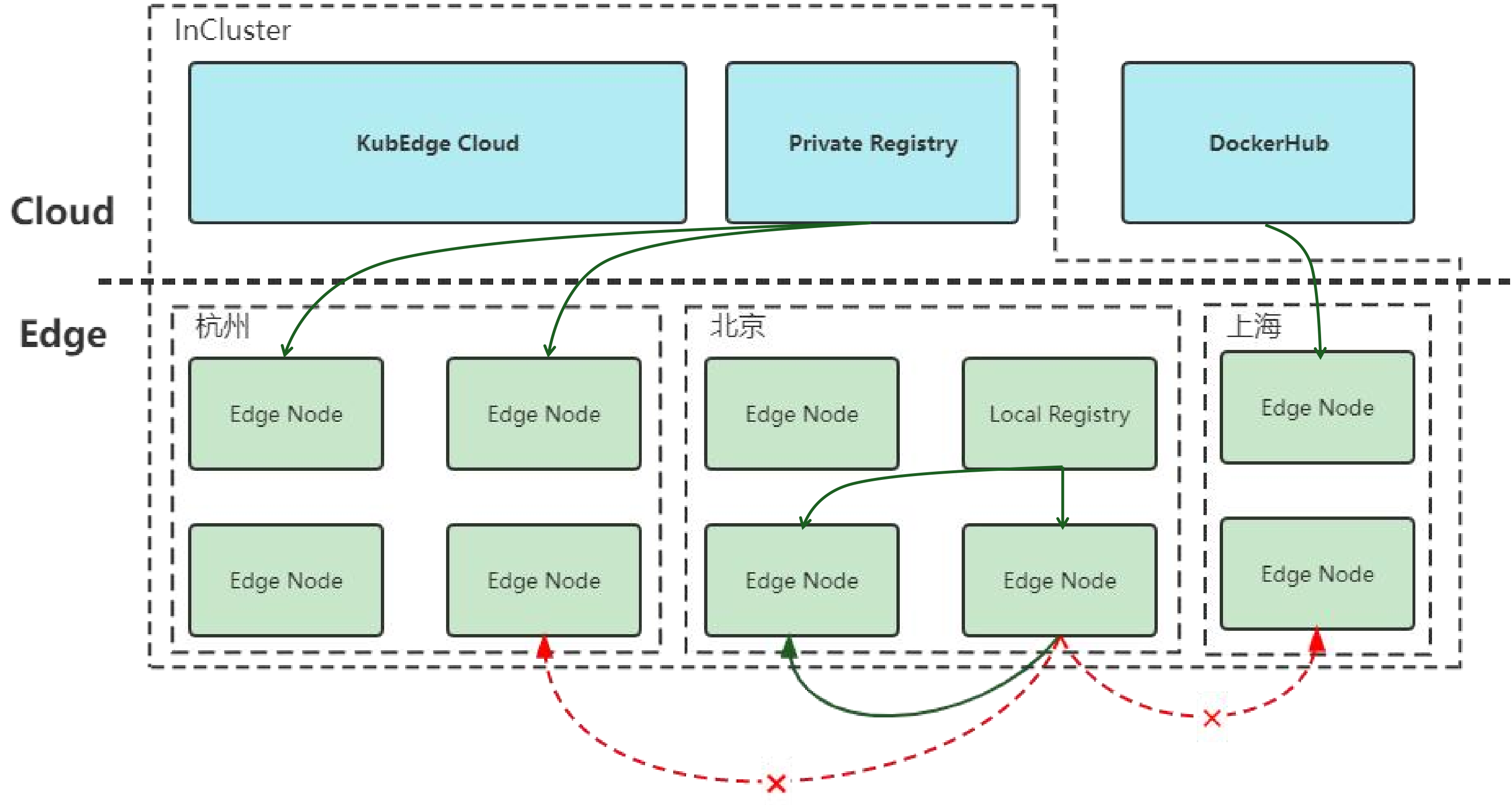
边缘节点分组管理

实例展示

边缘应用跨地域部署场景及问题

KubeEdge.io

边缘应用跨地域部署场景及问题



- 1. 边缘节点规模不同
- 2. 不同地域网络不互通
- 3. 不同地域对镜像的需求不同

边缘应用跨地域部署场景及问题

1. 边缘节点规模不同

—————>

指定副本数量

2. 不同地域网络不互通

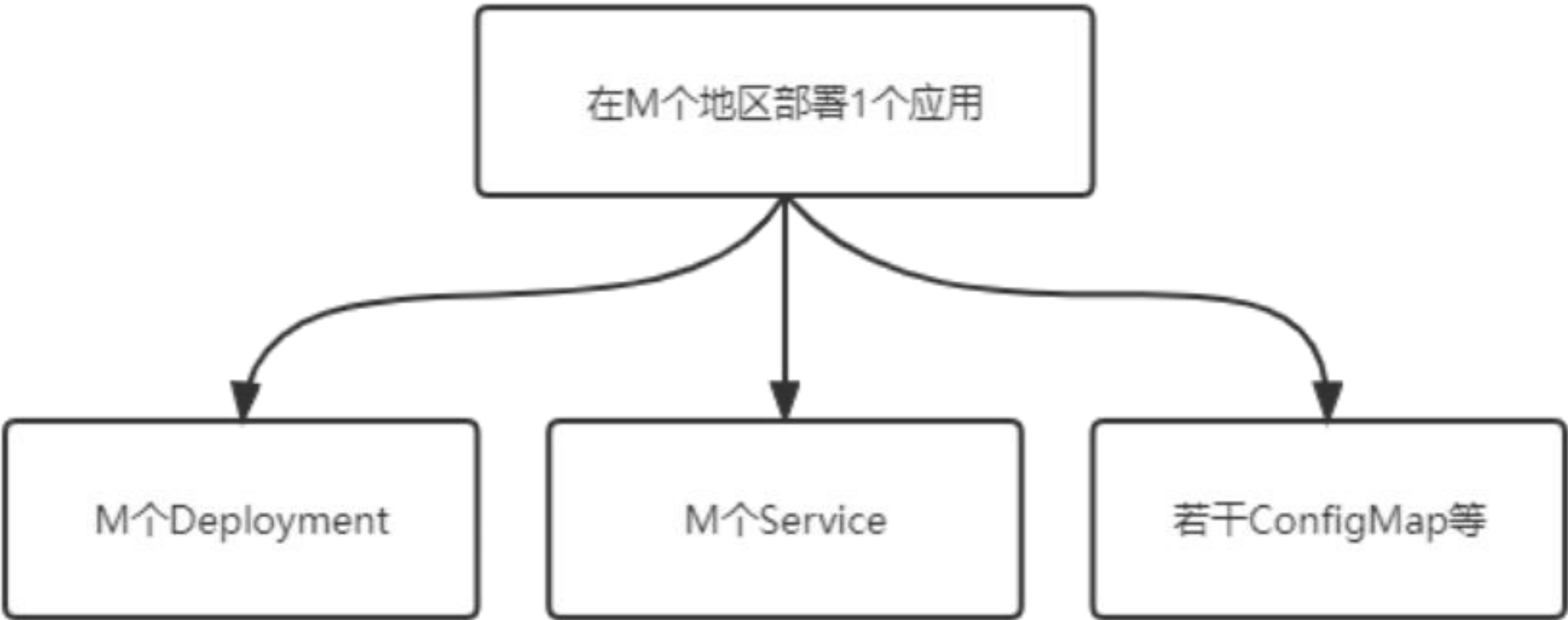
—————>

