



Technical Note 04-41

2004 で SOAP リクエストを解析する

By David Adams

Technical Note 04-41

(原題: Reading a Full SOAP Request with 4thDimension 2004)

概要

4th Dimension のバージョン 2003 以降では、メソッドを Web サービスとして提供し、Web サービスに対応している種々のアプリケーション、ツール、開発環境と 4D を融合させることが可能になりました。SOAP プロトコルは、XML 形式のメッセージで成り立っていますが、4D は XML 生成、解析、データタイプの変換などの面倒な処理をすべて内部的に行なってくれます。しかし場合によっては、デベロッパが自ら SOAP を処理したほうが望ましいかもしれません。たとえば、次のような状況が考えられます。

SOAP 通信をデバッグしたい

SOAP リクエストのログをとりたい

4D の変数、配列、引数にバインドされないデータを読み取りたい

空の値と存在しない入力値を区別したい

DOM Parse XMLvariable コマンドや他のサードパーティ製パーサーで解析したい

SOAP の構造に興味がある

GET WEB FORM VARIABLES コマンドのドキュメント化されていない動作を利用して SOAP メッセージを直接読み取る方法については Tech Note 04-19 で取り上げました。その後発表された 4D 2004 では、GET HTTP BODY という新しいコマンドができたので、さらに簡単に SOAP リクエストを読み取ることが可能になりました。

GET HTTP BODY(BLOB|text)

SOAP について

Web サービスは、その名称とは対照的に、理論上 Web を必要とはしていません。とはいえ現実にはほとんど SOAP リクエストは Web プロトコルである HTTP を介して送られており、4D も他の開発環境と同じく SOAP over HTTP だけをサポートしています。SOAP と HTTP の区別は重要です。HTTP にはどんなデジタル情報でも含められるのに対し、SOAP は SOAP のルールに従った XML 形式のメッセージでなければならないからです。言い換えると SOAP は XML ですが HTTP には XML を含めることができます。

| 説明 | 例文 |
|---|---|
| HTTP ヘッダ。 XML ではありません。 GET HTTP HEADER コマンド で取得することができます。 | POST /4DSOAP/ HTTP/1.1 User-Agent: 4D built-in SOAP Client Date: Sat, 20 Dec 2003 23:43:27 GMT Content-Type: text/xml; charset=utf-8 Host: 127.0.0.1 Connection: Close SoapAction: "A_WebService#ConvertFeetToMeters" Content-Length: 508 |
| HTTP ボディ。 SOAP メッセージです。 GET HTTP BODY で取得し、 DOM Parse XML variable で解 析することができます。 | <SOAP-ENV:Envelope SOAPENV: encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/1999/XMLSchema" xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"> <SOAP-ENV:Body> <mns:ConvertFeetToMeters xmlns:mns="http://www.4d.com/namespace/default"> <inFeet xsi:type="xsd:float">100.1</inFeet> </mns:ConvertFeetToMeters> </SOAP-ENV:Body> </SOAP-ENV:Envelope> |

前半部分は、純然たる HTTP であり、XML 形式ではありません。本体部分がフォームデータ、アップロードドキュメント、URL リクエストであろうと、大概、同じような形をしています。これに対しボディは空かもしれず、アップロードファイルや Web フォームのフィールドデータかもしれません。Web サービスでは、この部分に SOAP メッセージが格納されます。

内蔵 Web サービスの仕組み

4D は Web サービスリクエストを受け付けると、はじめに SOAP-ENV:Body セクションからメソッド名と入力データを読み取ります。

```
<mns:ConvertFeetToMeters xmlns:mns="http://www.4d.com/namespace/default"><inFeet xsi:type="xsd:float">100.1</inFeet></mns:ConvertFeetToMeters>
```

この場合は、ConvertFeetToMeters メソッドがコールされていることになります。入力データは SOAP DECLARATION コマンドを使用して任意のプロセス変数か配列にバインドすることができます。

```
C_REAL($1;$feet_r)
SOAP DECLARATION($1;Is Real;SOAP Input;"inFeet")
```

結果として ConvertFeetToMeters メソッドが起動するときには自動的に \$1 は実数タイプで値が 100.1 に設定されることになります。

オリジナルの SOAP メッセージを読む裏技

内蔵 Web サービスは便利ですが、オリジナルの SOAP メッセージがほとんど目に触れないような仕組みになっています。GET WEB FORM VARIABLES コマンドでは、=がひとつ足りない点を除けば、オリジナルのメッセージがほぼ完全な形で取得できます。GET WEB FORM VARIABLES は、Web フォームのデータを名=値という形式で受け取る仕組みになっています。同様に SOAP リクエストも途中に=があるので、このコマンドで読み取ることができるというわけです。

この方法は、SOAP リクエストのサイズがテキスト変数のサイズに収まっている限り動作します。この点、バージョン 2004 の GET HTTP BODY はテキストだけでなく BLOB も使用できる点が優れています。

