

CIS で JSON 応答構造が最上位要素でラップされない場合の JSON から XML への変換

目次

[概要](#)

[問題](#)

[解決策](#)

概要

このドキュメントでは、(Cisco Information Server (CIS) の) Representational State Transfer (REST) データソースが受信した JavaScript Object Notation (JSON) 応答を、CIS が受信した JSON 応答が最上位要素にラップされていない場合に、XML に変換する手順について説明します。

問題

場合によっては、JSON 応答が 1 つのドキュメントとしてラップされていないことがあります。たとえば次のような JSON 応答の場合です。

```
{ "status": "OK",
  "data": { "trends": { "55-64": [
    { "date": "201312", "value": "13.15" }
  ]
  },
  "query_cost": 6,
  "trends_frequency": "monthly"
}
```

これが応答の場合、デフォルトでの XML から JSON への変換では、次のような XML だけが生成されます。

```
<?xml version="1.0" encoding="UTF-8"?><test>
<status>OK</status>
```

Studio では最初のスニペットだけが返されます。ただし RestClient では完全な応答が返され、**cs_server.log** にも完全な応答が示されます。

望ましい XML 出力は次のとおりです。

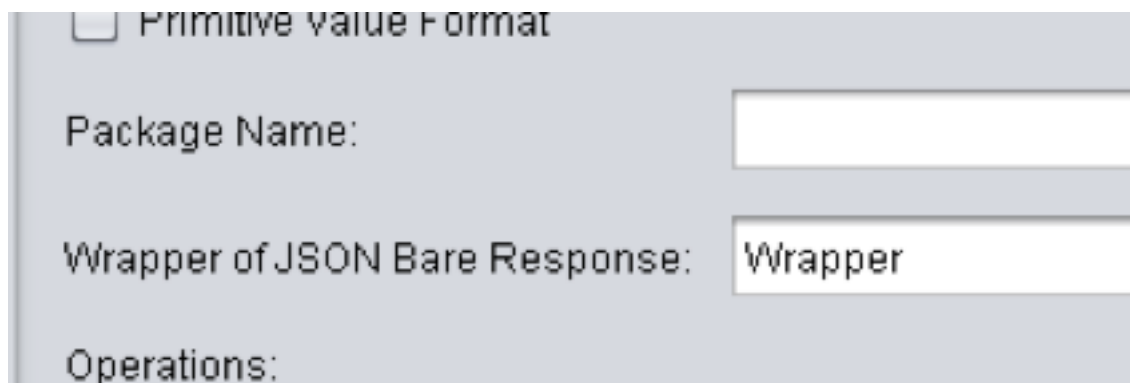
```
<?xml version="1.0" encoding="UTF-8"?>
<Wrapper>
<status>OK</status>
```

```
<data>
<trends>
<_55-64>
<date>201312</date>
<value>13.15</value>
</_55-64>
</trends>
<query_cost>6</query_cost>
<trends_frequency>monthly</trends_frequency>
</data>
</Wrapper>
```

解決策

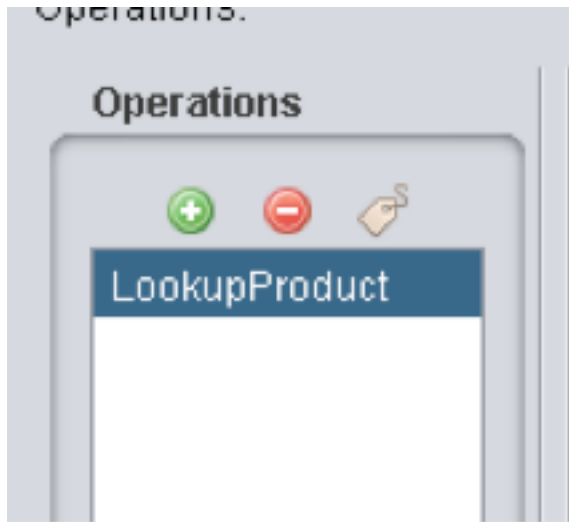
REST データ ソースは、JSON または XML 形式で Web サービスからデータを取得します。データが表形式で抽出可能な場合は、このために XSL 変換機能を使用する必要があります。基本的な手順を次に示します。

