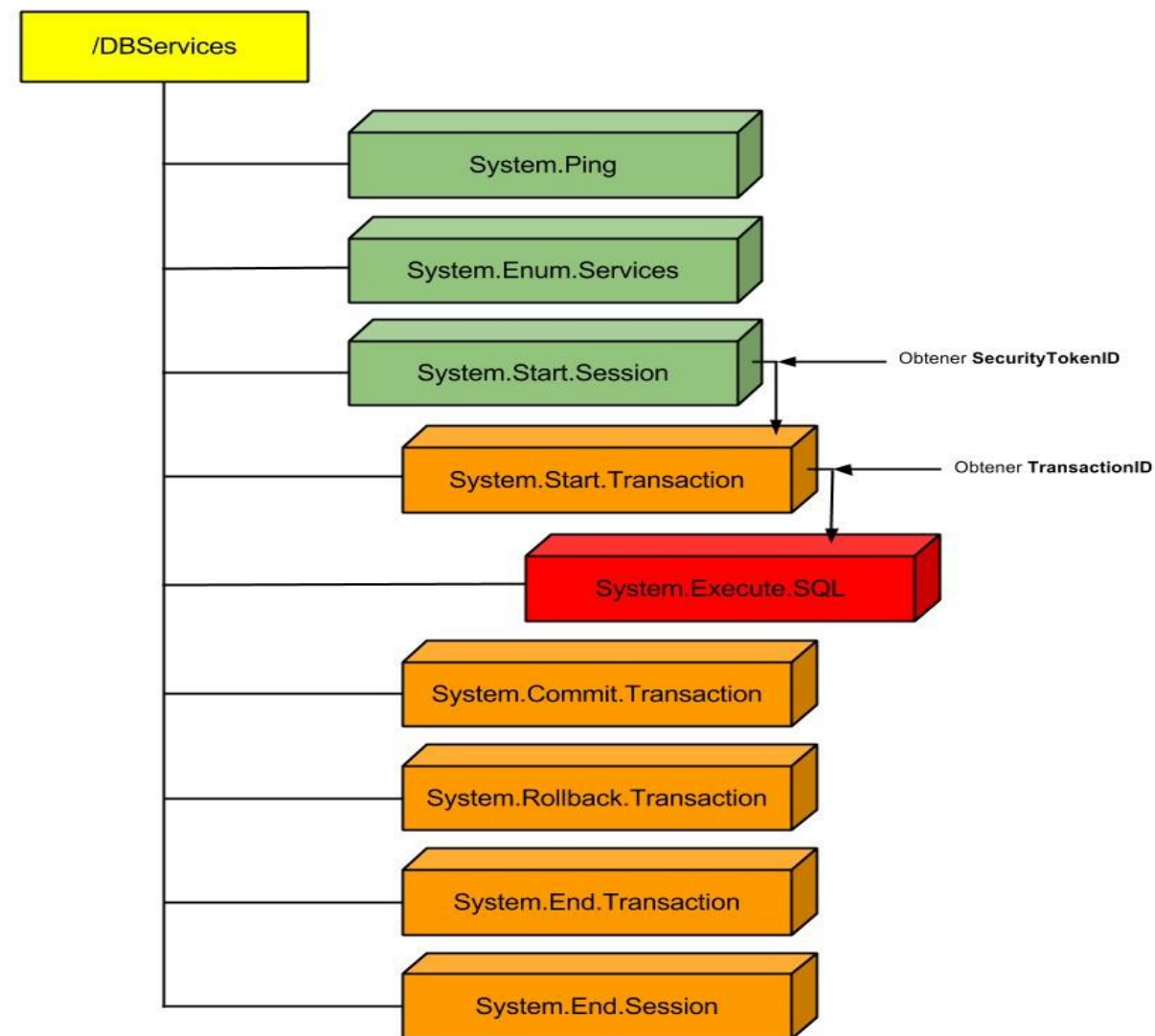
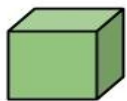


Servicios Disponibles por defecto en el manager DBServices o el contexto /DBServices

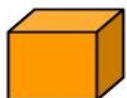
Todos los servicios son archivos .jar se encuentran ubicados en carpetas que tienen por nombre el nombre de su servicio la ruta de los mismos es **distro-bin/ServicesDaemon/Managers/DBServicesManager/DBServices/**.



