

Alibaba Cloud Elastic Container Instance

Image

Issue: 20200525









Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

- 1.** You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
- 2.** No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
- 3.** The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
- 4.** This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

- 5.** By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.
- 6.** Please contact Alibaba Cloud directly if you discover any errors in this document.

Document conventions

Style	Description	Example
	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: If the weight is set to 0, the server no longer receives new requests.
	A note indicates supplemental instructions, best practices, tips, and other content.	 Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click Settings > Network > Set network type.
Bold	Bold formatting is used for buttons, menus, page names, and other UI elements.	Click OK.
Courier font	Courier font is used for commands.	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid Instance_ID</code>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>

Style	Description	Example
{ } or {a b}	This format is used for a required value, where only one item can be selected.	switch {active stand}

Contents

Legal disclaimer.....	I
Document conventions.....	I
1 Use image caches.....	1
2 Use an image cache CRD to accelerate pod creation.....	5

1 Use image caches

To run containers, Elastic Container Instance (ECI) needs to pull the specified images. However, pulling large-sized images over an unstable network is time-consuming when you start ECIs. To resolve this issue, you can use the image cache feature.

To use the image cache feature, follow these steps:

1. Create an image cache.
2. Query the image cache.
3. Create an ECI by using the image cache.

Billing

You are charged for creating and using image caches.

Fees of creating an image cache

After you submit the request for creating an image cache, Alibaba Cloud uses 1 vCPU and 2 GiB memory of ECI resources to create an image cache. To create an image cache, you need to pay the following fees:

- The fee of 1 vCPU and 2 GiB memory. The billing stops after the image cache is created. For more information, see [Billing](#).
- The fee of the disk used for storing image snapshots. The default disk is an enhanced solid-state disk (ESSD) PL1 with 20 GiB memory. You can customize the size of the disk. The minimum size is 20 GiB. For more information about the price of disks, see [Disk pricing](#).
- The fee of an image snapshot. An image snapshot is generated each time you create an image cache. For more information, see [Snapshot billing methods](#).

You can log on to the [ECI console](#) to view the created image cache.

Fees of using an image cache

If you use an image cache to create an ECI, a disk created for storing image snapshots is mounted to the ECI. For more information, see [Billing](#).

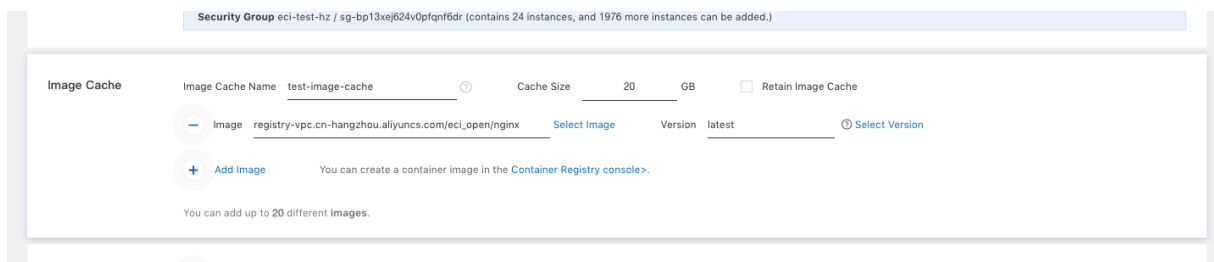
Assume that you want to create an ECI in the China (Beijing) region. The ECI needs an ESSD PL1 with 20 GiB memory to store image snapshots. You need to pay CNY 0.04 per hour for the disk.

Create an image cache

- By default, an ECI with 1 vCPU and 2 GiB memory is used to create an image cache. A fee will be generated during the creation.
- Currently, an image cache can contain up to 20 images. You can customize the size of the image cache. The default size is 20 GiB.
- You can set the retention period for the image cache. The image cache is automatically deleted after it expires. By default, the retention period is not specified. You can specify the retention period if needed.
- If you select private images that are not hosted by Alibaba Cloud Container Registry, specify the address, username, and password of the private image repository. ECI needs the information to pull images from the private image repository.
- If ECI needs to pull the selected images over the Internet, such as Docker images, you must enable Internet access for ECIs. For more information, see [Enable Internet access](#).

