

# Function Compute

Pricing

# Pricing

## Billing methods

Function Compute is billed on a pay-as-you-go basis. The fee consists of **executions cost**, **execution duration cost**, and optional **data transfer cost**. Function Compute also provides a certain amount of free processing quota every month.

We provide a pricing calculator. You can estimate the cost based on the number of function calls per day, the memory configured for the function, and the function execution duration.

**Note: A request is billed only after your function code is executed.** Therefore:

1. Requests whose response status code is 4XX or 5XX due to errors such as parameter, permission, and service errors will not be billed.
2. Requests whose response status code is 200 but errors occurred during the execution of function code will be billed.
3. Whether requests triggered by HTTP triggers will be billed is determined by the value of X-Fc-Error-Type in the response header. If the value is FCCommonError, the function is not executed and the corresponding request will not be billed. If the value is any other value, the corresponding request will be billed.

## Executions cost

The executions cost indicates the total number of function executions per hour.

Price: USD 0.2/1 million executions.

## Execution duration cost

Price: The price for 1 GB memory is USD 0.000016384/GB-s or USD 0.0000016384/GB-100 ms.

Note: The execution duration cost depends on the memory size that you have allocated to functions and the duration of function execution. An execution duration starts when your function starts to run and ends when the result is returned or the execution is terminated. The duration measurement granularity is 100 milliseconds. Therefore, your execution duration is rounded up to the nearest 100 milliseconds. For example, if your function execution duration is 1,010 milliseconds, the duration is

measured as 1,100 milliseconds.

Example: The function application of Alan's website is allocated 2 GB of memory, and a function runs for 1,010 milliseconds. Alan is billed  $0.000016384 \times (2 \times 1.1) = \text{USD } 0.0000360448$  for this function.

## Data transfer cost

Data transfer cost is only incurred when data is transferred on the public network in your function. Data that is transferred on the public network includes two parts:

1. Data that is returned from the **public network** when a function is called
2. Data that is transferred to the **public network** when a function is called

The total public network traffic is charged based on Elastic Compute Service's (ECS's) **pay-as-you-go** billing method. It is currently charged as **USD 0.117/GB**. For more information, see *ECS's billing method for public network bandwidth*.

For example, Alan calls a function and uploads a file of 10 MB to a public server in the function. The function returns the result of successful execution, which is a JSON string of 200 bytes. Then, the used public network traffic is 10 MB + 200 byte, and the cost is  $0.117 \times (10.0/1024 + 200.0/1024/1024/1024) = \text{USD } 0.00114$ .

Note:

1. Traffic used for accessing the internal network of Alibaba Cloud is not deemed public network traffic and therefore will not be charged.
2. If you use API Gateway to call a function and API Gateway is in the same region as Function Compute, the function is called within the internal network and no traffic cost is incurred.
3. If you call a function in different regions, data transfer cost will be charged.

## CDN back-to-origin data transfer cost

If you set content delivery network (CDN) back-to-origin processing to Function Compute, you need to pay for the traffic of transfer over the back-to-origin network.

The price of public network traffic in China is USD 0.117/GB. If your network traffic is heavy, submit a ticket for a quotation.

## Subscription

Function Compute supports subscription. You can purchase a resource consumption plan for a certain duration in advance. Within its lifecycle, the resource consumption plan is deducted by second for the resources consumed by the execution of a function. By subscribing to Function

Compute, you can spend less money on the same resources, substantially reducing your costs. For more information, see [Function Compute \(subscription\)](#).

## More information

If you use other Alibaba Cloud service instances in your functions, you will be charged additional fees. For example, if you use a function to write data to Alibaba Cloud Object Storage Service (OSS), or to activate a Log Service instance to store debugging information, these service instances are charged additionally. For more information, see [OSS billing items](#) and [Log Service billing methods](#).

## Free monthly quotas

Your primary account and RAM user account share the free monthly quotas of executions and execution duration. Free quotas are automatically cleared at the beginning of each calendar month, rather than accrued to the next month.

**Executions:** The first one million calls per month are free of charge.

**Execution duration:** The first 400,000 GB\*s per month are free of charge.

You can view the monthly free quotas, which is the number of free seconds for different memory sizes in the following table.

Memory (MB)	Free seconds per month
128	3,200,000
192	2,133,333
256	1,600,000
320	1,280,000
384	1,066,667
448	914,286
512	800,000
576	711,111
640	640,000
704	581,818
768	533,333
832	492,308
896	457,143
960	426,667

1,024	400,000
1,088	376,471
1,152	355,556
1,216	336,842
1,280	320,000
1,344	304,762
1,408	290,909
1,472	278,261
1,536	266,667

## Differences between old and new pricing methods

Function Compute is billed based on the number of requests, execution duration, amount of Internet downstream transfer, and amount of CDN back-to-origin transfer. Only the execution duration is metered in different ways in Function Compute V1 and V2.