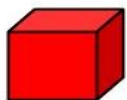
Leyenda



NO REQUIEREN de un **SecurityTokenID** para poder ser llamados



REQUIEREN de un **SecurityTokenID** para poder ser llamados



REQUIEREN de un **SecurityTokenID** y de un **TransactionID** para poder ser llamados

| | |
|-----------------------------------|---|
| Nombre del Servicio: | System.Enum.Services |
| Descripción: | Permite listar los servicios y parámetros de entrada, véalo como el equivalente a un dir en la línea de comando windows o ls en la familia de sistemas operativos unix/linux, para saber que archivos o directorios están dentro de una carpeta específica, de igual manera este servicio realiza la misma función. |
| Parámetros de entrada requeridos: | Ninguno |
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion |

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|---------------------------|------------------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Ejemplo de una llamada al servicio

ServiceName=System.Enum.Services
ResponseFormat=JSON
ResponseFormatVersion=1.0

Ejemplo de la salida en respuesta a la anterior llamada

Por motivos de longitud de la respuesta del servicio, vea el archivo localizado en la ruta dentro de la distribución binaria distro-bin/documentation/test/firebird/JSON/test01_service_output_case1.xml, en realidad es un archivo con sintaxis JSON la extensión es .XML por mantener la uniformidad de otros archivos de documentación.

Notas sobre el servicio

1. El servicio lista únicamente los servicios dentro del manager DBServices o el contexto /DBServices, otros managers deberían tener su propio servicio System.Enum.Services.
2. Si el servicio es llamado sin los parámetros opcionales **ResponseFormat** y **ResponseFormatVersion** se usara los valores por defecto dados en el archivo de configuración **DBServicesManager.conf** ubicado en el ruta **distro-bin/ServicesDaemon/Managers/DBServicesManager/**, específicamente los parámetros de configuración llamados **Default_Response_Format="XML-DataPacket"** **Default_Response_Format_Version="1.1"**.

| | |
|----------------------|---|
| Nombre del Servicio: | System.Start.Session |
| Descripción: | Permite iniciar sesión en una base de datos, también puede ser visto como iniciar una |

