ALIBABA CLOUD

Alibaba Cloud

Container Service for Kubernetes

Product Introduction

Document Version: 20201111

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 because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should 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 this document based on the "status quo", "being defective", and "existing functions" of its products and services. 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 take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, 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 directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	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.
Warning	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.
Notice	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.
Note	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 `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.What is Container Service for Kubernetes?

Container Service for Kubernetes (ACK) is one of the first services to participate in the Certified Kubernetes Conformance Program. ACK provides high-performance management services of containerized applications. You can manage enterprise-level containerized applications throughout the application lifecycle. This service allows you to run containerized applications in the cloud in an efficient manner.

Cluster types

ACK provides the following three types of Kubernetes clusters: dedicated Kubernetes clusters, managed Kubernetes clusters, and serverless Kubernetes clusters.

Item	Dedicated Kubernetes cluster	Managed Kubernetes cluster	Serverless Kubernetes
Features	You must create master nodes and worker nodes.	You only need to create worker nodes. ACK creates and manages master nodes.	You do not need to create master nodes or worker nodes.
	A dedicated Kubernetes cluster allows you to manage the cluster infrastructure in a more fine-grained manner. You must design, maintain, and upgrade the Kubernetes cluster on your own.	A managed Kubernetes cluster is easy-to-use, cost-efficient, and highly available. You do not need to manage master nodes.	A serverless Kubernetes cluster allows you to start applications directly. You do not need to manage nodes.
Billing method	You are not billed for cluster management features. However, you must pay for master nodes, worker nodes, and other infrastructure.	• ACK Standard clusters: You are not billed for cluster management features. However, you must pay for worker nodes and other infrastructure. • ACK Pro clusters: You are billed based on the number of clusters or the subscription method.	You are billed for the resources consumed by pods and the resource usage duration. The duration is measured in seconds.
Scenarios	Applies to all scenarios.	Applies to all scenarios.	Applies to batch tasks, urgent application scale-out, and continuous integration (CI) or continuous delivery (CD).

Item	Dedicated Kubernetes cluster	Managed Kubernetes cluster	Serverless Kubernetes
User profile			
Cluster creation procedure			

Features

- Cluster management
  - Create a cluster: You can create multiple types of clusters as needed, choose multiple types of worker nodes, and flexibly custom the configurations.
  - Upgrade a cluster: You can upgrade Kubernetes with a few clicks and manage the upgrade of system components in a unified manner.
  - Manage node pools: You can manage the lifecycle of node pools. You can configure node pools of different specifications in a cluster, such as VSwitches, runtimes, operating systems, and security groups.
  - Elastic scaling: You can scale in and scale out resources in the console with a few clicks based on your needs. You can also use service-level affinity rules and scale up resources.
  - Manage multiple clusters: You can manage applications in data centers and clusters in multiple clouds and regions in a unified manner.
  - Manage permissions: You can grant permissions to users in the RAM console or by using role-based access control (RBAC) policies.
- Application management
  - Application creation: You can create multiple types of applications based on images and templates. You can configure environment variables, application health checks, data disks, and logging.

  - Application lifecycle management: You can view, update, and delete applications, roll back application versions, view application events, perform rolling updates of applications, use new application versions to replace old application versions, and use triggers to redeploy applications.
  - Application pod scheduling: You can schedule application pods based on the following three policies: pod affinity, node affinity, and pod anti-affinity.
  - Application pod scaling: You can scale application pods manually or by using the Horizontal Pod Autoscaler (HPA).
  - Application release: Phased release and blue-green release are supported.
  - App Catalog: You can use App Catalog to simplify the integration of Alibaba Cloud services.
  - Application center: After an application is deployed, the application center displays the topology of the application on one page. You can also manage and roll back the application version in scenarios such as continuous deployment.

- Storage and network
  - Storage plug-ins: FlexVolume and CSI are supported.
  - Volumes and persistent volume claims (PVCs):
    - You can create Block Storage volumes, Apsara File Storage NAS volumes, Object Storage Service (OSS) volumes, and Cloud Paralleled File System (CPFS) volumes.
    - You can bind a volume to a PVC.
    - You can dynamically create and migrate volumes.
    - You can view and update volumes and PVCs by running scripts.
  - Network:
    - You can create clusters in VPCs and use the Flannel and Terway network plug-ins.
    - You can specify CIDR blocks of services and pods.
    - You can use the NetworkPolicy feature.
    - You can use ingresses to route requests from outside a cluster to services within the cluster.