実践編-SOAP のログ、解析、デバッグ

SOAP のヘッダやボディを直接、読み取ることができれば、ログの記録、解析やデバッグの点で多いに役立ちます。Web サービスが期待したように動作しないとき、SOAP DECLARATION に間違いがあるのか、リクエストが不正なのか、あるいは 4D の自動変換に問題があるのかを特定する助けになります。

SOAP を 4D で直接、読み取るのではなく、ネットワークパケットを解析することによって初めて分かる事柄もあります。詳細は Tech Note 03-21 を参照してください。

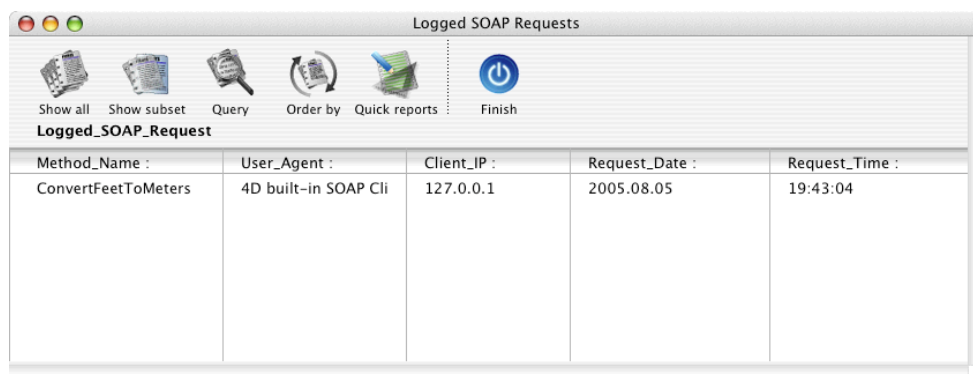
4D の Web サービスがコールされると、以下のような Web ログが記録されます。

209.238.253.9 - - [23/Dec/2003:17:04:31 +1100] "POST /4DSOAP/HTTP/1.1" 200 632

メソッド名などの詳細情報が必要な場合は、カスタムコードで SOAP ヘッダから取り出すと良いでしょう。

soap_LogRequest - ログの例題

SOAP リクエストごとに、時間、日付、クライアントの IP アドレス、4D のメソッド名、ユーザエージェントソフトウェアの名前を記録します。SOAP リクエストがあるたびに On Web Authentication データベースメソッドが起動するので、そこでログメソッドをコールします。



| Method_Name : | User_Agent : | Client_IP : | Request_Date : | Request_Time : |
|---------------------|----------------------|-------------|----------------|----------------|
| ConvertFeetToMeters | 4D built-in SOAP Cli | 127.0.0.1 | 2005.08.05 | 19:43:04 |

注記:

Web サービスに対応するコードはなるべく速くて実地的なものが望ましいことを考えると、サンプルコードには改良の余地があるといえます。サンプルではコードを簡潔にするためレコードにログを記録しました。多くの場合、ファイルに保存するほうが実的です。逆に複数の 4D Client を Web 公開している場合は、ログをレコードにとっておき、専用のプロセスあるいは専用の 4D Client にログを統合させるような戦略が効果的です。

未定義の入力値を読み取る

SOAP メッセージを直接、読み取れば SOAP DECLARATION で定義していない入力値を読み取ることも可能です。たとえば前述の SOAP が、次のように inRoundTo というオプションのフィールドを含んでいたとします。

```
<mns:ConvertFeetToMeters xmlns:mns="http://www.4d.com/namespace/default"><inFeet xsi:type="xsd:float">100</inFeet> <inRoundTo xsi:type="xsd:float">3</inRoundTo></mns:ConvertFeetToMeters>
```

これが毎回送られてくる情報であれば、普通は SOAP DECLARATION を訂正したいと思うはずです。

C_LONGINT(\$2)

SOAP DECLARATION(\$2;Is LongInt;SOAP Input;"inRoundTo")

注記:

コンパイラ宣言と SOAP 宣言には別々の目的があり、SOAP の入出力に使用されるものについてはどちらも必要です。

それでは、SOAP DECLARATION でパラメータを宣言しない理由にはどんなものがあるのでしょうか。

SOAP DECLARATION による定義をすると、上記の例では inRoundTo として値 0 が送られた場合、あるいはまったく送られて来なかった場合は共に \$2 が 0 になります。SOAP メッセージを直接、読み取れば、両者を区別することが可能です。

SOAP DECLARATION による定義で使用される引数、プロセス変数、プロセス配列は、Compiler_Web あるいは Compiler_Web からコールされたサブルーチンで宣言されていなければなりません。SOAP メッセージを直接、読み取れば、入力値を自由にフィールド、ドキュメント、ローカル/プロセス/インタープロセス変数あるいは配列に代入することが可能です。Compiler_Web で宣言する必要はありません。

特定の Web サービスは時代とともに進化するかもしれません。SOAP メッセージを直接、読み取れば、そのような更新にも対応しやすくなります。