

**FunctionGraph**

# Descripción general del servicio

**Edición** 01  
**Fecha** 2022-11-09



**Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2022. Todos los derechos reservados.**

Quedan terminantemente prohibidas la reproducción y/o la divulgación totales y/o parciales del presente documento de cualquier forma y/o por cualquier medio sin la previa autorización por escrito de Huawei Cloud Computing Technologies Co., Ltd.

## **Marcas registradas y permisos**



El logotipo  y otras marcas registradas de Huawei pertenecen a Huawei Technologies Co., Ltd. Todas las demás marcas registradas y los otros nombres comerciales mencionados en este documento son propiedad de sus respectivos titulares.

## **Aviso**

Es posible que la totalidad o parte de los productos, las funcionalidades y/o los servicios que figuran en el presente documento no se encuentren dentro del alcance de un contrato vigente entre Huawei Cloud y el cliente. Las funcionalidades, los productos y los servicios adquiridos se limitan a los estipulados en el respectivo contrato. A menos que un contrato especifique lo contrario, ninguna de las afirmaciones, informaciones ni recomendaciones contenidas en el presente documento constituye garantía alguna, ni expresa ni implícita.

Huawei está permanentemente preocupada por la calidad de los contenidos de este documento; sin embargo, ninguna declaración, información ni recomendación aquí contenida constituye garantía alguna, ni expresa ni implícita. La información contenida en este documento se encuentra sujeta a cambios sin previo aviso.

---

# Índice

---

|  |           |
|--|-----------|
| <b>1 ¿Qué es FunctionGraph?</b> .....                            | <b>1</b>  |
| <b>2 Características del producto</b> .....                      | <b>4</b>  |
| <b>3 Ventajas del producto</b> .....                             | <b>7</b>  |
| <b>4 Escenarios de aplicación</b> .....                          | <b>9</b>  |
| <b>5 Tipos de funciones</b> .....                                | <b>11</b> |
| 5.1 Funciones de evento.....                                     | 11        |
| 5.2 Funciones HTTP.....  | 12        |
| <b>6 Cuotas y restricciones de uso</b> .....                     | <b>13</b> |
| <b>7 Facturación</b> .....                                       | <b>16</b> |
| <b>8 Gestión de permisos</b> .....                               | <b>19</b> |
| <b>9 Conceptos</b> .....   | <b>22</b> |
| <b>10 Relaciones entre FunctionGraph y otros servicios</b> ..... | <b>24</b> |