| | | | | | | | | | | | | | |
|---|---|-----------------------|---|-------------------------------|---|-----------------------|------------|---------------------------|------------|-------------|------------|------------|------------|
| | conexión a la base de datos, validando primero si el usuario y contraseña es válido para la misma, según se indique en los parámetros adicionales definidos para la conexión. | | | | | | | | | | | | |
| Parámetros de entrada requeridos: | DBConnection Paramétros adicionales | | | | | | | | | | | | |
| Descripción de los parámetros requeridos | | | | | | | | | | | | | |
| <table><tr><td>DBConnection</td></tr><tr><td>Nombre de la base de datos definido en el archivo DBServicesManager.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/, en la sección <DBConnections>, cada base de datos tiene una entrada DBConnection, específicamente el atributo Name.</td></tr><tr><td>Parámetros Adicionales</td></tr><tr><td>Definidos en el archivo SystemStartSession.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/DBServices/System.Start.Session /, en la sección <DBConnections>, cada base de datos tiene una entrada DBConnection, específicamente el atributo Name y la sección <InputParams>, dentro de esa sección se definen los parámetros a ser enviados para cada una de las base de datos configuradas.</td></tr></table> | | DBConnection | Nombre de la base de datos definido en el archivo DBServicesManager.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/ , en la sección <DBConnections> , cada base de datos tiene una entrada DBConnection , específicamente el atributo Name . | Parámetros Adicionales | Definidos en el archivo SystemStartSession.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/DBServices/System.Start.Session / , en la sección <DBConnections> , cada base de datos tiene una entrada DBConnection , específicamente el atributo Name y la sección <InputParams> , dentro de esa sección se definen los parámetros a ser enviados para cada una de las base de datos configuradas. | | | | | | | | |
| DBConnection | | | | | | | | | | | | | |
| Nombre de la base de datos definido en el archivo DBServicesManager.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/ , en la sección <DBConnections> , cada base de datos tiene una entrada DBConnection , específicamente el atributo Name . | | | | | | | | | | | | | |
| Parámetros Adicionales | | | | | | | | | | | | | |
| Definidos en el archivo SystemStartSession.conf que es un archivo con sintaxis XML, ubicado en la ruta distro-bin/ServicesDaemon/Managers/DBServicesManager/DBServices/System.Start.Session / , en la sección <DBConnections> , cada base de datos tiene una entrada DBConnection , específicamente el atributo Name y la sección <InputParams> , dentro de esa sección se definen los parámetros a ser enviados para cada una de las base de datos configuradas. | | | | | | | | | | | | | |
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion | | | | | | | | | | | | |
| Posibles valores para los parámetros opcionales | | | | | | | | | | | | | |
| <table><tr><td>ResponseFormat</td><td>ResponseFormatVersion</td></tr><tr><td>XML-DATAPACKET</td><td>1.0</td></tr><tr><td>XML-DATAPACKET</td><td>1.1</td></tr><tr><td>JAVA-XML-WEBROWSET</td><td>1.0</td></tr><tr><td>JSON</td><td>1.0</td></tr><tr><td>CSV</td><td>1.0</td></tr></table> | | ResponseFormat | ResponseFormatVersion | XML-DATAPACKET | 1.0 | XML-DATAPACKET | 1.1 | JAVA-XML-WEBROWSET | 1.0 | JSON | 1.0 | CSV | 1.0 |
| ResponseFormat | ResponseFormatVersion | | | | | | | | | | | | |
| XML-DATAPACKET | 1.0 | | | | | | | | | | | | |
| XML-DATAPACKET | 1.1 | | | | | | | | | | | | |
| JAVA-XML-WEBROWSET | 1.0 | | | | | | | | | | | | |
| JSON | 1.0 | | | | | | | | | | | | |
| CSV | 1.0 | | | | | | | | | | | | |
| Ejemplo de una llamada al servicio | | | | | | | | | | | | | |
| ServiceName=System.Start.Session ResponseFormat=JSON ResponseFormatVersion=1.0 DBConnection=DB1 username=test1 password=123qwerty | | | | | | | | | | | | | |
| Ejemplo de la salida en respuesta a la anterior llamada | | | | | | | | | | | | | |
| [{ "Code" : 1, | | | | | | | | | | | | | |

```
"Description" : "Inicio de sesión exitoso",
"IDGROUP" : 1,
"IDUSER" : 1,
"IDVALID" : 1,
"LASTACCESSDATE" : null,
"LASTACCESSTIME" : null,
"SecurityTokenID" : 5200954554030646366
} ]
```

Notas sobre el servicio

1. Este servicio depende mucho de dos archivos de configuración que trabajan en conjunto para funcionar adecuadamente el **DBServicesManager.conf** y el **SystemStartSession.conf**.
2. El **DBServicesManager.conf** se encarga de definir la conexión a la base de datos como tal, tipo de driver JDBC según el sistema manejador de base de datos, el ip del servidor, el puerto, entre otros valores.
3. El **SystemStartSession.conf** define para cada conexión a base de datos configurada previamente en el **DBServicesManager.conf** el como se validará el usuario y que datos deseas obtener luego de validado.
4. Uno de los valores más importantes retornados por el servicio luego de un inicio de sesión exitoso es el **SecurityTokenID** el cual es un número, ese número es requerido como parámetro de entrada para otros servicios, como por ejemplo **System.Start.Transaction** o el **System.Execute.SQL**.
5. El número es único y solo sirve para la base de datos en la que se inicio sesión, no para ninguna otra base de datos configurada.
6. El número cambia con cada inicio de sesión en el tiempo, no será el mismo número si se inicia sesión hoy en una base de datos, que el que se generará en día de mañana en la misma base de datos, o incluso segundos después por otro cliente.
7. Si quiere acceder a otra base de datos debe llamar de nuevo a este servicio con los parámetros de entrada adecuados según su configuración y en caso de ser exitoso el inicio de sesión obtendrá otro SecurityTokenID distinto.
8. Puede iniciar sesión en tantas base de datos como estén configuradas y por cada una tendrá un SecurityTokenID distinto, puede usarlos todos al mismo tiempo según conveniencia.
9. Cada cliente (session http) puede iniciar solo una vez en cada base de datos.
10. El servicio contra parte de este servicio es el **System.End.Session**.