1. Studio を 6.2.3.00.22 パッチ以上にアップグレードします。これにより、新機能である Design by Example を利用できるようになります。この機能では、JSON 出力を表形式出力にマッピングするために使用する Extensible Stylesheet Language Transformation (XSLT) プロシージャのスキーマを容易に取得できます。
2. Studio で、その [XML text size] を大きな値 (例 : 50,000) に設定します。返された XML がこのサイズ制限を超えている場合、[Design by Example] をクリックすると失敗するため、これは重要です。Studio でこの設定値を増加できない場合は、返される XML のサイズを削減するため、異なる入力基準を使用することを検討します。
3. この手順は、データ ソースの作成時またはデータ ソースの作成後に実行できます。REST データ ソースの作成時に、[JSON Format] チェックボックスをオンにします。
4. JSON 構造が最上位要素でラップされていない場合、CIS はラッパーを追加できます。これにより、JSON 応答から作成される XML が適切に正規 XML に変換されます。応答がラップされているかどうか分からない場合は、ラッパーを追加して、これが必要であるかどうかを後で評価できます。これは [Wrapper of JSON Bare Response] フィールドを使用して設定します。このフィールドは、[JSON] > [XML] 出力のルート ノードを設定します。



The screenshot shows a configuration window with a checkbox for 'Primitive Value Format' which is unchecked. Below it, there are three input fields: 'Package Name:' (empty), 'Wrapper of JSON Bare Response:' (containing the text 'Wrapper'), and 'Operations:' (empty).

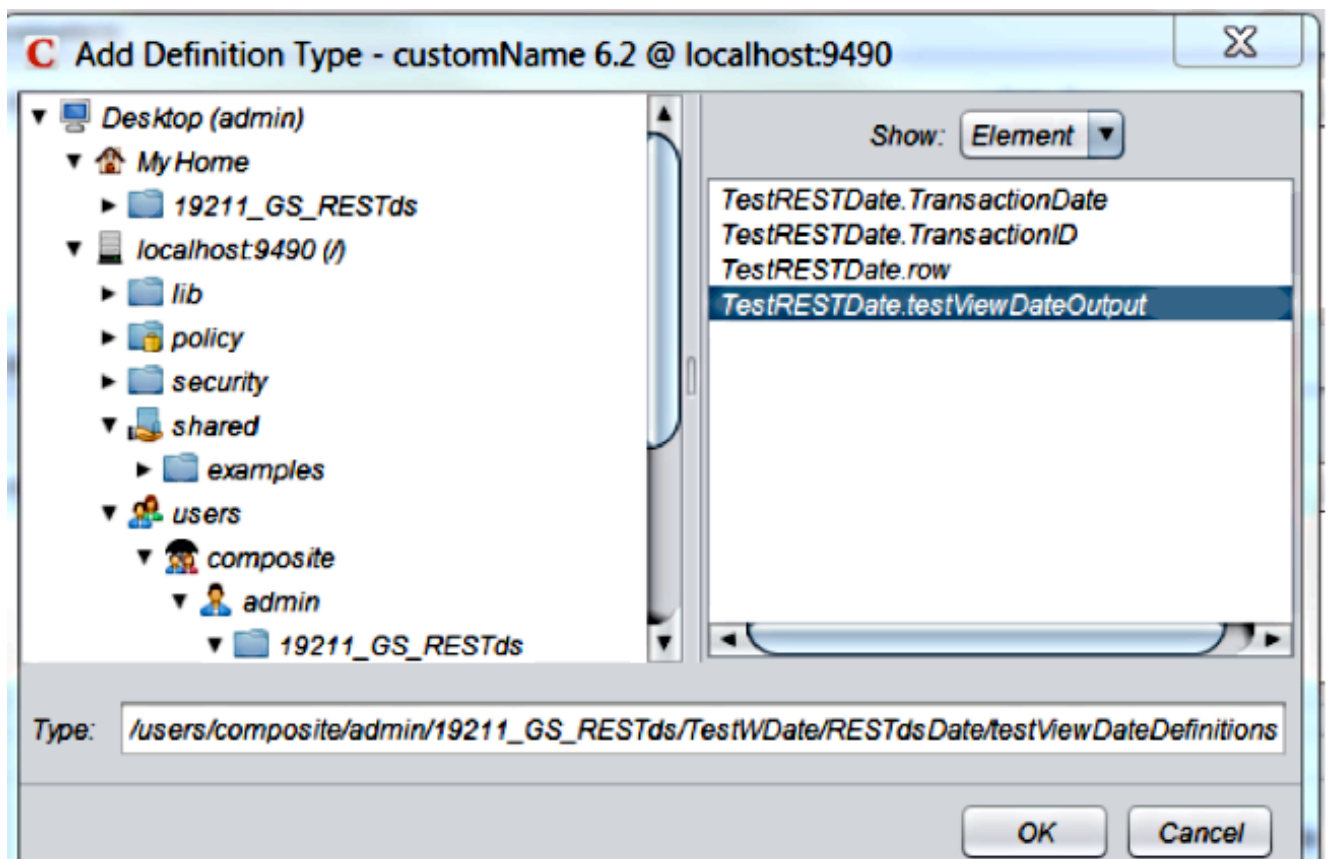
5. [Operations] セクションに追加されている操作のいずれかを選択します。