- O&M and security
  - Monitoring: You can monitor clusters, nodes, applications, and pods. You can use the Prometheus plug-in.
  - Logging: You can view cluster logs, pod logs, and application logs.
  - The Runtime Security page allows you to manage security policies of the container runtime, configure routine inspection of application security, and configure security monitoring and alerting on the runtime. This improves the overall security capabilities of containers.
  - Sandboxed-Container allows you to run an application in a sandboxed and lightweight virtual machine. This virtual machine has a dedicated kernel, isolates applications from each other, and provides enhanced security. Sandboxed-Container is suitable in scenarios such as untrusted application isolation, fault isolation, performance isolation, and load isolation among multiple users.
  - TEE-based confidential computing is a cloud-native and all-in-one solution based on Intel Software Guard Extensions (SGX). This solution ensures security, integrity, and confidentiality of data in use. It also lowers the costs of developing, delivering, and managing trusted applications and confidential computing applications. Confidential computing allows you to isolate sensitive data and code by using a trusted execution environment.

Product architecture

- Alibaba Cloud Container Registry (ACR)provides managed security services and lifecycle management of cloud-native assets. ACR distributes images to clusters in different scenarios and is seamlessly integrated with ACK to provide an all-in-one solution for cloud-native application management.
- Alibaba Cloud Service Mesh (ASM) is a managed service mesh platform that allows you to manage the traffic of an application that uses the microservices architecture in a unified manner. ASM is compatible with the open source Istio service mesh platform and allows you to manage the traffic of multiple Kubernetes clusters. ASM provides a unified way to manage the communications among containerized applications and applications on virtual machines.
- Alibaba Cloud Serverless Kubernetes (ASK) provides serverless Kubernetes clusters based on elastic computing. You can create containerized applications without managing or maintaining clusters.
- Alibaba Cloud Genomics Service (AGS) is a genome sequencing and secondary analysis service based on big data. It serves biotechnology industry users. AGS is an efficient, elastic, and reliable service that requires low costs.
- Alibaba Cloud Container Service for Kubernetes@Edge (ACK@Edge) is a Kubernetes cluster based on the standard Kubernetes runtime environment. It integrates the cloud, edge, and terminals to deliver, maintain, and manage applications. ACK@Edge also enhances node autonomy in edge clusters.

Use ACK

Click the button below to use ACK.

Use ACK

2.Benefits

This topic describes the advantages of Container Service for Kubernetes (ACK) clusters and the disadvantages of user-created Kubernetes clusters.

Advantages of ACK

Advantage	Description
Cluster management	• Provides the following three types of Kubernetes clusters: dedicated clusters, managed clusters, and serverless clusters. • Supports up to 5,000 Elastic Compute Service (ECS) instances in one cluster. ? **Note** If you want to create a cluster of more than 2,000 ECS instances, submit a ticket in Quota Center. • Supports multi-cluster management, cross-zone clusters, and the cluster federation feature. • Provides cross-zone high availability and disaster recovery.
Elastic resource scaling	• Automatically scales the number of containers based on container resource usage. • Scales up to thousands of nodes in minutes. • If your application is deployed on elastic container instances (ECIs) in a serverless Kubernetes cluster, up to 500 pods can be started in 30 seconds. • Supports vertical scaling with a few clicks. • Supports affinity policies and scale-out for your services. • Provides standard Horizontal Pod Autoscaler (HPA), Vertical Pod Autoscaler (VPA), and Cluster Autoscaler capabilities provided by the Kubernetes community. • Provides the scheduled scaling capability similar to cron HPA and provides serverless scalability by using virtual-kubelet-autoscaler. • Uses ack-kubernetes-elastic-workload to provide fine-grained scheduling for online business. • Provides alibaba-metrics-adapter to meet different scaling needs. Application scaling is optimized by using ingress gateways, Sentinel-based microservice rate limiting, and other methods.

