記事

Toshihiko Minamoto · 2022年2月1日 5m read

Open Exchange

DockerマイクロサーバーとしてIRIS Native APIを使用したWebSocketクライアントJS

動作中のデモを確認できるデモビデオを以下で公開しています。https://youtu.be/dSV-0RJ5Olg

皆さんこんにちは 完全に新しいIRISイメージと<mark>たった4行</mark> のDockerコマンドを使って実行するイメージを使ってマイクロサービスのデモを行いましょう。 2020年6月1日 - rcc

すべてのパーツを1つのコンテナイメージにまとめたコンパクトなオールインワンバージョンが公開されました。 詳細はこちら: <u>IRIS-NativeAPI-Nodejs-compact</u> 2020年5月24日 - rcc

Dockerを使った簡易インストールを追加しました。コンテキストを参照 2020年5月25日 - rcc

Linux & Windowsに最適な検証済みの強化スクリプトはこちら https://github.com/rcemper/WSockClientMicroSV/blob/master/READMEwindows.MD 2020年5月26日 - rcc

このデモは、Caché用にすでに存在するNode.jsに基づくWebSocketクライアントを再設計したものです。 主に以下のような変更点があります。

- 新しいIRIS Native API for Node.isの使用。特に<u>グローバル配列を操作</u>する場合
- 直接トリガーされたクライアントからサーバー設計への変更
- マイクロサービス/マイクロサーバーの例として、結果を別のdockerイメージに配置
- マイクロサービスの実行を制御するための単純なインターフェースをIRISに追加

ユーティリティをIRISホストで直接呼び出す代わりに、System Interoperability(別名 ENSEMBLE)で通常行うようにして、work-packagesをMicroServiceに送信します。もちろん、2つ以上のWebSocketサーバーオプションがあります。WebSocketクライアントサービスがジョブを完了したら、その結果を取得します。

組み込みのWebSocketクライアントに対するメリットは、すべてのネットワーク、セキュリティ、ファイアウォールの問題を主要なデータサーバーに寄せ付けないことです。 Node.js がこの分野で持っている経験と品質については言うまでもありません。

デモは デフォルトのechoサーバーとして使用します。 次に、テキストを数行入力します。 独自のテキストの任意の場所に「Lorem Ipsum」を追加して、コンテンツを長くすると良いでしょう。 そして、それをサービスに送信して、エコーを待ちます。 また、制御プログラムをExitまたはサービスをStopする前に、 テキストを変更するオプションもあります。 Published on InterSystems Developer Community (https://community.intersystems.com)

この処理はすべて非同期で実行されます。 完了を待たずに、それまでにechoサーバーから受信した内容を、リスナーが 定期的に表示します。

インストールするには、以下が必要です。

- IRISのdocker イメージ (intersystems/iris-community:2020.2.0.199.0)
- WebSocketマイクロサーバーのdockerイメージ(docker pull rcemper/rcc:demoJS)

簡略化されたinit: initに次のdockerコマンドを実行してください。 docker run --name ini1 --init -it --rm / -privileged -v \$(pwd):/external /rcemper/rcc:demoJS bash /rcc/init.sh ### 元のアプローチは有効ではありますが、必須ではありません。 ###

IRIS-Docker-micro-Durabilityを利用するためにOpen Exchangeまたはここから入手したWSockClientMicroSV.tar.gz

demoディレクトリの確認: Dockerイメージはnobodyであるため、保護をrwx (chmod 777)
 に設定します。###

これを**実行**するには、まずdemo**ディレクトリ**からIRISを起動します(-dまたは-itを実行して動作を観察します)。(!)

```
docker run --name iris1 --init --rm -d \
-p 52773:52773 -p 51773:51773 \
-v $(pwd):/external \
intersystems/iris-community:2020.2.0.199.0 \
-b /external/pre.copy
```

次にMicroServerを起動します。

```
docker run --name rcc1 --init -it --rm \
rcemper/rcc:demoJS /usr/bin/node /rcc/nodejs/WSockIris.js $(hostname -I)
```

.-itを使って起動したため、以下が表示されます。

```
platform = linux: ubuntu

    ******************
    Connect to IRIS on: 192.168.0.23
Successfully connected to InterSystems IRIS.
    *** wait 3sec for request ***
    ******* Startup done *******

    *** wait 3sec for request ***
    *** wait 3sec for request ***
```