# 1 ¿Qué es FunctionGraph?

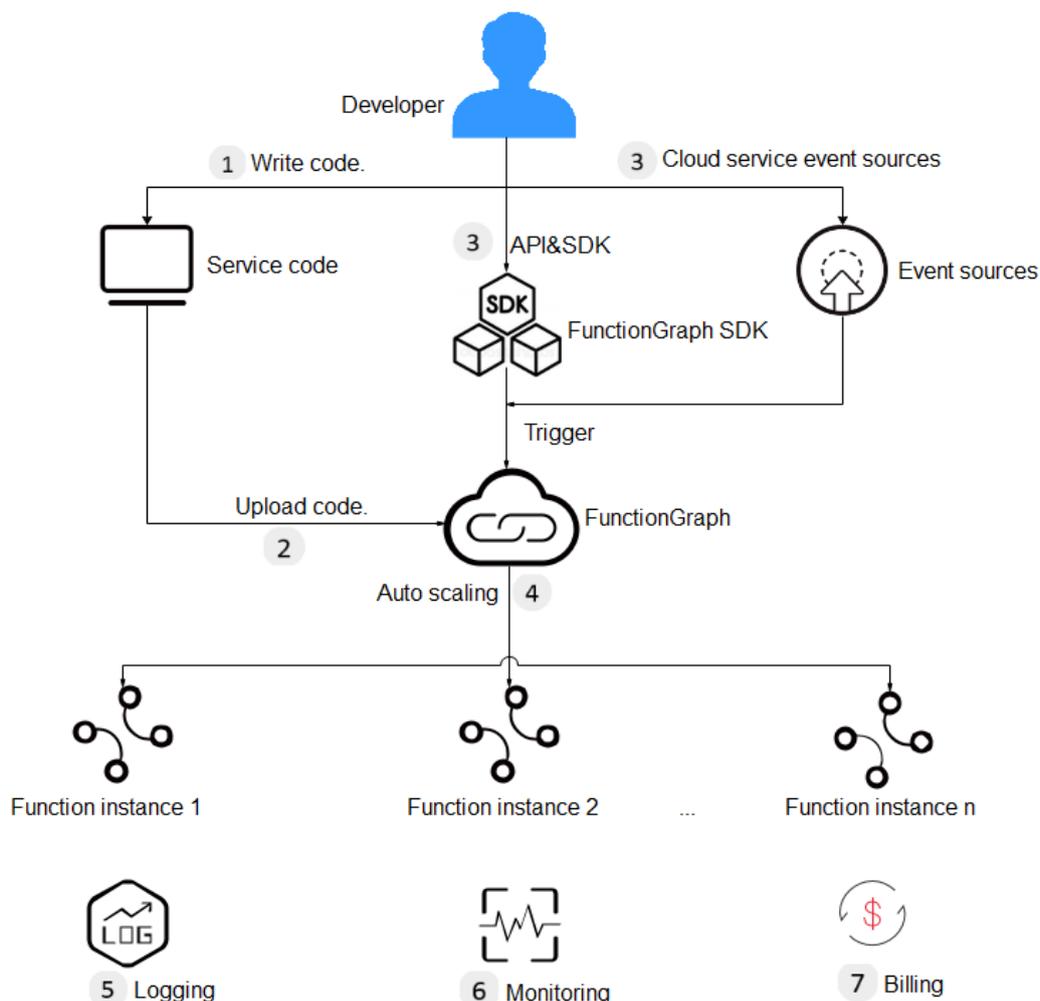
---

## Proceso de uso

FunctionGraph aloja y calcula funciones impulsadas por eventos en un contexto sin servidores, y garantiza altos niveles de disponibilidad, escalabilidad y cero tareas de mantenimiento. Todo lo que debe hacer es escribir su código y configurar las condiciones. Usted paga solo por lo que usa y no se le cobra cuando su código no se está ejecutando.

**Figura 1-1** muestra el proceso de uso de la "FunctionGraph".

Figura 1-1 Proceso de uso



1. Escribir código.

Escribe código en Node.js, Python, Java, Go, C# o PHP. Para obtener más información, consulte la [Guía para desarrolladores de FunctionGraph](#).

2. Cargar código.

Actualmente, puede editar código en línea, cargar un archivo ZIP o JAR, u obtener un archivo ZIP de OBS. Para obtener más detalles, consulte [Tabla 2-2](#).

3. Activar las funciones por llamadas de API o eventos de servicio en la nube.

Llame a RESTful API o use fuentes de eventos de servicios en la nube para activar la ejecución de funciones y generar instancias para implementar funciones de servicio.

4. Se implementa el escalado automático.

FunctionGraph implementa el escalado automático basado en el número de solicitudes recibidas. Para obtener más información, consulte [Cuotas y restricciones de uso](#).

5. Consultar registros.

Consulte los registros de ejecución de las funciones ya que FunctionGraph está interconectado con Log Tank Service (LTS). Para obtener más información, consulte [Consulta de registros de funciones](#).

6. Consultar información de monitoreo.

Consulte la información de monitoreo gráfico ya que FunctionGraph está interconectado con Cloud Eye. Para obtener más información, consulte [Consulta de métricas de funciones](#).

7. Modo de facturación

Después de ejecutar una función, se le facturará en función del número de solicitudes de ejecución de función y la duración de la ejecución. (v1: facturado por cada 100 ms; v2: facturado por cada 1 ms)

# 2 Características del producto

## Gestión de funciones

FunctionGraph ofrece gestión de funciones basada en consola.

- Se admiten Node.js, Java, Python, Go, C# y los tiempos de ejecución personalizados. [Tabla 2-1](#) proporciona los detalles.

### NOTA

Se recomienda utilizar la última versión de tiempo de ejecución.

**Tabla 2-1** Tiempos de ejecución

| Tiempo de ejecución | Versión admitida                             |
|---------------------|--|
| Node.js             | 6.10, 8.10, 10.16, 12.13 y 14.18             |
| Python              | 2.7, 3.6 y 3.9                               |
| Java                | 8.0 y 11                                     |
| Acceder             | 1.8 y 1.x                                    |
| C#                  | .NET Core 2.0, .NET Core 2.1 y .NET Core 3.1 |
| PHP                 | 7.3  |
| Personalizado       | -  |

- Múltiples modos de entrada de código  
FunctionGraph le permite editar código en línea, cargar un archivo ZIP desde el servicio de almacenamiento de objetos (OBS) o cargar directamente un archivo ZIP o JAR. [Tabla 2-2](#) enumera los modos de entrada de código admitidos para cada tiempo de ejecución.

**Tabla 2-2** Modos de entrada de código

| Tiempo de ejecución | Edición de código en línea | Carga de un archivo ZIP | Carga de un archivo JAR | Carga de un archivo ZIP desde OBS |
|---------------------|----------------------------|-------------------------|-------------------------|-----------------------------------|
| Node.js             | Se admite                  | Se admite               | No se admite            | Se admite                         |
| Python              | Se admite                  | Se admite               | No se admite            | Se admite                         |
| Java                | No se admite               | Se admite               | Se admite               | Se admite                         |
| Acceder             | No se admite               | Se admite               | No se admite            | Se admite                         |
| C#                  | No se admite               | Se admite               | No se admite            | Se admite                         |
| PHP                 | Se admite                  | Se admite               | No se admite            | Se admite                         |
| Personalizado       | Se admite                  | Se admite               | No se admite            | Se admite                         |

## Activador

FunctionGraph admite varios tipos de activadores, como notificaciones de mensajes simples (SMN), API Gateway (APIG) y tipos de activadores OBS. [Tabla 2-3](#) enumera los tipos de activador admitidos y el modo de invocación de función de cada tipo de activador.

**Tabla 2-3** Modos de invocación de función

| Activador  | Modo de invocación de función |
|--|-------------------------------|
| Activador SMN                                      | Invocación asincrónica        |
| Distributed Message Service (DMS)                  | Invocación asincrónica        |
| Activador APIG                                     | Invocación síncrona           |
| Activador OBS                                      | Invocación asincrónica        |
| Activador del servicio de ingestión de datos (DIS) | Invocación asincrónica        |
| Activador de temporizador                          | Invocación asincrónica        |
| Activador Kafka                                    | Invocación asincrónica        |

## Registros y Métricas

FunctionGraph muestra gráficamente las métricas de monitoreo de funciones y recopila registros de ejecución de funciones, lo que le permite ver los estados de funciones y localizar problemas consultando registros.

