

Alibaba Cloud

Auto Scaling Product Introduction

Document Version: 20220621

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 <code>cd /d C:/window</code> command to enter the Windows system folder.
<i>Italic</i>	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	This format is used for a required value, where only one item can be selected.	<code>switch {active stand}</code>

Table of Contents

1.Benefits	05
2.Scenarios	07
3.How Auto Scaling works	09
4.Limits	14
5.Terms	16

1. Benefits

Auto Scaling provides the following benefits: automation, cost-effectiveness, high availability, flexibility, intelligence, and easy audit.

Automation

Auto Scaling performs automatic scaling based on your configurations to prevent the errors caused by manual intervention.

- Scale-out activities:
 - Auto Scaling automatically creates the specified number of Elastic Compute Service (ECS) instances or elastic container instances and add them to your scaling group to provide sufficient computing power to meet your business requirements.
 - If your scaling group is associated with a Server Load Balancer (SLB) instance, Auto Scaling automatically attaches the ECS instances or elastic container instances in your scaling group to the SLB instance. This way, the SLB instance can forward requests to the attached ECS instances or elastic container instances.
 - If you set the Type parameter to ECS when you create your scaling group and associate an ApsaraDB RDS instance with your scaling group, Auto Scaling automatically adds the private IP addresses of the ECS instances in the scaling group to the whitelist that manages access to the ApsaraDB RDS instance. This way, application data on the ECS instances can be stored on the ApsaraDB RDS instance.
- Scale-in activities:
 - Auto Scaling automatically removes the specified number of ECS instances or elastic container instances from your scaling group to release idle resources at the earliest opportunity.
 - If your scaling group is associated with an SLB instance, Auto Scaling automatically detaches the ECS instances or elastic container instances in the scaling group from the SLB instance. In this case, the SLB instance no longer forwards requests to the ECS instances or elastic container instances in your scaling group.
 - If you set the Type parameter to ECS when you create your scaling group and associate an ApsaraDB RDS instance with your scaling group, Auto Scaling automatically removes the private IP addresses of ECS instances in the scaling group from the whitelist that manages access to the ApsaraDB RDS instance. As a result, application data on the ECS instances can no longer be stored on the ApsaraDB RDS instance.

Cost-effectiveness

Auto Scaling provisions resources when the demand for the resources increases and releases resources when the demand for the resources decreases. This improves resource utilization and significantly reduces costs.

- You do not need to prepare extra ECS instances or elastic container instances to ensure service availability during peak hours. You also do not need to worry about the waste of resources. Auto Scaling scales computing resources at the earliest opportunity. This significantly reduces your resource costs.
- Auto Scaling also helps you save manpower and time.

High availability

Auto Scaling can detect whether an ECS instance or elastic container instance in your scaling group is healthy or whether the instance runs as expected. If Auto Scaling detects that an ECS instance or elastic container instance in your scaling group is not in the running state, it considers the instance unhealthy and replaces the instance with a new instance to ensure service availability.

Flexibility and intelligence

Auto Scaling provides a rich set of intelligent features that are suitable for diverse business scenarios and help simplify your configurations. This improves operational efficiency.

- **Scaling modes:** Auto Scaling supports the fixed-number mode, health mode, scheduled mode, dynamic mode, and custom mode. You can combine these scaling modes based on your business requirements. In dynamic mode, Auto Scaling interconnects with the external monitoring system CloudMonitor or by using API operations. For more information, see [Scaling modes](#).
- **Instance configuration sources:**
 - Auto Scaling supports more than one instance configuration source. For example, you can specify an existing instance as the instance configuration source or create a scaling configuration from scratch. If you set the Type parameter to ECS when you create a scaling group, you can also specify a launch template as the instance configuration source.
 - Auto Scaling also allows you to specify multiple instance types. This improves the flexibility of templates and increases the success rate of scale-out activities. For example, you can specify multiple instance types and disk categories in a template that is used to create ECS instances. You can also specify multiple vCPUs and memory sizes in a template that is used to create elastic container instances to determine the range of instance types. For more information, see [Overview](#).
- **Scaling policies:** If you set the Type parameter to ECS when you create a scaling group, Auto Scaling provides various scaling policies. If you set the Type parameter to ECI when you create a scaling group, the default scaling policy is used.
 - **Priority policy:** Auto Scaling preferentially scales instances in zones that have the highest priority. If the scaling activity fails, Auto Scaling scales instances in zones that have the next highest priority.
 - **Balanced distribution policy:** Auto Scaling evenly creates ECS instances in multiple zones to ensure high availability of the instances.
 - **Cost optimization policy:** If you specify multiple instance types in the instance configuration source, Auto Scaling preferentially creates ECS instances that have the lowest unit price of vCPUs and removes ECS instances that have the highest unit price of vCPUs.
 - **Instance removal policy:** You can remove ECS instances that are created from the earliest scaling configuration. You can also remove ECS instances that are created at the earliest or latest point in time.
 - **Instance reclaim policy:** You can release all ECS instances. You can also retain specific resources to reduce your costs.