In Function Compute V1, the execution duration of requests is metered. Function Compute sums up the execution duration of all requests under your account, and multiplies it by the specification of the instance that processes the requests. The final value is used to calculate the charges within the current billing cycle. In Function Compute V2, the execution duration of instances is metered. Function Compute sums up the execution duration of all instances under your account, and multiplies it by the specification of the instances. The final value is used to calculate the charges within the current billing cycle. The execution duration of reserved and pay-as-you-go instances is metered using different standards. For more information, see the latest Pricing topic.

In Function Compute V1, the measure unit is GB-second. For example, it takes 3 seconds for an instance with 2 GB of memory to process a request. The amount of resources consumed is 6 GB-second ( $2 \text{ GB} \times 3 \text{ seconds}$ ). In Function Compute V2, the measure unit is CU-second. Capacity units (CUs) are units used to measure the computing capabilities of instances. For pay-as-you-go instances, one CU is equivalent to one pay-as-you-go instance with 1 GB of memory. 1 GB-second of resources consumed by a pay-as-you-go instance is equivalent to 1 CU-second of resources.

The following table compares the metering patterns of Function Compute V1 and V2.

Item	Function Compute V1	Function Compute V2
Object	Request	Instance
Unit	GB-second	CU-second

Subscription	Unsupported	Supported
Reserved instance	Unsupported	Supported

## Compatibility

Function Compute V1 is billed based on the number of requests. Each request is processed by a dedicated instance. The execution duration of a request equals the execution duration of the instance that processes the request. Therefore, Function Compute V2 complies with the pricing standards of Function Compute V1. Both Function Compute V1 and V2 return the same metering result for a given function.

## Appendix

1. Function Compute V1 pricing
2. Function Compute V2 pricing

## Overdue payment

Your Alibaba Cloud account has an overdue payment if your payment method, such as linked credit card or vouchers, cannot afford the cost of the last billing cycle.

Function Compute calculates fees on an hourly basis. At the start of every hour, Function Compute calculates the cost of your services in the last hour. The charges are automatically deducted from your any applicable payment methods, and a bill is generated for later inquiry and review. For more information, see [Billing method](#).

## Consequences

You can still use Function Compute normally after an overdue payment occurs within the 24x7 hours. However, all of your Function Compute services are frozen afterwards. The services and features are frozen as follows:

The execution of running functions is still carried out, and new requests are no longer processed after the tasks are complete.

Scheduled functions are no longer triggered.

All requests in the asynchronous task queue are paused. After 96 hours, requests which have

not been processed are gradually removed from the request queue.

## Restore service

To restore service functions, recharge your account and pay the overdue amount.

If a request has remained in the asynchronous task queue for less than 96 hours since the account became overdue, it is processed normally when the service functions are restored.

## See also

The Function Compute can be used flexibly with other Alibaba Cloud services, meeting diverse needs. Different cloud services use different billing methods. To guarantee that you can use Function Compute without interruption, we recommend that you pay close attention to linked credit limits, overdue alert emails, the usage of Function Compute and other cloud services. For more information, see *Alibaba Cloud Pricing*.

