

Re: how to deserialize variable element/node

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.xml/2006-11/msg00181.html>

- *From:* Gadget <gadget@xxxxxxxxxx>
 - *Date:* Sun, 19 Nov 2006 14:00:26 +1100
-

On 17 Nov 2006 20:48:34 -0800, wpmccormick@xxxxxxxxxx wrote:

Gadget wrote:

On 16 Nov 2006 19:54:06 -0800, wpmccormick@xxxxxxxxxx wrote:

wpmccormick@xxxxxxxxxx wrote:

Gadget wrote:

On 15 Nov 2006 19:28:42
-0800,
wpmccormick@xxxxxxxxxx
wrote:

I've a
complex
problem:

I need to re-explain this ...

Whether I deserialize this:

```
<foo>
.....
<bar>sometimes
a simple
string is
here</bar>
.....
</foo>
```

Re: how to deserialize variable element/node

Or this:

```
<foo>
.....
<bar>
<zip>sometimes</zip>
<zap>more
structure</zap>
<zoo>is
here</zoo>
</bar>
.....
</foo>
```

I want to end up with an object

```
class foo {
string bar;
baz[] Baz;
}
```

```
class {
string zip;
string zap;
string zoo;
}
```

In the first deserialization case, bar = "sometimes a simple string is here" and Baz is null. In the second bar is null and zip, zap and zoo are filled in.

Thanks

OK, so I think you're either misunderstanding XML or trying to use it in an unorthodox and incorrect manner.

Your two XML fragments would have to be represented in an XML schema by

specifying mixed content, as in one case <bar> contains TEXT and in the other it contains a set of tags.

Why do you feel the need to read this structure of data? Do you have an existing data set with this format strings, or are you responsible for creating the XML?

Your comment that your problem is 'rather simple' is not true. Perhaps your requirements are, but you seem to have created a very awkward problem for yourself.

Re: how to deserialize variable element/node

In XML your zip/zap/zoo should be inside a pair of <baz> tags, so you should have either of the following:

```
<foo>
<bar />
</baz>
<zip>sometimes</zip>
<zap>more structure</zap>
<zoo>is here</zoo>
</baz>
</foo>
```

```
<foo>
<bar>sometimes a simple string is here</bar>
<baz />
</foo>
```

This means you use the same class for 'foo'. In the first example there is no string in bar and in second case there is no object assigned to baz (it's null).

When the deserializer looks at the XML it sees <bar>, reads that it has no content, and sets it to be empty.

Next, it sees the tag <baz>, creates an instance of a baz, and reads the content tags, assigning each value to a property in the baz object via the reflection detected properties.

If baz is an empty tag then the object is set to null.

This weird mixing of meanings in XML produce ambiguity that the built-in serializer/deserializer can not resolve and won't even try to.

Cheers,
Gadget

I have an existing data set with this format and I have no control of it. The people that have forced it on me (and others who need to interface to their system) seem not to have a very good understanding and/or are using it in some unorthodox manner; perhaps in order to discourage deserialization?

So ... any ideas?

Yes, forget about serialization. You've been describing the classes you want to serialize/deserialize, but in fact you don't have the classes, you only have the dataset.

A better way of asking for a solution is to say "I have a set of XML of the following format. What would be the easiest way of reading/writing this to an object". Asking it by telling us the classes you've decided on is really muddying the water.

I find it difficult to believe any company with XML know-how would provide

Re: how to deserialize variable element/node

you with XML in both of formats you describe:

```
<foo>
.....
<bar>sometimes a simple string is here</bar>
.....
</foo>
```

OR

```
<foo>
.....
<bar>
<zip>sometimes</zip>
<zap>more structure</zap>
<zoo>is here</zoo>
</bar>
.....
</foo>
```

This is a very bizarre way for them to serialize data, so can I suggest you post two samples of the *actual* data so we can see the real structure?

Cheers,
Gadget

.