Para obtener más información sobre cómo consultar registros, consulte [Gestión de registros de funciones](#).

Para obtener más información sobre cómo ver las métricas de función, consulte [Consulta de métricas de función](#).

Para obtener detalles sobre la información de monitoreo a nivel de inquilino, consulte [Introducción al panel](#).

## Inicialización de función

La interfaz del inicializador se introduce en:

- Aísle la inicialización de funciones y el procesamiento de solicitudes para permitir una lógica de programa más clara y un código mejor estructurado y de mayor rendimiento.
- Garantice una actualización fluida de la función para evitar la pérdida de rendimiento durante la inicialización del arranque en frío de la capa de aplicación. Habilite las nuevas instancias de función para ejecutar automáticamente la lógica de inicialización antes de procesar las solicitudes.
- Identifique la sobrecarga de inicialización de la capa de aplicación y determine con precisión el tiempo de escalado de recursos y la cantidad de recursos necesarios. Esta característica hace que la latencia de solicitud sea más estable cuando la carga de la aplicación aumenta y se requieren más instancias de función.

## Funciones HTTP

Puede establecer **Function Type** en **HTTP Function** en la página de creación de funciones. Se diseñan las funciones HTTP para optimizar los servicios web. Puede enviar solicitudes HTTP a direcciones URL para activar la ejecución de la función. Las funciones HTTP solo admiten activadores APIG y API Connect (APIC).

### NOTA

Esta función solo es compatible con FunctionGraph v2.

## Rastreo

Puede habilitar el seguimiento de funciones. A continuación, puede ir a la consola de Application Performance Management (APM) para ver la información de seguimiento y JVM. Actualmente, solo se pueden rastrear funciones Java.

## Imágenes personalizadas

Puede empaquetar y cargar directamente imágenes de contenedores. Las imágenes son cargadas e iniciadas por la plataforma y se pueden llamar de manera similar a las funciones HTTP. A diferencia del modo de carga de código anterior, puede usar un paquete de código personalizado, que es flexible y reduce los costos de migración.

### NOTA

Esta función solo es compatible con FunctionGraph v2.

# 3 Ventajas del producto

---

## No hay servidores para gestionar

FunctionGraph ejecuta automáticamente su código y le libera del aprovisionamiento y administración de servidores, lo que le permite centrarse en la innovación empresarial.

## Escalado automático

FunctionGraph escala automáticamente para adaptarse a las fluctuaciones en la demanda de recursos, lo que asegura que los servicios sigan estando disponibles incluso durante los períodos de mayor actividad.

Escala automáticamente los recursos de entrada/salida según el número de solicitudes de servicio y distribuye las solicitudes a las instancias de función a través del equilibrio de carga automático.

## Activación basada en eventos

FunctionGraph se integra con múltiples servicios en la nube como SMN y OBS en un mecanismo de activación basado en eventos para cumplir con los requisitos de servicio.

Está interconectado con los servicios LTS y Cloud Eye, lo que le permite ver registros de funciones y métricas sin necesidad de ninguna configuración.

## Alta disponibilidad

Si una instancia presenta fallas, FunctionGraph iniciará otra instancia para procesar nuevas solicitudes y liberará los recursos de la instancia que no funciona correctamente.

## Pago por uso

Se le facturará según el número de solicitudes de función y la duración de ejecución, no se le cobrará cuando su código se esté dejado.

## Facturación de instancias reservadas

Las instancias reservadas se pueden crear para inicializar funciones y eliminar la influencia del inicio en frío en sus servicios. Las instancias reservadas siempre están activas en el entorno de ejecución.

Para el uso de instancias reservadas, se le facturará en función del número de solicitudes y la duración de ejecución de las instancias reservadas. La duración mínima de carrera es de 60s.

# 4 Escenarios de aplicación

FunctionGraph es adecuado para varios escenarios, como procesamiento de archivos en tiempo real, procesamiento de flujo de datos en tiempo real, backends de aplicaciones web y móviles y aplicaciones de IA.

## Situación 1: Aplicaciones controladas por eventos

Los servicios se ejecutan en modo controlado por eventos y los recursos se aprovisionan en función de las demandas. Los desarrolladores no necesitan preocuparse por los picos o valles de servicio. Los recursos inactivos no se facturan, lo que reduce los costos de operación y mantenimiento. Las aplicaciones controladas por eventos incluyen procesamiento de archivos, procesamiento de imágenes, transmisión/transcodificación en directo, procesamiento de flujo de datos en tiempo real y procesamiento de reglas/eventos de IoT.

- **Procesamiento de archivos en tiempo real**

Cuando se cargan archivos de un cliente a OBS, se pueden activar funciones para crear miniaturas de imágenes en tiempo real, convertir formatos de vídeo, agregar y filtrar archivos de datos o implementar otras operaciones de archivos.

