

WSDL2RPG – FAQ

FAQ Web Service sends Schema and Any Elements

Status of this document

Date: 20.06.2009
Version: 1.0

Question

What can I do if WSDL2RPG generates a 'schema' and an 'any' sub field in the data structure of the response message?

Answer

WSDL2RPG generates a 'schema' and an 'any' sub field if the web service returns a complete schema and a number of elements according to the schema.

The interesting part of the wsdl file may look similar to the following sample:

```
<xs:element name="responseMessage">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="s:schema" />
      <xs:any/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

The problem with that kind of web services is that the exact structure of the response data is not known at runtime. Even worse, the web service may (it hopefully will not really do that) send back a different schema and hence a different XML document each time it is called. Therefore all WSDL2RPG can do for you is to put the schema into one field and the XML message into another field. Guess what, these fields are named 'schema' and 'any'.

I do not like such web services and on my very own opinion these web service are no good programming practice. Actually the web service is just a huge container that can deliver all types of "goods". But you never know what it delivers until you look at the delivery note aka the XML schema.

So what to do first?

First generate the stub module and write a little test program to call the web service. Add a breakpoint right after the call to the web service. When the program stops at the breakpoint look at the content of 'schema' and 'any' to get an idea of what the web service returns to you.

Next step:

Depending of what you want to do, you have at least two options. First you can save the content of 'schema' and 'any' to files and then do whatever you want to do with that. Perhaps you want to verify the XML data against the schema but for sure you want to retrieve the values from the XML document. Retrieving the data from the saved XML document can be done with Scott Klement's HTTP API.

The second option you have is to pick up the data piece by piece right from the response stream instead of using the 'schema' and 'any' fields. If that is true you have to do the following steps:

- a) Modify the generate response data structure
- b) Replace the standard callback procedures "WSDL2R98_Any_start" and "WSDL2R98_Any_end" with your own ones.

So first remove 'schema' and 'any' from the response data structure and add your own fields and arrays to store the response data of the web service. It is your responsibility to decide what fields you need to store the response information. There is nothing I could advice for that.

Now locate that line of code where processing is delegated to "WSDL2R98_Any_start" and ""WSDL2R98_Any_end" and replace the procedure names with your own procedures that you have to develop right now. You probably want to name your own procedures "Local_Any_start" and ""Local_Any_end".

Open WSDL2R98 and copy "WSDL2R98_Any_start" and "WSDL2R98_Any_end" into the generated stub module. Do not forget to rename the procedures to "Local_Any_start" and "Local_Any_end" and to add prototypes for them.

Now remove all the code from these procedures and add your own logic to get the data. Use "depth", "name" and "path" to determine what kind of data arrived and put "value" into that field of the response data structure that you intended for to hold that piece of information.

You actually do not need to add any code to "Local_Any_start" because everything you need is delivered by "Local_Any_end". However you should keep "Local_Any_start" as an empty dummy procedure to prevent "WSDL2R98_Any_start" from being called.

Your comments are important to me! Please send me your comments about this FAQ. I will greatly appreciated it.

thomas.raddatz@tools400.de