For more information, see [Create a scaling group](#).

Easy audit

Auto Scaling records the details of each scaling activity. This way, you can effectively identify and troubleshoot issues.

Auto Scaling also provides the monitoring feature. You can use Auto Scaling together with CloudMonitor to monitor whether instances in your scaling group run as expected. This way, you can maintain service availability in an efficient manner.

2.Scenarios

Auto Scaling is suitable for applications that have fluctuating workloads and applications that have stable workloads. This topic describes the common application scenarios in which you can use Auto Scaling to scale Elastic Compute Service (ECS) instances and elastic container instances.

Unpredictable workload fluctuations

To demonstrate the benefits of Auto Scaling for scenarios that have unpredictable workload fluctuations, a news website is used as an example. The page views of the website increase sharply when breaking news is reported and decrease when the news becomes less hot. Traffic surges and drops are unpredictable. It is impractical and inefficient to manually adjust the number of instances, and you also may not know the exact number of instances that are required by your business.

In this case, you can use Auto Scaling to respond to unpredictable workload fluctuations. Auto Scaling allows you to create event-triggered tasks to monitor metrics such as CPU utilization, and then automatically scales your instances based on the monitoring results.

- Example 1: You can create two event-triggered tasks to trigger simple scaling rules. If the CPU utilization reaches 70% or more, one task triggers the simple scaling rule that adds three instances to your scaling group. If the CPU utilization drops below 30%, the other task triggers the simple scaling rule that removes three instances from your scaling group.
- Example 2: You can create an event-triggered task to trigger a target tracking scaling rule. This task keeps the CPU utilization of the instances at 50%.

Predictable workload fluctuations

To demonstrate the benefits of Auto Scaling for scenarios that have predictable workload fluctuations, a game company whose demand surges from 18:00 to 22:00 every day and drops after 22:00 is used as an example. It is impractical and inefficient for the company to manually adjust the number of instances.

In this case, you can use Auto Scaling to respond to predictable workload fluctuations. Auto Scaling allows you to create scheduled tasks, and then automatically scales your instances at specified points in time. You can create two scheduled tasks to trigger simple scaling rules. One task triggers the simple scaling rule that adds three instances at 17:55 each day. The other task triggers the simple scaling rule that removes three instances at 22:05 each day. Scheduled tasks help you handle the traffic spikes that occur during peak hours from 18:00 to 22:00 each day and automatically release instances during off-peak hours in an effective manner. When you use Auto Scaling, no instances remain idle, which can help you reduce costs.

Small workload fluctuations

To demonstrate the benefits of Auto Scaling for scenarios that have small workload fluctuations, a telecommunications company that has no obvious changes in demand during a course of time is used as an example. If the existing instances suddenly fail, the failed instances are difficult to be repaired or replaced in time, and the services of the telecommunications company are interrupted.

In this case, you can use Auto Scaling to perform health checks on your instances to ensure high availability of your service. Auto Scaling can automatically check the health status of your instances. If an instance is considered unhealthy, Auto Scaling automatically creates a new instance to replace the unhealthy instance to ensure your instances run as expected. You must configure the minimum number of instances in your scaling group. Auto Scaling ensures that your scaling group never goes below this size to keep your service uninterrupted.

Complex workload fluctuations

To demonstrate the benefits of Auto Scaling for scenarios that have complex workload fluctuations, a company whose daily traffic is stable but demand sometimes fluctuates is used as an example. If the company has some subscription instances and wants to adjust the number of instances only when the demand fluctuates, Auto Scaling can be used.

Auto Scaling allows you to manually add the subscription instances to your scaling group and create event-triggered tasks to monitor metrics such as CPU utilization for scaling activities. This way, Auto Scaling ensures that your instances run as expected and minimizes your costs.

Auto Scaling also allows you to create scheduled tasks and perform health checks on your instances based on your business requirements. You can use multiple Auto Scaling features at the same time to handle traffic spikes that occur in complex business scenarios and improve user experience.

