

# 04 Preguntas frecuentes

**Edición** 01  
**Fecha** 2025-01-21




**Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. 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 Preguntas frecuentes generales.....</b>	<b>1</b>
1.1 ¿Qué es FunctionGraph?.....	1
1.2 ¿Debo solicitar cualquier servicio de cómputo, de almacenamiento o de red al usar FunctionGraph?.....	1
1.3 ¿Necesito desplegar mi código después de programar?.....	1
1.4 ¿Cómo obtengo un token?.....	2
1.5 ¿Qué tiempos de ejecución admite FunctionGraph?.....	3
1.6 ¿Cuánto espacio en disco se asigna a cada función de FunctionGraph?.....	3
1.7 ¿FunctionGraph admite versiones de funciones?.....	3
1.8 ¿Cómo una función lee o escribe archivos?.....	3
1.9 ¿Cómo configuro un proxy cuando uso CLI?.....	4
1.10 ¿FunctionGraph admite la extensión de función?.....	4
1.11 ¿Qué permisos se requieren para que un usuario de IAM use FunctionGraph?.....	4
1.12 ¿Cómo puedo crear un paquete de dependencia de Python basado en una unidad de ODBC para la consulta de base de datos?.....	5
1.13 ¿Cuál es la cuota de FunctionGraph?.....	5
1.14 ¿Qué fuentes chinas admite FunctionGraph?.....	5
1.15 ¿Cómo resuelve FunctionGraph un nombre de dominio de DNS privado?.....	6
1.16 ¿Cómo resuelve una función basada en imagen de contenedor un nombre de dominio de DNS privado?.....	7
1.17 ¿Cómo uso un nombre de dominio para acceder a una API registrada con API Gateway (Dedicado)?.....	8
1.18 ¿Cuáles son los escenarios comunes de las aplicaciones de FunctionGraph?.....	8
1.19 ¿Por qué no se puede resolver el nombre de dominio del gateway de la API vinculado a un servicio durante la invocación de funciones?.....	8
1.20 ¿FunctionGraph admite la transmisión síncrona en el ancho de banda máximo de la intranet?.....	9
1.21 ¿Qué pasa si se agota la cuota de VPC?.....	9
1.22 ¿Cómo puedo imprimir logs de información, de errores o de aviso?.....	9
1.23 ¿Puedo establecer el nombre de dominio de una API en mi propio nombre de dominio?.....	9
1.24 ¿Puedo cambiar el tiempo de ejecución?.....	9
1.25 ¿Puedo cambiar el nombre de una función?.....	9
1.26 ¿Por qué se muestra el mensaje "failed to mount exist system path"?.....	9
1.27 ¿Cómo obtengo los archivos cargados?.....	10
1.28 ¿Por qué no puedo recibir respuestas por invocación síncrona?.....	10
1.29 ¿Qué debo hacer si no se recopilan los logs de ejecución del <code>os.system("command &amp;")</code> ?.....	10
1.30 ¿A qué directorios se puede acceder cuando se utiliza un tiempo de ejecución personalizado?.....	10
1.31 ¿Qué versiones menores de Python 3.6 y 3.9 son compatibles?.....	11

1.32 ¿Qué acciones se pueden utilizar en lugar de una delegación de gestión de VPC para el acceso a VPC?.....	11
1.33 ¿Cuáles son las posibles causas del tiempo de espera de la función?.....	11
1.34 ¿Cómo obtengo el código de una función?.....	11
1.35 ¿Tiene un código de ejemplo para inicializadores?.....	12
<b>2 Preguntas frecuentes sobre la creación de funciones.....</b>	<b>13</b>
2.1 ¿Puedo agregar hilos y procesos en el código de función?.....	13
2.2 ¿Cuáles son las reglas para empaquetar un proyecto de función?.....	13
2.3 ¿Cómo aísla el código FunctionGraph?.....	19
2.4 ¿Cómo creo el archivo Bootstrap para una función HTTP?.....	19
<b>3 Preguntas frecuentes sobre la gestión de activador.....</b>	<b>20</b>
3.1 ¿Qué eventos pueden activar una función de FunctionGraph?.....	20
3.2 ¿Qué pasa si se informa el código de error 500 cuando las funciones que usan activadores de APIG devuelven cadenas?.....	20
3.3 ¿Qué significan LATEST y TRIM_HORIZON en la configuración del activador de DIS?.....	21
3.4 ¿Por qué no puedo activar o desactivar los activadores de OBS invocando a las API?.....	21
3.5 ¿Cómo uso un activador de APIG para invocar una función?.....	22
3.6 ¿Qué permisos se requieren para crear un activador de APIG?.....	22
3.7 ¿Cómo obtiene una función la ruta de acceso o los parámetros de solicitud cuando se utiliza un activador de APIG?.....	22
3.8 ¿Puedo crear un activador de OBS con un bucket existente?.....	23
<b>4 Preguntas frecuentes sobre la gestión de dependencias.....</b>	<b>24</b>
4.1 ¿Qué es una dependencia?.....	24
4.2 ¿Cuándo necesito una dependencia?.....	24
4.3 ¿Cuáles son las precauciones para usar una dependencia?.....	24
4.4 ¿Qué dependencias admite FunctionGraph?.....	24
4.5 ¿FunctionGraph apoya las bibliotecas de clase?.....	26
4.6 ¿Cómo uso las dependencias de FunctionGraph de terceros?.....	26
4.7 ¿Cómo creo las dependencias de función?.....	26
4.8 ¿Cómo puedo crear una dependencia en la consola de FunctionGraph?.....	29
4.9 ¿Cómo agrego una dependencia a una función?.....	29
<b>5 Preguntas frecuentes sobre la ejecución de funciones.....</b>	<b>30</b>
5.1 ¿Cuánto tiempo se tarda en ejecutar una función de FunctionGraph?.....	30
5.2 ¿Qué pasos se incluyen en la ejecución de funciones?.....	30
5.3 ¿Cómo procesa FunctionGraph las solicitudes simultáneas?.....	30
5.4 ¿Qué sucede si las instancias de función no se han ejecutado durante mucho tiempo?.....	31
5.5 ¿Cómo puedo acelerar el acceso inicial a una función?.....	31
5.6 ¿Cómo sé la memoria real utilizada para la ejecución de funciones?.....	31
5.7 ¿Por qué mi primera solicitud es lenta?.....	31
5.8 ¿Qué hago si ocurre un error al invocar a una API?.....	31
5.9 ¿Cómo leo el encabezado de solicitud de una función?.....	32
5.10 ¿Puede una función invocar otra función?.....	32

5.11 ¿Se puede invocar la interfaz de ejecución síncrona en una red privada?.....	34
5.12 ¿Por qué una función utiliza más memoria de la estimada e incluso activa la alarma de falta de memoria?.....	34
5.13 ¿Cómo puedo comprobar el uso de memoria cuando veo "runtime memory limit exceeded"?.....	34
5.14 ¿Cómo soluciono los problemas de CrashLoopBackOff?.....	35
5.15 Después de actualizar una imagen con el mismo nombre, las instancias reservadas siguen utilizando la imagen antigua. ¿Qué puedo hacer?.....	35
<b>6 Preguntas frecuentes sobre la configuración de funciones.....</b>	<b>36</b>
6.1 ¿Puedo establecer variables de entorno al crear funciones?.....	36
6.2 ¿Puedo introducir información confidencial en variables de entorno?.....	36
6.3 ¿Cómo uso las versiones y los alias de una función HTTP con un activador de APIG para la actualización gris?....	36
<b>7 Preguntas frecuentes sobre el acceso a recursos externos.....</b>	<b>40</b>
7.1 ¿Cómo accede una función a la base de datos de MySQL?.....	40
7.2 ¿Cómo accede una función a Redis?.....	41
7.3 ¿Cómo configuro el acceso a la red externa?.....	42
<b>8 Otras preguntas frecuentes.....</b>	<b>43</b>
8.1 ¿Cómo puedo ver las reglas de alarma configuradas para una función?.....	43
8.2 ¿FunctionGraph admite la descompilación de ZIP durante la transcodificación de vídeo?.....	43
8.3 ¿Qué es una aplicación de FunctionGraph?.....	43
8.4 ¿Necesito pagar por el tiempo de inicio en frío?.....	43
8.5 ¿Por qué veo un mensaje que indica que mi cuenta fue suspendida al crear una función?.....	43
8.6 ¿Se facturarán las solicitudes de todas mis funciones en diferentes regiones?.....	44
<b>9 Migración de FunctionGraph de V1 a V2.....</b>	<b>45</b>
9.1 ¿Qué problemas de compatibilidad existen durante la migración?.....	45

# 1 Preguntas frecuentes generales

---

## 1.1 ¿Qué es FunctionGraph?

FunctionGraph permite ejecutar código sin aprovisionar ni gestionar servidores, mientras garantiza altos niveles de disponibilidad y escalabilidad. Todo lo que se necesita hacer es cargar el código y establecer las condiciones de ejecución: FunctionGraph se encarga del resto. Usted paga solo por lo que usa y no se le cobra cuando su código no se está ejecutando.

## 1.2 ¿Debo solicitar cualquier servicio de cómputo, de almacenamiento o de red al usar FunctionGraph?

Al utilizar FunctionGraph, no es necesario solicitar ni preconfigurar ningún servicio informático, de almacenamiento o de red, sino que debe cargar y ejecutar código en los tiempos de ejecución admitidos. FunctionGraph proporciona y administra los recursos informáticos subyacentes, incluidas las CPU de servidor, la memoria y las redes. Realiza mantenimiento de configuración y recursos, despliegue de código, ajuste automático, balanceo de carga, actualización segura y monitoreo de recursos.