Ventajas:

- FunctionGraph asigna automáticamente recursos para ejecutar más instancias de función a medida que aumenta el número de solicitudes recibidas.
- Los archivos se cargan al OBS para activar las funciones de procesamiento de archivos. FunctionGraph asigna automáticamente recursos para ejecutar más instancias de función a medida que aumenta el número de solicitudes recibidas.
- Se le facturará solo por los recursos utilizados para procesar archivos según sea necesario. (no se le facturan los recursos inactivos durante los mínimos de la demanda).

Por ejemplo, cree dos depósitos en OBS y cree una función para comprimir imágenes cargadas en el depósito especificado. Para obtener más información, consulte

[Compresión de imágenes](#).

- **Procesamiento de flujo de datos en tiempo real**

FunctionGraph trabaja con DIS para procesar flujos de datos en tiempo real. FunctionGraph admite el seguimiento de la actividad de la aplicación, el procesamiento secuencial de transacciones, el análisis de flujo de datos, la clasificación de datos, la generación de métricas, el filtrado de registros, la indexación, el análisis de redes sociales y la telemetría y medición de datos de dispositivos IoT.

Ventajas:

- Los datos se recopilan mediante estos flujos de DIS para activar las funciones de procesamiento de datos.
- FunctionGraph asigna automáticamente recursos para ejecutar más instancias de función a medida que aumenta el número de solicitudes recibidas.
- Se le facturará solo por los recursos utilizados para procesar archivos según sea necesario. (no se le facturan los recursos inactivos durante los mínimos de la demanda).

## Situación 2: Aplicaciones Web

Interconecte FunctionGraph con otros servicios en la nube o sus máquinas virtuales para crear rápidamente backends web y móviles escalables y de alta disponibilidad. Las aplicaciones web incluyen mini programas, páginas web/aplicaciones, chatbots y Backends para Frontends (BFF).

Ventajas:

- FunctionGraph garantiza la alta confiabilidad de los datos de los sitios web utilizando los servicios OBS y CloudTable, y la alta disponibilidad de la lógica de dichos sitios mediante el API Gateway.
- FunctionGraph asigna automáticamente recursos para ejecutar más instancias de función a medida que aumenta el número de solicitudes recibidas.
- Se le facturará solo por los recursos utilizados para procesar archivos según sea necesario. (no se le facturan los recursos inactivos durante los mínimos de la demanda).

## Situación 3: Aplicaciones de IA

La evolución de la inteligencia requiere que se integren varios servicios para un despliegue rápido. Las aplicaciones de IA incluyen integración de servicios de terceros, inferencia de IA, reconocimiento facial y reconocimiento de matrículas.

Ventajas:

- FunctionGraph trabaja con los servicios de la inteligencia empresarial (EI) para el reconocimiento de texto y la moderación de contenido para adaptarse a una amplia gama de escenarios. - realiza ajustes cuando lo necesite a medida que cambian las demandas.
- Solo necesita solicitar servicios relacionados y escribir código de servicio sin tener que aprovisionar o administrar servidores.
- Se le facturará solo por la ejecución de funciones y los servicios EI usados sin tener que pagar por recursos inactivos cuando las demandas de servicio son bajas.

# 5 Tipos de funciones

---

## 5.1 Funciones de evento

### NOTA

Debe elegir entre **Event Function** y **HTTP Function** al crear una función en la FunctionGraph v2.

### Resumen

FunctionGraph admite funciones de eventos. Un evento puede desencadenar la ejecución de funciones. Generalmente, está en formato JSON. Puede crear un evento para activar su función a través de la plataforma de servicios en la nube o CloudIDE. Todos los tipos de desencadenadores compatibles con FunctionGraph pueden desencadenar funciones de eventos.

### NOTA

1. En la página de creación de funciones, **Function Type** se establece como **Event Function** de forma predeterminada.
2. Durante la prueba, una función se puede activar simplemente introduciendo el evento especificado en formato JSON.
3. También puede usar disparadores para desencadenar funciones de eventos.

### Ventajas

- Fácil programación de nodo único  
Puedes editar funciones de eventos en CloudIDE o FunctionGraph o subir paquetes de código allí e implementarlos con unos pocos clics. No hay necesidad de que usted se preocupe por la concurrencia de funciones o la rectificación de fallas.
- Tiempos de ejecución de alta velocidad y alto rendimiento  
Las funciones de evento se pueden iniciar, escalar y llamar en milisegundos. Las fallas se pueden detectar y rectificar en cuestión de segundos.
- Cadena de herramientas completa  
FunctionGraph ofrece un registro, seguimiento, depuración y monitoreo completos, lo que permite a los desarrolladores implementar funciones en solo tres pasos.

## Restricciones

Las funciones de evento enfrentan restricciones de origen de eventos. Debe cumplir con las reglas de desarrollo de funciones de la plataforma de funciones.

## 5.2 Funciones HTTP

### NOTA

Esta función solo es compatible con FunctionGraph v2.

## Resumen

FunctionGraph admite funciones de eventos y funciones HTTP. Se diseñan las funciones HTTP para optimizar los servicios web. Puede enviar solicitudes HTTP a direcciones URL para activar la ejecución de la función. Las funciones HTTP solo admiten activadores APIG y APIC.

### NOTA

1. Las funciones HTTP admiten el protocolo HTTP/1.1.
2. En la página de creación de funciones, se agrega nuevamente **HTTP Function**.
3. La función HTTP debe establecerse en **bootstrap**. Puede escribir directamente el comando startup y **allow access over port 8000**.