Advantage	Description
All-in-one container management	• Application management: ◦ Supports phased release, blue-green release, application monitoring, and application autoscaling. ◦ Provides a built-in application store that allows you to deploy applications with a few clicks by using Helm. ◦ Supports Service Catalog to simplify cloud service integration. • Repository (Alibaba Cloud Container Registry): ◦ Provides high availability and high concurrency of image pull requests. ◦ Supports accelerated image retrieval. ◦ Supports large-scale P2P image distribution. By using the optimized image distribution process, ACK can automatically distribute images to a maximum of 10,000 nodes. This improves the distribution efficiency by four times. **Note** User-created image repositories may fail to respond when millions of clients attempt to pull images at the same time. Alibaba Cloud Container Registry (ACR) improves the reliability of image repositories and reduces the O&M and upgrade costs. • Logs: ◦ Allows you to collect logs and deliver the collected logs to Log Service. ◦ Supports the integration with third-party open source logging solutions. • Monitoring: ◦ Supports both container-level and VM-level monitoring. ◦ Supports the integration with open source monitoring solutions from third-party providers.
Multiple types of nodes	• The following node types are supported: ◦ x86-based computing resources: ECS instances based on the x86 architecture ◦ Heterogeneous computing resources: GPU ECS instances, NPU ECS instances, and FPGA ECS instances ◦ Bare metal computing resources: ECS Bare Metal (EBM) instances ◦ Serverless computing resources: ACK virtual nodes ◦ Edge nodes: Alibaba Cloud Container Service for Kubernetes@Edge (ACK@Edge) supports centralized management of cloud and edge nodes and unified application release. It increases the release efficiency by three times. • The following billing methods are supported: ◦ Preemptible instances ◦ Subscription ◦ Pay-as-you-go

Advantage	Description
IaaS capabilities	● Network: ○ Provides a high-performance plug-in to assign elastic network interfaces (ENIs) to pods. The performance of this plug-in is 20% higher than that of regular network solutions. ○ Supports container access policies and throttling. ● Storage: ○ Supports Alibaba Cloud disks, Apsara File Storage NAS volumes, and Object Storage Service (OSS) buckets, and provides standard CSI drivers. ○ Allows you to dynamically create and migrate volumes. ● Load balancing: Allows you to create public-facing and internal Server Load Balancer (SLB) instances. **Note** If you use user-created ingresses to control access to a user-created Kubernetes cluster, frequent service releases may negatively affect the performance of the ingress and increase the error rate. ACK allows you to create SLB instances, which provide high-availability load balancing and can automatically modify network configurations to suit your business needs. This solution is widely used by a large number of users over a long period of time. It provides much higher stability and reliability than user-created ingresses.

Advantage	Description
Enterprise-level security and stability	ACK is integrated with a multi-layer security solution that secures cloud-native ACK clusters. This solution protects the underlying infrastructure, intermediate software supply chains, and top-layer runtime environments. • The end-to-end security solution ensures the security of the following resources: ◦ Infrastructure: ACK supports comprehensive network isolation and end-to-end data encryption. ACK associates Alibaba Cloud accounts and RAM users with the Kubernetes Role-Based Access Control (RBAC) system. It also provides fine-grained permission management and comprehensive audit capabilities. ◦ Software supply chains: ACK has a complete DevSecOps process, including image scan, cloud-native delivery chain, image signing, and image synchronization. ◦ Runtime: ACK provides runtime security capabilities, including security policy management of applications, configuration routine inspection, runtime monitoring and alerting, and key encryption and management. • Default security capabilities: ◦ ACK provides optimized OS images of Kubernetes containers and supports Kubernetes clusters and Docker versions that have high stability and enhanced security. ◦ Enhances the security compliance of cluster configurations, system components, and system images based on the CIS Benchmark and container security best practices. ◦ Grants minimum permissions on managing default cloud resources on nodes. • Sandboxed-Container allows you to run an application in a sandboxed and lightweight virtual machine. This virtual machine has a dedicated kernel, isolates applications from each other, and provides enhanced security. Sandboxed-Container is suitable in scenarios such as untrusted application isolation, fault isolation, performance isolation, and load isolation among multiple users. • TEE-based confidential computing is a cloud-native and all-in-one solution based on Intel Software Guard Extensions (SGX). This solution ensures security, integrity, and confidentiality of data in use. It also lowers the costs of developing, delivering, and managing trusted applications and confidential computing applications. Confidential computing allows you to isolate sensitive data and code by using a trusted execution environment.
24/7 technical support	Provides you with 24/7 technical support in the ticket system.

