
記事

[Mihoko Iijima](#) · 2023年3月5日 4m read

VSCode : RESTサービス実行中プロセスにアタッチしてデバッグする方法

開発者の皆さん、こんにちは！

1つ前の記事

では、VSCodeのObjectScriptエクステンションに追加されたRESTサービスのデバッグツールについてご紹介しましたが、この記事では、RESTクライアントなどからHTTP要求を実行し、処理中のIRIS内プロセスにアタッチしながらデバッグを行う方法についてご紹介します。

事前準備などは、[前回の記事](#)と一緒にになりますのでこの記事では割愛します。

図解のクラスでお試しいただく場合は、こちら <https://github.com/Intersystems-jp/RESTDebug-VSCode> をご利用ください。

この方法では、**HTTP要求時に実行しているIRIS内プロセスを特定する必要があります。**

そのため、デバッグ対象のメソッドで処理を待機させるため、hang 20のようにhangコマンドをコード中に追加して試します。

手順は以下の通りです。

1) プロセスにアタッチしてデバッグを行うため、launch.jsonを作成する。

VSCodeのデバッグ用ファイル：launch.jsonファイルを作成し、アタッチ用の設定を行います。

作成方法詳細は、[VSCode : プロセスにアタッチしてデバッグする方法](#)の「手順1 : launch.jsonの用意」をご参照ください。

例)

```
{  
    // Use IntelliSense to learn about possible attributes.  
    // Hover to view descriptions of existing attributes.  
    // For more information, visit: https://go.microsoft.com/fwlink/?LinkId=830387  
    "version": "0.2.0",  
    "configurations": [  
        {  
            "type": "objectscript",  
            "request": "attach",  
            "name": "???????????",  
            "system": true,  
            "processId": "${command:PickProcess}"  
        }  
    ]
```

}

2) デバッグ対象メソッドにHangコマンド追加します。

Hangを利用して処理を数秒待機させ、その間に実行中プロセスを特定します。ブレークポイントを置きたい行の前あたりにhangコマンドを追加してください（2030秒待機するように記述します）。

```
hang 20
```

VSCodeでコードを編集するためには、ワークスペース内にクラス定義用ファイル（*.cls）を配置する必要があります。

サーバ側にあるコードに追記したい場合は、ワークスペースにエクスポートしてから追記します。

エクスポートは、サーバ側の対象コードを右クリックし「Export」を選択します。（下図参照）

Export実行後、ワークスペースに移動します。エクスポートファイルは、/src/パッケージ名/クラス名.clsの形式でワークスペースにエクスポートされます。

対象クラスを開き、コードを追記します。

コードの追記が終わったら、Ctrl + Sで保存します（保存 + コンパイルが実施されます）。

3) 外部クライアントからHTTP要求を実行します。

例は、PostmanからPOST要求でクエリパラメータとBodyに文字列を指定しています。

Sendボタンを実行すると、2) で指定したHangコマンドが実行されるため、指定秒数待機します。

この間に、アタッチ対象プロセスを探します。

4) アタッチ対象プロセスにアタッチする。

3) の実行後すぐ、VSCodeのデバッグウィンドウを開き、launch.jsonで指定したデバッグ名（例では、「アタッチデバッグ練習」）をクリックし、プロセスリストを表示させます。

プロセスリストには「RESTディスパッチクラス名.*」（*には数字が入ります）の形式で処理中ルーチン名が表示されます。

対象プロセスを見つけたら行をクリックしプロセスにアタッチします。

注意：指定したhangの秒数の間にプロセスを見つけてアタッチしてください。

5) デバッグを実行する。

Hangコマンドの実行が終わると、ブレークマークのところでデバッグが一時停止します。

6) デバッグを終了する。

最後に、デバッグ停止のボタンを押して終了します。

後は、デバッグ用アイコンを利用してデバッグを進めるだけです。

メモ:

1. アタッチ用プロセス一覧に対象プロセスが表示されない場合は、一旦VSCodeを再起動して再度お試しください。
2. アタッチのデバッグを実行するために追加したhangコマンドは
デバッグ終了時必ず削除 / コメント化してください。

[#REST API](#) [#デバッグ](#) [#ヒントとコツ](#) [#開発環境](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)

ソースURL:

<https://jp.community.intersystems.com/post/vscode%EF%BC%9Arest%E3%82%B5%E3%83%BC%E3%83%93%E3%82%B9%E5%AE%9F%E8%A1%8C%E4%B8%AD%E3%83%97%E3%83%AD%E3%82%BB%E3%82%B9%E3%81%AB%E3%82%A2%E3%82%BF%E3%83%83%E3%83%81%E3%81%97%E3%81%A6%E3%83%87%E3%83%90%E3%83%83%E3%82%B0%E3%81%99%E3%82%8B%E6%96%B9%E6%B3%95>
