
Artículo

[Pierre-Yves Duq...](#) · 5 ene, 2022 Lectura de 1 min

[Open Exchange](#)

Python Gateway. Parte II: Instalación

Esta serie de artículos describe el uso del [Python Gateway](#) para InterSystems. Ejecuta el código de Python y mucho más desde InterSystems IRIS. Este proyecto te trae toda la potencia de Python directamente a tu entorno InterSystems IRIS:

- Ejecuta cualquier código de Python
- Transfiere datos de forma transparente desde InterSystems IRIS a Python
- Crea procesos empresariales de interoperabilidad inteligentes, con el Adaptador de interoperabilidad de Python
- Guarda, examina, modifica y restaura el contexto de Python desde InterSystems IRIS

Otros artículos

El plan para la serie hasta el momento es el siguiente (sujeto a cambios):

- [Parte I: Resumen, entorno e introducción](#)
- Parte II: Instalación y resolución de problemas <-- estamos aquí
- Parte III: Funcionalidades básicas
- Parte IV: Adaptador de interoperabilidad
- Parte V: Cómo ejecutar una función
- Parte VI: Jupyter Notebook
- Parte VII: Soporte a Jupyter

Instalación

Este artículo te ayudará a instalar el Python Gateway. Hay varias formas de instalarlo y utilizarlo:

- Host
 - Windows
 - Linux
 - Mac
- Docker
 - Extrae la imagen prediseñada
 - Crea tu propia imagen

Independientemente del método que utilices para la instalación, necesitarás el código. El único sitio donde se puede descargar es en la [página de actualizaciones](#), que contiene versiones estables probadas. Coge siempre la más reciente. No clones/descargues el repositorio, descarga la última versión.

Host

Si instalas Python Gateway en el sistema operativo del host, primero (independientemente del sistema operativo) debes instalar Python.

1. [Instala Python 3.6.7 64 bits](#). Se recomienda instalar Python en la ubicación predeterminada.
2. Instala el módulo dill: `pip install dill`.
3. Carga el código ObjectScript (es decir, ejecutar `$system.OBJ.ImportDir("C:/InterSystems/Repos/Python/isc/py/", "*.cls", "c", 1)`) en el namespace Production (Ensemble-enabled). En caso de que quieras habilitar Production en la llamada del namespace: escribe `##class(%EnsembleMgr).EnableNamespace($Namespace, 1)`.
4. Sitúa [callout DLL/SO/DYLIB](#) en la carpeta bin en la que está instalado InterSystems IRIS. El archivo de la librería se debe colocar en una ruta que devuelve `write ##class(isc.py.Callout).GetLib()`.

Windows

5. Revisa que tu variable de entorno PYTHONHOME señala a Python 3.6.7.
6. Verifica que tu variable de entorno SYSTEM PATH incluye la variable PYTHONHOME (o el directorio al que señala).

Linux

5. Comprueba que tu variable de entorno SYSTEM PATH incluye /usr/lib y /usr/lib/x86_64-linux-gnu, preferiblemente al principio. Utiliza el archivo /etc/environment para establecer las variables de entorno.
6. En caso de error, consulta la sección Troubleshooting (Resolución de problemas) undefined symbol: PyTrueStruct y especifica la propiedad PythonLib.

Mac

5. Por el momento, solo está soportada la versión python 3.6.7 de [Python.org](#). Comprueba la variable PATH.

Si modificaste las variables de entorno, reinicia tu producto de InterSystems.

Docker

Hay muchas ventajas en utilizar contenedores:

- Portabilidad
- Eficiencia
- Aislamiento
- Diseño ligero
- Inmutabilidad

Consulta [esta](#) serie de artículos para obtener más información sobre Docker.

Todas las creaciones de Docker para Python Gateway se basan en contenedores 2019.1.