## 1.3 ¿Necesito desplegar mi código después de programar?

Después de la programación, solo necesita empaquetar su código en un archivo ZIP (Java, Node.js, Python y Go) o archivo JAR (Java), y cargar el archivo a FunctionGraph para su ejecución.

Cuando cree un archivo ZIP, coloque el archivo de controlador en el directorio **root** para asegurarse de que su código se puede ejecutar normalmente después de ser descomprimido.

Si edita código en Go, comprima el archivo compilado y asegúrese de que el nombre del archivo de biblioteca dinámica sea coherente con el nombre del complemento del controlador. Por ejemplo, si el nombre del archivo de biblioteca dinámica es **testplugin.so**, establezca el controlador en **testplugin.Handler**.

## 1.4 ¿Cómo obtengo un token?

Puede usar un token para la autenticación al invocar a las API. Para obtener un token, utilice la API estándar de Identity and Access Management (IAM).

- Ejecute el siguiente comando para obtener el token en la región **CN South-Guangzhou**:

```
curl -k -i -X POST https://iam.cn-south-1.myhuaweicloud.com/v3/auth/tokens -H 'Content-Type: application/json' -d '{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "HUAWEI CLOUD account",
          "password": "Login password",
          "domain": {
            "name": "HUAWEI CLOUD account"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "cn-south-1"
      }
    }
  }
}'
```

- Ejecute el siguiente comando para obtener el token en la región **CN North-Beijing1**:

```
curl -k -i -X POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens -H 'Content-Type: application/json' -d '{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "HUAWEI CLOUD account",
          "password": "Login password",
          "domain": {
            "name": "HUAWEI CLOUD account"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "cn-north-1"
      }
    }
  }
}'
```

**📖 NOTA**

El valor de **X-Subject-Token** en la cabecera de respuesta es el token. Un token obtenido en una región solo se puede usar para invocar a FunctionGraph en esta región.

Para obtener detalles, consulte [Obtención de un token de usuario](#).

## 1.5 ¿Qué tiempos de ejecución admite FunctionGraph?

**Tabla 1-1** enumera los tiempos de ejecución soportados por FunctionGraph.

**Tabla 1-1** Tiempos de ejecución y versiones compatibles

Tiempo de ejecución	Versión
Python	2.7, 3.6, 3.9
Node.js	6.10, 8.10, 10.16, 12.13, 14.18
Java	8 y 11
Go	1.x
C#.NET Core	2.0, 2.1 y 3.1
PHP	7.3

## 1.6 ¿Cuánto espacio en disco se asigna a cada función de FunctionGraph?

A cada función de FunctionGraph se le asigna un espacio de disco efímero de 512 MB. Puede cargar los paquetes de despliegue hasta 10 GB de tamaño. Para obtener más información, consulte la sección [Restricciones de cuotas y de uso](#).

## 1.7 ¿FunctionGraph admite versiones de funciones?

Sí. Para obtener más información, consulte la sección [Gestión de versiones](#).

## 1.8 ¿Cómo una función lee o escribe archivos?

### Fondo

Una función puede leer archivos en el directorio de código. El directorio de trabajo de una función es el directorio de nivel superior del archivo de controlador. Se supone que ha subido una carpeta llamada **backend**. Para leer su archivo **test.conf** en el mismo nivel de directorio que el archivo de controlador, utilice la ruta relativa **code/backend/test.conf** o utilice una ruta absoluta (es decir, el valor de la variable de entorno **RUNTIME\_CODE\_ROOT**). Para escribir un archivo (por ejemplo, para crear o descargar un archivo), vaya al directorio **/tmp** o utilice la función de montaje del sistema de archivos proporcionada por FunctionGraph.



 **NOTA**

- Si se reclaman los contenedores, el contenido de archivo leído/escrito no será válido.
- Actualmente, FunctionGraph no admite la persistencia de instancias.

## Escenarios típicos

- Descargue los archivos almacenados en el servicio de almacenamiento de objetos (OBS) en el directorio **/tmp** para su procesamiento.
- Para almacenar los datos de ejecución de funciones en OBS, cree un archivo en el directorio **/tmp**, escriba los datos en el archivo y, a continuación, cargue el archivo en OBS.

## 1.9 ¿Cómo configuro un proxy cuando uso CLI?

### Pregunta

Cuando uso CLI para cargar un paquete de código ZIP, ¿cómo puedo configurar un servidor de proxy y la información de identidad para completar la autenticación a través del gateway de proxy en la red interna?

### Respuesta

Ejecute el siguiente comando para establecer un proxy:

```
export HTTP_PROXY="http://user:password@proxyIp:proxyPort"
```

Para obtener más información, consulte [https://www.cyberciti.biz/faq/unix-linux-export-variable-http\\_proxy-with-special-characters/](https://www.cyberciti.biz/faq/unix-linux-export-variable-http_proxy-with-special-characters/).

## 1.10 ¿FunctionGraph admite la extensión de función?

FunctionGraph ha integrado las bibliotecas no estándar como redis, http y obs\_client. Puede utilizar directamente estas bibliotecas al desarrollar funciones. Para obtener más información, consulte la [Guía del desarrollador](#).

Alternativamente, utilice sus propias dependencias. Para obtener más información, consulte la sección [Gestión de la dependencia](#).

## 1.11 ¿Qué permisos se requieren para que un usuario de IAM use FunctionGraph?

Al iniciar sesión en FunctionGraph y agregar, eliminar, modificar y consultar una función y sus activadores como usuario de IAM, conceda permisos al grupo de usuarios al que pertenece el usuario de IAM según sea necesario. Por ejemplo, para crear un bucket de OBS y un activador, conceda el permiso **Tenant Administrator** para OBS. Otro ejemplo, para cargar las dependencias de OBS, conceder los permisos **FunctionGraph FullAccess** y **OBS Administrator** al usuario de IAM y establecer el ámbito en servicios globales. Gestione los permisos según el principio de permisos mínimos (PoLP). Para obtener más información, consulte [Descripción de permisos](#).

## 1.12 ¿Cómo puedo crear un paquete de dependencia de Python basado en una unidad de ODBC para la consulta de base de datos?

Para los paquetes dependientes del sistema operativo (por ejemplo, unixODBC), descargue el código fuente para compilar los paquetes dependientes.

1. Inicie sesión en su ECS en la consola de ECS (asegúrese de que las herramientas de GCC y de Make han sido instaladas) y ejecute el siguiente comando para descargar el paquete de código fuente:

```
wget source code path
```

Si descargó un archivo **.zip**, ejecute el siguiente comando para descomprimirlo:

```
unzip xxx/xx.zip
```

Si descargó un archivo **tar.gz**, ejecute el siguiente comando para descomprimirlo:

```
tar -zxvf xxx/xx.tar.gz
```

2. Run the following command to create the **/opt/function/code** directory:

```
mkdir /opt/function/code
```

3. Vaya al directorio de destino y ejecute el siguiente comando:

```
./configure --prefix=/opt/function/code --sysconfdir=/opt/function/code;make;make install
```

4. Vaya a **/opt/function/code/lib/pkgconfig** y compruebe si el directorio de prefijo es **/opt/function/code**.

```
cd /opt/function/code/lib/pkgconfig
```

5. Copie todos los archivos de **/opt/function/code/lib** a **/opt/function/code**.

```
cp -r /opt/function/code/lib/* /opt/function/code
```

6. Cambie a **/opt/function/code** y comprima todos los archivos en un paquete de **.zip**.

```
cd /opt/function/code  
zip -r xxx.zip *
```

## 1.13 ¿Cuál es la cuota de FunctionGraph?

Para obtener detalles sobre la cuota de recursos de FunctionGraph, consulte la sección [Restricciones de cuotas y de uso](#). Para obtener más información sobre cómo aumentar la cuota, consulte [¿Cómo solicito una cuota más alta?](#)

## 1.14 ¿Qué fuentes chinas admite FunctionGraph?

FunctionGraph admite las siguientes fuentes:

- NotoSansTC-Regular.otf
- NotoSerifTC-Regular.otf
- NotoSansSC-Regular.otf
- NotoSerifSC-Regular.otf

## 1.15 ¿Cómo resuelve FunctionGraph un nombre de dominio de DNS privado?

FunctionGraph no puede analizar directamente los nombres privados de dominio de DNS de Huawei Cloud. Para analizarlos, invoque a las API de DNS y realice los siguientes pasos.

### Resolver un nombre de dominio de DNS privado

Asegúrese de que se han creado una VPC y un nombre de dominio de DNS privado antes de realizar los siguientes pasos:

**Paso 1** Asociar una VPC con el nombre de dominio privado y agregue conjuntos de registros.

Inicie sesión en la consola de DNS y asocie una VPC con el nombre de dominio privado.

Click the domain name, and add a type A record set.

**Paso 2** Crear una función.

Cree una función cuyo tiempo de ejecución sea Python 2.7. El siguiente es un código de ejemplo.

```
# -*- coding:utf-8 -*-
import json
import os
def handler(event, context):
    os.system("curl -iv www.test.com")
```

**Paso 3** Configurar una delegación con permisos de DNS y de VPC para la función.

En la consola de IAM, cree una delegación con permisos **DNS ReadOnlyAccess** y **VPC Administrator** para FunctionGraph.

