Subject: [solved] Extensions to Xmlize for arrays
Posted by cbpporter on Fri, 20 May 2011 09:59:43 GMT
View Forum Message <> Reply to Message

I have small problem with Xmlize and containers. The XML will look like this:

```
<cats>
   <item>...</item>
   <item>...</item>
   ...
</cats>
```

That is not very XMLish. I want:

```
<cats>
   <cat>...</cat>
   <cat>...</cat>
   ...
</cats>
```

Could we add this to XmlIO class:

```
class XmlIO {
...

public:
...
 template <class T> XmlIO operator()(const char *tag, const char *itemtag, T& var);
```

and after:

```
template <class T>
void Xmlize(XmlIO xml, const char* itemtag, T& var)
{
 var.Xmlize(xml);
}

...

template <class T> XmlIO XmlIO::operator()(const char *tag, const char *itemtag, T& var) {
 XmlIO n(*this, tag);
 Xmlize(n, itemtag, var);
 return *this;
}
```

...

```
template<class T>
void Xmlize(XmlIO xml, const char* itemtag, Vector<T>& data)
{
 XmlizeContainer(xml, itemtag, data);
}

...

template<class T>
void Xmlize(XmlIO xml, const char* itemtag, Array<T>& data)
{
 XmlizeContainer(xml, itemtag, data);
}
```

This change is only meant for Array and Vector marshaling, but could be extended to rest of the containers.

So now I can do:

```
xml("cats", cats);
```

for the original XML format, or:

```
xml("cats", "cat", cats);
```

for the format I want.

Subject: Re: Extensions to Xmlize for arrays
Posted by tojocky on Fri, 20 May 2011 15:20:53 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 20 May 2011 12:59
This change is only meant for Array and Vector marshaling, but could be extended to rest of the containers.

So now I can do:

```
xml("cats", cats);
```

for the original XML format, or:

xml("cats", "cat", cats);

for the format I want.

What about if your Array cats have items with array type?

Subject: Re: Extensions to Xmlize for arrays
Posted by mirek on Sun, 22 May 2011 17:18:57 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 20 May 2011 05:59I have small problem with Xmlize and containers. The XML will look like this:

```
<cats>
   <item>...</item>
   <item>...</item>
   ...
</cats>
```

That is not very XMLish. I want:

```
<cats>
   <cat>...</cat>
   <cat>...</cat>
   ...
</cats>
```

Could we add this to XmlIO class:

```
class XmlIO {
...

public:
...
 template <class T> XmlIO operator()(const char *tag, const char *itemtag, T& var);
```

and after:

```
template <class T>
```

```
void Xmlize(XmlIO xml, const char* itemtag, T& var)
{
 var.Xmlize(xml);
}

...

template <class T> XmlIO XmlIO::operator()(const char *tag, const char *itemtag, T& var) {
 XmlIO n(*this, tag);
 Xmlize(n, itemtag, var);
 return *this;
}

...

template<class T>
void Xmlize(XmlIO xml, const char* itemtag, Vector<T>& data)
{
 XmlizeContainer(xml, itemtag, data);
}

...

template<class T>
void Xmlize(XmlIO xml, const char* itemtag, Array<T>& data)
{
 XmlizeContainer(xml, itemtag, data);
}
```

This change is only meant for Array and Vector marshaling, but could be extended to rest of the containers.

So now I can do:

```
xml("cats", cats);
```

for the original XML format, or:

```
xml("cats", "cat", cats);
```

for the format I want.

I have no problem adding this. Could you post a complete patch please?

Mirek

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Mon, 23 May 2011 07:53:50 GMT
View Forum Message <> Reply to Message

tojocky wrote on Fri, 20 May 2011 18:20cbpporter wrote on Fri, 20 May 2011 12:59
This change is only meant for Array and Vector marshaling, but could be extended to rest of the containers.

So now I can do:

xml("cats", cats);

for the original XML format, or:

xml("cats", "cat", cats);

for the format I want.

What about if your Array cats have items with array type?

Not a problem. My patch does not affect your serialization, since you need to write extra code to change the sub item name. But if you do, you will end up with:

```
<cats>
  <cat>
    ...
    <whiskers>
      <whisker>...
    </whiskers>
  </cat>
  <cat>
  ...
</cats>
```

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Mon, 23 May 2011 07:57:36 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 22 May 2011 20:18

I have no problem adding this. Could you post a complete patch please?

Mirek

Sue, I'll upload the file as an attachment.

File Attachments
1) Xmlize.h, downloaded 2359 times

Subject: Re: Extensions to Xmlize for arrays
Posted by mirek on Tue, 24 May 2011 16:01:29 GMT
View Forum Message <> Reply to Message

Patch applied, thank you.

Mirek

Subject: Re: Extensions to Xmlize for arrays
Posted by sergeynikitin on Wed, 02 Nov 2011 07:33:44 GMT
View Forum Message <> Reply to Message

Can you do analog decision for ArrayMap (maybe also VectorMap ...)?

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Wed, 02 Nov 2011 09:31:00 GMT
View Forum Message <> Reply to Message

sergeynikitin wrote on Wed, 02 November 2011 09:33Can you do analog decision for ArrayMap (maybe also VectorMap ...)?

Sure, I'll look over it after I can check out a new nightly build.

Subject: Re: Extensions to Xmlize for arrays
Posted by sergeynikitin on Wed, 02 Nov 2011 12:43:55 GMT
View Forum Message <> Reply to Message

I tried to do it myself, but there is some error. I did not understand, so far, due to lack of time and put off for later.

Until replaced by a direct search for Array. This is certainly the wrong way.

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Mon, 30 Jul 2018 14:31:56 GMT
View Forum Message <> Reply to Message

This is one crazy necro, but I'm still using "wrong" labels for vector sub-items.

What could we do to solve this issue?

Subject: Re: Extensions to Xmlize for arrays
Posted by mirek on Mon, 30 Jul 2018 15:13:42 GMT
View Forum Message <> Reply to Message

What is your code?

Mirek

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Tue, 31 Jul 2018 08:01:22 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 30 July 2018 18:13What is your code?

Mirek
Well, here is the thing. I posted it way back. It worked for me. You said you applied it. sergeynikitin found some erros or other problems.

Time passed and I was forced to update U++ and remove the use of this code from the project. The code is lost. And there is something similar now in U++, but don't know exactly how to use it.

What does this do exactly:
template <class T> XmlIO List(const char *tag, const char *itemtag, T& var);

Best I can do is link to my old post: https://
www.ultimatepp.org/forums/index.php?t=msg&th=6032&go to=32509&#msg_32509

Subject: Re: Extensions to Xmlize for arrays
Posted by mirek on Tue, 31 Jul 2018 09:06:21 GMT
View Forum Message <> Reply to Message

Yes, List solves this for arrays. If I remember well, there was some overloading issue that prevented plain operator() to be used here.

Now I guess you will request something similar for maps, correct?

Subject: Re: Extensions to Xmlize for arrays
Posted by cbpporter on Tue, 31 Jul 2018 09:40:10 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 31 July 2018 12:06Yes, List solves this for arrays. If I remember well, there was some overloading issue that prevented plain operator() to be used here.

Now I guess you will request something similar for maps, correct?
Somewhere between "soon" and another 7 years .

Thank you! I'll give List a try. I've spent the last 3 days updating our XML configuration system and found that the resulting XMLs had this problem. It would be great if I can finally fix this.