| | |
|---|---|
| Nombre del Servicio: | System.Start.Transaction |
| Descripción: | Permite iniciar una transacción en una base de datos, también puede ser visto como iniciar una conexión a la base de datos, solo para la base de datos a la que hace referencia el SecurityTokenID . |
| Parámetros de entrada requeridos: | SecurityTokenID |
| Descripción de los parámetros requeridos | |
| SecurityTokenID | |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . | |

| | |
|-----------------------------------|---|
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion |
|-----------------------------------|---|

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|---------------------------|------------------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Ejemplo de una llamada al servicio

ServiceName=System.Start.Transaction
ResponseFormat=JSON
ResponseFormatVersion=1.0
SecurityTokenID=5200954554030646366

Ejemplo de la salida en respuesta a la anterior llamada

```
[ { "Code" : 1,
  "Description" : "Transacción iniciada correctamente en la base de datos de nombre: [DB1]",
  "TransactionID" : 7497785048145135526
} ]
```

Notas sobre el servicio

1. Para poder llamar a este servicio se requiere de **SecurityTokenID**, el cual puede ser obtenido llamando a **System.Start.Transaction**.
2. El servicio puede ser llamado múltiples veces para iniciar varios contextos de transacciones independiente por lo que que las inserciones, eliminaciones, modificaciones solo serán vistas por los otros contextos de transacción solo si son aplicadas llamadas a el servicios **System.Commit.Transaction** y dependiendo del aislamiento configurado por defecto en el manejador de base de datos, ejemplo: committed, uncommitted, consitency entre otros (Depende estrictamente del manejador Firebird, Mysql el comportamiento no del Services Daemon).
3. Con cada llamada se genera un **TransactionID** distinto, el cual internamente es una conexión JDBC a la base de datos, motivados a que java en el JDBC no permite múltiples transacciones por conexión. El modelo de JDBC es una una conexión y una transacción activa, debe ser cuidadoso cuando abra muchas transacciones por que realmente se están abriendo muchas conexiones a la base de datos con las implicaciones de uso de recursos en el servidor.
4. Los servicios contra parte de este servicio son **System.Commit.Transaction**, **System.Rollback.Transaction**, **System.End.Transaction**.

| | |
|----------------------|---|
| Nombre del Servicio: | System.Execute.SQL |
| Descripción: | Permite ejecutar una consulta SQL, solo para la |

| | |
|-----------------------------------|---|
| | base de datos a la que hace referencia el SecurityTokenID , en el contexto de transacción indicado por TransactionID . Este es el servicio más importante desde el punto de vista funcional, ya que permite la interacción real con la base de datos y la información que contiene. |
| Parámetros de entrada requeridos: | SecurityTokenID TransactionID SQL |

Descripción de los parámetros requeridos

| SecurityTokenID |
|--|
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . |
| TransactionID |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. |
| SQL |
| Cadena de caracteres representando la instrucción SQL a ser enviada al sistema manejador de base de datos, ejemplo: Select * From tblUsersDB Where IdUser=1 . |

| | |
|-----------------------------------|---|
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion Commit Parámetros Adicionales |
|-----------------------------------|---|

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|---------------------------|------------------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Descripción de los parámetros opcionales

| Commit |
|---|
| Indica con un 1 si se debe hacer un aplicar (Commit) a la operación sin tener que llamar al servicio System.Commit.Transaction , el no enviar el parámetro implica que no se hará el commit, es lo mismo que enviar un 0. Sin embargo si en el DBServicesManager.conf en la sección de DBConnection de la base de datos que hace referencia el SecurityTokenID específicamente el parámetro Auto_Commit es igual a 1, la operación se aplicará sin necesidad de enviar este parámetro a 1. |

Parámetros Adicionales

Estos parámetros adicionales son enviados cuando en el parámetro SQL se requiere enviar información de tipo BLOB, codificada en base64, o cuando se desea hacer múltiples operaciones al mismo tiempo.