#### NOTA

Necesita configurar el permiso para leer datos de recursos de DNS porque la función necesita obtener dichos datos al analizar un nombre de dominio. De lo contrario, se muestra el siguiente mensaje de error, que indica que no se han podido obtener los datos del recurso de DNS.

```
2020/08/20 10:37:12 GMT+08:00 Start invoke request
'a2f105b4-2e72-4fda-94a5-86d3837e961d', version: latest
[GET /v2/zones/{zone_id}/recordsets] failed, response:
{"code": "DNS.1802", "message": "Policy doesn't allow dns:recordset:list to be
performed."}
2020/08/20 10:37:13 GMT+08:00 Finish invoke request
'a2f105b4-2e72-4fda-94a5-86d3837e961d', duration: 1030.072ms, billing
duration: 1100ms, memory used: 77.039MB.
```


**Paso 4** Configurar la función.


En la página de detalles de la función creada en el [Paso 2](#) haga clic en la ficha **Configuration** y configure las siguientes opciones:


1. En el caso de **Permissions**, seleccione la delegación creada en [Paso 3](#).
2. Habilite el acceso a la VPC y seleccione la VPC creada, la subred y el nombre de dominio. Para más detalles, véase [Figura 1-1](#).



**Figura 1-1** Configuración de la función

**VPC Access** ?

Status 

VPC   [Create VPC](#)

Subnet   [Create Subnet](#)

Domain Name ?    
  [config.vpc.dns.aEle](#)

[Save](#)

**Paso 5** Comprobar el resultado de la ejecución.

A continuación, se pueden analizar todos los nombres de dominio IPv4 configurados.

----Fin

 **NOTA**

Los cambios en las direcciones IP correspondientes a los nombres de dominio de VPC que configure entrarán en vigor en 10 minutos.

## 1.16 ¿Cómo resuelve una función basada en imagen de contenedor un nombre de dominio de DNS privado?

Las funciones de FunctionGraph creadas con una imagen contenedora no pueden analizar directamente los nombres de dominio privados del servicio de nombres de dominio (DNS). Sin embargo, puede invocar a las API de DNS para lograr este propósito.

### Resolver un nombre de dominio de DNS privado

1. Obtener un nombre de dominio privado y un ID de zona.

Este procedimiento utiliza un nombre de dominio con un conjunto de registros como ejemplo.

- a. Inicie sesión en la consola de DNS.
- b. Obtenga un ID de zona.

Haga clic en  y seleccione **Domain Name** en el cuadro de búsqueda para obtener un ID de zona.

- c. Obtenga el nombre de dominio privado correspondiente a un conjunto de registros.  
Haga clic en el nombre de dominio para ir a la lista de conjuntos de registros y seleccione un conjunto de registros.

2. Compilar la lógica de resolución.

Depurar la API utilizada para [consultar los conjuntos de registros en una zona](#).

- Establezca **zone\_id** en el ID de zona obtenido en el paso anterior y haga clic en **Debug**. La dirección IP del nombre de dominio privado se muestra en el cuerpo de la respuesta.
- Cambie a la pestaña **Sample Code** para obtener el código completo. Para obtener más información acerca de las dependencias, haga clic en **View SDK Details**.

## 1.17 ¿Cómo uso un nombre de dominio para acceder a una API registrada con API Gateway (Dedicado)?

El nombre de dominio **www.test.com** se utiliza como ejemplo. El procedimiento es el siguiente:

- Paso 1** Inicie sesión en la consola de API Gateway, elija **Dedicated Gateways** en el panel de navegación y haga clic en el nombre del gateway de destino. En la página **Gateway Information**, vea **EIP** en el área **Inbound Access** para obtener la dirección IP del gateway de API.
- Paso 2** En la consola de DNS, configure una regla IPv4 para asignar **www.test.com** a una dirección del gateway de API.
- Paso 3** Configurar la resolución de nombres de dominio haciendo referencia a [¿Cómo resuelve FunctionGraph un nombre de dominio de DNS privado?](#). De esta manera, puede acceder a la API registrada con el gateway de la API usando **www.test.com** de nombre de dominio.

----Fin

## 1.18 ¿Cuáles son los escenarios comunes de las aplicaciones de FunctionGraph?

1. Aplicaciones web: mini programas, páginas web/apps, chatbots y Backends for Frontends (BFF).
2. Aplicaciones basadas en eventos: procesamiento de archivos, procesamiento de imágenes, transmisión/transcodificación de video en directo, procesamiento de flujo de datos en tiempo real y procesamiento de reglas/eventos de IoT.
3. Aplicaciones de IA: integración de servicios de terceros, inferencia de IA y reconocimiento de matrículas.

Para obtener más información, consulte la sección [Escenarios de aplicaciones](#).

## 1.19 ¿Por qué no se puede resolver el nombre de dominio del gateway de la API vinculado a un servicio durante la invocación de funciones?

Actualmente, FunctionGraph solo resuelve los nombres de dominio de DNS de Huawei Cloud y los nombres de dominio de POD.

## 1.20 ¿FunctionGraph admite la transmisión síncrona en el ancho de banda máximo de la intranet?

No en este momento.

## 1.21 ¿Qué pasa si se agota la cuota de VPC?

Un tenant puede crear hasta 4 VPC. Para crear más VPC, envíe un ticket de servicio.

## 1.22 ¿Cómo puedo imprimir logs de información, de errores o de aviso?

Se tome Java como ejemplo. Puede usar [esta demostración](#) para imprimir logs.

## 1.23 ¿Puedo establecer el nombre de dominio de una API en mi propio nombre de dominio?

Sí. El procedimiento es el siguiente:

**Paso 1** Inicie sesión en la consola de APIG y vincule un nombre de dominio haciendo referencia a [Vinculación de un nombre de dominio](#).

**Paso 2** En la página de la ficha **Domain Names** del grupo de API creado, haga clic en **Bind Independent Domain Name**. Por ejemplo, establezca `xxxx.apig.cn-north` en `test.com/user/get`.

----Fin

## 1.24 ¿Puedo cambiar el tiempo de ejecución?

No. Una vez creada una función, no se puede cambiar su tiempo de ejecución.

## 1.25 ¿Puedo cambiar el nombre de una función?

No. El nombre de una función no se puede cambiar una vez creada.

## 1.26 ¿Por qué se muestra el mensaje "failed to mount exist system path"?

Cuando vea este mensaje, monte el archivo en una nueva ruta.

ID de usuario/ID de grupo de usuarios: Puede ser cualquier número excepto 1000. El valor `-1` se convertirá automáticamente a **1003**. Los dos ID controlan los permisos de directorio para acceder a un sistema de archivos remoto.

Nombre del sistema de archivos/ECS: Nombre del sistema de archivos o ECS que se va a crear. Asegúrese de que ha especificado una VPC y una delegación a la que se le ha autorizado a acceder.

Directorio compartido: Para configurar un directorio compartido remoto para el ECS montado, consulte la [Creación de un directorio compartido de NFS en ECS](#).

Ruta de acceso: Ubicación donde se va a montar el sistema de archivos en la función. Establezca un nuevo directorio de dos niveles que comience por `/mnt`. Por ejemplo, `/mnt/test`.

## 1.27 ¿Cómo obtengo los archivos cargados?

Se tome Python como ejemplo. Si utiliza `os.getcwd()` para consultar el directorio actual, el directorio será `/opt/function`. Sin embargo, el código se ha subido a `/opt/function/code` en realidad.

Puede utilizar cualquiera de los siguientes métodos para obtener archivos cargados:

1. Ejecute el comando `cd` para cambiar a `/opt/function/code`.
2. Acceda a la ruta absoluta (valor de la variable de entorno `RUNTIME_CODE_ROOT`).

### NOTA

Puede obtener los archivos cargados haciendo referencia a los métodos anteriores cuando se utilizan otros idiomas.

## 1.28 ¿Por qué no puedo recibir respuestas por invocación síncrona?

Si la latencia de ejecución de la función E2E supera los 90s, se recomienda la invocación asincrónica. Si se utiliza una invocación síncrona, no se pueden recibir respuestas después de 90s debido a restricciones del gateway.

## 1.29 ¿Qué debo hacer si no se recopilan los logs de ejecución del `os.system("command &")`?

No utilice `os.system("command &")`. El resultado del comando en segundo plano no se recopilará. Para obtener la salida del comando, utilice `subprocess.Popen` en su lugar.

## 1.30 ¿A qué directorios se puede acceder cuando se utiliza un tiempo de ejecución personalizado?

Por defecto, solo se puede acceder al directorio `/tmp`, por ejemplo, para crear o descargar archivos.

## 1.31 ¿Qué versiones menores de Python 3.6 y 3.9 son compatibles?

3.6.8 y 3.9.2.

## 1.32 ¿Qué acciones se pueden utilizar en lugar de una delegación de gestión de VPC para el acceso a VPC?

Se pueden utilizar las acciones enumeradas en la [Tabla 1-2](#).

**Tabla 1-2** Acciones

Permiso	Acción
Supresión de un puerto	vpc:ports:delete
Consulta de un puerto	vpc:ports:get
Creación de un puerto	vpc:ports:create
Consulta de una VPC	vpc:vpcs:get
Consulta de una subred	vpc:subnets:get

## 1.33 ¿Cuáles son las posibles causas del tiempo de espera de la función?

- Se agotó el tiempo de espera de la lógica de código. En este caso, optimice el código o aumente el tiempo de espera.
- Se agotó el tiempo de espera de la red. Para solucionar este problema, aumente el tiempo de espera.
- Tomó mucho tiempo cargar las clases de Java durante el inicio en frío. En este caso, aumente el tiempo de espera o la memoria.