Create an image cache in the ECI console

You can create an image cache in the [ECI console](#) through simple operations.



Create an image cache by using an API operation

You can view and debug the API operation used to create an image cache in [OpenAPI Explorer](#).

Parameter description:

```
ImageCacheName=normalimage // The name of the image cache.
Image.1=registry-vpc.cn-beijing.aliyuncs.com/eci_open/nginx:1.15.10
Image.2=registry-vpc.cn-beijing.aliyuncs.com/eci_open/node:11
Image.3=registry-vpc.cn-beijing.aliyuncs.com/eci_open/mysql:5.6
RetentionDays=30 // The retention period of the image cache. Valid values: 1 to 65536.
Unit: days.
```

ImageCacheSize=20 // The size of the image cache. Valid values: 20 to 500. Default value : 20. Unit: GiB.

Query the image cache

You can view and debug the API operation used to query an image cache in [OpenAPI Explorer](#).

Sample response:

```
{
  "ImageCaches": [
    {
      "CreationTime": "2019-12-18T03:18:02Z",
      "Status": "Ready",
      "ContainerGroupId": "eci-2zeh184ra0em1rvku1f",
      "ImageCacheId": "imc-2zeh184ra0em1rvku1f", // The ID of the image cache.
      "ImageCacheName": "normalimage",
      "Events": [...],
    },
    {
      "SnapshotId": "s-2ze10h8c369eb7spa2oe",
      "Images": [
        "registry-vpc.cn-beijing.aliyuncs.com/eci_open/nginx:1.15.10",
        "registry-vpc.cn-beijing.aliyuncs.com/eci_open/node:11",
        "registry-vpc.cn-beijing.aliyuncs.com/eci_open/mysql:5.6"
      ],
      "ExpireDateTime": "2020-01-17T20:49:47Z", // The time when the image cache expires.
      "Progress": "100%" // The percentage of the creation progress.
    }
  ],
  "RequestId": "1ABEBF6E-6780-49CA-8C34-EEC6017F3012"
}
```

Create an ECI by using the image cache

You can use the image cache to create an ECI in one of the following ways:

- Specify an image cache.
- Enable automatic image cache match. ECI automatically selects and uses the optimal image cache.

You can view and debug the API operation used to create an ECI in [OpenAPI Explorer](#).



Notice:

When you create an ECI by using the image cache, set the ImagePullPolicy parameter to IfNotPresent for the containers. Otherwise, the effect of using an image cache to accelerate ECI creation will be greatly affected.

Specify an image cache

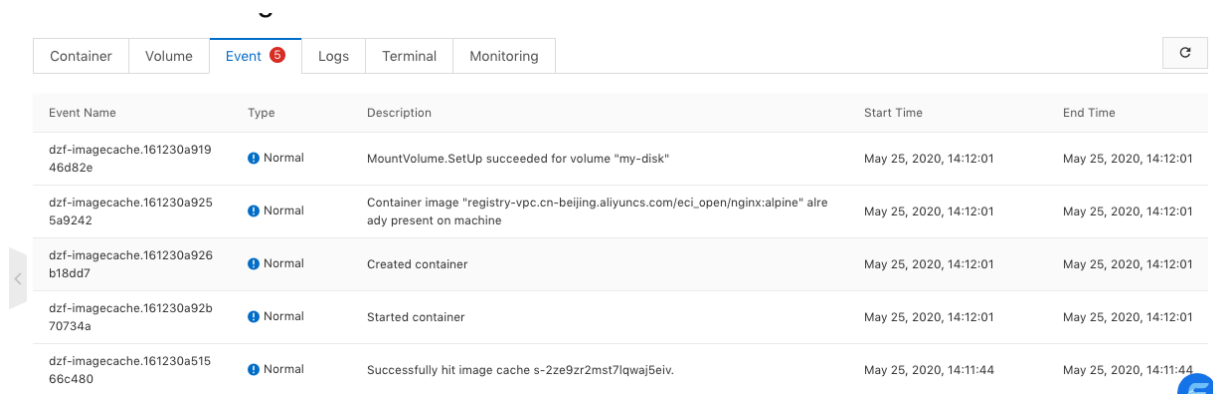
```
ImageSnapshotId=imc-2zeh184ra0em1rvku1f
```

