Technical Note 04-12
SOAP でデータを返す

By David Adams
Technical Note 04-12

（原題: Returning Data Through SOAP）

概要

バージョン 2003 以降、4D のメソッドを Web サービスとして提供できるようになりました。特に変更を加えなくても同じメソッドをローカルとリモートの両方から利用できますが、Web サービスとしてコールした場合は、$0 以外に任意のプロセス変数や配列を返すわけにはいきません。この Tech Note では、Web サービスでデータを返す方法について考慮します。4D の特徴を例示するためのサンプルデータベースも付属しています。

Web サービス、SOAP そして XML

Web サービスは、4D に組み込まれた Web サーバを通して送受信される SOAP（Simple Object Access Protocol）メッセージで成り立っています。SOAP メッセージは XML 形式で構成されています。4D は以下の処理をすべて自動で行なっています。

SOAP メッセージを受信する
XML データタイプを 4th Dimension のデータタイプに変換する
セキュリティチェックを実行する
SOAP リクエストを当該プロジェクトメソッドに渡す
4th Dimension の値を XML 要素にバインドする
4th Dimension の値を XML の値に変換する
SOAP レスポンスを生成して送信する

SOAP はすべて内部で処理されるので、特に意識しなくても Web サービスを使用することができますが、やはり SOAP メッセージについて理解しておくことは大切です。SOAP メッセージを直接分析する方法については Tech Note 03-21 が参考になるはずです。

Web サービスを起動する

メソッドを Web サービスで公開するためには、次の条件が満たされなくてはなりません。

4D の Web サーバが起動している
「環境設定/Wen サービス/Web サービスリクエストを許可する」が有効である
「対象メソッドのプロパティ/Web サービスとして提供する」が有効である

返された値を定義する

SOAP レスポンスで返すデータを定義する方法は次のふたつです。
メソッドの引数$0 は自動的に出力されます。
プロセス変数また配列は SOAP DECLARATION コマンドで出力にバインドされます。

$0 を利用する場合

以下は、$0 に値を代入する簡単なメソッドです。

```
C_TEXT($0;$1)
$0:=$1
```

上記のメソッドに対応する SOAP メソッドは（$1 が"Hello world!"だとして）次のようになります。

```
<SOAP-ENV:Envelope SOAP-ENV: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/2001/XMLSchema"
	xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

 <SOAP-ENV:Body>

	<ns1:returnInputTextResponse xmlns:ns1="http://www.4d.com/namespace/default"
	SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

		<FourD_arg0 xsi:type="xsd:string">Hello world!</FourD_arg0>

	</ns1:returnInputTextResponse>
</SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

$0 の値は FourD_arg0 という XML 要素として返されていますが、この要素名は SOAP DECLARATION コマンドで定義することができます。

```
SOAP DECLARATION($0;Is Text ;SOAP Output ;"outEchoInputString")
```

この場合、出力される SOAP は次のように変わります。

```
<outEchoInputString xsi:type="xsd:string">Hello world!</outEchoInputString>
```

SOAP DECLARATION コマンドの用途は、パラメータに分かりやすい名前をつけることだけではありません。これから解説するように、複数のデータを返すときにもこの同じコマンドを使用することになります。

SOAP DECLARATION で$0 以外の値を利用する場合

SOAP DECLARATION コマンドは、ひとつのメソッドの中で何回も実行することができます。以下の例では 3 個のプロセス変数を SOAP レスポンスのために定義しています。

```
SOAP DECLARATION(returnLongint_l;Is Longint ;SOAP Output ;"outSecondsSinceMidnight")
SOAP DECLARATION(returnReal_r;Is Real ;SOAP Output ;"outPi")
SOAP DECLARATION(returnString_s;Is String Var ;SOAP Output ;"outServerVersion")