## 1.34 ¿Cómo obtengo el código de una función?

1. Inicie sesión en la consola de FunctionGraph y haga clic en el nombre de la función de destino para ir a la página de detalles. Elija **Operation** > **Export function** en la parte superior derecha y haga clic en **Export Code**.
2. Alternativamente, invoque a la API de exportación de función.



## 1.35 ¿Tiene un código de ejemplo para inicializadores?

Sí. Vea los siguientes ejemplos:

- Node.js ([Introducción al inicializador](#))

```
exports.initializer = function(context, callback) {  
    callback(null, '');  
};
```

- Python ([Introducción al inicializador](#))

```
def my_initializer(context):  
    print("hello world!")
```

- Java ([Introducción al inicializador](#))

```
public void my_initializer(Context context)  
{  
    RuntimeLogger log = context.getLogger();  
    log.log(String.format("ak:%s", context.getAccessKey()));  
}
```

- PHP ([Introducción al inicializador](#))

```
<?php  
Function my_initializer($context) {  
    echo 'hello world' . PHP_EOL;  
}  
?>
```

# 2 Preguntas frecuentes sobre la creación de funciones

---

## 2.1 ¿Puedo agregar hilos y procesos en el código de función?

Puede crear los hilos y procesos adicionales en su función mediante las funciones del tiempo de ejecución y de las características del sistema operativo.

## 2.2 ¿Cuáles son las reglas para empaquetar un proyecto de función?

Además de la edición de código en línea, puede crear una función cargando un archivo ZIP o JAR, o cargando un archivo ZIP desde OBS. Para más detalles, véase [Reglas de empaquetado](#) y [Ejemplo de paquetes de proyecto ZIP](#).

### Reglas de empaquetado

Además de editar código en línea, puede crear una función cargando un archivo ZIP local o un archivo JAR, o cargando un archivo ZIP desde Object Storage Service (OBS). [Tabla 2-1](#) describe las reglas para empaquetar un proyecto de función.

**Tabla 2-1** Reglas de empaquetado del proyecto de función

Tiempo de ejecución	Archivo JAR	Archivo ZIP	Archivo ZIP en OBS
Node.js	No se admite.	<ul style="list-style-type: none"> <li>● Si los archivos del proyecto de función se guardan bajo el directorio <code>~/Code/</code>, seleccione y empaquete todos los archivos bajo este directorio para asegurarse de que el controlador de función está bajo el directorio raíz después de descomprimir el archivo ZIP.</li> <li>● Si el proyecto de función utiliza las dependencias de terceros, empaquete las dependencias en un archivo ZIP e importe el archivo ZIP en la página de código de función. Alternativamente, empaquete las dependencias de terceros y los archivos de proyecto de función juntos.</li> </ul>	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.

Tiempo de ejecución	Archivo JAR	Archivo ZIP	Archivo ZIP en OBS
PHP	No se admite.	<ul style="list-style-type: none"><li>● Si los archivos del proyecto de función se guardan bajo el directorio <code>~/Code/</code>, seleccione y empaquete todos los archivos bajo este directorio para asegurarse de que el controlador de función está bajo el directorio raíz después de descomprimir el archivo ZIP.</li><li>● Si el proyecto de función utiliza las dependencias de terceros, empaquete las dependencias en un archivo ZIP e importe el archivo ZIP en la página de código de función. Alternativamente, empaquete las dependencias de terceros y los archivos de proyecto de función juntos.</li></ul>	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.

Tiempo de ejecución	Archivo JAR	Archivo ZIP	Archivo ZIP en OBS
Python 2.7	No se admite.	<ul style="list-style-type: none"> <li>● Si los archivos del proyecto de función se guardan bajo el directorio <code>~/Code/</code>, seleccione y empaquete todos los archivos bajo este directorio para asegurarse de que el controlador de función está bajo el directorio raíz después de descomprimir el archivo ZIP.</li> <li>● Si el proyecto de función utiliza las dependencias de terceros, empaquete las dependencias en un archivo ZIP e importe el archivo ZIP en la página de código de función. Alternativamente, empaquete las dependencias de terceros y los archivos de proyecto de función juntos.</li> </ul>	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.

Tiempo de ejecución	Archivo JAR	Archivo ZIP	Archivo ZIP en OBS
Python 3.6	No se admite.	<ul style="list-style-type: none"> <li>● Si los archivos del proyecto de función se guardan bajo el directorio <code>~/Code/</code>, seleccione y empaquete todos los archivos bajo este directorio para asegurarse de que el controlador de función está bajo el directorio raíz después de descomprimir el archivo ZIP.</li> <li>● Si el proyecto de función utiliza las dependencias de terceros, empaquete las dependencias en un archivo ZIP e importe el archivo ZIP en la página de código de función. Alternativamente, empaquete las dependencias de terceros y los archivos de proyecto de función juntos.</li> </ul>	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.
Java 8	Si la función no hace referencia a los componentes de terceros, compila solo los archivos de proyecto de función en un archivo JAR.	Si la función hace referencia a componentes de terceros, compila los archivos de proyecto de función en un archivo JAR y comprime todos los componentes de terceros y el archivo JAR de función en un archivo ZIP.	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.

Tiempo de ejecución	Archivo JAR	Archivo ZIP	Archivo ZIP en OBS
Go 1.x	No se admite.	Comprima el archivo compilado y asegúrese de que el nombre del archivo binario es consistente con el del controlador. Por ejemplo, si el nombre del archivo binario es <b>Handler</b> , establezca el nombre del controlador en <b>Handler</b> .	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.
C#	No se admite.	Compress project files into a ZIP file. The ZIP file must contain the following files: <i>Project_name.deps.json</i> , <i>Project_name.dll</i> , <i>Project_name.runtimeconfig.json</i> , <i>Project_name.pdb</i> , and <b>HC.Serverless.Function.Common.dll</b> .	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.
Custom	No se admite.	Comprima los archivos de proyecto en un archivo ZIP. El archivo ZIP debe contener un archivo de arranque.	Comprima los archivos de proyecto en un archivo ZIP y subirlo a un bucket de OBS.

## Ejemplo de paquetes de proyecto ZIP

- Ejemplo de directorio de un paquete de proyecto Nods.js

```

Example.zip
|--- lib
|--- node_modules
directory
|--- index.js
|--- package.json
Example project package
Service file directory
NPM third-party component
.js handler file (mandatory)
NPM project management file

```

- Ejemplo de directorio de un paquete de proyecto PHP

```

Example.zip
|--- ext
|--- pear
repository
|--- index.php
Example project package
Extension library directory
PHP extension and application
PHP handler file

```

- Ejemplo de directorio de un paquete de proyecto Python

```

Example.zip
|--- com
|--- PLI
directory
Example project package
Service file directory
Third-party dependency PLI

```

--- index.py	.py handler file (mandatory)
--- watermark.py	.py file for image watermarking
--- watermark.png	Watermarked image

- **Ejemplo de directorio de un paquete de proyecto Java**

Example.zip	Example project package
--- obstest.jar	Service function JAR file
--- esdk-obs-java-3.20.2.jar file	Third-party dependency JAR file
--- jackson-core-2.10.0.jar file	Third-party dependency JAR file
--- jackson-databind-2.10.0.jar file	Third-party dependency JAR file
--- log4j-api-2.12.0.jar file	Third-party dependency JAR file
--- log4j-core-2.12.0.jar file	Third-party dependency JAR file
--- okhttp-3.14.2.jar file	Third-party dependency JAR file
--- okio-1.17.2.jar file	Third-party dependency JAR file

- **Ejemplo de directorio de un paquete de proyecto Go**

Example.zip	Example project package
--- testplugin.so	Service function package

- **Ejemplo de directorio de un paquete de proyecto C#**

Example.zip	Example project package
--- fssExampleCsharp2.0.deps.json	File generated after project compilation
--- fssExampleCsharp2.0.dll	File generated after project compilation
--- fssExampleCsharp2.0.pdb	File generated after project compilation
--- fssExampleCsharp2.0.runtimeconfig.json	File generated after project compilation
--- Handler	Help file, which can be directly used
--- HC.Serverless.Function.Common.dll	.dll file provided by FunctionGraph

- **Personalizado**

Example.zip	Example project package
--- bootstrap	Executable boot file

## 2.3 ¿Cómo aísla el código FunctionGraph?

Cada función de FunctionGraph se ejecuta en su propio entorno y tiene sus propios recursos y sistema de archivos.

## 2.4 ¿Cómo creo el archivo Bootstrap para una función HTTP?

Para crear una función HTTP, cree un archivo de arranque. Para obtener más información, consulte [Creación de un archivo de arranque](#).



# 3 Preguntas frecuentes sobre la gestión de activador

---

## 3.1 ¿Qué eventos pueden activar una función de FunctionGraph?

Para obtener más información, consulte [Orígenes de eventos compatibles](#).

## 3.2 ¿Qué pasa si se informa el código de error 500 cuando las funciones que usan activadores de APIG devuelven cadenas?

Asegúrese de que la respuesta de función para una invocación por API Gateway se ha encapsulado y contiene **body(String)**, **statusCode(int)**, **headers(Map)** y **isBase64Encoded(boolean)**.