# Best practice for cost optimization

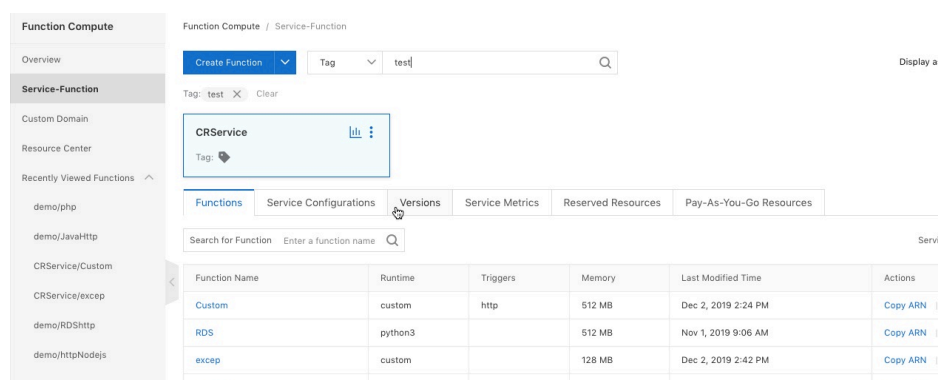
## Billing methods

Different applications may have different requirements on the type and scalability of compute resources. Function Compute supports two billing methods: subscription and pay-as-you-go. You can choose an appropriate billing method based on the scenario to save costs. Subscription means that you must pay for the resources before you can use them. In Function Compute, you can use reserved instances to offset the resource costs. The benefit of subscription is cost-effectiveness. In comparison to pay-as-you-go, it saves 70% of the costs. Subscription also has its own limits. In most cases, the number of resources that you purchase is determined based on the peak load. It is difficult to maintain a high utilization of the purchased resources during off-peak hours. Pay-as-you-go means that you only pay for the resources that you have used. Pay-as-you-go instances in Function Compute are charged based on the amount of time that it takes the instances to process requests, which is accurate to milliseconds. The pay-as-you-go instances are not charged when they are idle. Therefore, the utilization of pay-as-you-go instances is considered 100%. The benefit of pay-as-you-go is high resource utilization. However, pay-as-you-go instances are charged higher than reserved instances. The elastic scaling feature of Function Compute allows you to use reserved and pay-as-you-go instances simultaneously to gain benefits from both instance types.

Your reserved instances are shared by all the functions in the same region under your account. Function Compute sums up the amount of resources used by all the functions per second and uses reserved instances to offset the costs. For more information about subscription and pay-as-you-go, see [Pricing](#).

## Cost structure

Before you optimize the costs, you must analyze the types and proportions of the costs that your workloads incur. Fine-grained resource usage tracking can help you optimize costs more efficiently. Function Compute allows you to query resource usage information on a function, service, or user account basis. You can also classify services or functions by using tags. For example, you can use tags to classify and track functions by environment (test or production environment), application, or owner.



## Use reserved instances to optimize costs

As described in the preceding sections, Function Compute provides multiple features for you to analyze and optimize your costs. Based on these features, a variety of approaches can be applied to improve resource utilization and reduce costs.

### Approach 1: choose an appropriate billing method to fit your applications

It is important to choose an appropriate billing method based on the way how your applications use the resources. For example, some online applications in a production environment are sensitive to the response latency. You must reserve a certain number of resources to guarantee a low response latency. In this case, you can use reserved instances, which can help you reduce costs. Tasks running test cases or running at a scheduled time require high throughput. Additional resources may be required to handle unexpected loads when the end date takes priority. In this case, you can use pay-as-you-go instances to balance resource utilization and costs. As shown in the following figure, you can analyze the usage of both reserved and pay-as-you-go resources from the Function Compute console. The resource usage information can be used as a reference for resource sizing, helping you

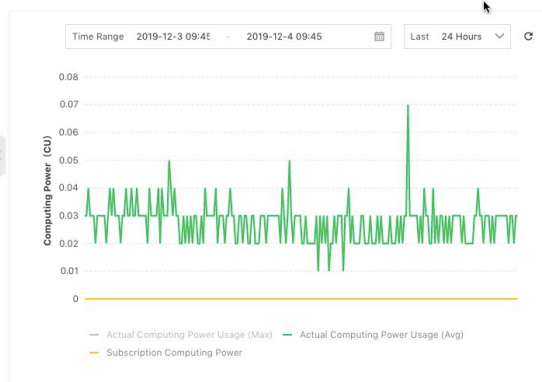


allocate reserved and pay-as-you-go resources properly.