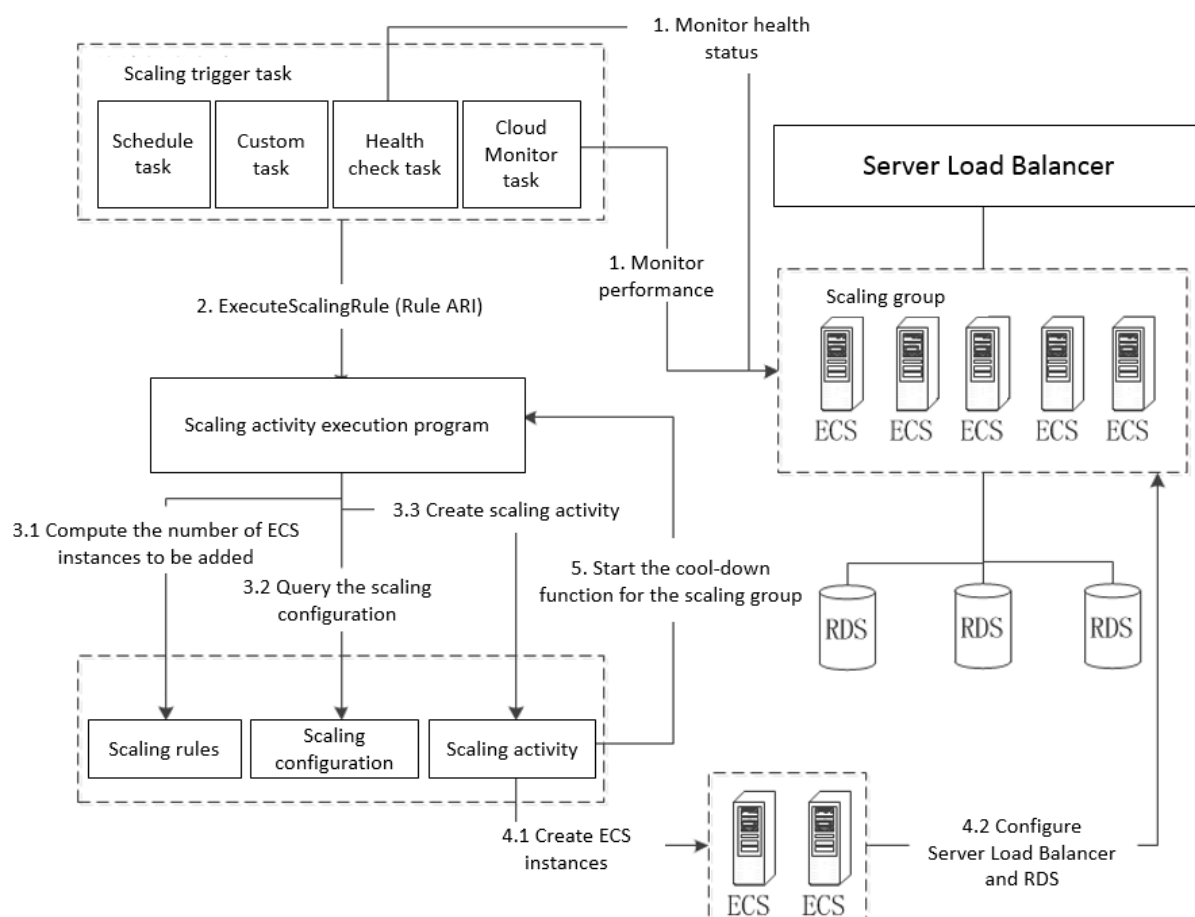
3. How Auto Scaling works

This topic describes how Auto Scaling works and how to configure scaling modes. This topic provides the workflow diagrams of Auto Scaling.

This topic describes how Auto Scaling works for Elastic Compute Service (ECS) instances. If you use a scaling group to manage elastic container instances, you cannot associate the scaling group with an ApsaraDB RDS instance. You also cannot manually add elastic container instances to or delete elastic container instances from the scaling group. Except for the preceding cases, you can manage elastic container instances in a scaling group by using the methods that you use to manage ECS instances in a scaling group.

Workflow


The following figure shows how Auto Scaling adds ECS instances.




In the example, a web application is used. The web application has a three-layer system architecture and uses ECS instances to process requests, as shown in the black dotted line box on the right side of the preceding figure. In the system architecture, the Server Load Balancer (SLB) instance at the top layer forwards the requests from the client to the ECS instances in the scaling group, which are at the middle layer. The ECS instances process the requests from the client. ApsaraDB RDS instances at the bottom layer store business data from the ECS instances.