## Ventajas

- Soporte para múltiples marcos  
Puede usar marcos web comunes, como Node.js Express y Koa, para escribir funciones web y migrar sus servicios de marco web local a la nube con menos modificaciones.
- Menos pasos de procesamiento de solicitudes  
Las funciones pueden recibir y procesar directamente solicitudes HTTP, eliminando la necesidad de API Gateway para convertir el formato JSON. Esto acelera el procesamiento de solicitudes y mejora el rendimiento del servicio web.
- Experiencia de escritura de primera calidad.  
La escritura de funciones HTTP es similar a la de los servicios web nativos. También puede usar las API nativas de Node.js para disfrutar de una experiencia de desarrollo local.

## Restricciones

- Las funciones HTTP solo admiten activadores APIG (compartido), APIG (dedicado) y APIC.
- Se pueden vincular varios activadores de API a la misma función, pero todas las API deben pertenecer al mismo servicio APIG.
- Para las funciones HTTP, el tamaño del cuerpo de respuesta HTTP no puede exceder los 6 MB.

# 6 Cuotas y restricciones de uso

## Restricciones de recursos de cuenta

En la siguiente tabla se muestran las cuotas de los recursos de la cuenta. Para obtener más información sobre cómo consultar y modificar las cuotas, consulte [Cuotas](#).

**Tabla 6-1** Restricciones de recursos de cuenta

| Recurso   | Restricción  |
|---|--|
| Número máximo de funciones que se pueden crear en una cuenta  | 400  |
| Número máximo de versiones permitidas para una función  | 10   |
| Número máximo de alias permitidos para una función  | 10   |
| Número máximo de activadores de DIS, DMS, Kafka y Timer permitidos para una versión de función  | 10   |
| Tamaño de un paquete de implementación de código (en formato ZIP o JAR) que se puede mostrar en la consola FunctionGraph                        | 3 MB   |
| Tamaño de un paquete de implementación de código (en formato ZIP o JAR) que se puede cargar en la consola FunctionGraph                         | 10 MB  |
| Tamaño de un paquete de implementación de código (en formato ZIP o JAR) que se puede editar en línea durante la invocación de la API de función | 50 MB  |
| Tamaño de un paquete de implementación de código original permitido durante la invocación de la API de función                                  | <ul style="list-style-type: none"><li>● Paquete ZIP que se carga directamente: 1500 MB</li><li>● Carga de paquetes a través de OBS: 300 MB</li></ul> |

| Recurso  | Restricción   |
|--|---|
| Tamaño máximo de los paquetes de implementación permitidos para una cuenta | 10 GB   |
| Número de ejecuciones simultáneas por cuenta                               | 100   |
| Número máximo de instancias reservadas que una cuenta puede crear          | 90 (Número de ejecuciones simultáneas por cuenta x 90%) |
| Tamaño de todas las variables de entorno de una función                    | 2 KB  |

## Restricciones de recursos en ejecución de función

**Tabla 6-2** Restricciones de recursos en ejecución de función

| Recurso  | Restricción            |
|--|------------------------|
| Espacio efímero en disco (espacio de <code>/tmp</code> )                                     | 512 MB                 |
| Número de descriptores de fichero  | 1024                   |
| Número total de procesos y subprocesos   | 1024                   |
| Duración máxima de ejecución por solicitud   | 900s                   |
| Tamaño válido de la carga útil del cuerpo de solicitud de invocación (invocación sincrónica) | 6 MB                   |
| Tamaño válido de la carga útil del cuerpo de respuesta de invocación (invocación sincrónica) | 6 MB                   |
| Tamaño válido de la carga útil del cuerpo de solicitud de invocación (invocación asíncrona)  | 256 KB                 |
| Tamaño de los recursos importados  | ≤ Archivo ZIP de 50 MB |
| Tamaño de los recursos exportados  | ≤ de 50 MB             |

 **NOTA**

- Tamaño válido de la carga útil del cuerpo de respuesta de invocación (invocación sincrónica): La cadena de caracteres devuelta o la cadena de caracteres JSON del cuerpo de respuesta serializada es menor o igual a 6 MB de forma predeterminada. El tamaño real de los datos varía en función de la configuración de back-end de FunctionGraph. El back-end determina el tamaño de los datos serializados con una desviación de nivel de byte. El tamaño real de la carga útil válida es de 6 MB  $\pm$  100 bytes.
- No se recomienda invocar una función cuyo tiempo de ejecución supere 90 segundos en la consola FunctionGraph. Para invocar tal función, utilice la invocación asincrónica.
- El tamaño de carga útil válido de un cuerpo de solicitud es de 6 MB cuando se usa un disparador Kafka, DDS, DIS o GaussDB (for Mongo) y es de 4 MB cuando se usa un disparador APIG.

# 7 Facturación

FunctionGraph utiliza la facturación de pago por uso, y el precio se calcula en base al número de solicitudes y la duración de la ejecución. No se requieren gastos mínimos.

**Precio total = Precio de las solicitudes + Precio de la duración de la ejecución**

## Facturación de instancias comunes

### Precio de las solicitudes

- Se le facturará en función del número de solicitudes, que es el número total de veces que se invocan todas las funciones.
- El primer millón de solicitudes al mes son gratuitas. Después de agotar el millón de solicitudes gratuitas, se le facturará según las tarifas especificadas en [Detalles de precios de FunctionGraph](#).