访问本地的Endpoints

3. 不同地域的镜像不同

—————>

指定使用的镜像
-
- 为每个地区应用并管理单独的Deployment和服务资源



应用生命周期管理复杂导致运维成本提高

02

边缘节点分组管理

KubeEdge.io

边缘节点分组管理

1.节点分组

将不同地区的边缘节点按照节点组的形式组织

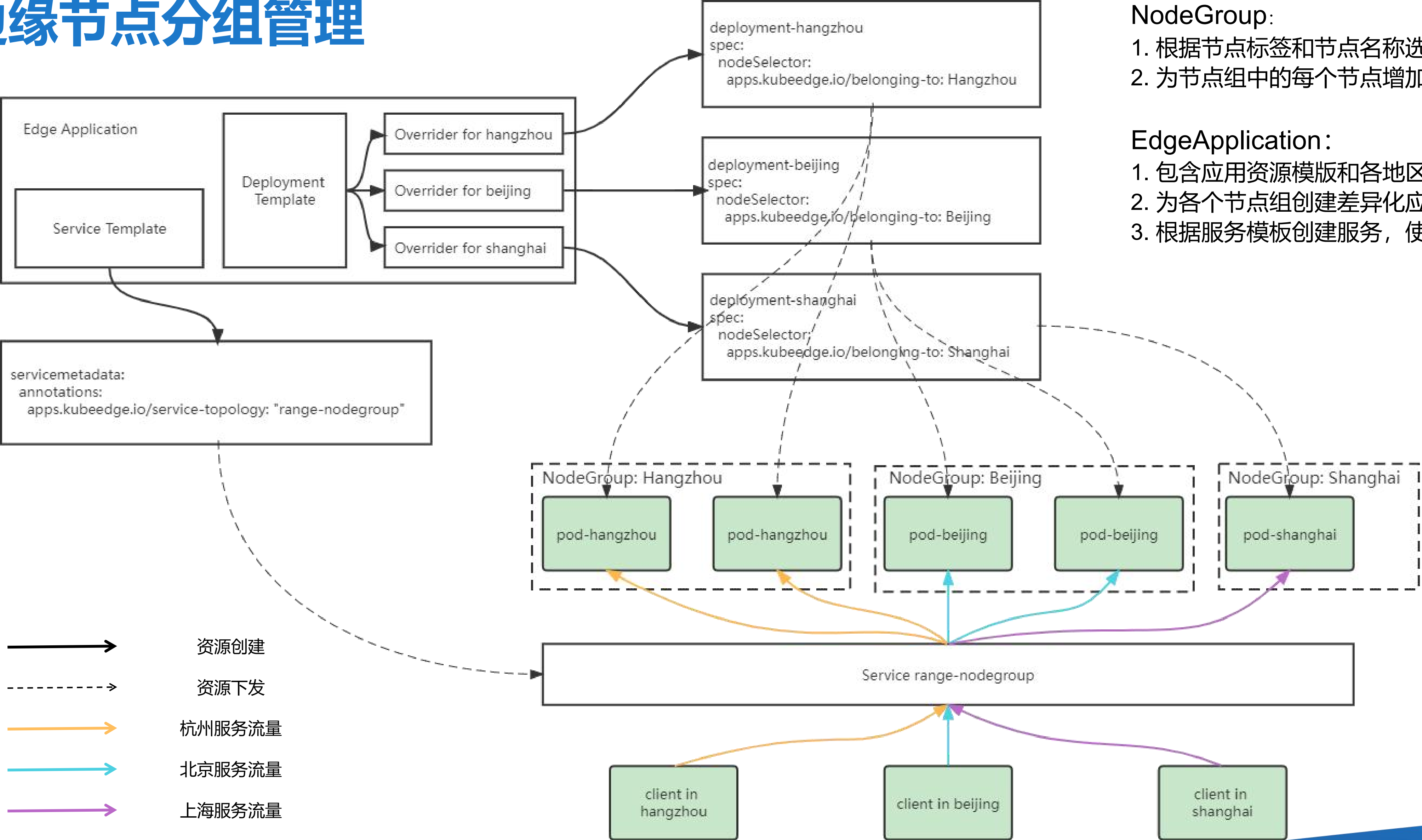
2.边缘应用

将应用资源整体打包并满足不同节点组之间的差异化部署需求

3.流量闭环