You can use Auto Scaling to adjust the number of ECS instances at the middle layer based on your business requirements. The following procedure describes how Auto Scaling adjusts the number of ECS instances:

1. Auto Scaling triggers scaling activities if the conditions specified in scaling modes are met. For information about how to configure scaling modes, see [Configure a scaling mode](#). Auto Scaling supports the following scaling modes:
 - Fixed-number mode:
 - If you configure the **Minimum Number of Instances** parameter when you create a scaling group, Auto Scaling automatically adds ECS instances to the scaling group to maintain the specified minimum number of ECS instances in the scaling group.
 - If you configure the **Maximum Number of Instances** parameter when you create a scaling group, Auto Scaling automatically removes the excess ECS instances from the scaling group to maintain the specified maximum number of ECS instances in the scaling group.
 - If you configure the **Expected Number of Instances** parameter when you create a scaling group, Auto Scaling automatically adds ECS instances to or removes ECS instances from the scaling group to maintain the expected number of ECS instances in the scaling group.
 - Health mode: If you enable the health check feature when you create a scaling group, Auto Scaling checks the status of the ECS instances in the scaling group at specified intervals. If an ECS instance is not in the Running state, Auto Scaling considers the instance unhealthy and removes the instance from the scaling group.
 - Scheduled mode: You can create a scheduled task to automatically execute a scaling rule at a specified point in time.
 - Custom mode: You can manually perform scaling operations. For example, you can manually execute scaling rules, or add, remove, or delete ECS instances.
 - Dynamic mode: You can create an event-triggered task based on a performance metric that is monitored by CloudMonitor, such as the CPU utilization. If the metric value of the scaling group meets the specified alert condition, an alert is triggered and the specified scaling rule is executed. For example, if the average CPU utilization of all ECS instances in a scaling group exceeds 60%, an alert is triggered, and the specified scaling rule is executed.

 **Note** You can use the preceding scaling modes together based on your business requirements. For example, if your business loads significantly increase from 12:00:00 every day, you can create a scheduled task to automatically create 20 ECS instances at 12:00:00 every day. To make sure that the number of ECS instances meets your business requirements, you can use the scheduled mode together with other scaling modes such as the dynamic mode and the custom mode.

2. Auto Scaling calls the `ExecuteScalingRule` API operation to trigger scaling activities. In this API operation, Auto Scaling specifies the unique identifier of the scaling rule that you want the system to execute. Example: `ari:acs:ess:cn-hangzhou:140692647406****:scalingrule/asr-bp1dvirgwkoowxk7****`.

 **Note** If the metrics that are used in the dynamic mode are the metrics reported by your monitoring system to CloudMonitor, you must call the `ExecuteScalingRule` API operation in your program.

- If a scaling rule is created in the Auto Scaling console, you can find the scaling rule in the scaling rule list and click the ID of the scaling rule in the **Scaling Rule ID/Name** column to view the unique identifier of the scaling rule on the page that appears. Sample scaling rule ID: `asr-bp14u7kzh8442w9z****`. For more information about how to create scaling rules, see [Create a](#)

scaling rule.

- If a scaling rule is called by calling an API operation, you can call the [DescribeScalingRules](#) API operation to query the unique identifier of the scaling rule.
3. Auto Scaling uses the unique identifier to query the information about the scaling rule, scaling group, and scaling configuration and then triggers scaling activities.
 - i. Auto Scaling uses the unique identifier to query the information about the scaling rule and the scaling group to which the scaling rule applies, and then calculates the number of ECS instances that are required. Auto Scaling also queries the information about the SLB instance and the ApsaraDB RDS instance to which the required ECS instances are attached.
 - ii. Auto Scaling queries the information about the scaling configuration in the scaling group. The information includes the vCPUs, memory, and bandwidth of the ECS instances that are required.
 - iii. Auto Scaling triggers scaling activities based on the required number of ECS instances, instance configuration, SLB instance, and ApsaraDB RDS instance.
 4. During the scaling activities, Auto Scaling creates the required number of ECS instances and attaches the ECS instances to the SLB instance and ApsaraDB RDS instance.
 - i. Auto Scaling creates the required number of ECS instances based on the instance configuration information.
 - ii. Auto Scaling adds the private IP addresses of the ECS instances that are created to the whitelist that manages access to the ApsaraDB RDS instance, and then adds the ECS instances as the backend servers of the specified SLB instance.
 5. After a scaling activity is complete, Auto Scaling enables the cooldown feature for the scaling group.

The scaling group can receive new requests to execute scaling rules only after the cooldown time expires.