Enable automatic image cache match

```
AutoMatchImageCache=true
```

Verify the result in the ECI console

View the events of the ECI in the ECI console.



The screenshot shows the 'Event' tab in the ECI console. The table below represents the data shown in the screenshot.

Event Name	Type	Description	Start Time	End Time
dzf-imagecache.161230a91946d82e	Normal	MountVolume.SetUp succeeded for volume "my-disk"	May 25, 2020, 14:12:01	May 25, 2020, 14:12:01
dzf-imagecache.161230a9255a9242	Normal	Container image "registry-vpc.cn-beijing.aliyuncs.com/eci_open/nginx:alpine" already present on machine	May 25, 2020, 14:12:01	May 25, 2020, 14:12:01
dzf-imagecache.161230a926b18dd7	Normal	Created container	May 25, 2020, 14:12:01	May 25, 2020, 14:12:01
dzf-imagecache.161230a92b70734a	Normal	Started container	May 25, 2020, 14:12:01	May 25, 2020, 14:12:01
dzf-imagecache.161230a51566c480	Normal	Successfully hit image cache s-2ze9zr2mst7lqwaj5eiv.	May 25, 2020, 14:11:44	May 25, 2020, 14:11:44

The ECI is created from existing images. No time is spent in pulling images from a remote image repository.

For more information about how to use the image cache feature in Alibaba Cloud Container Service for Kubernetes, see [Use an image cache CRD to accelerate pod creation](#).

2 Use an image cache CRD to accelerate pod creation

Downloading images is time-consuming when you create an Elastic Container Instance (ECI). ECI allows you to use an image cache to accelerate ECI creation. Before you create an ECI, you can cache the required images as an image snapshot on a disk. Then, ECI creates the ECI based on the image snapshot to download fewer or no images. This accelerates the creation of the ECI. Assume that you create an ECI based on Flink images from Docker Hub. The total size of the images is 386.26 MB. During testing, downloading images without using an image snapshot takes 50s, whereas downloading images by using an image snapshot only takes 5s. Using an image snapshot saves 45s when you create an ECI. The saved time depends on the number and size of images used to create an ECI and the network over which ECI connects to the image repository.

ECI provides an image cache custom resource definition (CRD) for you to use the image cache feature in Kubernetes. Note that the image cache CRD is a **cluster-level** resource and is similar to a persistent volume (PV). The image cache CRD can be shared by all namespaces in a cluster.

Before you begin

Verify that your Kubernetes cluster supports the image cache feature.

```
$ kubectl get crd/imagecaches.eci.alibabacloud.com
NAME                                CREATED AT
imagecaches.eci.alibabacloud.com  2019-09-27T01:15:07Z
```

If the preceding information is returned, your Kubernetes cluster supports the image cache feature.

```
kubectl get crd/imagecaches.eci.alibabacloud.com
Error from server (NotFound): customresourcedefinitions.apiextensions.k8s.io "
imagecaches.eci.alibabacloud.com" not found
```

If the preceding information is returned, your Kubernetes cluster does not support the image cache feature. Upgrade [Virtual Kubelet](#) to enable the image cache feature.

The user who is responsible for upgrading Virtual Kubelet varies according to the cluster type.

- Serverless Kubernetes cluster: The administrator upgrades Virtual Kubelet.
- Managed Kubernetes cluster: You need to upgrade Virtual Kubelet by yourself.