returnLongint_l:=Current time(*)+0  ` Convert time to seconds since midnight.
returnReal_r:=Pi
returnString_s:="Version #"+Application version
```

SOAP DECLARATION コマンドによって、プロセス変数の値はそれぞれ XML 要素にバインドされます。利用可能なリソースとデータタイプが許す限り、何度でもコールすることができます。この方法は、配列を含め、多くの値を返せるという点が$0 を利用する場合よりも優れています。

サポートされているデータタイプ

データソース	可否
$0	●
4D 表記	
フィールド	
インタープロセス配列	
インタープロセス変数	
直接表記	
ローカル配列	
ローカル変数	
$0 以外のパラメータ	
プロセス配列	●
プロセス変数	●

4D タイプ	変数可否	配列可否	SOAP タイプ	例
BLOB	●		base64Binary	U09BUF9CTE9C
ブール	●	●	boolean	0
日付	●	●	date	2003-08-23
整数	●	●	int	1
倍長整数	●	●	int	1
ピクチャ				
ポインタ	●			
実数	●	●	float	1.1
文字列	●	●	string	Hello world!
テキスト	●	●	string	Hello world!
時間	●		time	16:54:5510:00

ピクチャは BLOB に変換すれば扱うことが可能です。Tech Note 04-13 をご覧ください。（今回のサンプルも同様の手法でピクチャを Web サービスで扱っています。）

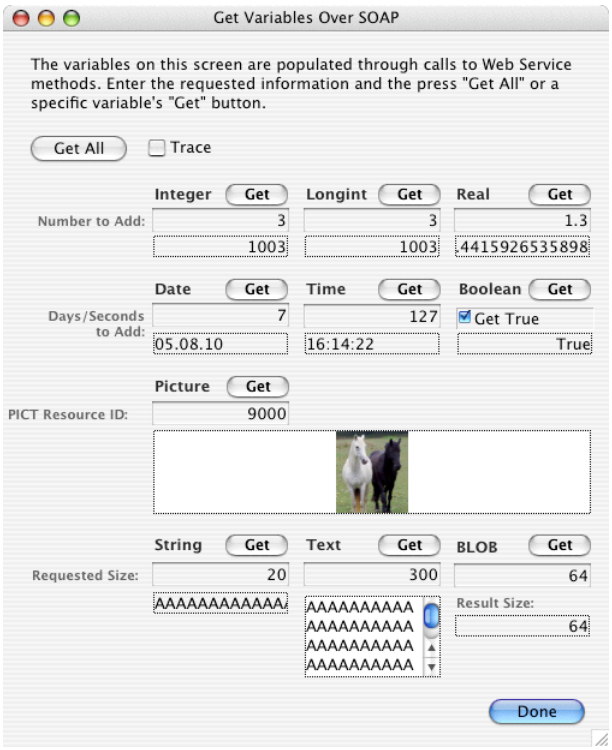
4D には時間の配列というものがないので倍長整数の配列で代用します。

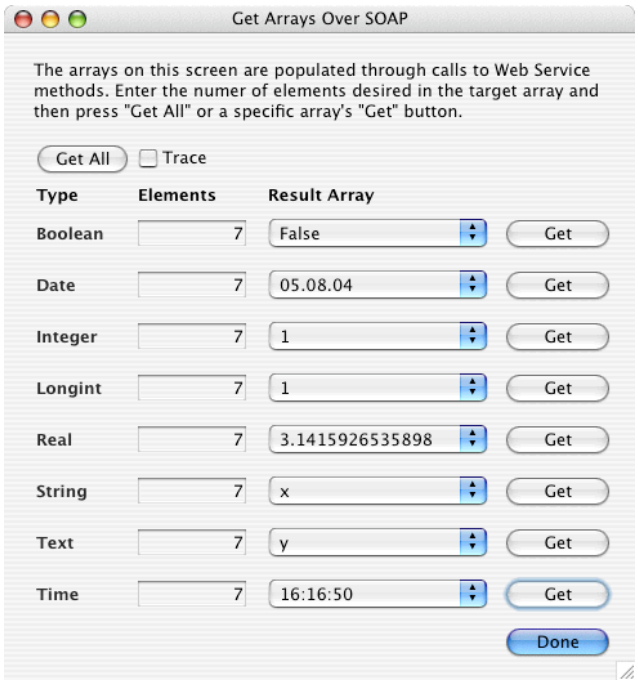
サンプルデータベース

サンプルデータベースでは 4th Dimension の Web サービスで交換可能なデータタイプの送受信をテストすることができます。

変数

数値はリクエストした値を加算したもの、画像はリクエストした PICT リソース、文字はリクエストした長さのデータが返されます。





配列

リクエストしたサイズの配列が返され、ポップアップ/ドロップダウンリストに納められます。

プロキシメソッドのカスタマイズ

4D の SOAP プロキシメソッドは、Web サービスの URL を直接指定します。サンプルのではローカルマシンにアクセスするので URL は http://127.0.0.1/です。それぞれのメソッドにこのURLをハードコーディングする代わりにrequestGetAccessURLメソッドでURLを定義しているので、別のサーバを使用する場合はここを書き換えます。実際のシステムでは、レコード/リスト/リソースなどに保存すると良いでしょう。

ポート番号

サンプルではポート 8080 で Web リクエストを受け付けるように設定されています。OS で許可されている未使用のポート番号であれば、どれでも使用することができます。

Compiler_Web

はじめて Web サービスを使用する際、しばしば問題になるのが Compiler_Web メソッドです。このメソッドでは Web フォームや SOAP リクエストとして入ってくる値を 4D の変数や配列にバインドするためのものです。レスポンスつまり出力に使用する値はここで宣言しなくても大丈夫です。

まとめ

SOAP DECLARATION コマンドに関連してポイントとなる情報をまとめてみました。

SOAP DECLARATION コマンドは、別ルーチンではなく、Web サービスとしてコールされるメソッドの中で使用しましょう。

XML の要素名には使用できない文字があります。中でもスペースが使用できない点に留意してください。SOAP DECLARATION コマンドでは、不正な要素名を定義しないように注意しましょう。

要素名をいい加減につけると SOAP クライアントが困ってしまいます。要素名はデータの意味が分かるようなものをつけるように心がけましょう。

SOAP DECLARATION コマンドは、変数や配列のタイプを宣言するわけではありません。パラメータ、変数、配列の定義は必ず別途、宣言するようにしましょう。

SOAP DECLARATION コマンドは、メソッドが Web サービスとしてコールされたのでない限り無視されます。この仕組みがあるおかげで、同じメソッドをローカルで呼び出すことも Web サービスとしてコールすることもできるようになっています。