Configure a scaling mode

Auto Scaling can automatically trigger scaling activities to add ECS instances to or remove ECS instances from a scaling group based on your configurations. You can configure scaling modes for Auto Scaling to trigger scaling activities. The following table describes the scaling modes.

Scaling mode	Configuration method	Description
Fixed-number mode	Scaling group + Instance configuration source ¹	The scaling in the fixed-number mode varies based on the values of the following parameters: <ul style="list-style-type: none">• Minimum Number of Instances• Maximum Number of Instances• (Optional) Expected Number of Instances
Health mode	Scaling group + Instance configuration source ¹	You must turn on Instance Health Check for the scaling group.
Scheduled mode	Scaling group + Instance configuration source + Scaling rule + Scheduled task ²	The scaling in the scheduled mode varies based on the configurations of scheduled tasks.

Scaling mode	Configuration method	Description
Dynamic mode	Scaling group + Instance configuration source + Scaling rule + Event-triggered task ³	The scaling in the dynamic mode varies based on the configurations of event-triggered tasks.
Custom mode	Custom configuration method	In this mode, you can manually add, remove, or delete ECS instances. You can also manually execute the scaling rules that you created.
Multiple modes	Combination of the preceding configuration methods	<p>The configurations that take effect vary based on the scaling modes that are used. The scaling modes that are used are independent of each other. The scaling modes have no priorities. Auto Scaling first executes the configurations of the scaling mode that is first triggered.</p> <p>For example, if you use the scheduled mode together with the dynamic mode to cope with business demands, you must create both scheduled tasks and event-triggered tasks. If the scheduled mode is triggered earlier than the dynamic mode, Auto Scaling executes the scheduled task before it executes the event-triggered task.</p>

The following items describe each configuration method:

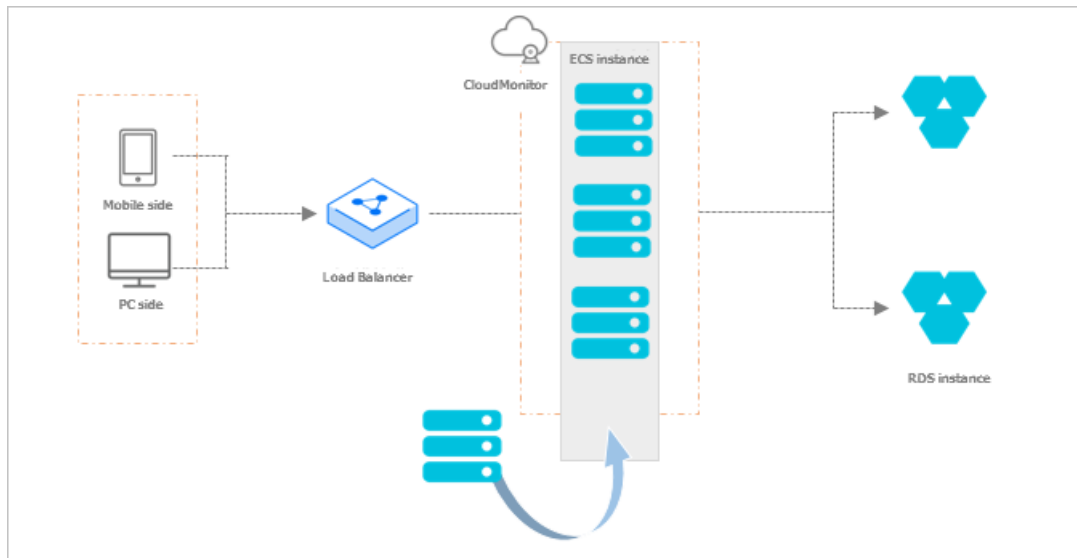
- 1. Scaling group + Instance configuration source. You must create a scaling group, configure an instance configuration source for the scaling group, and then enable the instance configuration source and the scaling group. Auto Scaling can automatically scale instances only after the preceding operations are complete. Scaling group and instance configuration source are the basic configuration unit. You must configure the basic configuration unit.
- 2. Scaling group + Instance configuration source + Scaling rule + Scheduled task. In addition to the basic configuration unit in Method 1, you must create a scaling rule and a scheduled task. Auto Scaling executes the scaling rule based on the scheduled task.
- 3. Scaling group + Instance configuration source + Scaling rule + Event-triggered task. In addition to the basic configuration unit in Method 1, you must create a scaling rule and an event-triggered task. Auto Scaling executes the scaling rule based on the event-triggered task.