### Precio por duración de ejecución

FunctionGraph proporciona las instancias reservadas y las a pedido, que se miden de manera diferente.

- **Reserved instances:** Consulte [Facturación de instancias reservadas](#).
- **On-demand instances:** Este tipo de instancia es creada y liberada por FunctionGraph y se le facturará en base a la duración real de la ejecución. El tiempo de ejecución es desde el momento en que el código de función comienza a ejecutarse hasta el momento en que devuelve una respuesta o termina.

### Gastos adicionales:

Si FunctionGraph utiliza otros servicios en la nube, se generarán gastos adicionales. Para obtener más información, consulte [Detalles de precios de FunctionGraph](#).

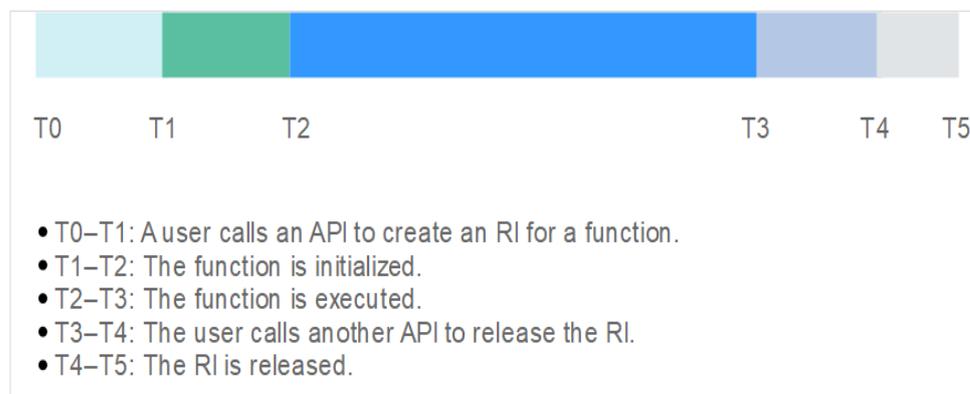
## Facturación de instancias reservadas

Puede crear y liberar instancias reservadas y se facturan en función de la duración de su ejecución. Las instancias reservadas permanecen activas en el entorno de ejecución, eliminando la influencia de los arranques en frío en la latencia.

- Si llama a una API para crear una instancia reservada, la facturación comenzará tan pronto como se complete la creación de la instancia reservada.

- Si llama a otra API para liberar una instancia reservada, las nuevas solicitudes no se encaminarán a la instancia reservada. La instancia reservada se liberará dentro de un período determinado y la facturación se detendrá.

**Figura 7-1** Ciclo de vida de una instancia reservada



Como se muestra en **Figura 7-1**, la facturación dura de T1 a T4.

Las instancias reservadas se miden con una granularidad de segundo. Si una instancia reservada se ejecuta durante cualquier fracción de minuto, se le facturará el minuto completo. De lo contrario, se le facturará en función de la duración real de la ejecución.

Por ejemplo, si una instancia reservada se ejecuta durante 51s, se le facturará durante 1 minuto. Si la instancia reservada se ejecuta por 61s, se le facturará por 61s.

- La unidad de duración de ejecución es de GB-segundo, lo que significa 1 GB por segundo.

## Nivel gratuito

FunctionGraph ofrece un nivel gratuito, que puede compartir con sus usuarios de IAM.

- **Solicitudes:** 1 millón de solicitudes gratuitas cada mes.
- **Duración de la ejecución:** 400,000 GB por segundos de ejecución gratuita cada mes. Por ejemplo, si la memoria para una función es de 1 GB, la duración de ejecución libre cada mes es de 400,000s. Si la memoria es de 512 MB, la duración de ejecución libre es de 800,000s.

### AVISO

El nivel gratuito se restablece al principio de cada mes calendario.

**Tabla 7-1** muestra la duración de ejecución libre para diferentes especificaciones de memoria.

**Tabla 7-1** Duración de ejecución gratuita

| Memoria (MB) | Duración (s) de ejecución gratuita por Mes |
|--------------|--|
| 128          | 3,200,000                                  |
| 256          | 1,600,000                                  |
| 512          | 800,000                                    |
| 768          | 533,333                                    |
| 1024         | 400,000                                    |
| 1280         | 320,000                                    |
| 1536         | 266,667                                    |
| N            | 1024 x 400,000/N                           |

## Renovación

Para renovar la suscripción, elija Renewals en la consola de administración. Para obtener más información, consulte Gestión de renovación.

## Vencimiento y pago atrasado

Si su cuenta está en mora, podrá visualizar los detalles de los pagos en mora en el Centro de facturación. Para evitar que los recursos relacionados se detengan o liberen, recargue su cuenta lo antes posible. Para obtener más información, consulte Realización de reembolsos (clientes directos de pospago).

# 8 Gestión de permisos

---

Si necesita asignar diferentes permisos a los empleados de su empresa para acceder a sus recursos de FunctionGraph, IAM es una buena opción para la gestión de permisos detallada. IAM proporciona autenticación de identidad, gestión de permisos y control de acceso, ellos le ayudan a asegurar el acceso a sus recursos en la nube.