A continuación se muestra una respuesta de ejemplo devuelta por una función de Node.js que utiliza un activador de APIG:

```
exports.handler = function (event, context, callback) {
  const response = {
    'statusCode': 200,
    'isBase64Encoded': false,
    'headers': {
      "Content-type": "application/json"
    },
    'body': 'Hello, FunctionGraph with APIG',
  }
  callback(null, response);
}
```

A continuación se muestra una respuesta de ejemplo devuelta por una función Java que utiliza un activador de APIG:

```
import java.util.Map;

public HttpTriggerResponse index(String event, Context context){
```

```
String body = "<html><title>FunctionStage</title>"
    + "<h1>This is a simple APIG trigger test</h1><br>"
    + "<h2>This is a simple APIG trigger test</h2><br>"
    + "<h3>This is a simple APIG trigger test</h3>"
    + "</html>";
int code = 200;
boolean isBase64 = false;
Map<String, String> headers = new HashMap<String, String>();
headers.put("Content-Type", "text/html; charset=utf-8");
return new HttpTriggerResponse(body, headers, code, isBase64);
}

class HttpTriggerResponse {
    private String body;
    private Map<String, String> headers;
    private int statusCode;
    private boolean isBase64Encoded;
    public HttpTriggerResponse(String body, Map<String,String> headers,
int statusCode, boolean isBase64Encoded){
        this.body = body;
        this.headers = headers;
        this.statusCode = statusCode;
        this.isBase64Encoded = isBase64Encoded;
    }
}
```

### 3.3 ¿Qué significan LATEST y TRIM\_HORIZON en la configuración del activador de DIS?

Los cursores **LATEST** y **TRIM\_HORIZON** especifican los puntos de inicio para leer datos en flujos de servicio de ingestión de datos (DIS).

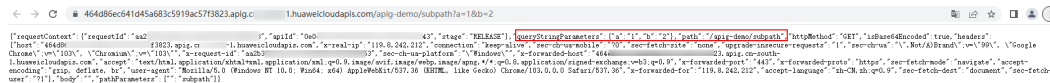
- **TRIM\_HORIZON**: Los datos se leen del registro válido más antiguo almacenado en la partición.  
Por ejemplo, un tenant usó un flujo de DIS para cargar tres piezas de datos A1, A2 y A3. Suponiendo que A1 expira, pero A2 y A3 siguen siendo válidos después de un periodo de tiempo, si el tenant especifica **TRIM\_HORIZON** para descargar datos, solo A2 y A3 pueden descargarse.
- **LATEST**: Los datos se leen del último registro de la partición. Esta opción asegura que se lean los datos más recientes de la partición.

Para obtener más información, consulte [Obtención de un cursor](#).

### 3.4 ¿Por qué no puedo activar o desactivar los activadores de OBS invocando a las API?

OBS no admite activadores de pull. Por lo tanto, los activadores de OBS no se pueden habilitar o deshabilitar.





## 3.8 ¿Puedo crear un activador de OBS con un bucket existente?

Sí. Si se muestra un mensaje que indica que la configuración del activador actual entra en conflicto con la de otro, los dos activadores tienen el mismo bucket, prefijo y sufijo. Si aún desea usar este bucket para el activador actual, modifique el prefijo o el sufijo.

# 4 Preguntas frecuentes sobre la gestión de dependencias

---

## 4.1 ¿Qué es una dependencia?

Una dependencia es un paquete de programa y también un entorno necesario para ejecutar un paquete de software. El paquete de software se basa en y solo puede ejecutarse en el entorno.

## 4.2 ¿Cuándo necesito una dependencia?

Cuando instala un programa o desarrolla código que depende de un entorno para ejecutarse, debe introducir la dependencia.

## 4.3 ¿Cuáles son las precauciones para usar una dependencia?

- El nombre de cada archivo de una dependencia no puede terminar con una tilde (~).
- No debe haber más de 30,000 archivos en una dependencia.
- Puede cargar un archivo ZIP de dependencia dentro de 10 MB en la página de detalles de la función. Para una dependencia mayor (máximo 300 MB), subirlo usando OBS.
- Si su función utiliza una dependencia privada grande, aumente el tiempo de espera seleccionando **Configuration > Basic Settings** en la página de detalles de la función.

## 4.4 ¿Qué dependencias admite FunctionGraph?

### Dependencias admitidas

FunctionGraph admite las bibliotecas estándar y dependencias de terceros.

- Bibliotecas estándar

Si utiliza las bibliotecas estándar, puede importarlas a su código en línea o empaquetarlas y subirlas a FunctionGraph.

- Se admita las bibliotecas no estándares  
FunctionGraph ofrece los componentes integrados de terceros, como se describe en [Tabla 4-1](#) y [Tabla 4-2](#). Puede importar estos componentes al código en línea de la misma manera que importa las bibliotecas estándar.

**Tabla 4-1** Componentes de terceros integrados con el tiempo de ejecución de Node.js

Nombre	Descripción	Versión
q	Encapsulación de método asincrónico	1.5.1
co	Control de proceso asincrónico	4.6.0
lodash	Biblioteca común de herramientas y métodos	4.17.10
esdk-obs-nodejs	SDK de OBS	2.1.5
express	Marco de desarrollo de aplicaciones basado en web simplificado	4.16.4
fgs-express	Proporciona un marco de aplicación de Node.js para que FunctionGraph y APIG ejecuten aplicaciones sin servidor y las API de REST. Este componente proporciona un ejemplo del uso del framework de Express para crear aplicaciones o servicios web sin servidor y API RESTful.	1.0.1
request	Simplifica la invocación de HTTP y admite HTTPS y redirección.	2.88.0

**Tabla 4-2** Bibliotecas no estándares soportadas por el tiempo de ejecución de Python

Módulo	Descripción	Versión
dateutil	Fecha y hora de procesamiento	2.6.0
requests	Biblioteca de HTTP	2.7.0
httplib2	httpcliente	0.10.3
numpy	Computación matemática	1.13.1
redis	Ciente de Redis	2.10.5

Módulo	Descripción	Versión
obsclient	Cliente de OBS	-
smnsdk	Acceso de SMN (nube pública)	1.0.1

- Otras bibliotecas de terceros

Para otras bibliotecas de terceros no enumeradas en las tablas anteriores, empaquéte las y cárguelas en un bucket de OBS o en la página de detalles de función. Para obtener más información, consulte [¿Cómo puedo crear una dependencia en la consola de FunctionGraph?](#). Estas bibliotecas se usarán en su código de función.

## 4.5 ¿FunctionGraph apoya las bibliotecas de clase?

Sí. FunctionGraph admite las bibliotecas estándar y bibliotecas no estándares de terceros. Para más detalles, véase [¿Qué dependencias admite FunctionGraph?](#)

## 4.6 ¿Cómo uso las dependencias de FunctionGraph de terceros?

1. Empaquete las bibliotecas de terceros en un paquete ZIP haciendo referencia a [¿Cómo creo las dependencias de función?](#)
2. Cree una dependencia en la consola de FunctionGraph haciendo referencia a [¿Cómo puedo crear una dependencia en la consola de FunctionGraph?](#)
3. En la página de detalles de la función, haga clic en la ficha **Code** y agregue la dependencia haciendo referencia a [¿Cómo agrego una dependencia a una función?](#). Luego puede usar la dependencia en el código de función.

## 4.7 ¿Cómo creo las dependencias de función?

**Se recomienda crear las dependencias de función en EulerOS.** Si se utilizan otros sistemas operativos, puede producirse un error debido a las bibliotecas dependientes subyacentes. Por ejemplo, no se puede encontrar la biblioteca de enlaces dinámicos.

### NOTA

Si los módulos que se van a instalar necesitan las dependencias como .dll, .so y .a, guárdelos en un paquete .zip.

## Configuración del entorno de EulerOS

EulerOS es un sistema operativo de Linux de código abierto para empresas. Cuenta con alta seguridad, escalabilidad y rendimiento, cumpliendo con los requisitos de los clientes para infraestructura de TI y servicios de cómputo en la nube. Se recomienda [Huawei Cloud EulerOS](#).

1. Compre un ECS de EulerOS en Huawei Cloud haciendo referencia a [Comprar e iniciar sesión en un ECS de Linux](#). En la página **Configure Basic Settings**, seleccione **Public Image** y seleccione **Huawei Cloud EulerOS** y una versión de imagen.
2. Descargue la imagen de EulerOS y utilice el software de virtualización para configurar la VM de EulerOS en un PC local.

## Creación de una dependencia para una función de Python

Asegúrese de que la versión de Python del entorno de empaquetado es la misma que la de la función. Para Python 2.7, se recomienda Python 2.7.12 o posterior. Para Python 3.6, se recomienda Python 3.6.3 o posterior.

Para instalar la dependencia de PyMySQL para una función de Python 2.7 en el directorio **/tmp/pymysql** local, ejecute el siguiente comando:

```
pip install PyMySQL --root /tmp/pymysql
```

Después de que el comando se ejecute correctamente, vaya al directorio **/tmp/pymysql**:

```
cd /tmp/pymysql/
```

Vaya al directorio **site-packages** (generalmente **usr/lib64/python2.7/site-packages/**) y ejecute el siguiente comando:

```
zip -rq pymysql.zip *
```

Se genera la dependencia requerida.