Workflow diagrams

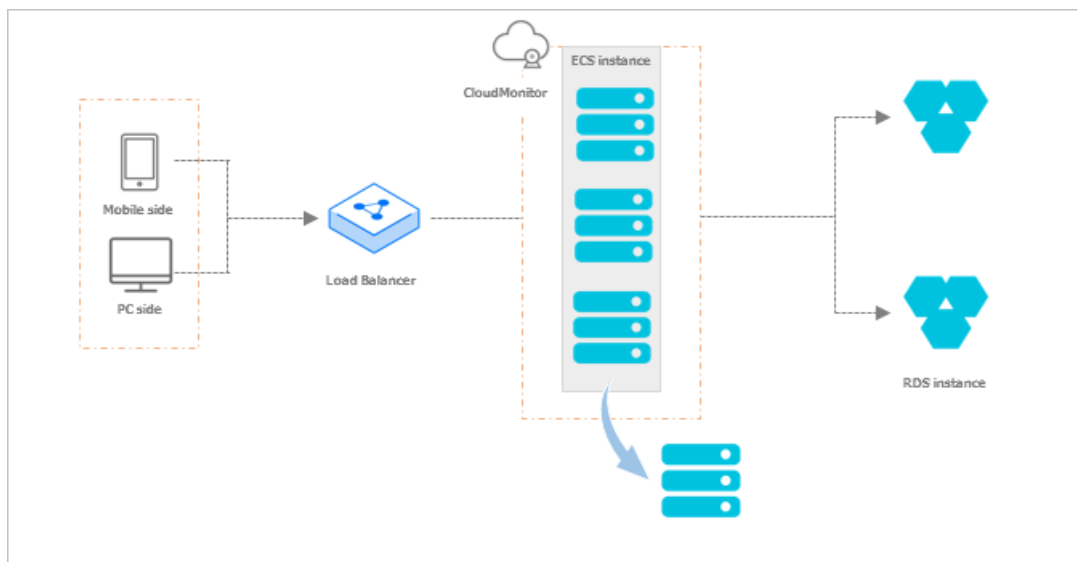
Auto Scaling allows you to associate your scaling groups with SLB instances and ApsaraDB RDS instances. When you send a request from a mobile device or a PC, the associated SLB instance forwards the request to one ECS instance in the scaling group. The ECS instance receives and processes the request. The associated ApsaraDB RDS instance stores the application data.

Auto Scaling adjusts the number of ECS instances in your scaling group based on your business requirements and the scaling modes that you configured. The following figures show the processes of scale-out, scale-in, and elastic recovery (health check) events in Auto Scaling.

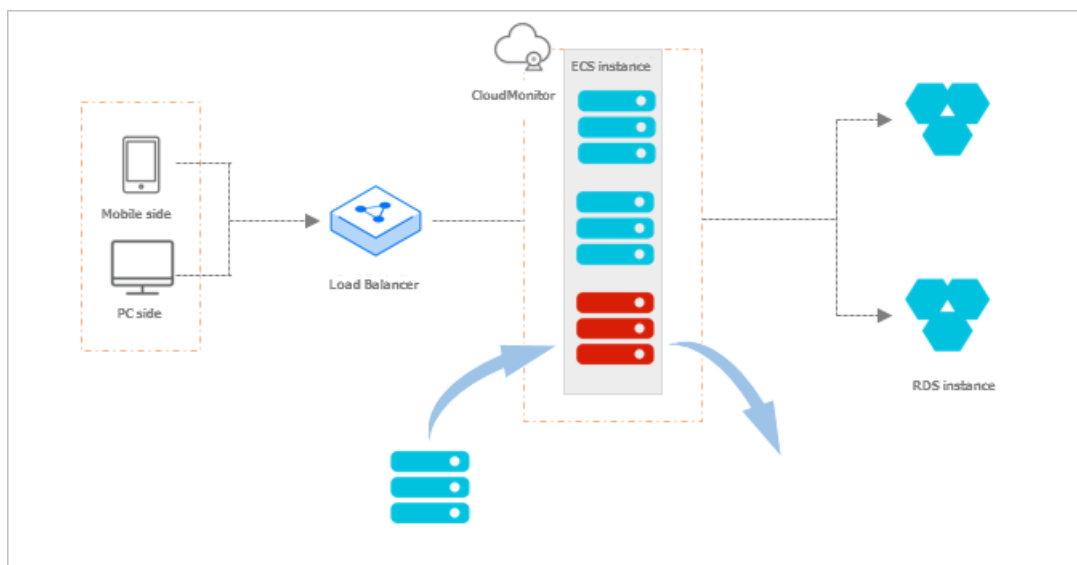
Scale-out in Auto Scaling



Scale-in in Auto Scaling



Elastic recovery in Auto Scaling




4.Limits

This topic describes the limits on the features and resource quantities of Auto Scaling.