Cómo extraer la imagen prediseñada

Ejecuta: `docker run -d -p 52773:52773 --name irispy intersystemscommunity/irispy:latest` para extraer y ejecutar Python Gateway sobre InterSystems IRIS Community Edition. Eso es todo.

Crea tu propia imagen

Para crear la imagen de Docker, ejecuta en el repositorio raíz: `<code>docker build --force-rm --tag intersystemscommunity/irispy:latest .`

De forma predeterminada, la imagen se crea a partir de la imagen intersystems/iris:2019.1.0.510.0-1. Sin embargo, se puede cambiar si se proporciona la variable IMAGE.

Si quieres crearla desde InterSystems IRIS Community Edition, ejecuta: `docker build --build-arg IMAGE=store/intersystems/iris:2019.1.0.510.0-community --force-rm --tag intersystemscommunity/irispy:latest`.

Después, puedes ejecutar tu imagen de Docker:

```
docker run -d \
  -p 52773:52773 \
  -v /<HOST-DIR-WITH-iris.key>:/mount \
  --name irispy \
  intersystemscommunity/irispy:latest \
  --key /mount/iris.key
```

Si estás ejecutando una imagen basada en InterSystems IRIS Community Edition, puedes omitir el proporcionar la clave.

Notas de Docker

Información sobre Docker:

- El proceso de prueba `isc.py.test.Process` guarda la **image artifact** en el directorio temporal. Tal vez quieras cambiar esa ruta a un directorio que ya esté instalado. Para ello, edita la nota correspondiente a la llamada de la Correlation Matrix: Graph, donde se especifica una ruta de archivo válida para la función `f.savefig`.
- Para acceder al terminal, ejecuta: `docker exec -it irispy sh`.
- Accede al Multiprocesamiento simétrico (SMP) con el usuario/contraseña `SuperUser/SYS` o `Admin/SYS`.
- Para detener el contenedor, ejecuta: `docker stop irispy && docker rm --force irispy`.

Cómo validar tu instalación

Después de que hayas instalado Python Gateway es hora de comprobar que funciona. Ejecuta este código:

```
set sc = ##class(isc.py.Callout).Setup()
set sc = ##class(isc.py.Main).SimpleString("x='HELLO'", "x", , .var)
write var
```

La salida debería ser HELLO - valor de la variable x de Python. En caso de que el estado de respuesta/devolución sea un error o que var esté vacío, consulta la [sección de resolución de problemas del archivo Readme](#).

Conclusión

La instalación de Python Gateway es rápida y sencilla. Si estás familiarizado con Docker, coge la última imagen. Si no, hay paquetes para los principales sistemas operativos de InterSystems IRIS: Windows, Linux y Mac. En el siguiente artículo comenzaremos a utilizar Python Gateway.

Guía ilustrada

También hay una guía de instalación ilustrada en el grupo de usuarios de ML Toolkit. El grupo de usuarios de ML Toolkit es un repositorio privado de GitHub que se estableció como parte del GitHub corporativo de InterSystems. Está dirigido a los usuarios externos que instalan, aprenden o ya utilizan los componentes del ML Toolkit. Si quieres unirte al grupo de usuarios de ML Toolkit, envía un email a la dirección MLToolkit@intersystems.com e indica los siguientes datos (son necesarios para que los miembros del grupo te puedan conocer e identificar durante los debates):

- Nombre de usuario de GitHub
- Nombre completo (tu nombre seguido de tu apellido en caracteres latinos)
- Organización (para la que trabajas, estudias, o tu oficina en casa)
- Cargo (tu cargo real en tu empresa, o "Estudiante", o "Independiente")
- País (en el que resides)

[#Inteligencia Artificial](#) [#Lenguajes](#) [#Machine learning](#) [#Python](#) [#InterSystems IRIS](#)
[Ir a la aplicación en InterSystems Open Exchange](#)

URL de fuente: <https://es.community.intersystems.com/post/python-gateway-parte-ii-instalaci%C3%B3n>