
Subject: Vector a user defined struct for XML persistency

Posted by [sinpeople](#) on Sat, 17 Oct 2020 08:20:12 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi folks,

I am trying to put a user-defined Phase struct into a vector before using XML for persistency. It worked fine before I put it into a vector.

The code is below:

```
struct Phase
{
    int    id;
    double split;
    Vector<int> signal_groups;
    int    min_green;
    int    all_red;
    int    amber;
    int    skip;
    void Xmlize(XmlIO& xml);
};

void Phase::Xmlize(XmlIO& xml)
{
    xml
    ("id", id)
    ("split", split)
    .List("signal_groups", "signal_group", signal_groups)
    ("min_green", min_green)
    ("all_red", all_red)
    ("amber", amber)
    ("skip", skip)
    ;
}

struct JunctionCFG {
    Size          sz;
    int           lang;
    String        text;
    Vector<Size>   vector;
    VectorMap<String, int> score;
    VectorMap<String, String> map;
    ArrayMap<int, Size>   sizemap;
    double        number;
    bool          option;

    Phase phase;
```

```

// Comment here, other compiling error
// Vector<Phase> phase_grp;

void Init();
void Load();
void Save();
//void Dump();
void Xmlize(XmlIO& xml);
};

void JunctionCFG::Xmlize(XmlIO& xml)
{
xml
  ("Dimension", sz)
  ("text", text)
  .List("vector", "size", vector)
  ("Scores", score)
  ("map", map)
  ("sizemap", sizemap)
  ("number", number)
  ("option", option)
  ("phase", phase)
// .List("phase_group", "phase", phase_grp)
;
  XmlizeLang(xml, "Language", lang);
}

```

After the Vector<Phase> is added into the code, it has compiling error like the following:

Quote:

Core/Topt.h line 171 required from 'void Upp::AssertMoveable(T*)[with T= Phase]'
 Core/Topt.h line 157 error: no match for 'operator*'(operand type is 'Phase')

Ideally, I want the XML looks like

```

<phases>
  <phase>
    <id value="2"/>
    <split value="0.25"/>
    <signal_groups>
      <signal_group value="25"/>

```

```

    <signal_group value="50"/>
  </signal_groups>
  <min_green value="-986896"/>
  <all_red value="20"/>
  <amber value="4"/>
  <skip value="-1"/>
</phase>
<phase>
  <id value="2"/>
  <split value="0.25"/>
  <signal_groups>
    <signal_group value="25"/>
    <signal_group value="50"/>
  </signal_groups>
  <min_green value="-986896"/>
  <all_red value="20"/>
  <amber value="4"/>
  <skip value="-1"/>
</phase>
<phase>
  <id value="2"/>
  <split value="0.25"/>
  <signal_groups>
    <signal_group value="25"/>
    <signal_group value="50"/>
  </signal_groups>
  <min_green value="-986896"/>
  <all_red value="20"/>
  <amber value="4"/>
  <skip value="-1"/>
</phase>
<phase>
  <id value="2"/>
  <split value="0.25"/>
  <signal_groups>
    <signal_group value="25"/>
    <signal_group value="50"/>
  </signal_groups>
  <min_green value="-986896"/>
  <all_red value="20"/>
  <amber value="4"/>
  <skip value="-1"/>
</phase>
</phases>

```

Thank you very much for your support!

Best Regards

David WANG

File Attachments

1) [Capture.JPG](#), downloaded 421 times

Subject: Re: Vector a user defined struct for XML persistency

Posted by [Oblivion](#) on Sat, 17 Oct 2020 08:33:53 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

You have to mark the struct as Moveable. (Vector requires its elements to be moveable. Array doesn't.)

```
struct Phase : Moveable<Phase> // <--
{
    int id;
    double split;
    Vector<int> signal_groups;
    int min_green;
    int all_red;
    int amber;
    int skip;
    void Xmlize(XmlIO& xml);
};
```

Best regards,
Oblivion

Subject: Re: Vector a user defined struct for XML persistency

Posted by [mirek](#) on Sat, 17 Oct 2020 12:59:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Oblivion wrote on Sat, 17 October 2020 10:33Hello,

You have to mark the struct as Moveable. (Vector requires its elements to be moveable. Array doesn't.)

Actually, I guess the first reply should be "use Array instead of Vector"... :)