Disadvantages of user-created Kubernetes clusters

- The operations to create a Kubernetes cluster are complicated.
  You must manually configure the components, configuration files, certificates, keys, plug-ins, and tools. This process takes professional engineers up to several weeks.
- Significant costs are required to integrate user-created Kubernetes cluster with public cloud services.

You must use your own resources to integrate your system with other Alibaba Cloud services, such as Log Service, monitoring services, and storage management services.

- A container is a systematic project that involves various technologies, such as network, storage, operating systems, and orchestration. Professional engineers are required to use containers.
- The container technology is under constant development. To keep up with the frequent version iterations, you must continuously upgrade and test your containerized applications.

3.Scenarios

DevOps & continuous delivery

Optimized continuous delivery pipeline

Container Service integrates with Jenkins to automate the DevOps pipeline that ranges from code submission to application deployments. The pipeline ensures that code is submitted for deployment only after passing automated testing, and provides a better alternative to traditional delivery models that involve complex deployments and slow iterations.

Benefits

- DevOps pipeline automation
  Automates the DevOps pipeline, from code updates to code builds, image builds, and application deployments.
- Environment consistency
  Allows you to deliver code and runtime environments based on the same architecture.
- Continuous feedback
  Provides immediate feedback on each integration or delivery.

Related services

ECS

Microservice architecture

Agile development and deployment to speed up the evolution of business models

Your workload in the production environment is divided into multiple microservice applications, which are managed by Alibaba Cloud image repositories. Alibaba Cloud can schedule, orchestrate, deploy, and implement the canary releases of microservice applications and you only need to focus on feature updates.

Benefits

- Load balancing and service discovery
  Forwards layer 4 and layer 7 requests and binds the requests to backend containers.
- Multiple scheduling and disaster recovery policies
  Supports different levels of affinity scheduling policies, and cross-zone high availability and disaster recovery.
- Microservice monitoring and auto scaling
  Supports microservice and container monitoring, and microservice auto scaling.

Related services

ECS, ApsaraDB for RDS, and OSS

Hybrid cloud architecture

Unified O&M of cloud resources

You can centrally manage cloud and on-premises resources in the Container Service console. Containers hide the differences between infrastructures. This enables you to use the same images and orchestration templates to deploy applications in the cloud and on premises.

Benefits

- Application scaling in the cloud
  During peak hours, Container Service can scale up applications in the cloud and forward traffic to the scaled-up resources.
- Disaster recovery in the cloud
  Business systems can be deployed on premises for service provisioning and in the cloud for disaster recovery.
- On-premises development and testing
  Applications that are developed and tested on premises can be seamlessly released to the cloud.

Related services

ECS, VPC, and Express Connect

Automatic scaling architecture

Traffic-based scalability

Container Service enables workloads to auto-scale their resources based on traffic. This prevents traffic spikes from bringing down your system and eliminates idle resources during off-peak hours.

Benefits

- Quick response
  A scale-out event can be triggered within seconds when traffic reaches the scale-out threshold.
- Auto scaling
  The scaling process is fully automated without human interference.
- Cost efficiency
  Containers are automatically scaled in when traffic decreases to avoid resource waste.

Related services

ECS and CloudMonitor

4.Terms

4.1. Terms

Basic terms

Cluster

A collection of cloud resources that are required to run containers. Several cloud resources, such as Elastic Compute Service (ECS) instances, Server Load Balancer (SLB) instances, and Virtual Private Clouds (VPCs), are associated together to form a cluster.

Managed Kubernetes cluster

A cluster for which you only need to create worker nodes. Container Service for Kubernetes creates and manages master nodes. This type of Kubernetes cluster is easy to use with low cost and high availability. You do not need to manage the master nodes of the Kubernetes cluster.

Dedicated Kubernetes cluster

A cluster for which you must create three master nodes and several worker nodes to achieve high availability. This type of Kubernetes cluster allows you to manage the cluster infrastructure in a more fine-grained manner. It requires you to plan, maintain, and upgrade the Kubernetes cluster on your own.