次に、新しいLinuxターミナルで制御アプリケーションを実行します。

docker exec -it iris1 iris session iris ZSocket

Published on InterSystems Developer Community (https://community.intersystems.com)

以下が表示されます。

```
*** Welcome to WebSocket Micoservice demo ***
Known Hosts (*=Exit) [1]:
1 wss://echo.websocket.org/
2 --- server 2 ----
3 --- server 3 ----
select (1): ==> wss://echo.websocket.org/
Enter text to get echoed from WebSocketClient Service
Terminate with * at first position
or get generated text by %
or append new text with @
    hello socket microServer
    now you got 2 lines
3
Select action for WebClient Service
New EchoServer (E), Send+Listen(S), New Text(N), Exit(X), Exit+Stop Client(Z) [S]s
****** 0 Replies *****
***** 2 Replies ******
    hello socket microServer
    now you got 2 lines
Select action for WebClient Service
マイクロサービスでは以下が表示されます。
*** wait 3sec for request ***
echoserver: wss://echo.websocket.org/
** Lines to process: 1 **
****** next turn ******
* WebSocket Client connected *
***** Client is ready *****
Line: 1 text> 'hello socket microServer '
Received: 1 > 'hello socket microServer '
Line: 2 text> 'now you got 2 lines '
Received: 2 > 'now you got 2 lines '
****** lines sent: 2 *****
*** replies received: 2 ****
*** wait 3sec for request ***
```

事後警告。 イメージのバージョンを必ず確認してくださ!!! 私はntersystems/iris-community:2020.2.0.204にハマってしま!!ました。 Published on InterSystems Developer Community (https://community.intersystems.com)

注意: すべてのスクリプトはLinux でのみテストされています!

```
*** wait 3sec for request ***
*** Welcome to WebSocket Micoservice demo ***
                                                                     *** wait 3sec for request ***
Known Hosts (*=Exit) [1]:
                                                                     *** wait 3sec for request ***
1 wss://echo.websocket.org/
                                                                     *** wait 3sec for request ***
2 --- server 2 ----
                                                                     *** wait 3sec for request ***
  --- server 3 ----
                                                                     *** wait 3sec for request ***
select (1): ==> wss://echo.websocket.org/
                                                                     *** wait 3sec for request ***
                                                                     *** wait 3sec for request ***
Enter text to get echoed from WebSocketClient Service
                                                                     *** wait 3sec for request ***
Terminate with * at first position
                                                                     *** wait 3sec for request ***
or get generated text by %
                                                                     *** wait 3sec for request ***
or append new text with @
                                                                     *** wait 3sec for request ***
                                                                     echoserver: wss://echo.websocket.org/
    hi this is the final chek
                                                                      ** Lines to process: 3 **
2
3
    what did you expect ?
                                                                      ****** next turn ******
    so go
                                                                      * WebSocket Client connected *
                                                                      ***** Client is ready *****
                                                              Line: 1 text> 'hi this is the final chek
Select action for WebClient Service
                                                              Received: 1 > 'hi this is the final chek
New EchoServer (E), Send+Listen(S),New Text(N),Exit(X), Exit+
                                                              Line: 2 text> 'what did you expect ? '
Stop Client(Z) [S]S
                                                              Received: 2 > 'what did you expect ?
Line: 3 text> 'so go '
                                                              Received: 3 > 'so go
****** 0 Replies ******
                                                                      ****** lines sent: 3 *****
****** 3 Replies ******
                                                                      *** replies received: 3 ****
1
    hi this is the final chek
2
3
    what did you expect ?
                                                                      *** wait 3sec for request ***
    so go
                                                                      *** wait 3sec for request ***
                                                                      *** wait 3sec for request ***
```

#Docker #JavaScript #Microservices #Node.js #ObjectScript #グローバル #コンテスト #コンテナ化 #InterSystems IRIS

InterSystems Open Exchangeで関連アプリケーションを確認してください

ソースURL:

https://jp.community.intersystems.com/post/docker%E3%83%9E%E3%82%A4%E3%82%AF%E3%83%AD%E3%82%B5%E3%83%BC%E3%83%BC%E3%83%BC%E3%81%A8%E3%81%97%E3%81%A6iris-native-api%E3%82%92%E4%BD%BF%E7%94%A8%E3%81%97%E3%81%9Fwebsocket%E3%82%AF%E3%83%A9%E3%82%A4%E3%82%A2%E3%83%B3%E3%83%88is%C2%A0