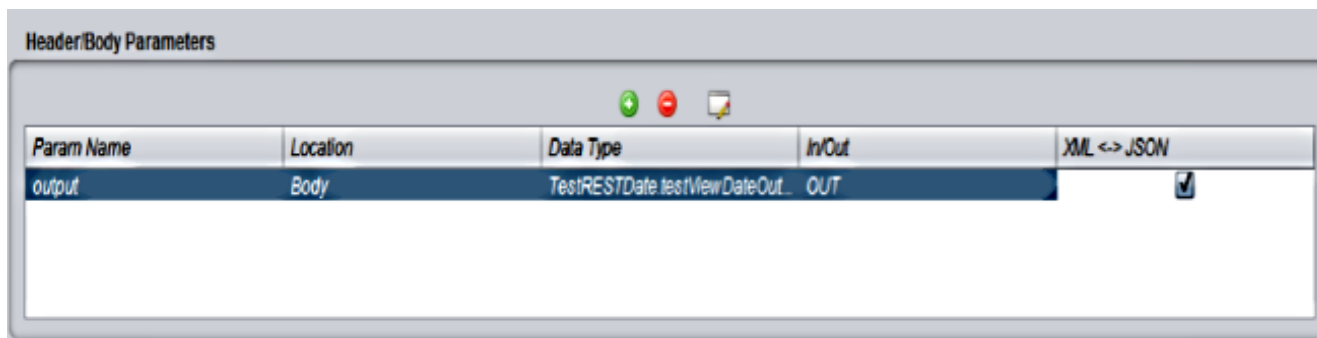
6. [Header/Body Parameters] セクションの [Design by Example] をクリックします。この操作を追加したばかりの場合は、このボタンをアクティブにするため、REST データ ソース パネルを閉じてから再び開く必要があることがあります。



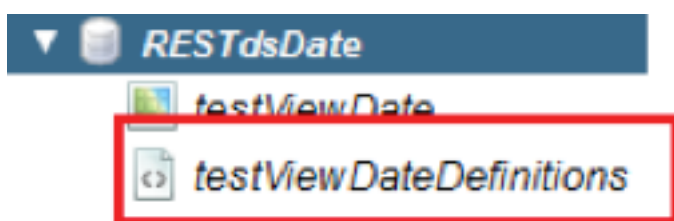
7. JSON 応答の最上位要素を選択し、[OK] をクリックします。



8. パラメータに名前を指定し、[Save] をクリックします。



これにより、XML スキーマ定義セットが作成されます。この定義セットはデータ ソース リソースに含まれます。



9. これにより、JSON 出力が XML に変換されるため、XSL 変換機能で処理できるようになります。操作を開き、出力構造を調べます。この出力には、XSLT プロシージャで使用する構造が示されています。

Name	Type / Reference	Native Type
output	TestRESTDate testViewDateOutput	TestRESTDate testViewDateOutput
TestRESTDate.row[1..*]	<Anonymous>	
TestRESTDate.TransactionID	DECIMAL(32,0)	integer
TestRESTDate.TransactionDate	VARCHAR(2147483647)	NMTOKEN

10. 操作を実行します。JSON 出力が XML に変換されます。