#### Resource Center

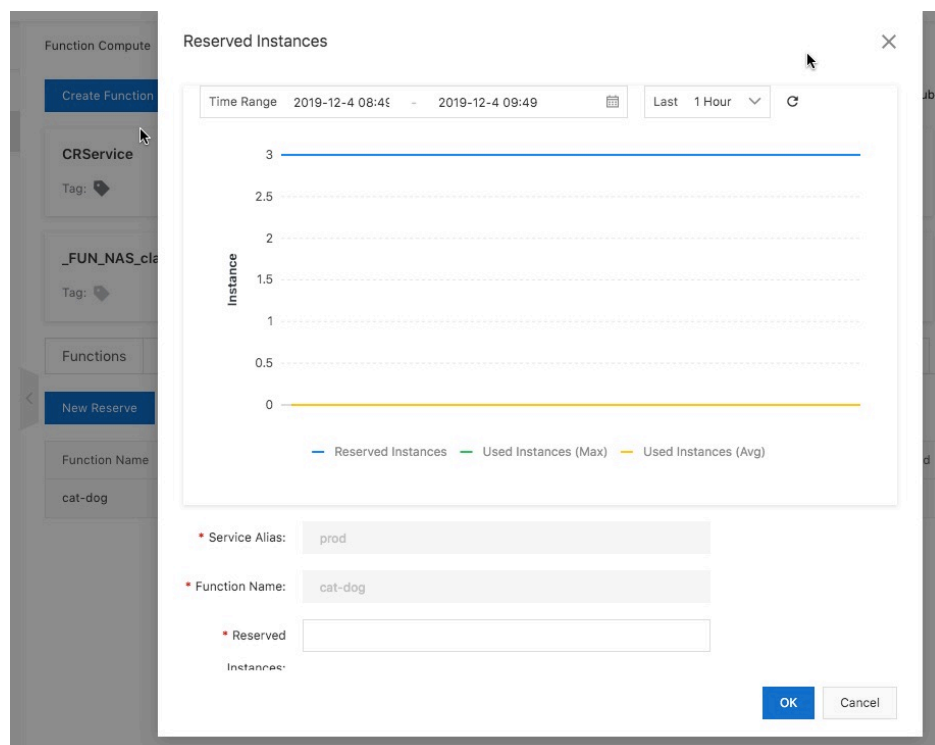
##### Resource Usage Details

Last updated: Dec 4, 2019 5:55 AM



## Approach 2: release reserved instances that are idle

Function Compute supports fine-grained and multi-dimensional resource usage monitoring. You can check the usage of reserved and pay-as-you-go resources for each function, and release reserved resources that are idle to save costs. In addition, you can use tags to filter functions when you query resource usage information. For example, you can list all the functions that are used for testing purposes and then check whether they have idle resources. As shown in the following figure, the reserved instances allocated to the function have remained idle for a long period of time. In this case, you need to release the reserved instances to save costs.



## Approach 3: use load leveling to deal with fluctuations

To handle a large number of pending requests where the end date is flexible, you can use asynchronous calls to process the requests, and throttle the maximum number of instances launched concurrently. This can help you deal with fluctuations with ease. As shown in the following figure, the resource demand barely fluctuates. You can use reserved instances to offset the overheads, which is cost-effective. Function Compute will automatically cache requests in queues and process requests at a rate that corresponds to the specified maximum resource limit. It also uses load balancing and resource isolation to guarantee the overall performance of function invocation. You only need to specify the maximum number of instances to be launched concurrently.

## Approach 4: run workloads during off-peak hours

For function invocation tasks such as regression tests, their end dates are flexible. You may want to run these tasks during off-peak hours, such as a time period between 21:00 of the current day and 9:00 of the next day. Function Compute provides triggers for you to run your tasks at a scheduled time. In this way, you are able to use reserved instances to run these tasks with low costs.

## Summary

The variety of billing methods, competitive pricing, and fine-grained resource usage monitoring of Function Compute can help you manage resources for serverless applications with ease and save resource costs.

# Function Compute (Subscription)

## Function Compute (Subscription)

### Function Compute (subscription)

Function Compute supports subscription. You can purchase a resource consumption plan for a certain duration in advance. Within its lifecycle, the resource consumption plan is deducted by second for the resources consumed by the execution of a function. By subscribing to Function Compute, you can spend less money on the same resources, substantially reducing your costs.

## What is a resource consumption plan?

- Purpose: A resource consumption plan is deducted by second within its lifecycle when a Function Compute instance consumes resources during the execution of the function.
- Unit: CU. 1 CU is equal to 1 GB of memory, which means that a resource consumption plan of 1 CU can match the 1 GB\*s of resources consumed in each second.
- Type: constant. If the resource consumption plan is not upgraded within its lifecycle, the resource consumption plan is deducted for resources consumed in each second in a constant manner, and not in a decreasing manner. The nature of the resource consumption plan is similar to bandwidth, but not to a data transfer plan.
- Regional restriction: Your resource consumption plan can only be used in the region that is specified when you purchase the resource consumption plan.
- Matching objects: A resource consumption plan can match both resources purchased with subscription and pay-as-you-go resources.

For example, the following figure shows a new order page for Function Compute (subscription). If you have purchased 3 CUs for one month in the China (Shenzhen) region, 3 GBs *resources can be consumed in each second within one month. If the resources actually consumed by your instance in one second do not exceed 3 GBs*, you do not need to pay again for the resources consumed in this second. Otherwise, you need to pay for the part of resources that exceeds 3 GB\*s. If the resource consumption plan is not fully used in one second, it will not be accrued to the next second.