Serverless Kubernetes cluster

A cluster for which you do not need to create and manage any master nodes or worker nodes. You can use the Container Service console or command-line interface to configure resources for containers, specify container images for applications, provide methods for external access, and start applications.

Node

A virtual machine (VM) or a physical server that has Docker Engine installed and is used to deploy and manage containers. The Agent program of Container Service is installed on a node and registered with a cluster. The number of nodes in a cluster can be scaled based on your requirements.

Container

A runtime instance created from a Docker image. A single node can run multiple containers.

Image

A standard packaging format of a containerized application in Docker. An image from Docker Hub, Alibaba Cloud Container Registry, or your private registry can be specified to deploy its packaged containerized application. An image ID is a unique identifier composed of the image repository URI and image tag. The default tag is latest.

Kubernetes terms

Master node

The manager of a Kubernetes cluster. It runs components such as kube-apiserver, kube-scheduler, kube-controller-manager, etcd, and container network. Generally, three master nodes are deployed to ensure high availability.

Worker node

A node in a Kubernetes cluster that carries workloads. It can be either a VM or a physical server. A worker node schedules pods and communicates with the master node. Components running on a worker node include the Docker runtime environment, kubelet, kube-proxy, and other optional add-on components.

Namespace

A method used in Kubernetes to divide cluster resources between multiple users. By default, Kubernetes starts with three initial namespaces: default, kube-system, and kube-public. Administrators can also create new namespaces as required.

Pod

The smallest deployable computing unit that can be created and managed in Kubernetes. A pod encapsulates one or more containers, storage resources, a unique network IP address, and options that specify how the containers run.

Replication controller (RC)

A feature that monitors running pods to ensure that a specified number of pod replicas are running at any given time. One or more pod replicas can be specified. If the number of pod replicas is smaller than the specified value, an RC starts new pod replicas. If the number of pod replicas exceeds the specified value, the RC stops the redundant pod replicas.

Replica set (RS)

The upgraded version of RC. Compared with RCs, RSs support more selector types. RS objects are not used independently, but are used as deployment parameters under ideal conditions.

Deployment

An update operation performed on a Kubernetes cluster. Deployment is more widely applied than RS. You can use deployments to create, update, or perform rolling updates for services. A new RS is created when you perform a rolling update for a service. A compound operation is performed to increase the number of replicas in the new RS to the desired value while decreasing the number of replicas in the original RS to zero. This kind of compound operation is better performed by a deployment than through RS. We do not recommend that you manage or use the RS created by a deployment.

Service

The basic operation unit of Kubernetes. It is an abstraction of real application services. Each service has multiple containers that support it. The kube-proxy port and service selector determine the back-end container to which a service request is forwarded, and a single access interface is provided externally. The back end can be scaled or maintained without the awareness of users.

Labels

A collection of key-value pairs attached to resource objects. Labels are intended to specify identifying attributes of objects that are meaningful and relevant to users, but do not directly imply semantics to the core system. 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, and each key must be unique for a specified object.

Volume

Kubernetes volumes are similar to Docker volumes. However, they are different in one key aspect. Docker volumes are used to persist data in Docker containers, while Kubernetes volumes share the same lifetime as the pods that enclose them. The volumes declared in each pod are shared by all containers in the pod. The actual back-end storage technology used is irrelevant when you use persistent volume claim (PVC) logical storage. The specific configurations for persistent volumes (PVs) are completed by storage administrators.

PV and PVC

PVs and PVCs allow Kubernetes clusters to provide a logical abstraction over the storage resources, so that the actual configurations of back-end storage can be ignored in the pod configuration logic, and instead completed by the PV configurators. The relationship between PVs and PVCs is similar to that between nodes and pods. PVs and nodes are resource providers which can vary by cluster infrastructure, and are configured by the administrators of a Kubernetes cluster. PVCs and pods are resource consumers that can vary based on service requirements, and are configured by either the users or service administrators of a Kubernetes cluster.