Ejemplo 1 de una llamada al servicio (Consulta Plana)

ServiceName=System.Execute.SQL
SecurityTokenID=5200954554030646366
TransactionID=7497785048145135526
ResponseFormat=JSON
ResponseFormatVersion=1.0
SQL=Select * From tblUsersDB Where IdUser=1

Ejemplo 1 de la salida en respuesta a la anterior llamada (Consulta Plana)

```
[
  [
    {
      "IDUSER": 1,
      "IDGROUP": 1,
      "DISABLED": 0,
      "USERNAME": "test1",
      "PASSWD": "123qwerty",
      "FIRSTNAME": "System user firstname",
      "LASTNAME": "System user lastname",
      "ACCESS_DATE": "2013-08-17",
      "ACCESS_TIME": "12:15:14"
    }
  ],
  []
]
```

Ejemplo 2 de una llamada al servicio (Consulta Parametrizada)

ServiceName=System.Execute.SQL
SecurityTokenID=5200954554030646366
TransactionID=7497785048145135526
ResponseFormat=JSON
ResponseFormatVersion=1.0
SQL=Select * From tblUsersDB Where IdUser=[paramvalue]IdUser[/paramvalue]
IdUser=[array datatype="integer"] [value]1[/value] [value]2[/value] [/array]

Ejemplo 2 de la salida en respuesta a la anterior llamada (Consulta Parametrizada)

```
[
  [
    {
      "IDUSER": 1,
      "IDGROUP": 1,
      "DISABLED": 0,
```

```

"USERNAME": "test1",
"PASSWD": "123qwerty",
"FIRSTNAME": "System user firstname",
"LASTNAME": "System user lastname",
"ACCESS_DATE": "2013-08-17",
"ACCESS_TIME": "12:15:14"
},
{
  "IDUSER": 2,
  "IDGROUP": 1,
  "DISABLED": 0,
  "USERNAME": "test2",
  "PASSWD": "12345678",
  "FIRSTNAME": null,
  "LASTNAME": null,
  "ACCESS_DATE": null,
  "ACCESS_TIME": null
}
],
[]
]

```

Notas sobre el servicio

1. En el ejemplo 2, se observa un parámetro adicional llamado IdUser, este parámetro es definido por la consulta, se realizan dos llamadas a la consulta la primera con el IdUser=1 y la otra IdUser=2, dando como resultado una respuesta con dos tuplas registros.
2. Cada llamada al System.Execute.SQL está protegida por un semáforo asociado a su conexión, por lo que puede tener un cliente que llame en dos hilos de ejecución distintos a este servicio usando el mismo **SecurityTokenID** y **TransactionID**, sin tener que preocuparse por la concurrencia de ambas llamadas, una se ejecutará primero y luego la otra.
3. Si quiere una ejecución simultanea de dos llamadas a este servicio cada llamada deberá tener un **TransactionID** distinto, lo cual permite que cada llamada use su propia conexión.