Limits on features

Auto Scaling features have the following limits:

- The applications that are hosted on ECS instances or elastic container instances in a scaling group must be stateless and horizontally scalable.
- ECS instances or elastic container instances in a scaling group may be automatically released. We recommend that you do not store information such as sessions, application data, or logs on ECS instances or elastic container instances in a scaling group. If the applications that are hosted on ECS instances or elastic container instances require persistent storage of data, we recommend that you store status information such as sessions on independent ECS instances or elastic container instances, application data in ApsaraDB RDS, and logs in Log Service. For more information, see [What is ApsaraDB RDS?](#) and [What is Log Service?](#).
- Auto Scaling cannot automatically add the IP addresses of the ECS instances or elastic container instances to the whitelist that manages access to ApsaraDB for Memcache instances. Therefore, you must manually add the IP addresses of the ECS instances or elastic container instances to the whitelist. For more information, see [Configure an IP address whitelist](#).
- If you delete the ApsaraDB RDS instance, the Application Load Balancer (ALB) server group, the Classic Load Balancer (CLB) instance, or the backend server group of the CLB instance that is associated with a scaling group, the scaling group is automatically disassociated from the deleted resource.
- If a scaling activity that is automatically triggered in a scaling group fails for 30 consecutive days or more, Auto Scaling disables the auto-triggered scaling feature for the scaling group during system inspections and notifies you by using text messages or emails.

 **Note** If a scaling task fails consecutively, we recommend that you handle the issue at the earliest opportunity. You can log on to the [Auto Scaling console](#). You can also [submit a ticket](#) to contact technical support.

Limits on resource quantities

The following table describes the limits on the quantities of Auto Scaling resources within a single account.

Item	Quota
Scaling groups in a single region	The quota is determined based on the amount of Auto Scaling resources that you use. For more information, go to Quota Center .
Scaling configurations in a single scaling group	
Scaling rules in a single scaling group	
ApsaraDB RDS instances that can be associated with a scaling group	
CLB instances that can be associated with a scaling group	

Item	Quota	Note
ALB server groups that can be associated with a scaling group		You can submit a ticket to request a quota increase.
vServer groups that can be associated with a scaling group		
Maximum number of instances that are allowed in a single scaling group		
Scheduled tasks in a single region		
ECS instances or elastic container instances that can be scaled at a time	500	
Instance types in a single scaling configuration	10	
Event notifications in a single scaling group	6	
Lifecycle hooks in a single scaling group	10	



5.Terms

This topic describes the common terms used in Auto Scaling.

Terms

Term	Description	References
Auto Scaling	Auto Scaling is a service that automatically changes the number of instances to adjust the computing power based on the business requirements and policies. Auto Scaling can scale Elastic Compute Service (ECS) instances and elastic container instances. Auto Scaling automatically adds instances of the specified type during peak hours to ensure sufficient computing power. Auto Scaling automatically removes instances of the specified type during off-peak hours to reduce resource costs.	什么是弹性伸缩Auto Scaling
scaling group	A scaling group is a group of instances that can be used for the same scenario. Instances within a scaling group must be of the same type. You can specify the minimum and maximum numbers of instances in a scaling group, and associate Server Load Balancer (SLB) instances and ApsaraDB RDS instances with the scaling group.	Overview
ECS instance	An ECS instance is a virtual server that includes basic computing components such as vCPU, memory, operating system, bandwidth, and disk. ECS eliminates the need for upfront investments in IT hardware and allows you to scale computing resources on demand. This makes ECS instances more convenient and efficient than physical servers.	What is ECS?
elastic container instance	Elastic Container Instance is a container service provided by Alibaba Cloud that combines container and serverless technologies.	What is Elastic Container Instance?
SLB instance	SLB is a service that forwards network traffic to backend servers to increase the throughput of your applications. You can use SLB to prevent service interruptions that are caused by single points of failure (SPOFs) and improve the availability of applications.	SLB Overview
ApsaraDB RDS instance	ApsaraDB RDS is a stable and reliable online database service that supports elastic scaling. ApsaraDB RDS supports mainstream database engines and provides a variety of database solutions, such as disaster recovery, backup, restoration, monitoring, and migration.	What is ApsaraDB RDS?