Con IAM, puede usar su cuenta a crear usuarios de IAM para sus empleados y asignar permisos a los usuarios para controlar su acceso a tipos de recursos específicos. Por ejemplo, algunos desarrolladores de software de su empresa necesitan usar recursos de FunctionGraph pero no deben eliminarlos ni realizar operaciones de alto riesgo. Para lograr este resultado, puede crear usuarios de IAM para los desarrolladores de software y concederles solo los permisos necesarios para usar los recursos de FunctionGraph.

Si su cuenta no necesita usuarios individuales de IAM para la gestión de permisos, puede omitir este capítulo.

IAM se puede utilizar de forma gratuita. Solo paga por los recursos de tu cuenta. Para obtener más información acerca de IAM, consulte Descripción general del servicio IAM.

## Permisos de FunctionGraph

De forma predeterminada, los nuevos usuarios de IAM no tienen ningún permiso asignado. Debe agregar un usuario a uno o más grupos y asignar políticas de permisos a estos grupos. A continuación, el usuario hereda los permisos de los grupos a los que pertenece. Este proceso se llama autorización. Después de la autorización, el usuario puede realizar operaciones especificadas en FunctionGraph en función de los permisos.

FunctionGraph es un servicio a nivel de proyecto implementado y accedido en regiones físicas específicas. Para asignar permisos de FunctionGraph a un grupo de usuarios, especifique el ámbito como proyectos específicos de la región y seleccione proyectos para que los permisos surtan efecto. Si se selecciona **All projects**, los permisos surtirán efecto para el grupo de usuarios en todos los proyectos específicos de la región. Al acceder a FunctionGraph los usuarios tienen que cambiar a una región en la que se les haya autorizado a utilizar el servicio FunctionGraph.

Puede conceder permisos a los usuarios mediante roles y directivas.

- **Roles:** Tipo de mecanismo de autorización de grano grueso que define permisos relacionados con las responsabilidades del usuario. Este mecanismo proporciona solo un número limitado de roles de nivel de servicio para la autorización. Al utilizar roles para conceder permisos, es posible que también deba asignar otros roles de los que dependen

los permisos. Sin embargo, los roles no son una opción ideal para la autorización detallada y el control de acceso seguro.

- Políticas: Tipo de mecanismo de autorización detallado que define los permisos necesarios para realizar operaciones en recursos de nube específicos bajo ciertas condiciones. Este mecanismo permite una autorización más flexible basada en políticas, cumpliendo los requisitos para un control de acceso seguro.

**Tabla 8-1** enumera todas las políticas de sistema admitidas por FunctionGraph.

**Tabla 8-1** Descripción de permisos

| Nombre de rol/<br>política     | Descripción   | Categoría                        | Dependencia     |
|--------------------------------|---|----------------------------------|-----------------|
| Administrador de FunctionGraph | Este rol tiene los permisos para gestionar funciones, flujos y activadores e invocar funciones. | Rol definido por el sistema      | Invitado tenant |
| Invocador de FunctionGraph     | Este rol tiene los permisos para consultar funciones, flujos, activadores e invocar funciones.  | Rol definido por el sistema      | No disponible   |
| FunctionGraph FullAccess       | Esta política otorga todos los permisos para FunctionGraph.                                     | Política definida por el sistema | No disponible   |
| FunctionGraph ReadOnlyAccess   | Esta política otorga permisos de solo lectura para FunctionGraph.                               | Política definida por el sistema | No disponible   |
| FunctionGraph CommonOperations | Esta política otorga permisos para consultar funciones y activadores e invocar funciones.       | Política definida por el sistema | No disponible   |

**Tabla 8-2** enumera las operaciones comunes soportadas por cada política definida por el sistema de FunctionGraph. Elija las políticas adecuadas definidas por el sistema de acuerdo con esta tabla.

**Tabla 8-2** Operaciones comunes apoyadas por cada política definida por el sistema

| Operación             | Invocador de FunctionGraph | Administrador de FunctionGraph | FunctionGraph ReadOnly Access | FunctionGraph CommonOperations | FunctionGraph FullAccess |
|-----------------------|----------------------------|--------------------------------|-------------------------------|--------------------------------|--------------------------|
| Creación de funciones | ×                          | √                              | ×                             | ×                              | √                        |

| Operación                          | Invocador de FunctionGraph | Administrador de FunctionGraph | FunctionGraph ReadOnly Access | FunctionGraph CommonOperations | FunctionGraph FullAccess |
|------------------------------------|----------------------------|--------------------------------|-------------------------------|--------------------------------|--------------------------|
| Consulta de funciones              | ✓                          | ✓                              | ✓                             | ✓                              | ✓                        |
| Modificación de funciones          | ×                          | ✓                              | ×                             | ×                              | ✓                        |
| Eliminación de funciones           | ×                          | ✓                              | ×                             | ×                              | ✓                        |
| Invocación de funciones            | ✓                          | ✓                              | ×                             | ✓                              | ✓                        |
| Consulta de registros de funciones | ✓                          | ✓                              | ✓                             | ✓                              | ✓                        |
| Consulta de métricas de función    | ✓                          | ✓                              | ✓                             | ✓                              | ✓                        |

### Enlaces útiles

- [Descripción general del servicio IAM](#)
- [Creación de un usuario y concesión de permisos](#)
- [Políticas de permisos y acciones admitidas](#)

# 9 Conceptos

---

## Función

Las funciones son códigos definidos para manejar eventos.

## Origen del evento

Un origen de eventos es un servicio de nube pública o una aplicación personalizada que publica eventos.