- Dedicated Kubernetes cluster: You need to upgrade Virtual Kubelet by yourself.
- User-created Kubernetes cluster: You need to upgrade Virtual Kubelet by yourself.

After you verify that the image cache feature is supported, you can create image caches in your Kubernetes cluster and use them to create pods.

To use the newly added parameters, you must upgrade the image cache CRD in your Kubernetes cluster to the latest version. We recommend that you run the following command after Virtual Kubelet is upgraded:

```
kubectl apply -f imagecache-crd-sample.yaml
```

imagecache-crd-sample.yaml:

```
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
  name: imagecaches.eci.alibabacloud.com
spec:
  group: eci.alibabacloud.com
  version: v1
  names:
    kind: ImageCache
    plural: imagecaches
    shortNames:
      - ic
  categories:
    - all
  scope: Cluster
  subresources:
    status: {}
  validation:
    openAPIV3Schema:
      required:
        - spec
      properties:
        spec:
          type: object
          required:
            - images
          properties:
            imagePullSecrets:
              type: array
              items:
                type: string
            images:
              minItems: 1
              type: array
              items:
                type: string
            imageCacheSize:
              type: integer
            retentionDays:
              type: integer
      additionalPrinterColumns:
        - name: Age
          type: date
          JSONPath: .metadata.creationTimestamp
```

```

- name: CacheId
  type: string
  JSONPath: .status.imageCacheId
- name: Phase
  type: string
  JSONPath: .status.phase
- name: Progress
  type: string
  JSONPath: .status.progress

```

Manage an image cache

After the image cache CRD is upgraded to the latest version, you can manage image caches

.

YAML file of an image cache:

```

apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
spec:
  images:
    - centos:latest
    - busybox:latest
  imagePullSecrets:
    - default:secret1
    - default:secret2
    - kube-system:secret3
  imageCacheSize:
    25
  retentionDays:
    7

```

Parameter description

Parameter	Type	Required	Description
spec.images	[]String	Yes	The images in the image cache.

Parameter	Type	Required	Description
spec.imagePullSecrets	[]String	No	The secrets of the image repositories. Specify each secret in the namespace:secretName format. If the image cache contains images from a private image repository, you must create a secret for the repository and add the secret to the value of this parameter. If all images in the image cache are public images, you do not need to set this parameter.
spec.imageCacheSize	Integer	No	The size of the disk used to store image snapshots. Valid values: 20 to 32768 . Default value: 20. Unit: GiB.
spec.retentionDays	Integer	No	The retention period of the image cache . Valid values: 1 to 65536. By default, the parameter is not specified, indicating that the image cache does not expire.

Create an image cache

Create an image cache.

```
# kubectl create -f imagecache-secrets-test.yaml
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample-liu
```



```
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
    25
  retentionDays:
    7
```

View the creation status of the image cache.

```
liumihustdeMacBook-Pro:vk-debug liumihust$ kubectl get imagecache imagecache-
sample-liu
NAME                AGE  CACHEID                PHASE  PROGRESS
imagecache-sample-liu 20h  imc-2zeditzeoemfhqor4rf8  Ready  100%
```

Query image caches

You can run the following command to query all image caches in the Kubernetes cluster:

```
kubectl get imagecache imagecache-sample-liu -o yaml
```