### NOTA

Para instalar el paquete de instalación de la rueda local, ejecute el siguiente comando:

```
pip install piexif-1.1.0b0-py2.py3-none-any.whl --root /tmp/piexif  
//Replace piexif-1.1.0b0-py2.py3-none-any.whl with the actual installation  
package name.
```

## Creación de una dependencia para una función Node.js

Asegúrese de que la versión de Node.js correspondiente se ha instalado en el entorno.

Para instalar la dependencia MySQL para una función Node.js 8.10, ejecute el siguiente comando:

```
npm install mysql --save
```

La carpeta **node\_modules** se genera bajo el directorio actual.

- SO Linux

Ejecute el siguiente comando para generar un paquete ZIP.

```
zip -rq mysql-node8.10.zip node_modules
```

Se genera la dependencia requerida.

- SO Windows

Comprima **node\_modules** en un archivo ZIP.

Para instalar varias dependencias, cree primero un archivo **package.json**. Por ejemplo, introduzca el siguiente contenido en el archivo **package.json** y, a continuación, ejecute el siguiente comando:

```
{  
  "name": "test",
```



```
"version": "1.0.0",
"dependencies": {
  "redis": "~2.8.0",
  "mysql": "~2.17.1"
}
}
npm install --save
```

#### NOTA

No ejecute el comando **CNPM** para generar dependencias de Node.js.

Comprima **node\_modules** en un paquete ZIP. Esto genera una dependencia que contiene tanto MySQL como Redis.

Para otras versiones de Node.js, puede crear dependencias de la manera indicada anteriormente.

## Creación de una dependencia para una función Java

Cuando compila una función usando Java, las dependencias deben compilarse localmente. Para obtener más información sobre cómo agregar dependencias, consulte [Desarrollo de funciones en Java \(Usando un IDEA Java Project\)](#).

## Creación de una dependencia para una función PHP

Se recomienda EulerOS 2.9.6.

Por defecto, Composer y PHP 7.3 se han instalado en el entorno. Instalar Protobuf 3.19 usando Composer.

Cree el archivo **composer.json** con el siguiente contenido:

```
{
  "require": {
    "google/protobuf": "^3.19"
  }
}
```

Ejecute el siguiente comando:

```
Composer install
```

La carpeta **vendor** se genera con las subcarpetas **autoload.php**, **composer** y **google** en el directorio actual.

- Linux

Ejecute el siguiente comando para generar un paquete ZIP.

```
zip -rq vendor.zip vendor
```

- Windows

Comprima **vendor** en un archivo ZIP.

Si es necesario instalar varias dependencias, indíquelas en el archivo **composer.json**, comprima la carpeta **vendor** en un archivo ZIP y súbalo.

#### NOTA

Para usar dependencias de terceros descargadas usando Composer en código de proyecto de PHP, cargue las dependencias a través de **solicitar "./vendor/autoload.php"**. De forma predeterminada, los archivos descomprimidos del paquete ZIP cargado se colocan en un directorio al mismo nivel que el código del proyecto.

## 4.8 ¿Cómo puedo crear una dependencia en la consola de FunctionGraph?

1. Inicie sesión en la consola de FunctionGraph y elija **Functions > Dependencias** en el panel de navegación.
2. Haga clic en **Create Dependency**.
3. Configure los siguientes parámetros.

**Tabla 4-3** Parámetros de configuración de dependencia

Parámetro	Descripción
Name	Nombre de la dependencia.
Code Entry Mode	Subir un archivo ZIP directamente o a través de OBS. <ul style="list-style-type: none"> <li>● <b>Upload ZIP file:</b> Haga clic en <b>Select File</b> para cargar un archivo ZIP.</li> <li>● <b>Upload from OBS:</b> Especifica un URL de enlace de OBS. Para obtener más información sobre cómo obtener la dirección URL, consulte <a href="#">Acceso a un objeto mediante su URL</a>.</li> </ul>
Runtime	Seleccione un tiempo de ejecución.
Description	Descripción de la dependencia. Este parámetro es opcional.

4. Haga clic en **OK**. De forma predeterminada, una nueva dependencia es la versión **1**.
5. Haga clic en el nombre de la dependencia y vea todas las versiones e información relacionada en la página mostrada. Cada dependencia puede tener varias versiones.
  - Para crear una versión de dependencia, haga clic en **Create Version** en la esquina superior derecha de la página.
  - Para ver la dirección de una versión, haga clic en la versión.
  - Para eliminar una versión, haga clic en el icono de eliminación en la misma fila.

Version	Name	Description	Last Modified	Description
1	00-00-000	00000000	2020-01-14 10:10	last month

## 4.9 ¿Cómo agrego una dependencia a una función?

1. En la página de detalles de la función, haga clic en la ficha **Code** y haga clic en **Add** en el área **Dependencies**.
  - **Public:** Las dependencias públicas son proporcionadas por FunctionGraph y se pueden agregar directamente.
  - **Private:** Las dependencias privadas son las que creaste y subiste.
2. Haga clic en **OK**.

# 5 Preguntas frecuentes sobre la ejecución de funciones

---

## 5.1 ¿Cuánto tiempo se tarda en ejecutar una función de FunctionGraph?

Para la invocación de función, la duración de ejecución está dentro de 900s. El tiempo de espera de ejecución predeterminado es 3s. Puede establecer la duración del tiempo de espera (unidad: s) en un entero comprendido entre 3 y 900. Si establece la duración del tiempo de espera de ejecución de una función en 3s, se terminará después de 3s.

## 5.2 ¿Qué pasos se incluyen en la ejecución de funciones?

La ejecución de la función incluye dos pasos:

1. Seleccione una instancia inactiva con la memoria requerida.
2. Ejecute el código especificado.

## 5.3 ¿Cómo procesa FunctionGraph las solicitudes simultáneas?

FunctionGraph escala automáticamente las instancias de función según el número de solicitudes. Si el número de solicitudes simultáneas aumenta, FunctionGraph asigna más instancias de función para procesar las solicitudes. Si ese número disminuye, FunctionGraph asigna menos instancias de función en consecuencia.

Número de instancias de función = Concurrencia de función/Concurrencia por instancia

- Concurrencia de función: el número de solicitudes ejecutadas simultáneamente por una función en un determinado punto de tiempo.
- Concurrencia por instancia: el número máximo de solicitudes simultáneas permitidas por una sola instancia. Esto es equivalente al parámetro **Max. Requests per Instance** de la página **Concurrency**.

 **NOTA**

Para obtener detalles acerca de los servicios relacionados que pueden estar involucrados en solicitudes simultáneas, consulte [Fuentes de eventos compatibles](#).

## 5.4 ¿Qué sucede si las instancias de función no se han ejecutado durante mucho tiempo?

Si una función no se ha ejecutado durante un período de tiempo, se liberarán todas las instancias relacionadas con la función.

## 5.5 ¿Cómo puedo acelerar el acceso inicial a una función?

C# y Go soportan una velocidad de inicio más baja que otros idiomas debido a problemas de mecanismo. Puede utilizar los siguientes métodos para acelerar el acceso inicial a una función:

- Asigne más memoria a la función.
- Simplifique el código de función, por ejemplo, elimine paquetes de dependencias innecesarios.
- Al usar C# en los escenarios no simultáneos, también puede:  
Cree un activador temporizador de un minuto para asegurarse de que haya al menos una instancia activa.

## 5.6 ¿Cómo sé la memoria real utilizada para la ejecución de funciones?

La información devuelta sobre una función contiene la memoria máxima consumida. Para obtener más información, consulte las [API de SDK](#) in la *Guía de desarrollador de FunctionGraph*. Alternativamente, compruebe el uso de memoria en el resultado de ejecución.

## 5.7 ¿Por qué mi primera solicitud es lenta?

Las funciones se inician en frío. Si se realiza una inicialización o una operación larga durante la primera ejecución de la función, la primera solicitud se retrasará. Sin embargo, las solicitudes posteriores antes de la eliminación del contenedor serán más rápidas. Si no hay ninguna solicitud en un minuto, el contenedor se eliminará.

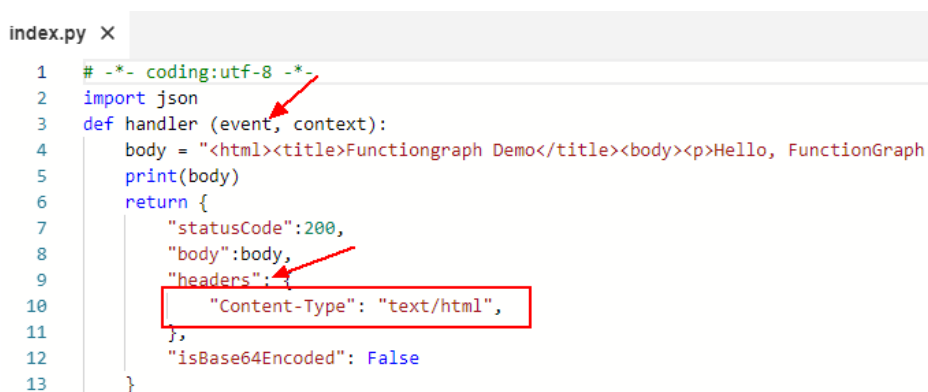
## 5.8 ¿Qué hago si ocurre un error al invocar a una API?

Rectifique el error haciendo referencia a [Códigos de error](#). Si la falla persiste, póngase en contacto con el servicio de asistencia técnica.

## 5.9 ¿Cómo leo el encabezado de solicitud de una función?

El primer parámetro en el controlador de función contiene el encabezado de solicitud. Puede imprimir el resultado de la ejecución de la función para obtener los campos requeridos.