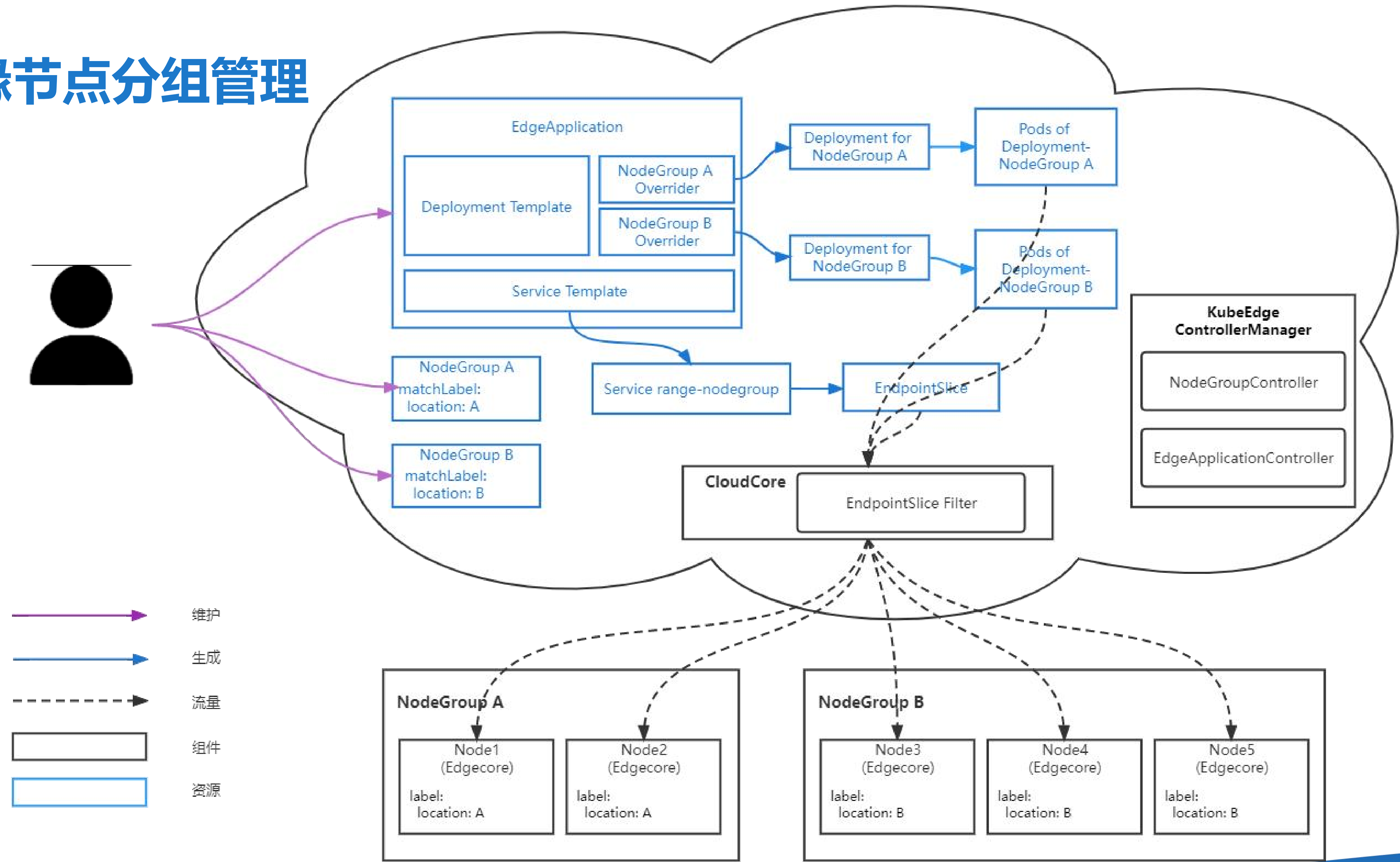
将服务流量限制在同一节点组中

边缘节点分组管理



- NodeGroup:**
1. 根据节点标签和节点名称选取节点形成节点组
 2. 为节点组中的每个节点增加belonging-to标签
- EdgeApplication:**
1. 包含应用资源模版和各地区差异化配置信息
 2. 为各个节点组创建差异化应用实例
 3. 根据服务模板创建服务，使其拓扑范围为节点组内

边缘节点分组管理



03

实例展示

KubeEdge.io

实例展示

在开始之前:

1. 开启AKE (Autonomic Kube-API Endpoint) 特性

AKE特性可以使边缘的kube-proxy组件正常运行, 通过cloudcore-edgecore之间建立的云边信道list/watch所需资源。

CloudCore配置:

开启DynamicController组件, 在cloudcore配置中:

```
modules:
  ...
  dynamicController:
    enable: true
  ...
```

EdgeCore配置:

开启MetaServer组件, 在edgecore配置中 (通常为/etc/kubeedge/config/edgecore.yaml) :

```
modules:
  ...
  metaManager:
    metaServer:
      enable: true
  ...
```

2. 配置Kube-Proxy

将kubeconfig的server地址配置为http://127.0.0.1:10550 (10550为MetaServer端口), 通常通过修改kube-proxy的configmap进行。

```
data:
  ...
  kubeconfig.conf: |-
  ...
  clusters:
  - cluster:
    ...
    server: http://127.0.0.1:10550
    name: default
  ...
```

3. 配置完成后重启上述组件

实例展示

启用分组管理特性：

1. 在k8s中创建节点分组管理CRD

```
kubectl apply -f build/crds/apps
```

该命令会安装NodeGroup和EdgeApplication两个CRD到集群中，如果使用keadm init来初始化kubedge，则该CRD会自动安装；否则需要执行上述命令来手动安装。

2. 部署controllers

```
kubectl apply -f build/controllermanager
```

该命令会在集群的云端节点上部署controllermanager，其中包含NodeGroupController和EdgeApplicationController，分别用于两个CRD的管理，同时也配套安装了相关RBAC规则。

实例展示

NodeGroup用法说明:

NodeGroup会选取节点组成节点组，主要有两种选取方式：节点名和标签。被选取的节点会被额外加上 apps.kubeedge.io/belonging-to 的标签。

1. 根据节点名选取节点

可以通过指定需要被放入节点组的节点名。

2. 根据标签选取节点

多个标签之间是AND关系，节点需要同时具有这些标签才会被放入节点组中。当新接入的节点拥有满足条件的标签时，会被自动地放入相应节点组中。

3. NodeGroup生命周期

当删除节点组时，节点组中节点上的 belonging-to 标签会被自动删除，即节点组中的节点会自动退出节点组。

```
apiVersion: apps.kubeedge.io/v1alpha1
kind: NodeGroup
metadata:
  name: beijing
spec:
  nodes:
    - node110
```

```
apiVersion: apps.kubeedge.io/v1alpha1
kind: NodeGroup
metadata:
  name: hangzhou
spec:
  matchLabels:
    location: hangzhou
```

实例展示

EdgeApplication用法说明

1. WorkloadTemplate

WorkloadTemplate中存放的是应用所需要的资源模板，如Deployment，Service，ConfigMap等。

对于Deployment：

会根据WorkloadScope中的差异化配置生成各节点组的实例。

对于Service：

会为其增加range-nodegroup的标签。

对于其他资源：

会直接在集群中创建。

2. WorkloadScope

WorkloadScope中存放的是节点组的差异化配置信息，可以为特定节点组设置Override。目前支持设置replicas override和image override。

replicas override：

设置的值会覆盖Deployment模板中的replicas字段。

image override：

可以修改镜像地址的Registry、Repository、Tag三个部分
每个部分可以进行的修改有add、remove、replace。



3. EdgeApplication生命周期

更新：

当更新EdgeApplication时，其创建的子资源会被自动更新，同时兼容Deployment的滚动更新策略，在同节点组的Pod实例间进行滚动更新。

删除：

当删除EdgeApplication时，所有该EdgeApplication创建的资源都会被自动删除。

```
apiVersion: apps.kubeedge.io/v1alpha1
kind: EdgeApplication
metadata:
  name: hostname-app
spec:
  workloadTemplate:
    manifests:
      - apiVersion: apps/v1
        kind: Deployment
        metadata:
          name: hostname-app
          namespace: default
        ...
      - apiVersion: v1
        kind: Service
        metadata:
          name: hostname-svc
          namespace: default
        ...
  workloadScope:
    targetNodeGroups:
      - name: hangzhou
        overrides:
          replicas: 3
        ...
      - name: beijing
        overrides:
          imageOverrides:
            - component: "Tag"
              operator: "replace"
              value: "1.2"
        ...
```

添加社区小助手

课后答疑 | 大咖交流 | 社区礼品



Github: <https://github.com/kubeedge>



Website: <https://kubeedge.io>



Slack Channel: <https://kubeedge.slack.com>



KubeEdge公众号



课程小助手