You can view the details of an image cache, including its properties, status, and events through the following YAML file:

```
$ kubectl get imagecache/imagecache-secrets-test -o yaml
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  creationTimestamp: "2019-10-24T13:40:56Z"
  generation: 1
  name: imagecache-sample-liu
  resourceVersion: "212484735"
  selfLink: /apis/eci.alibabacloud.com/v1/imagecaches/imagecache-sample-liu
  uid: e83a1c53-f663-11e9-a0d4-fa13fc175b32
spec:
  imageCacheSize: 25
  images:
  - busybox:latest
  - nginx
status:
  events:
  - count: 1
    eventTime: null
    firstTimestamp: "2019-10-24T13:41:40Z"
    involvedObject: {}
    lastTimestamp: "2019-10-24T13:41:40Z"
    message: Success pull image busybox:latest
    metadata:
      creationTimestamp: null
      name: busybox:latest
      reportingComponent: ""
      reportingInstance: ""
    source:
      component: kubelet
```

```

host: eci
type: Normal
- count: 1
eventTime: null
firstTimestamp: "2019-10-24T13:42:04Z"
involvedObject: {}
lastTimestamp: "2019-10-24T13:42:04Z"
message: Success pull image nginx
metadata:
  creationTimestamp: null
  name: nginx
reportingComponent: ""
reportingInstance: ""
source:
  component: kubelet
  host: eci
  type: Normal
- count: 1
eventTime: null
firstTimestamp: "2019-10-24T13:42:06Z"
involvedObject: {}
lastTimestamp: "2019-10-24T13:42:06Z"
message: Success pull image registry-vpc.cn-beijing.aliyuncs.com/acs/pause-amd64:3.0
:3.0
metadata:
  creationTimestamp: null
  name: registry-vpc.cn-beijing.aliyuncs.com/acs/pause-amd64:3.0
reportingComponent: ""
reportingInstance: ""
source:
  component: kubelet
  host: eci
  type: Normal
imageCacheId: imc-2zeditzeoemfhqor4rf8
expireDateTime: "2019-10-25T13:40:54Z"
lastUpdatedTime: "2019-10-24T13:44:17Z"
phase: Ready
progress: 100%
startTime: "2019-10-24T13:40:58Z"

```

Quickly create a pod based on an image cache

An image cache is a cluster-level resource. You can use it to create pods in different namespaces.

You can use an image cache to create a pod in one of the following ways:

- Specify an image cache. When you create a pod, you can specify an image cache.
- Enable automatic image cache match. When you create a pod, ECI automatically selects the optimal image cache based on matching rules.

Specify an image cache

You must specify an image cache in the `Ready` state to accelerate pod creation. Otherwise, the pod fails to be created.

Specify an image cache in the YAML file of a deployment

Add an annotation in the YAML file of a deployment to specify an image cache.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
      annotations:
        k8s.aliyun.com/eci-image-snapshot-id: imc-2ze5tm5gehgtiigaz5jh
    spec:
      nodeName: virtual-kubelet
      containers:
      - name: nginx
        image: nginx:1.7.9
        imagePullPolicy: IfNotPresent
```

Specify an image cache in the YAML file of a pod

Add an annotation in the YAML file of a pod to specify an image cache.

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.aliyun.com/eci-image-snapshot-id: imc-2ze5tm5gehgtiigaz5jh
  name: nginx-imagecache-id
spec:
  containers:
  - image: nginx:1.7.9
    imagePullPolicy: IfNotPresent
    name: nginx
  resources:
    limits:
      cpu: 300m
      memory: 200Mi
    requests:
      cpu: 200m
      memory: 100Mi
  nodeName: virtual-kubelet
```

Enable automatic image cache match

You can enable automatic image cache match through an annotation to accelerate pod creation. ECI selects the optimal image cache from existing image caches based on matching rules. If no suitable image cache is found, ECI creates a pod by downloading required images.

The following matching rules are listed based on their priorities in descending order:

- The matching degree of the images.
- The size of the images.
- The time when the image cache was created.

Enable automatic image cache match in the YAML file of a deployment

Add an annotation in the YAML file of a deployment to enable automatic image cache match.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
      annotations:
        k8s.aliyun.com/eci-image-cache: "true"
    spec:
      nodeName: virtual-kubelet
      containers:
      - name: nginx
        image: nginx:1.7.9
        imagePullPolicy: IfNotPresent
```

Enable automatic image cache match in the YAML file of a pod

Add an annotation in the YAML file of a pod to enable automatic image cache match.

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.aliyun.com/eci-image-cache: "true"
  name: nginx-auto-match
spec:
  containers:
  - image: nginx:1.7.9
    imagePullPolicy: IfNotPresent
    name: nginx
  resources:
    limits:
      cpu: 300m
      memory: 200Mi
    requests:
      cpu: 200m
      memory: 100Mi
```

```
nodeName: virtual-kubelet
```

Recommended configurations

1. We recommend that you specify an image in the image cache for a container to improve the matching degree.
2. We recommend that you set the `ImagePullPolicy` parameter to `IfNotPresent` for a container to avoid downloading duplicate images.