Term	Description	References
scaling mode	A scaling mode specifies when to add or remove a specific number of instances for a scaling group. Scaling modes include scheduled mode, dynamic mode, fixed quantity mode, custom mode, health mode, and multi-mode.	Scaling modes
instance configuration source	Auto Scaling uses the instance configuration source that you select to create instances. The instance configuration source can be a scaling configuration or a launch template.	Overview
scaling configuration	A scaling configuration is a type of instance configuration source and includes the configuration information of instances.	Create a scaling configuration (ECS)
scaling rule	<ul style="list-style-type: none"> Step scaling rules, target tracking scaling rules, and simple scaling rules are used to add or remove instances when scaling activities are triggered. Predictive scaling rules are used to predict the future metric values based on historical monitoring data and intelligently specify boundary values for scaling groups. 	Overview
scaling task	Scaling tasks are categorized into scheduled tasks and event-triggered tasks. A scheduled task can be used to scale instances at a specified time. An event-triggered task can be used to dynamically scale instances based on specified monitoring metrics.	<ul style="list-style-type: none"> Create a scheduled task Overview
scaling activity	A scaling activity records the changes in the number of instances within a scaling group, the boundary values of the scaling group, and the expected number of instances. Scaling activities are triggered when scaling rules are executed, the boundary values of a scaling group are modified, or the expected number of instances is modified.	View the details of a scaling activity
expected number of instances	<p>After the Expected Number of Instances feature is enabled, Auto Scaling automatically maintains the number of instances at the expected value.</p> <div>  Note To enable the Expected Number of Instances feature, you must set the Expected Instances parameter when you create a scaling group. The specified expected number of instances in the scaling group can be modified. </div>	Expected number of instances

Term	Description	References
Parallel scaling activity	<p>A scaling activity triggered by using one of the following methods is a parallel scaling activity:</p> <ul style="list-style-type: none"> • Execute a scaling rule manually or by using a scheduled task. • Manually add or remove ECS instances. • Perform a check on the instance health, or the expected, minimum, or maximum number of instances. <p>A parallel scaling activity can be executed if the ongoing scaling activities are also parallel scaling activities.</p> <div> <p> Note Parallel scaling activities and non-parallel scaling activities are differentiated only if the Expected Number of Instances feature is enabled. If the Expected Number of Instances feature is disabled, no other scaling activities can be executed when a scaling activity is in progress.</p> </div>	<p>Expected number of instances</p>
Non-parallel scaling activity	<p>Scaling activities other than parallel activities are non-parallel scaling activities. No other scaling activities can be triggered when a non-parallel scaling activity is in progress.</p> <div> <p> Note Parallel scaling activities and non-parallel scaling activities are differentiated only if the Expected Number of Instances feature is enabled. If the Expected Number of Instances feature is disabled, no other scaling activities can be executed when a scaling activity is in progress.</p> </div>	<p>Expected number of instances</p>
stable instance	<p>A stable instance refers to an ECS instance that is in the In Service, Protected, or Standby state in a scaling group.</p>	<p>ECS instance lifecycle in a scaling group</p>
scaling process	<p>A scaling process refers to a process that you can manually suspend and resume, such as the scale-out, scale-in, health check, scheduled task, and event-triggered task. Scaling processes help you control scaling groups at the process level.</p>	<ul style="list-style-type: none"> • Suspend a scaling process • Suspend and resume scaling processes

Term	Description	References
ECS instance lifecycle	<p>The lifecycle of an ECS instance in a scaling group refers to the process from the time when the instance is created to the time when it is released. The lifecycle management mode of an ECS instance depends on how the instance is created.</p> <ul style="list-style-type: none">• If the instance is automatically created by Auto Scaling, the lifecycle of the instance is managed by the scaling group.• If the instance is manually created and you enable the scaling group to manage the lifecycle of the instance, the lifecycle of the instance is managed by the scaling group. If you do not enable the scaling group to manage the lifecycle of the instance, you must manually manage the lifecycle of the instance.	ECS instance lifecycle in a scaling group
lifecycle hook	<p>A lifecycle hook allows ECS instances that are being added to or removed from a scaling group to enter the Pending state. After the ECS instances enter the Pending state, you can perform custom operations on them. For example, after an ECS instance is created, you can use a lifecycle hook to allow the instance to enter the Pending state. You can perform tests on the instance to ensure its service availability. Then, Auto Scaling adds the instance as a backend server to an associated SLB instance.</p>	配置生命周期挂钩
cooldown time	<p>The cooldown time refers to a period during which Auto Scaling cannot execute new scaling activities after one scaling activity is complete in a scaling group. During the cooldown time, Auto Scaling rejects all scaling activity requests of event-triggered tasks from CloudMonitor. This prevents scaling activities from being frequently triggered due to fluctuations in the metric value.</p>	Cooldown time