## Invocación síncrona

Los clientes esperan respuestas explícitas a sus solicitudes desde una función. Las respuestas se devuelven solo después de invocar la función.

## Invocación asincrónica

Los clientes no se preocupan por los resultados de invocación de funciones de sus solicitudes. Después de recibir una solicitud, el FunctionGraph la pone en una cola, devuelve una respuesta y procesa otras solicitudes cuando hay recursos inactivos.

## Activador

Un activador es un evento que desencadena la ejecución de una función.

## Multi-Concurrencia de una sola instancia

Número de solicitudes que una instancia puede procesar simultáneamente.

## Imágenes personalizadas

Puede empaquetar y cargar directamente imágenes de contenedores. A continuación, la plataforma carga e inicia estas imágenes para crear funciones.

## Ejecución de funciones personalizadas

Puede personalizar scripts y archivos para ejecutar funciones.

## Registros de funciones

Registros generados durante la invocación de la función.

## Monitoreo de funciones

Monitorización de la información generada durante la ejecución de la función.

## Versión de la función

FunctionGraph le permite publicar una o más versiones a lo largo de los procesos de desarrollo, pruebas y producción para administrar su código de función. El código y las variables de entorno de cada versión se guardan como una instantánea. Una vez publicado el código de función, modifique la configuración cuando sea necesario.

## Alias de función

Puede crear un alias para una versión de función específica. Para volver a una versión anterior, utilice el alias correspondiente para representar la versión en lugar de modificar el código de función.

Cada alias de función se puede vincular a una versión principal y a una versión adicional para el desplazamiento del tráfico.

## Paquete de dependencia

FunctionGraph le permite administrar dependencias de manera unificada. Puede cargar dependencias desde una ruta local o a través de OBS si son demasiado grandes, y especificar nombres para ellas.

Para obtener más información sobre cómo generar dependencias de función, consulte [¿Cómo creo las dependencias de función?](#)

## Archivo de Bootstrap

El archivo **bootstrap** es el archivo de inicio de una función HTTP. La función HTTP solo puede leer **bootstrap** como el nombre del archivo de inicio. Si el nombre del archivo no es **bootstrap**, el servicio no se puede iniciar.

# 10 Relaciones entre FunctionGraph y otros servicios

**Tabla 10-1** describe los servicios en la nube que se interconectan con FunctionGraph.

**Tabla 10-1** Servicios interconectados

| Servicio    | Función   |
|-------------|---|
| SMN         | Las funciones de FunctionGraph se construyen para procesar notificaciones de SMN. Para obtener más información, consulte <a href="#">Guía del usuario de SMN</a> .  |
| DMS         | Las funciones de FunctionGraph están configuradas para sondear automáticamente las colas de DMS en busca de mensajes y procesar cualquier mensaje nuevo. Para obtener más información, consulte <a href="#">Guía del usuario de DMS</a> .   |
| Gateway API | Las funciones de FunctionGraph se invocan a través de HTTPS definiendo REST API con servicios de back-end especificados. Para obtener más información, consulte <a href="#">Guía del usuario de API Gateway</a> .   |
| OBS         | Las funciones FunctionGraph se crean para procesar eventos de depósito OBS, como eventos de creación o eliminación de objetos. Por ejemplo, cuando se carga una imagen en el depósito especificado, OBS invoca la función para leer la imagen y crear una miniatura. Para obtener más información, consulte <a href="#">Guía del usuario de OBS</a> .                 |
| DIS         | Las funciones de FunctionGraph se crean para sondear periódicamente transmisiones de DIS en busca de nuevos registros, como transmisiones de clics en sitios web, transacciones financieras, transmisiones de redes sociales, registros de TI y eventos de seguimiento de ubicación. Para obtener más información, consulte <a href="#">Guía del usuario de DIS</a> . |
| Cloud Eye   | FunctionGraph está interconectado con Cloud Eye para informar métricas de monitoreo, lo que le permite ver métricas de funciones y mensajes de alarma a través de Cloud Eye.  |

| Servicio | Función  |
|----------|--|
| VPC      | Las funciones se pueden configurar para acceder a recursos en las Nubes Privadas Virtuales (VPC) o para acceder a Internet a través de la traducción de direcciones de red de origen (SNAT) vinculando direcciones de IP elásticas. Para obtener más información, consulte <a href="#">Guía del usuario de VPC</a> . |

**Tabla 10-2** Operaciones de FunctionGraph que pueden ser grabadas por CTS

| Operación                                  | Tipo de recurso       | Nombre del seguimiento |
|--|-----------------------|------------------------|
| Creación de una función                    | Funciones             | CreateFunction         |
| Eliminación de una función                 | Funciones             | DeleteFunction         |
| Modificación de información de función     | Funciones             | ModifyFunctionMetadata |
| Publicación de una versión de función      | FunctionVersions      | PublishFunctionVersion |
| Eliminación de un alias de función         | FunctionVersionsAlias | DeleteVersionAlias     |
| Eliminación de un activador de función     | Activador             | DeleteTrigger          |
| Creación de un activador de función        | Activador             | CreateTrigger          |
| Deshabilitación de un activador de función | Activador             | DisabledTrigger        |
| Habilitación de un activador de función    | Activador             | enabledTrigger         |
| Ejecución de una función                   | FunctionInstance      | invokeFunction         |