If you add the `k8s.aliyun.com/eci-image-snapshot-id` and `k8s.aliyun.com/eci-image-cache` annotations at the same time, the former takes precedence over the latter. The specified image cache is used to create a pod.



Notice:

In the YAML file of a deployment, you must add the annotation for a pod. If you add the annotation for the deployment, the annotation does not take effect.

Delete an image cache

If you delete an image cache, the pods created based on the image cache are not affected.

Run the following commands to delete an image cache:

```
$ kubectl get imagecache/imagecache-secrets-test
NAME          AGE  CACHEID          PHASE  PROGRESS
imagecache-secrets-test 14h  imc-2ze5tm5gehgtjru0kfm8  Ready  100%
$ kubectl delete imagecache/imagecache-secrets-test
imagecache.eci.alibabacloud.com "imagecache-secrets-test" deleted
$ kubectl get imagecache/ imagecache-secrets-test
Error from server (NotFound): imagecaches.eci.alibabacloud.com "imagecache-secrets-test" not found
```

API reference related to the image cache feature in Kubernetes

Group	Version	Kind
eci.alibabacloud.com	v1	ImageCache

ImageCache v1 eci.alibabacloud.com

ImageCacheList

Parameter	Type	Description
apiVersion	String	The version of the API. Set the value to <code>eci.alibabacloud.com/v1</code> .

Parameter	Type	Description
kind	String	The type of the resource. Set the value to ImageCacheList.
metadata	ListMeta	The metadata that the resource must have.
Items	[]ImageCache	The image caches.

ImageCache

Parameter	Type	Description
apiVersion	String	The version of the API. Set the value to eci.alibabacloud.com/v1.
kind	String	The type of the resource. Set the value to ImageCache.
metadata	ObjectMeta	
spec	ImageCacheSpec	The details of the image cache.
status	ImageCacheStatus	The status of the image cache.

ImageCacheSpec

Parameter	Type	Required	Description
images	[]String	Yes	The images in the image cache.
imagePullSecrets	[]String	No	The secrets of the private image repositories. Specify each secret in the namespace:secretName format.
imageCacheSize	Integer	No	The size of the image cache. Valid values: 20 to 32768. Default value: 20. Unit: GiB.

Parameter	Type	Required	Description
retentionDays	Integer	No	The retention period of the image cache . Valid values: 1 to 65536. By default, the parameter is not specified, indicating that the image cache does not expire.

ImageCacheStatus

Parameter	Type	Description
phase	String	The status of the image cache.
progress	String	The creation progress of the image cache.
imageCacheId	String	The ID of the image cache . You can use the image cache ID returned by the CreateImageCache operation.
startTime	Time	The time when the image cache was created.
lastUpdatedTime	Time	The most recent time when the image cache was updated.
events	[]Event	The events generated during the creation of the image cache.
expireDateTime	Time	The time when the image cache expires. The image cache expires when the specified retention period ends or 24 hours after the image cache fails to be created.

CreateImageCache

Creates an image cache.

HTTP request URL

POST /apis/eci.alibabacloud.com/v1/imagecaches

Sample request

```
curl -X POST -H 'Content-Type: application/yaml' --data '
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-secrets-test
spec:
  images:
  - registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0
  imagePullSecrets:
  - default:acr-test
' http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches
```

Sample response

```
{
  "apiVersion": "eci.alibabacloud.com/v1",
  "kind": "ImageCache",
  "metadata": {
    "creationTimestamp": "2019-10-10T03:50:47Z",
    "generation": 1,
    "name": "imagecache-secrets-test",
    "resourceVersion": "647052328",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test11",
    "uid": "2506ba37-eb11-11e9-9576-f2ef49bf1a08"
  },
  "spec": {
    "imagePullSecrets": [
      "default:acr-test"
    ],
    "images": [
      "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
    ]
  }
}
```

ListImageCache

Queries the image caches in a Kubernetes cluster.