| | | | |
|---|--|------------------------|---|
| Nombre del Servicio: | System.Commit.Transaction | | |
| Descripción: | Permite aplicar (commit) una transacción en una base de datos y para todas las llamadas anteriores a el servicio System.Execute.SQL , solo para la base de datos a la que hace referencia el SecurityTokenID y en el contexto indicado por el TransactionID . | | |
| Parámetros de entrada requeridos: | SecurityTokenID TransactionID | | |
| Descripción de los parámetros requeridos | | | |
| <table><tr><td>SecurityTokenID</td></tr><tr><td>Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session.</td></tr></table> | | SecurityTokenID | Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . |
| SecurityTokenID | | | |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . | | | |

| <table border="1"> <tr> <th colspan="2">TransactionID</th></tr> <tr> <td colspan="2">Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción.</td></tr> </table> | | TransactionID | | Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. | | | | | | | | | |
|---|---|----------------|-----------------------|---|-----|----------------|-----|--------------------|-----|------|-----|-----|-----|
| TransactionID | | | | | | | | | | | | | |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. | | | | | | | | | | | | | |
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion | | | | | | | | | | | | |
| Posibles valores para los parámetros opcionales | | | | | | | | | | | | | |
| <table border="1"> <tr> <th>ResponseFormat</th><th>ResponseFormatVersion</th></tr> <tr> <td>XML-DATAPACKET</td><td>1.0</td></tr> <tr> <td>XML-DATAPACKET</td><td>1.1</td></tr> <tr> <td>JAVA-XML-WEBROWSET</td><td>1.0</td></tr> <tr> <td>JSON</td><td>1.0</td></tr> <tr> <td>CSV</td><td>1.0</td></tr> </table> | | ResponseFormat | ResponseFormatVersion | XML-DATAPACKET | 1.0 | XML-DATAPACKET | 1.1 | JAVA-XML-WEBROWSET | 1.0 | JSON | 1.0 | CSV | 1.0 |
| ResponseFormat | ResponseFormatVersion | | | | | | | | | | | | |
| XML-DATAPACKET | 1.0 | | | | | | | | | | | | |
| XML-DATAPACKET | 1.1 | | | | | | | | | | | | |
| JAVA-XML-WEBROWSET | 1.0 | | | | | | | | | | | | |
| JSON | 1.0 | | | | | | | | | | | | |
| CSV | 1.0 | | | | | | | | | | | | |
| Ejemplo de una llamada al servicio | | | | | | | | | | | | | |
| ServiceName= System.Commit.Transaction SecurityTokenID= 5200954554030646366 TransactionID= 7497785048145135526 ResponseFormat= JSON ResponseFormatVersion= 1.0 | | | | | | | | | | | | | |
| Ejemplo de la salida en respuesta a la anterior llamada | | | | | | | | | | | | | |
| <pre>[{ "TransactionID": 7497785048145135526, "Code": 1, "Description": "Transacción aplicada con éxito para el id: [7497785048145135526]" }]</pre> | | | | | | | | | | | | | |
| Notas sobre el servicio | | | | | | | | | | | | | |
| <ol style="list-style-type: none"> 1. Cuando se aplica la transacción el número del TransactionID sigue siendo válido, por lo que puede ser usado para otras llamadas a los servicios como System.Execute.SQL e incluso System.Commit.Transaction. 2. La única manera de invalidar el TransactionID para que no pueda ser usado más, es llamando al servicio System.End.Transaction o el servicio System.End.Session. | | | | | | | | | | | | | |

| | |
|----------------------|--|
| Nombre del Servicio: | System.Rollback.Transaction |
| Descripción: | Permite revertir (rollback) una transacción en una base de datos y para todas las llamadas anteriores a el servicio System.Execute.SQL , solo para la base de datos a la que hace referencia el SecurityTokenID y en el contexto indicado por el |