Function Compute (prepaid service)

Region	China (Hangzhou)	China (Shanghai)	China (Qingdao)	China (Beijing)	China (Zhangjiakou)	China (Hohhot)
	China (Shenzhen)	China (Hong Kong)	Singapore	Australia (Sydney)	Indonesia (Jakarta)	Japan (Tokyo)
	India (Mumbai)	Germany (Frankfurt)	US (Silicon Valley)	US (Virginia)		

Current Selected

Region: China (Zhangjiakou)

Computing Power: 1 CU

Purchase Cycle: 1 Month(s)

Fee: \$12.16

Buy Now

价格有疑问点这里

Computing Power: 250CU 500CU 1000CU 1 CU

Note: 1CU computing power ≈ 1GB memory and 0.66 core CPU. details Prepaid Resource Documentation.

Purchase Cycle: 1 month 2 3 4 5 6 7 8 9 1 yr 2 yr 3 yr Auto Renew

## Features

**Order ID:** An order is generated when a request for a new order, renewal, or upgrade is initiated. Each order has a globally unique ID.

**Instance ID:** An instance ID is generated for each new order. This ID is used in all subsequent operations, such as renewal and upgrade, for the order, and is a unique ID that is used in the entire lifecycle of the new order.

- New order: You can purchase a certain number of CUs in a specified region and specify the duration. For more information, see [Manage new orders in Function Compute \(subscription\)](#).
- Upgrade: You can add more CUs to a specified instance so that more resources can be

consumed in a unit of time. For more information, see [Manage upgrades in Function Compute \(subscription\)](#).

- Renewal: You can extend the lifecycle of a specified instance. For more information, see [Manage renewal in Function Compute \(subscription\)](#).
  - Manual renewal: You can manually renew an instance.
  - Auto renewal: You can enable auto renewal so that the system automatically initiates a renewal request before the lifecycle of your instance ends.

## Pricing

Unit	Price
1 CU*month	12.16USD

## Purchase restrictions

Limit on the number of CUs in a resource consumption plan: Function Compute has no limit on the number of CUs in a resource consumption plan. Currently, the only limit is on the maximum number of CUs that can be purchased for a single order, which is 5,000 CUs. If you need more than 5,000 CUs, you can place multiple orders to increase the total number of CUs in the resource consumption plan for a single region. If you feel it is inconvenient to manage multiple orders, contact us by submitting a ticket and we will adjust the upper limit on the number of CUs in the resource consumption plan for a single order based on your requirements.

## FAQ

When will my resource consumption plan take effect?

There is a slight delay after you successfully pay for the resource consumption plan and before the resource consumption plan takes effect, of about 1 to 15 minutes. The effective lifecycle starts counting when the resource consumption plan takes effect, but not when you successfully pay for the resource consumption plan.

How can I identify whether the resource package takes effect after I purchase a resource consumption plan?

After purchasing the resource consumption plan, you can log on to the Function Compute console and click Resource Center in the left-side navigation pane. On the Resource Center page, view the Purchased Resources list, which displays active instances. If you do not know whether the resources that you subscribed to have taken effect, contact us and we will check this for you. If the resource consumption plan does not take effect for up to 4 hours due to system reasons, we will refund the full amount. After receiving the refund, you can make another purchase.

If I purchase a resource package of one month or one year, when will the resource package expire?

- Subscription by months: The system will count the lifecycle by natural months. If you purchase resources of one month, the expiration time will be 00:00 on the day following the day of the month when you purchased the resources, in the next month.
  - Example 1: If you place an order of one month at 15:00 on August 14, the expiration time will be 00:00 on September 15 and the lifecycle is 31 days and 9 hours.
  - Example 2: If you place an order of two months at 15:00 on August 14, the expiration time will be 00:00 on October 15 and the lifecycle is 61 days and 9 hours.
  - Example 3: If you place an order of one month at 15:00 on January 29, January 30, or January 31, the expiration time will be 00:00 on March 1.
- Subscription by years: If you purchase resources of one year, the expiration time will be 00:00 on the day following the day of the year when you purchased the resources, in the next year.
  - Example: If you place an order at 15:00 on August 14, 2019, the expiration time will be 00:00 on August 15, 2020 and the lifecycle is 366 days and 9 hours.

Can I renew a resource consumption plan after expiration?

You can manually renew the resource consumption plan within seven days after expiration. After seven days have passed, you can no longer renew the resource consumption plan. (Note: We recommend that you renew instances before they expire, because we cannot assure that renewal will be successful after expiration.)

Is cross-region deduction supported?

No.