Ingress

A collection of rules that allow inbound access to cluster services. An Ingress can be configured to provide services with externally-reachable URLs, load-balance traffic, terminate SSL, and offer name-based virtual hosting. You can request the Ingress by posting Ingress resources to API servers. An Ingress controller is responsible for fulfilling an Ingress, usually with a load balancer. It can also be used to configure your edge router or additional front ends to help handle the traffic.

References

- Docker glossary
- Kubernetes concepts

4.2. Comparison between Container Service for Kubernetes terms and native Kubernetes terms

This topic compares Container Service for Kubernetes terms with native Kubernetes terms.

Container Service for Kubernetes	Native Kubernetes
cluster	cluster
node	node
container	container
image	image
namespace	namespace
deployment	deployment
StatefulSet	StatefulSet
job	job
cron job	CronJob
service	service
Ingress	Ingress
label	label

Container Service for Kubernetes	Native Kubernetes
ConfigMap	ConfigMap
secret	secret
persistent volume (PV)	PersistentVolume
persistent volume claim (PVC)	PersistentVolumeClaim
HPA	HPA
cluster IP	cluster IP
node port	NodePort
Server Load Balancer	LoadBalancer
node affinity	NodeAffinity
pod affinity	PodAffinity
pod anti affinity	PodAntiAffinity
selector	LabelSelector
annotation	annotation
trigger	webhook
endpoint	endpoint
resource quota	resource quota
resource limit	limit range
template	template

5.Limits

This topic describes the limits that apply when you use Container Service for Kubernetes clusters.

Overview

- After a Kubernetes cluster is created, the following limits apply:
  - You cannot change the VPC network.
  - You cannot change the cluster type. For example, you cannot convert a dedicated cluster to a managed cluster.
  - You cannot change the network plug-in.
  - You cannot change the volume plug-in.
- You cannot migrate applications across namespaces.
- Currently, ECS instances support two billing methods: pay-as-you-go and subscription. You can change the billing method of ECS instances from pay-as-you-go to subscription in the ECS console. Other resources such as SLB instances only support the pay-as-you-go billing method.
- Operations such as cluster creation, expansion, and auto scaling are subject to resource quota limits and inventory availability. The number of nodes you specified may not be created.
- If you choose to create subscription instances when you create a cluster, instance creation may fail due to resource quota limits and inventory availability. After a subscription instance is created, you cannot release it before the subscription expires. To avoid these issues, we recommend that you choose to create pay-as-you-go instances. You can change the billing method from pay-as-you-go to subscription in the ECS console if necessary.

Quota limits on ACK

Item	Limit	Quota increase
Real-name verification	Real-name verification	No exception
Create pay-as-you-go instances	None	None
Total number of Kubernetes clusters and managed Kubernetes clusters under an account	50 to 2,000	Navigate to the Quota Center page to submit a ticket.
Total number of master nodes and worker nodes in a cluster	100 to 5,000	Navigate to the Quota Center page to submit a ticket.
Maximum number of pods on a worker node	Configurable option. Maximum is 256.	No exception
Total number of serverless Kubernetes clusters under an account	2	Navigate to the Quota Center page to submit a ticket.

Quota limits on underlying cloud resources

Resource	Item	Limit	Quota increase
Computing	Total vCPU quota of pay-as-you-go instances	500 vCPUs	Submit a ticket.
	vCPU quota of a pay-as-you-go instance	Less than 16 vCPUs	Submit a ticket.
	Total vCPU quota of preemptible instances	800 vCPUs	Submit a ticket.
	Convert from pay-as-you-go to subscription	Not supported by the following instance families: t1, s1, s2, s3, c1, c2, m1, m2, n1, n2, and e3.	Submit a ticket.
	Maximum number of ECS instances managed by a scaling group	1,000	Submit a ticket.
Networking	Maximum number of custom route entries in a route table	48	Submit a ticket.
	Maximum number of VSwitches in a VPC network	24	Submit a ticket.
	Maximum number of VPC networks	10	Submit a ticket.
	Maximum number of internal IP addresses in a VPC network	65,535	No exception
	Maximum number of IP addresses attached to a basic security group	2,000	No exception
	Maximum number of Elastic Network Interfaces (ENIs)	50,000	No exception
	Maximum number of Elastic IP addresses (EIPs)	20	Submit a ticket.
	Maximum number of SLB instances under an account	60	Submit a ticket.