| | | | | | | | | | | | | | |
|--|---|------------------------|--|-----------------------|---|-----------------------|------------|---------------------------|------------|-------------|------------|------------|------------|
| | TransactionID. | | | | | | | | | | | | |
| Parámetros de entrada requeridos: | SecurityTokenID TransactionID | | | | | | | | | | | | |
| Descripción de los parámetros requeridos | | | | | | | | | | | | | |
| <table><tr><td>SecurityTokenID</td></tr><tr><td>Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session.</td></tr><tr><td>TransactionID</td></tr><tr><td>Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción.</td></tr></table> | | SecurityTokenID | Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session. | TransactionID | Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. | | | | | | | | |
| SecurityTokenID | | | | | | | | | | | | | |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session. | | | | | | | | | | | | | |
| TransactionID | | | | | | | | | | | | | |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. | | | | | | | | | | | | | |
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion | | | | | | | | | | | | |
| Posibles valores para los parámetros opcionales | | | | | | | | | | | | | |
| <table><tr><td>ResponseFormat</td><td>ResponseFormatVersion</td></tr><tr><td>XML-DATAPACKET</td><td>1.0</td></tr><tr><td>XML-DATAPACKET</td><td>1.1</td></tr><tr><td>JAVA-XML-WEBROWSET</td><td>1.0</td></tr><tr><td>JSON</td><td>1.0</td></tr><tr><td>CSV</td><td>1.0</td></tr></table> | | ResponseFormat | ResponseFormatVersion | XML-DATAPACKET | 1.0 | XML-DATAPACKET | 1.1 | JAVA-XML-WEBROWSET | 1.0 | JSON | 1.0 | CSV | 1.0 |
| ResponseFormat | ResponseFormatVersion | | | | | | | | | | | | |
| XML-DATAPACKET | 1.0 | | | | | | | | | | | | |
| XML-DATAPACKET | 1.1 | | | | | | | | | | | | |
| JAVA-XML-WEBROWSET | 1.0 | | | | | | | | | | | | |
| JSON | 1.0 | | | | | | | | | | | | |
| CSV | 1.0 | | | | | | | | | | | | |
| Ejemplo de una llamada al servicio | | | | | | | | | | | | | |
| ServiceName=System.Rollback.Transaction SecurityTokenID=5200954554030646366 TransactionID=7497785048145135526 ResponseFormat=JSON ResponseFormatVersion=1.0 | | | | | | | | | | | | | |
| Ejemplo de la salida en respuesta a la anterior llamada | | | | | | | | | | | | | |
| <pre>[{ "TransactionID": 7497785048145135526, "Code": 1, "Description": "Transacción revertida con éxito para el id: [7497785048145135526]" }</pre> | | | | | | | | | | | | | |
| Notas sobre el servicio | | | | | | | | | | | | | |
| 1. Cuando se revierte la transacción el número del TransactionID sigue siendo válido, por lo que puede ser usado para otras llamadas a los servicios como System.Execute.SQL e incluso System.Commit.Transaction o System.Rollback.Transaction . | | | | | | | | | | | | | |

2. La única manera de invalidar el **TransactionID** para que no pueda ser usado más, es llamando al servicio **System.End.Transaction** o el servicio **System.End.Session**.

| | |
|-----------------------------------|---|
| Nombre del Servicio: | System.End.Transaction |
| Descripción: | Permite terminar y revertir (rollback) una transacción en una base de datos y para todas las llamadas anteriores a el servicio System.Execute.SQL , solo para la base de datos a la que hace referencia el SecurityTokenID y en el contexto indicado por el TransactionID , adicionalmente invalida el TransactionID para que no pueda ser usado para ninguna otra llamada a servicios. |
| Parámetros de entrada requeridos: | SecurityTokenID TransactionID |

Descripción de los parámetros requeridos

| |
|---|
| SecurityTokenID |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . |
| TransactionID |
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Transaction y representa el contexto de la transacción. |

| | |
|-----------------------------------|---|
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion |
|-----------------------------------|---|

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|---------------------------|------------------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Ejemplo de una llamada al servicio

ServiceName=System.End.Transaction
SecurityTokenID=5200954554030646366
TransactionID=7497785048145135526
ResponseFormat=JSON
ResponseFormatVersion=1.0

Ejemplo de la salida en respuesta a la anterior llamada

```
[
  {
    "TransactionID": 7497785048145135526,
    "Code": 1,
    "Description": "Transacción revertida y terminada con éxito para el id: [7497785048145135526]"
  }
]
```

Notas sobre el servicio

1. Cuando se termina la transacción el número del **TransactionID** deja de ser válido, por lo que no puede ser usado para otras llamadas a los servicios como **System.Execute.SQL**.
2. Internamente este servicio cierra realmente la conexión JDBC al manejador de base de datos, por lo que libera recursos en el servidor.