Can I use a resource consumption plan in an overlapped manner?

Yes.

What can I do if I want a refund when using Function Compute (subscription)?

Function Compute (subscription) does not support features such as the refund of your balance, no-reason refund within five days, and the return of renewal fees. For more information, submit a ticket for consultation.

# Purchase Operation Management

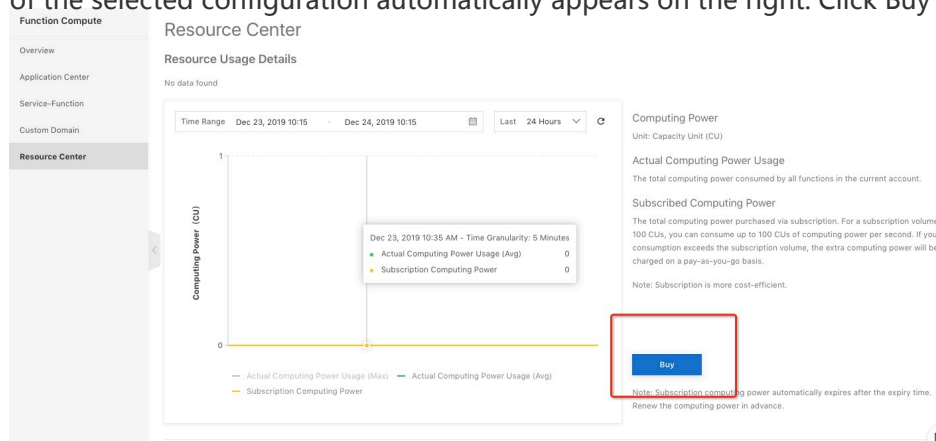
## Place a new order

You can purchase a resource consumption plan for a specified duration for a specified region. After the instances take effect, the resource consumption plan can be deducted by second for the resources consumed by your reserved instances and pay-as-you-go instances within the lifecycle of the instances. If the resource consumption plan can cover the resources that are consumed, you do not need to pay for the resources. Otherwise, you need to pay for the part of the resources that is not covered by the resource consumption plan.

Example: User A has function A and function B in Shanghai. At 19:35:56 on August 30, 2019, function A matches two reserved instances of 1 GB and has consumed three pay-as-you-go instances of 1 GB. Function B matches four reserved instances of 2 GB and has consumed five pay-as-you-go instances of 2 GB. In the second of 19:35:56 on August 30, 2019, user A consumed resources of  $(2 \times 1 \text{ GB} + 3 \times 1 \text{ GB}) + (4 \times 2 \text{ GB} + 5 \times 2 \text{ GB}) \times 1 \text{ s} = 23 \text{ GB} \cdot \text{S}$  in total. If user A has subscribed resources of 20 CUs that are active at 19:35:56 on August 30, 2019, user A only needs to pay for resources of 3 GB·S in this second. If user A has subscribed resources of more than 23 CUs that are active, user A does not need to pay for the resources.

## Procedure for placing a new order

Select the region, number of CUs, and duration for the resource consumption plan. The cost of the selected configuration automatically appears on the right. Click Buy Now.



Function Compute ( prepaid service )

Region	China (Hangzhou)	China (Shanghai)	China (Qingdao)	China (Beijing)	China (Zhangjiakou)	China (Hohhot)
China (Shenzhen)	China (Hong Kong)	Singapore	Australia (Sydney)	Indonesia (Jakarta)	Japan (Tokyo)	
India (Mumbai)	Germany (Frankfurt)	US (Silicon Valley)	US (Virginia)			

Current Selected

Region: China (Hangzhou)  
 Computing Power: 1 CU  
 Purchase Cycle: 1 Month(s)  
 Fee: **\$0.12**  
[Buy Now](#)  
[价格有疑问点这里](#)

Computing Power: 250CU 500CU 1000CU 1 CU

Note: 1CU computing power = 1GB memory and 0.66 core CPU, details: Prepaid Resource Documentation.

Purchase Cycle: 1 month 2 3 4 5 6 7 8 9 1 yr 2 yr 3 yr ☐ Auto Renew

Click **Billing Center**. In the left-side navigation pane, choose **Orders**. On the **Orders** page, check whether the order that you created is in the order list.

- Find the created order and click **Details** in the **Action** column. On the page shown in the following figure, you can view the details of the order, including the payment history, instance ID, configuration details, and lifecycle of the order.

## Upgrade Operation Management

### Upgrade

If your current resource consumption plan is insufficient, you can upgrade a specified instance and only need to pay the price difference.

