Artículo
Eduardo Anglada · 7 sep, 2021 Lectura de 3 min

Open Exchange

ObjectScript sobre ODBC

```
Este es un ejemplo de código que funciona en IRIS 2020.1 y en Caché 2018.1.3
No se mantendrá sincronizado con las nuevas versiones.
Y NO cuenta con el servicio de soporte de InterSystems.
```

De vez en cuando, puedes encontrarte una situación en la que, por diferentes razones, ODBC es la única opción para acceder a un sistema remoto. Lo cual es suficiente mientras necesites examinar o cambiar tablas. Pero no puedes ejecutar directamente algunos comandos o cambiar algunos *globals*.

En este artículo vamos a ver 3 procedimientos SQL que permiten acceder a los globals usando ODBC.

1. SQLprocedure **Ping()** devuelve Server::Namespace::$ZV. Permite verificar la conexión.
2. SQLprocedure **Xcmd(<commandline>,<resultvar>)** ejecuta la línea de comandos que envías y devuelve el resultado en <resultvar>.
3. SQLprocedure **Gset(<global>,<subscript>,<value>,<$data>)** te permite establecer o eliminar un global.

- <global> es un nodo del global perteneciente al *namespace* del servidor remoto. Hay que incluir el símbolo inicial, por ejemplo, '^MyGlobal' .
- <subscript> representa el subíndice completo incluyendo los paréntesis, por ejemplo: '(1,3,"something",3)' .
- <value> es el valor que queremos establecer.
- <$data> controla si se establece el Nodo global o se ejecuta un ZKILL en él, por ejemplo:
    - 1, 11 para establecer el valor.
    - 0,10 para borrar (ZKILL) el valor.

El procedimiento Gset está diseñada para hacer uso del Global Scanning descrito en otro artículo. Gset permite copiar globals a través de cualquier conexión ODBC.

Instalación:

- En el sistema remoto necesitas la clase que se encuentra en OpenExchange.
- Además necesitas definir los procedimientos como "Linked SQL Procedures", para ello emplea este asistente:
  SMP>System>SQL> Wizards>Link Procedure
  En los ejemplos se usa el namespace rccEX.
- Si quieres ejecutar la copia de globals también necesitas instalar la clase Global Scanning desde OEX

Ejemplos:

```
USER>do $system.SQL.Shell()
SQL Command Line Shell
[SQL]USER>>select rccEX.Ping()
Expression_1
cemper9::CACHE::IRIS for Windows (x86-64) 2020.1 (Build 215U) Mon Mar 30 2020 20:14:3
3 EDT
```

Verifica la existencia del global ^rcc

```
[SQL]USER>>select rccEX.Xcmd('set %y=$d(^rcc)','%y')
ok: 10
```

Establece algún valor en ^rcc4(1,"demo",3,4)

```
[SQL]USER>>select rccEX.Gset('^rcc4','(1,"demo",3,4)','this is a demo',1)
Expression_1
ok: ^rcc4(1,"demo",3,4)
```

Haz una copia de ^rcc2 a ^rcc4.

Primero vemos el contenido de ^rcc2:

```
USER>>select reference,value,"$DATA" from rcc_G.Scan where rcc_G.scan(^rcc2,4)=1
Reference       Value   $Data
 ^rcc2                  10
(1)             1       1
(2)             2       11
(2,"xx")                10
(2,"xx",1)      "XX1"   1
(2,"xx",10)     "XX10"  1
(2,"xx",4)      "XX4"   1
(2,"xx",7)      "XX7"   1
(3)             3       1
(4)             4       11
(4,"xx")                10
(4,"xx",1)      "XX1"   1
(4,"xx",10)     "XX10"  1
(4,"xx",4)      "XX4"   1
(4,"xx",7)      "XX7"   1
(5)             5       1
16 Rows(s) Affected
```

Ahora ejecuta la copia del contenido de ^rcc2 en ^rcc4:

```
[SQL]USER>>select rccEX.Gset('^rcc4',reference,value,"$DATA")  from rcc_G.Scan where
rcc_G.scan('^rcc2',4)=1
Expression_1
ok: ^rcc4
ok: ^rcc4(1)
ok: ^rcc4(2)
ok: ^rcc4(2,"xx")
ok: ^rcc4(2,"xx",1)
ok: ^rcc4(2,"xx",10)
ok: ^rcc4(2,"xx",4)
ok: ^rcc4(2,"xx",7)
ok: ^rcc4(3)
ok: ^rcc4(4)
ok: ^rcc4(4,"xx")
ok: ^rcc4(4,"xx",1)
ok: ^rcc4(4,"xx",10)
ok: ^rcc4(4,"xx",4)
ok: ^rcc4(4,"xx",7)
ok: ^rcc4(5)
 16 Rows(s) Affected
```

Un agradecimiento especial para @Anna Golitsyna por inspirarme a publicar esto.

#Globals #ObjectScript #ODBC #SQL #Caché #Ensemble #InterSystems IRIS
Ir a la aplicación en InterSystems Open Exchange

URL de fuente:https://es.community.intersystems.com/post/objectscript-sobre-odbc