| | |
|-----------------------------------|---|
| Nombre del Servicio: | System.End.Session |
| Descripción: | Permite terminar una sesión, solo para la base de datos a la que hace referencia el SecurityTokenID , adicionalmente invalida el SecurityTokenID y todos los TransactionID dependiente del mismo, revirtiendo los e impidiendo su uso. |
| Parámetros de entrada requeridos: | SecurityTokenID |

Descripción de los parámetros requeridos

| SecurityTokenID |
|---|
| Es un número obtenido de una llamada previa de inicio de sesión exitosa al servicio System.Start.Session . |

| | |
|-----------------------------------|---|
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion |
|-----------------------------------|---|

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|---------------------------|------------------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Ejemplo de una llamada al servicio

ServiceName=System.End.Session
SecurityTokenID=5200954554030646366

ResponseFormat=JSON
ResponseFormatVersion=1.0

Ejemplo de la salida en respuesta a la anterior llamada

```
[
  {
    "Code": 1,
    "Description": "Sesión terminada con éxito para el token de seguridad id: [5200954554030646366]"
  }
]
```

Notas sobre el servicio

1. Todas las transacciones y operaciones pendientes son revertidas y cerradas.
2. Los **TransactionID** abiertos a partir del **SecurityTokenID** son inválidos luego de llamado el servicio.
3. El **SecurityTokenID** se vuelve inválido luego de llamado el servicio.

| | |
|-----------------------------------|--|
| Nombre del Servicio: | System.Ping |
| Descripción: | Servicio pensado para el monitoreo de conectividad con el servidor, tiene tres funciones básicas. <ol style="list-style-type: none">1. Verificar conectividad TCP/IP y HTTP.2. Obtener la fecha y hora del servidor.3. Verificar velocidad de respuesta. |
| Parámetros de entrada requeridos: | PING |

Descripción de los parámetros requeridos

| | |
|---|---|
| PING | |
| Número entero positivo largo, que es enviado al servicio con la única finalidad de ser devuelto incrementado en 1, como respuesta del servicio. | |
| Parámetros de entrada opcionales: | ResponseFormat ResponseFormatVersion |

Posibles valores para los parámetros opcionales

| ResponseFormat | ResponseFormatVersion |
|--------------------|-----------------------|
| XML-DATAPACKET | 1.0 |
| XML-DATAPACKET | 1.1 |
| JAVA-XML-WEBROWSET | 1.0 |
| JSON | 1.0 |
| CSV | 1.0 |

Ejemplo de una llamada al servicio

ServiceName=System.Ping
Ping=10
ResponseFormat=JSON
ResponseFormatVersion=1.0

Ejemplo de la salida en respuesta a la anterior llamada

```
[
  {
    "Pong": 11,
    "DateRequest": "2013-09-14",
    "TimeRequest": "13:10:57"
  }
]
```

Notas sobre el servicio

1. El servicio no requiere de autenticación previa en ninguna base de datos.
2. En el ejemplo de salida del ejemplo anterior note que el campo **Pong** es el valor del parámetro de entrada **Ping** + 1.
3. Es fácil hacer un cliente que constantemente realice una llamada al servicio lea el valor de Pong y luego lo vuelva a enviar al servicio de manera repetitiva para conservar un secuencia, al estilo del famoso comando ping.

El orden de los parámetros de llamada a los servicios no afecta en nada, sean estos parámetros requeridos o no.

Da lo mismo lo siguiente:

Llamada ordenada

ServiceName=System.Execute.SQL
SecurityTokenID=5200954554030646366
TransactionID=7497785048145135526
ResponseFormat=JSON
ResponseFormatVersion=1.0
SQL=Select * From tblUsersDB Where IdUser=1

Llamada desordenada

ResponseFormatVersion=1.0
SQL=Select * From tblUsersDB Where IdUser=1
TransactionID=7497785048145135526
SecurityTokenID=5200954554030646366
ServiceName=System.Execute.SQL
ResponseFormat=JSON

El resultado es exactamente el mismo y ambos son llamadas válidas