Note: Function Compute (subscription) does not support down-grading.

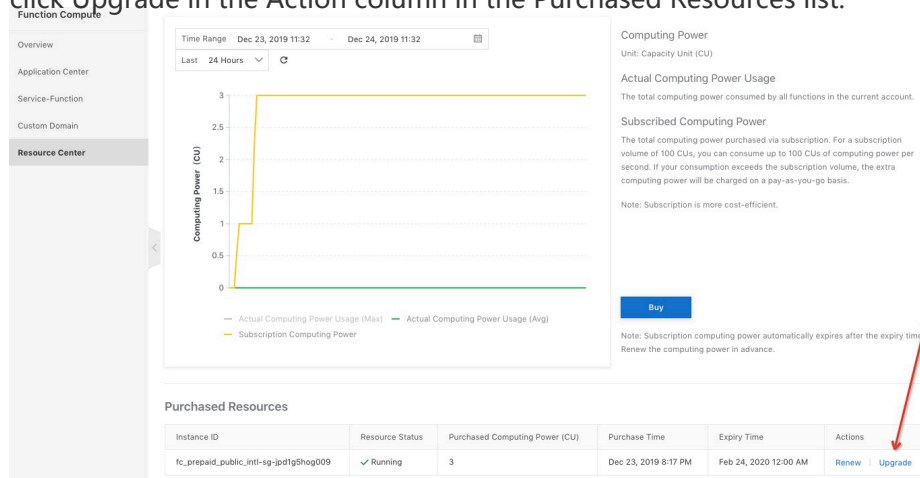
### Calculate the upgrade cost

Price of an upgraded order = Remaining instance duration (accurate to seconds) x (Number of new CUs - Number of old CUs) x Price per second (price of 1 CU\*s)

Example: If you purchase 10 CUs for one month at 15:00 on August 15, 2019, the expiration time will be 00:00:00 on September 16, 2019 and the lifecycle is 31 days and 9 hours. If you upgrade the instance to 15 CUs after using the instance for n hours, you need to pay for the extra 5 CUs in the remaining lifecycle. The amount to be paid is calculated as follows: USD (31 x 24 + 9 - n) x 3600 x (15 - 10) x 12.16/30/24/3600.

### Procedure for upgrade

Log on to the Function Compute console. In the left-side navigation pane, choose Resource Center. On the Resource Center page, locate the row that contains the target instance and click Upgrade in the Action column in the Purchased Resources list.



The Upgrade page shows the current configuration and expiration time of an instance. When changing the configuration, you can only set the number of CUs to a value larger than the current value. Click Function Compute (Subscription) Agreement of Service in the lower-right corner, and then click Pay.

The screenshot shows the 'Upgrade' page. It has a 'Current Config' section with fields for Instance Name, Computing Power (3CU), and Region (Japan (Tokyo)). Below this is the 'Expiration Time' (02/24/2020 00:00:00). The 'Configuration Upgrade' section shows a slider for 'Computing Power' with options 250CU, 500CU, 1000CU, and a dropdown set to 4 CU. A note states: 'Note: 1CU computing power = 1GB memory and 0.66 core CPU, details: Prepaid Resource Documentation.' At the bottom right, there is a 'Coupon' section showing a credit of \$8,413.92, a 'Price' of \$0.00, and a 'Saving' of \$0.25. Below this is a checkbox for 'Function Compute (prepaid service) Agreement of Service' and a 'Pay' button.

Click Billing Center. In the left-side navigation pane, choose Orders. On the Orders page, check whether the order that you created is in the order list. As shown in the following figure, the type of the first order is upgraded, which is the order generated for the upgrade.

Find the upgraded order and click Details in the Action column. On the page shown in the following figure, you can view the details of the order, including the payment history, instance ID, configuration details, and lifecycle of the order.



# Renew Operation Management

Function Compute (subscription) supports both manual renewal and auto renewal. You can configure a specified renewal method as needed.