Como se muestra en la siguiente figura, **event** es el primer parámetro en el controlador de función, y **headers** es el encabezado de solicitud.



```

index.py x
1  # -*- coding:utf-8 -*-
2  import json
3  def handler (event, context):
4      body = "<html><title>Functiongraph Demo</title><body><p>Hello, FunctionGraph!
5      print(body)
6      return {
7          "statusCode":200,
8          "body":body,
9          "headers": {
10             "Content-Type": "text/html",
11         },
12         "isBase64Encoded": False
13     }

```

## 5.10 ¿Puede una función invocar otra función?

Sí. Una función puede invocar otra función usando una API o un SDK en la misma región. Para la invocación entre regiones, la persona que invoca debe tener el acceso público.

### Uso de una API

Para obtener más información, consulte el [Ejemplo 4: Uso de una función para invocar una subfunción](#).

### Uso de un SDK (Python)

1. Cree una función llamada **FuncB** que será invocada.

Cree la función desde cero con el tiempo de ejecución de Python 3.6 y el siguiente código:

```

# -*- coding:utf-8 -*-
import json
def handler (event, context):
    # Prints input parameters to verify the invocation result.
    logger = context.getLogger()
    logger.info(event)
    return {
        "statusCode": 200,
        "isBase64Encoded": False,
        "body": json.dumps(event),
        "headers": {
            "Content-Type": "application/json"
        }
    }

```

2. Cree una función invocadora llamada **FuncA**.

1. Cree la función desde cero con el tiempo de ejecución de Python 3.6 y el siguiente código. Reemplace *<Your Function Region>* y *<Your Function URN>* con la región y URN de **FuncB**.

```

# -*- coding:utf-8 -*-
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkfunctiongraph.v2.region.functiongraph_region import
FunctionGraphRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkfunctiongraph.v2 import *
def handler (event, context):
    logger = context.getLogger()
    ak = context.getAccessKey()
    sk = context.getSecretKey()

    credentials = BasicCredentials(ak, sk)
    client = FunctionGraphClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(FunctionGraphRegion.value_of("<Your Function Region>")) \
        .build()

    try:
        request = InvokeFunctionRequest()
        request.function_urn = "<Your Function URN>"
        # Input parameters.
        request.body = '{"input_key':'input_value'}"
        # Response body format. v0: text; v1: JSON.
        request.x_cff_request_version = "v1"
        response = client.invoke_function(request)
        logger.info(response)
    except exceptions.ClientRequestException as e:
        logger.error(e.status_code)
        logger.error(e.request_id)
        logger.error(e.error_code)
        logger.error(e.error_msg)
    return {
        "statusCode": 200,
        "isBase64Encoded": False,
        "body": "",
        "headers": {
            "Content-Type": "application/json"
        }
    }

```

- a. Configurar las dependencias para la función.

Configure las dependencias **huawei-cloud-sdk-core-python-3.x** y **huaweicloud-sdk-python-v3**, que se pueden descargar desde [SDK](#). Para obtener más información sobre cómo crear una dependencia, vea [¿Cómo creo dependencias de función?](#)

- b. Conceder permisos a la función.

Conceda el permiso **FunctionGraph CommonOperation** a la función creando una delegación para que la función pueda invocar otras funciones. Para obtener más información, consulte la sección [Creación de una delegación](#).

3. Comprobar el resultado de la ejecución.

Configure un evento de prueba en blanco para **FuncA** y haga clic en **Test**. Se muestra un resultado similar al siguiente.

**Figura 5-1** Consulta del resultado de la ejecución

```

2022-07-12T12:11:01Z Start invoke request '1b588617-ab5e-4bbc-bb9f-913b4eb21b97', version: latest
2022-07-12T12:11:02Z 1b588617-ab5e-4bbc-bb9f-913b4eb21b97 INFO {"request_id": "ee52f806-fad2-443a-9527-f5451106510b", "result": {"statusCode": 200, "isBase64Encoded": false, "body": ""}, {"input_key": "input_value"}}, {"headers": {"Content-Type": "application/json"}, "status": 200}
2022-07-12T12:11:02Z Finish invoke request '1b588617-ab5e-4bbc-bb9f-913b4eb21b97', duration: 1558.074ms, billing duration: 1559ms, memory used: 35.035MB, billing memory: 128MB

```

Vea los logs de ejecución de **FuncB's** generados en el mismo período.

```

2022-07-12T12:11:02Z Start invoke request 'ee52f806-fad2-443a-9527-f5451106510b', version: latest
2022-07-12T12:11:02Z ee52f806-fad2-443a-9527-f5451106510b {"input_key": "input_value"}
2022-07-12T12:11:02Z Finish invoke request 'ee52f806-fad2-443a-9527-f5451106510b', duration: 36.095ms, billing duration: 37ms, memory used: 20.754MB, billing memory: 1

```

## 5.11 ¿Se puede invocar la interfaz de ejecución síncrona en una red privada?

Sí, la interfaz se invoca en una red privada de forma predeterminada. Para la invocación entre regiones, habilite el acceso a la red pública.

## 5.12 ¿Por qué una función utiliza más memoria de la estimada e incluso activa la alarma de falta de memoria?

1. El análisis de eventos y la caché consumen memoria adicional durante la invocación de la función.
2. Una vez completada la invocación, la memoria recuperada a menudo se coloca en el grupo interno en lugar de volver al sistema operativo, lo que resulta en un alto uso de memoria. Esto es más evidente en el caso de una alta concurrencia.

## 5.13 ¿Cómo puedo comprobar el uso de memoria cuando veo "runtime memory limit exceeded"?

Compruebe la memoria utilizada en la respuesta.

**Figura 5-2** Comprobación de la memoria utilizada

```
2022-07-21T07:10:22Z Start invoke request '696ca9a9-22d9-4602-a354-decf8c99aac3',  
version: latest  
2022-07-21T07:10:23Z Finish invoke request '696ca9a9-22d9-4602-a354-  
decf8c99aac3'(invoke Failed:RuntimeMemoryExceedLimit), duration: 1266.043ms, billing  
duration: 1267ms, memory used: 510.953MB, billing memory: 512MB
```

## 5.14 ¿Cómo soluciono los problemas de CrashLoopBackOff?

El mensaje "CrashLoopBackOff: The application inside the container keeps crashing" se muestra cuando se produce un error en la ejecución de una imagen personalizada. En este caso, realice las siguientes operaciones:

1. Analizar las causas.

**Figura 5-3** Consulta del resultado de la ejecución

```
function invocation exception, error: CrashLoopBackOff: The application inside the container keeps crashing:  
Traceback (most recent call last):  
  File "app.py", line 1, in <module>  
    from flask import Flask, request, g  
ModuleNotFoundError: No module named 'flask'
```

2. Verifique la imagen de contenedor haciendo referencia a [Despliegue de una función mediante una imagen de contenedor](#).
3. Compruebe si la imagen utiliza la arquitectura de x86 de Linux. Actualmente, solo se soportan las imágenes de x86 de Linux.

## 5.15 Después de actualizar una imagen con el mismo nombre, las instancias reservadas siguen utilizando la imagen antigua. ¿Qué puedo hacer?

Utilice una etiqueta no más reciente para gestionar las actualizaciones de imágenes y no utilice el mismo nombre de imagen.



# 6 Preguntas frecuentes sobre la configuración de funciones

---

## 6.1 ¿Puedo establecer variables de entorno al crear funciones?

Sí. Establezca variables para pasar dinámicamente la configuración a su código de función y bibliotecas sin cambiar su código. Para obtener más información, consulte la sección [Configuración de variables de entorno](#).

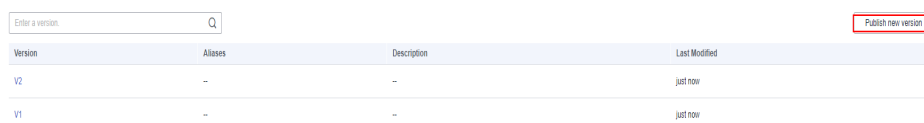
## 6.2 ¿Puedo introducir información confidencial en variables de entorno?

FunctionGraph muestra toda la información introducida en texto sin formato. Por lo tanto, no introduzca información insensible, como contraseñas, cuando defina variables de entorno.

## 6.3 ¿Cómo uso las versiones y los alias de una función HTTP con un activador de APIG para la actualización gris?

1. Cree versiones V1 y V2 basadas en **latest** y cree un alias llamado **alias** para V1 con un peso del 100%, y cree un activador de APIG para **alias**.

**Figura 6-1** Versiones de publicación V1 y V2



Version	Aliases	Description	Last Modified
V2	-	-	just now
V1	-	-	just now

**Figura 6-2** Creación de alias y vinculación de V1 a él

Alias: (1)

Enter an alias:

Name	Version	Description	Modified
alias	V1 (Weight: 100 %)	-	just now

**Figura 6-3** Creación de un activador para alias

< testapig | Aliases: alias

Function Info

Code Monitoring Configuration

Basic Settings

Trigger

Permissions

Disk Mounting

Concurrency

Advanced Settings

Trigger Total triggers: 1

APIG (Subtotal: (0))

**API\_testapigV1** Enabled

Created: Jul 20, 2022 09:58:28 GMT+08:00

URL: https://huaweicloudapis.com/testapig-V1

API Group: group\_test

Environment: RELEASE

Security Authentication: IAM

Method: ANY

Path: /testapig-V1

Timeout: 5000 ms

**NOTA**

Puede crear activadores de APIG para una versión o alias de función. De forma predeterminada, el nombre de un activador es **API\_{Function name}{Version}**, y la ruta de solicitud es **/{Function name}{Version}**.