Resource	Item	Limit	Quota increase
Server Load Balancing (SLB)	Maximum number of backend servers attached to an SLB instance	200	No exception
	Maximum number of listeners of an SLB instance	50	Submit a ticket.
	Maximum number of times that an ECS instance can be repeatedly added as a backend server to SLB instances	50	No exception
Block storage	Maximum number of pay-as-you-go disks across all regions under an account	Number of ECS instances across all regions × 5 You can create at least 10 pay-as-you-go disks under an account.	Submit a ticket.
	Maximum capacity of pay-as-you-go disks that are used as data disks under an account	This limit is subject to ECS resource usage, region, and disk type. You can go to the Privileges & Quotas page in the ECS console to view detailed information. For more information, see View quotas (old version).	Submit a ticket.

Supported regions

The following table lists the regions where Container Service for Kubernetes is available.

- Asia Pacific

Region name	City	Region ID
China (Beijing)	Beijing	cn-beijing
China (Zhangjiakou)	Zhangjiakou	cn-zhangjiakou
China (Hohhot)	Hohhot	cn-huhehaote
China (Hangzhou)	Hangzhou	cn-hangzhou
China (Shanghai)	Shanghai	cn-shanghai
China (Shenzhen)	Shenzhen	cn-shenzhen

Region name	City	Region ID
China (Chengdu)	Chengdu	cn-chengdu
China (Hong Kong)	Hong Kong	cn-hongkong
Japan (Tokyo)	Tokyo	ap-northeast-1
Singapore	Singapore	ap-southeast-1
Australia (Sydney)	Sydney	ap-southeast-2
Malaysia (Kuala Lumpur)	Kuala Lumpur	ap-southeast-3
Indonesia (Jakarta)	Jakarta	ap-southeast-5

- Americas and Europe

Region name	City	Region ID
US (Silicon Valley)	Silicon Valley	us-west-1
US (Virginia)	Virginia	us-east-1
UK (London)	London	eu-west-1
Germany (Frankfurt)	Frankfurt	eu-central-1

- Middle East and India

Region name	City	Region ID
UAE (Dubai)	Dubai	me-east-1
India (Mumbai)	Mumbai	ap-south-1

6.Kubernetes version support policy

This topic describes the Kubernetes version support policy used by Container Service for Kubernetes.

Supported versions

Container Service for Kubernetes supports the latest releases of two major Kubernetes versions. For example, v1.14.8 and v1.16.6. You can create Kubernetes clusters through the console and get technical support with ease. To continue to use an out-of-support Kubernetes version, for example, v1.12.6, you can submit a ticket to apply for a grace period, which expires when the next major version is released.

> **Note** In the following text, versions consisting of two numbers are referred to as major versions, such as v1.14. Versions consisting of three numbers are referred to as minor versions, such as v1.14.8.

Version release cycle

- Container Service for Kubernetes updates major Kubernetes versions roughly every six months.
- After a major version is released, Container Service for Kubernetes will release minor versions that include feature updates or fixes for security vulnerabilities.

Version upgrade and deprecation

- Container Service for Kubernetes supports upgrading Kubernetes versions through the console. After a new major version is launched, the oldest version will be immediately removed. After the upgrade to the new major version is released, the oldest version will be deprecated after a six-month grace period.

  > **Note**
  >
  > - After a Kubernetes version is **removed**, you cannot select this version when you create Kubernetes clusters through the console.
  > - After a Kubernetes version is **deprecated**, you cannot upgrade this Kubernetes version and no technical support is provided for this version. You have a six-month grace period to upgrade the old Kubernetes version.
  >
  > Assume that Container Service for Kubernetes currently supports Kubernetes v1.11 and v1.12. After v1.14 is launched, v1.11 will be removed. After the upgrade to v1.14 is released, v1.11 will be deprecated after a certain period of time. Note that after a new major version is launched, the upgrade to this version will be available in about one month. For more information, see ACK Kubernetes version supporting roadmap.

- For major versions, you can upgrade to the next major version only. This restriction does not apply to minor version upgrades.
- After a new minor version is launched, technical support is provided for this version only. We recommend that you upgrade old minor versions to the latest version as soon as possible.