HTTP request URL

GET /apis/eci.alibabacloud.com/v1/imagecaches

Sample request

```
curl -X GET 'http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches'
```

Sample response

```
{
  "apiVersion": "eci.alibabacloud.com/v1",
  "items": [
```



```

{
  "apiVersion": "eci.alibabacloud.com/v1",
  "kind": "ImageCache",
  "metadata": {
    "annotations": {
      "kubectrl.kubernetes.io/last-applied-configuration": "{\"apiVersion\":\"eci.alibabacloud.com/v1\", \"kind\":\"ImageCache\", \"metadata\":{\"annotations\":{}}, \"name\":\"imagecache-secrets-test\", \"spec\":{\"imagePullSecrets\":[\"default:acr-test\"], \"images\":[\"registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0\"]}\""}
    },
    "creationTimestamp": "2019-10-10T01:43:48Z",
    "generation": 1,
    "name": "imagecache-secrets-test",
    "resourceVersion": "646375979",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test",
    "uid": "67460461-eaff-11e9-9774-b2c06862f69a"
  },
  "spec": {
    "imagePullSecrets": [
      "default:acr-test"
    ],
    "images": [
      "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
    ]
  },
  "status": {
    "imageCacheId": "imc-2ze5xnx4yipl807zo0o",
    "lastUpdatedTime": "2019-10-10T01:47:14Z",
    "phase": "Ready",
    "progress": "100%",
    "startTime": "2019-10-10T01:43:49Z"
  }
},
{
  "kind": "ImageCacheList",
  "metadata": {
    "continue": "",
    "resourceVersion": "647062964",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches"
  }
}

```

GetImageCache

Queries the details of an image cache.

HTTP request URL

GET /apis/eci.alibabacloud.com/v1/imagecaches/{name}

Sample request

```
curl -X GET 'http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test'
```

Sample response

```

{
  "apiVersion": "eci.alibabacloud.com/v1",
  "kind": "ImageCache",

```

```

"metadata": {
  "annotations": {
    "kubectrl.kubernetes.io/last-applied-configuration": "{\"apiVersion\":\"eci.alibabacloud.com/v1\", \"kind\":\"ImageCache\", \"metadata\":{\"annotations\":{}, \"name\":\"imagecache-secrets-test\"}, \"spec\":{\"imagePullSecrets\":[\"default:acr-test\"], \"images\":[\"registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0\"]}\""}
  },
  "creationTimestamp": "2019-10-10T01:43:48Z",
  "generation": 1,
  "name": "imagecache-secrets-test",
  "resourceVersion": "646375979",
  "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test",
  "uid": "67460461-eaff-11e9-9774-b2c06862f69a"
},
"spec": {
  "imagePullSecrets": [
    "default:acr-test"
  ],
  "images": [
    "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
  ]
},
"status": {
  "imageCacheId": "imc-2ze5xnx4yiplly807zo0o",
  "lastUpdateTime": "2019-10-10T01:47:14Z",
  "phase": "Ready",
  "progress": "100%",
  "startTime": "2019-10-10T01:43:49Z"
}
}

```

DeleteImageCache

Deletes an image cache.

HTTP request URL

```
DELETE /apis/eci.alibabacloud.com/v1/imagecaches/{name}
```

Sample request

```
curl -X DELETE -H 'Content-Type: application/yaml' 'http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test'
```

Sample response

```

"kind": "Status",
"apiVersion": "v1",
"metadata": {},
"status": "Success",
"details": {
  "name": "imagecache-secrets-test",
  "group": "eci.alibabacloud.com",
  "kind": "imagecaches",
  "uid": "67460461-eaff-11e9-9774-b2c06862f69a"
}
}

```