## Manual renewal

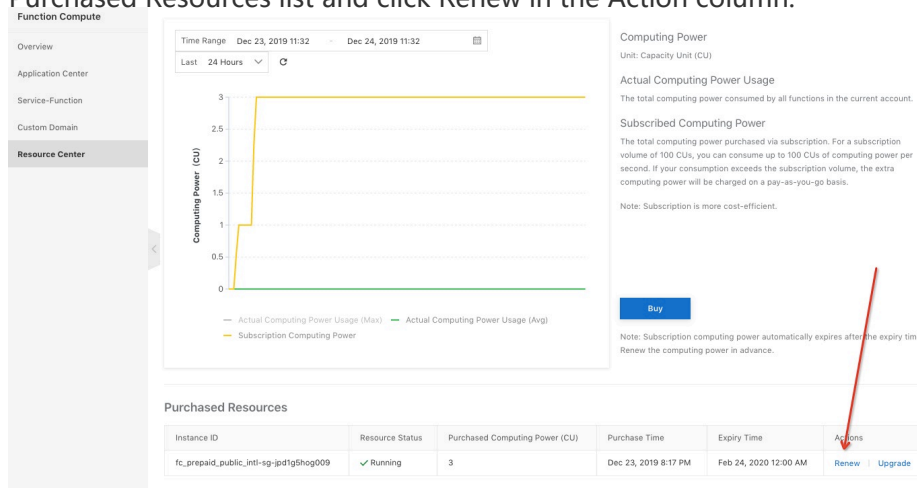
An instance owner manually extends a specified duration for a specified instance.

### Restrictions on manual renewal

- Restriction on instances: You cannot manually renew instances that expired more than seven days previously.
- Restriction on duration: Currently, you can select 1 to 11 months, 1 year, 2 years, or 3 years for a single renewal. If you have special requirements, please contact us.

### Procedure for manual renewal

- Log on to the Function Compute console. In the left-side navigation pane, choose Resource Center. On the Resource Center page, locate the row that contains the target instance in the Purchased Resources list and click Renew in the Action column.



On the Manually Renew page, select a duration. Then, a prompt appears, indicating the expiration time after the renewal. Click OK and select Function Compute (Subscription) Agreement of Service in the lower-right corner. Then, click Pay.

The screenshot shows the 'Renew' console for a Function Compute instance. The 'Current Config' section displays the instance name 'fc\_prepaid\_public\_intl-sg-jpd1g5hog009', computing power '3CU', and region 'Japan (Tokyo)'. The expiration time is '02/24/2020 00:00:00'. Below this, the 'Renewal Duration' section shows a timeline from 1 to 9 months, with options for 1, 2, and 3 years. The expiration time after renewal is '03/24/2020 00:00:00'. On the right, a coupon section shows a credit of '\$8,414.01' and a price of '\$0.00' with a saving of '\$0.36'. A 'Pay' button is visible at the bottom right.

After the payment succeeds, you can click **Billing Center**. In the left-side navigation pane, choose **Orders**. On the **Orders** page, check whether the order that you renewed is in the order list. As shown in the following figure, the type of the first order is renewed, which is the order generated for the renewal.

Find the renewed order and click **Details** in the **Action** column. On the page shown in the following figure, you can view the details of the order, including the payment history, instance ID, and expiration times before and after the renewal.

## Auto renewal

If you enable auto renewal for an instance, the system will renew your instance at the specified time according to certain rules. You can configure auto renewal for your instance by using any of the following methods:

- Configure for a new order: On the new order page, select **Auto Renew** on the right of the duration. When you place an order by month, the period of auto renewal is one month. When you place an order by year, the period of auto renewal is one year.
- Configure in the Renew console: If you do not select **Auto Renewal** for a new order, you can log on to the **Renew** console for configuration. In the **Renew** console, you can perform the following operations:
  - Manually renew an instance.
  - Enable auto renewal for an instance that is manually renewed.
  - Disable auto renewal for an instance.
  - Perform the preceding operations for multiple instances in batches.

## Auto-renewal rules

- On the Chinese site, auto renewal is generally triggered nine days before expiration to one

day before expiration until the renewal succeeds.

- On the international site, auto renewal is generally triggered three days before expiration, one day before expiration, on the expiration day, six days after expiration, and 14 days after expiration.
- On the Japanese site, auto renewal is generally triggered on the expiration day, six days after expiration, and 14 days after expiration.
- Auto renewal starts from 8:00 (UTC+8) every day and is executed for each instance in turn. Therefore, the actual renewal time of an instance may be later than 8:00 and is generally not later than 18:00 of the same day.
- Calculation of the expiration time: If an instance expires at 00:00:00 on December 11, the day before expiration indicates December 10, while six days after expiration indicates December 17.

#### Notes:

1) Auto renewal is not triggered immediately after auto renewal is enabled, but triggered at preceding fixed time points. 2) Auto renewal will not be triggered if auto renewal is enabled after expiration on the Chinese site and 14 days after expiration on the international site. Note that if you enable auto renewal after 8:00 the day before expiration on the Chinese site or after 8:00 14 days after expiration on the international site, auto renewal may fail to take effect. We recommend that you enable auto renewal in advance.