Un URL del activador de APIG solo se usará para un alias o su versión coincidente. Por ejemplo, supongamos que V1 se ha vinculado a **alias**. Si crea un activador con el nombre predeterminado **API\_testapigV1** para V1 y luego crea otro activador con el mismo nombre y URL para **alias**, el activador **API\_testapigV1** no se mostrará en la lista de activadores de V1.

- Vuelva a la página de ficha **Alias** de **latest** y haga clic en **Edit** en la fila de **alias** y establezca **Additional Version** en **V2** con un peso personalizado. Esto es para la actualización gris de V1 a V2.

**NOTA**

Las ponderaciones indican el porcentaje de datos recibidos que se asignarán a las versiones correspondientes. Puede establecer un peso que cumpla con sus requisitos de servicio.

**Figura 6-4** Edición de un alias

★ Alias ?

Enter 1 to 63 characters, starting with a letter and ending with a letter or digit. Only letters, digits, hyphens (-), and underscores (\_) are allowed.

★ Version  Weight 90 %

Traffic Shifting

Requests of the major version will be forwarded to an additional version according to the specified weight. [Learn more](#)

Additional Version  ★ Weight  %

Description

0/512

3. Cree la versión V3, haga clic en **Edit** en la fila de **alias** en la página de fichas **Alias** y cambie **Version** a **V2** y **Additional Version** a **V3** con una ponderación personalizada. Esto es para la actualización gris de V2 a V3.

**Figura 6-5** Creación de V3

Version	Aliases	Description	Last Modified
V3	-	-	just now
V2	-	-	3 minutes ago
V1	-	-	3 minutes ago

**Figura 6-6** Edición de un alias

★ Alias ?

Enter 1 to 63 characters, starting with a letter and ending with a letter or digit. Only letters, digits, hyphens (-), and underscores (\_) are allowed.

★ Version  Weight 10 %

Traffic Shifting

Requests of the major version will be forwarded to an additional version according to the specified weight. [Learn more](#)

Additional Version  ★ Weight  %

Description

0/512

**AVISO**

Después de publicar una versión basada en **latest**, puede publicar más versiones solo después de realizar un cambio de configuración o código en la función.

Los alias enlazados con activadores no se pueden eliminar.

---

# 7 Preguntas frecuentes sobre el acceso a recursos externos

## 7.1 ¿Cómo accede una función a la base de datos de MySQL?

Realice las siguientes operaciones:

1. Compruebe si la base de datos de MySQL está desplegada en una VPC.
  - Si la base de datos de MySQL se despliega en una VPC, configure la misma VPC y subred que la base de datos de MySQL para la función haciendo referencia a la [Configuración de acceso de VPC](#).
  - Si la base de datos de MySQL se construye en una red pública, obtenga su dirección IP pública.
2. Compilar código para conectar una función a la base de datos de MySQL.
  - a. Cree una función y agregue la dependencia **pymysql** a la función en la página de pestañas **Code**.

Para obtener más información sobre cómo cargar la dependencia **pymysql**, consulte la [Creación de una dependencia](#).

### NOTA

FunctionGraph proporciona **pymysql** de la dependencia pública. Por lo tanto, puede usarlo directamente.

- b. Edite el siguiente código para conectar la función a la base de datos de MySQL:

```
# -*- coding:utf-8 -*-
import pymysql.cursors

def handler(event, context):
    # Connect to the database
    connection = pymysql.connect(host='host_ip',
                                user='user',
                                password='passwd',
                                db='db',
                                charset='utf8mb4',
                                cursorclass=pymysql.cursors.DictCursor)
```

```

try:
    with connection.cursor() as cursor:
        # Create a new record
        sql = "INSERT INTO `users` (`email`, `password`)
VALUES (%s, %s)"
        cursor.execute(sql, ('webmaster@python.org', 'very-
secret'))

        # connection is not autocommit by default. So you must
commit to save
        # your changes.
        connection.commit()

    with connection.cursor() as cursor:
        # Read a single record
        sql = "SELECT `id`, `password` FROM `users` WHERE
`email`=%s"
        cursor.execute(sql, ('webmaster@python.org',))
        result = cursor.fetchone()
        print(result)
finally:
    connection.close()
    output = '^_^'
return output

```

#### NOTA

Si la función necesita acceder a las API de RDS, [crear una delegación](#) y conceder los permisos requeridos.

## 7.2 ¿Cómo accede una función a Redis?

Realice las siguientes operaciones:

1. Comprobar si la instancia de Redis se despliega en una VPC.
  - Si la instancia de Redis se implementa en una VPC, configure la misma VPC y subred que la instancia de Redis para la función haciendo referencia a la [Configuración de acceso de VPC](#).
  - Si la instancia de Redis se construye en una red pública, obtenga su dirección IP pública.
2. Compilar el código para conectar una función a la instancia de Redis.

FunctionGraph ha integrado la biblioteca de terceros [redis-py](#) en sus tiempos de ejecución de Python 2.7 y Python 3.6. Por lo tanto, no es necesario descargar ninguna otra biblioteca de Redis.

```

# -*- coding:utf-8 -*-
import redis
def handler (event, context):
    r = redis.StrictRedis(host="host_ip",password="passwd",port=6379)
    print(str(r.get("hostname")))
    return "^_^"

```

 **NOTA**

- Si la función no puede acceder a la instancia de Redis en una red pública, realice las siguientes operaciones:
  - Modificar el archivo **redis.conf** para permitir el acceso desde cualquier dirección IP.
  - Establecer una contraseña para acceder a la instancia de Redis en el archivo **redis.conf**.
  - Deshabilitar el firewall.
- Si la función necesita acceder a las API de DCS, **crear una delegación** y conceder los permisos requeridos.

## 7.3 ¿Cómo configuro el acceso a la red externa?

De forma predeterminada, las funciones desplegadas en una VPC están aisladas de Internet. Si una función necesita acceder a redes internas y externas, agregue un gateway de NAT pública para la VPC.

### Requisitos previos

1. Ha creado una VPC y una subred de acuerdo con [Creación de una VPC](#).
2. Ha obtenido una dirección IP elástica según [la asignación de un EIP](#).

### Procedimiento de creación de un gateway de NAT pública

**Paso 1** Inicie sesión en la consola de NAT Gateway y haga clic en **Comprar NAT pública Gateway**.

**Paso 2** En la página mostrada, introduzca la información del gateway, seleccione una VPC y una subred (por ejemplo, **vpc-01**), y confirme y envíe la configuración para comprar NAT Gateway. Para obtener más información, consulte [Comprar un NAT Gateway público](#).

**Paso 3** Haga clic en el nombre de NAT Gateway público. En la página de detalles que se muestra, haga clic en [Add an SNAT Rule](#) y haga clic en **OK**.

----Fin

# 8 Otras preguntas frecuentes

---

## 8.1 ¿Cómo puedo ver las reglas de alarma configuradas para una función?

Inicie sesión en la consola de Cloud Eye y vea las reglas de alarma.

## 8.2 ¿FunctionGraph admite la descompilación de ZIP durante la transcodificación de vídeo?

No. Por favor, descompile sus archivos antes de subirlos.

## 8.3 ¿Qué es una aplicación de FunctionGraph?

Una aplicación actúa como una carpeta. Después de crear una función, se categoriza automáticamente en la aplicación **default** y no se puede cambiar a otras aplicaciones. En el futuro, las funciones se gestionarán por etiqueta para una mejor experiencia.

## 8.4 ¿Necesito pagar por el tiempo de inicio en frío?

No. El tiempo de inicio en frío no se medirá y no hace falta pagar por ello.

## 8.5 ¿Por qué veo un mensaje que indica que mi cuenta fue suspendida al crear una función?

Su cuenta está en mora.



## **8.6 ¿Se facturarán las solicitudes de todas mis funciones en diferentes regiones?**

Sí.

# 9 Migración de FunctionGraph de V1 a V2

---

## 9.1 ¿Qué problemas de compatibilidad existen durante la migración?

### 1. Diferencia de `args`

V1:

```
args = parser.parse_args()
```

Después de la migración a V2, será:

```
args = parser.parse_args(args=[])
```

El `sys.argv` del tiempo de ejecución de Python varía entre las dos versiones.

Para V2, es `['/home/snuser/runtime/python3.6/server.py', '127.0.0.1:31536', '/opt/function/code']`, cuyos dos últimos parámetros no están disponibles para V1.

### 2. Diferencia de `asyncio`

V1:

```
loop = asyncio.get_event_loop()
loop.run_until_complete(func(arg1, arg2))
loop.close()
```

Después de la migración a V2, será:

```
loop_tmp = asyncio.new_event_loop()
asyncio.set_event_loop(loop_tmp)
loop = asyncio.get_event_loop()
loop.run_until_complete(func(arg1, arg2))
loop.close()
```

En V1, `asyncio.get_event_loop()` obtiene el bucle de evento actual del subproceso del SO (subproceso principal). Lanzará `RuntimeError` en V2, porque el tiempo de ejecución de Python de V2 no ejecuta funciones en el subproceso principal.

Para utilizar `asyncio` en V2, cree un bucle de eventos.