Subject: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 13:37:35 GMT
View Forum Message <> Reply to Message

1>C:\uppdevel\uppsrc\Core/Vcont.h(132) : error C2512: 'Gdiplus::Pen' : no appropriate default constructor available
C:\uppdevel\uppsrc\Core/Vcont.h(132) : while compiling class template member function 'void Upp::Array<T>::Init(void **,void **)'
with
[T=Gdiplus::Pen]
.\Layers.cpp(39) : see reference to class template instantiation 'Upp::Array<T>' being compiled with [T=Gdiplus::Pen]

Pen dosn't have Pen() constructor so how i can use any container with Pen , Vector , Array ?

Layers.cpp(39)
pens.Reserve(Globals::RoadCategories);

Any suggestions ?

Subject: Re: Array & Gdiplus::Pen problem
Posted by mirek on Wed, 01 Aug 2007 14:14:48 GMT
View Forum Message <> Reply to Message

I do not think it is Reserve.

Can you post more Array related code please?

Mirek

Subject: Re: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 14:30:06 GMT
View Forum Message <> Reply to Message

It was exaclty Reserve
When i mark out Reserve i got another place

Pen *p;
.....
p = new Pen(this->m_pOutlin.....
.....
pens.Add(p);

And then it failed at Add

So finaly i had no solution and switched
from
void MetaLayer::GetPens(int type, Array<Pen> & pens) const

to
void MetaLayer::GetPens(int type, Vector<Pen *> & pens) const

and by hand managing memory of Pens

Subject: Re: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 14:46:44 GMT
View Forum Message <> Reply to Message

Tested more and:
When i comment out other method below

```
 void MetaLayer::GetSimplePens(int type, Array<Pen> & pens) const
 {
  pens.SetCount(Globals::RoadCategories);
  if (type == 0)
  {
   for (int i = 0; i < Globals::RoadCategories; i++)
   {
    pens.Set(i, new Pen(Color_(0,0,0), 3.));
    pens[i].SetEndCap(Gdiplus::LineCapDiamondAnchor);
    pens[i].SetStartCap(Gdiplus::LineCapDiamondAnchor);
   }
  }
  else if (type == 1)
  {
   for (int i = 0; i < Globals::RoadCategories; i++)
   {
    pens.Set(i, new Pen(Color_(255,0,0), 1.));
    pens[i].SetStartCap (Gdiplus::LineCapDiamondAnchor);
    pens[i].SetEndCap (Gdiplus::LineCapDiamondAnchor);
   }
  }
  else
   for (int i = 0; i < Globals::RoadCategories; i++)
    pens.Set(i, new Pen(Color_(0,0,0), 3));

 }
```

I can compile code with earlier metohod
When i enable this metod the compiler fails at other metod below!!!!

void MetaLayer::GetPens(int type, Array<Pen> & pens) const

WHats wrong with Array ?

Subject: Re: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 14:51:29 GMT
View Forum Message <> Reply to Message

After changing code to below everything is ok
Propably SetCount was the error but complier message pointed to incorect line in code ...

```
void MetaLayer::GetSimplePens(int type, Array<Pen> & pens) const
{
 pens.Reserve(Globals::RoadCategories);
 Pen *p;
 if (type == 0)
 {
  for (int i = 0; i < Globals::RoadCategories; i++)
  {
   p = new Pen(Color_(0,0,0), 3.);
   p->SetEndCap(Gdiplus::LineCapDiamondAnchor);
   p->SetStartCap(Gdiplus::LineCapDiamondAnchor);
   pens.Add(p);
  }
 }
 else if (type == 1)
 {
  for (int i = 0; i < Globals::RoadCategories; i++)
  {
   p = new Pen(Color_(255,0,0), 1.);
   p->SetStartCap (Gdiplus::LineCapDiamondAnchor);
   p->SetEndCap (Gdiplus::LineCapDiamondAnchor);
   pens.Add(p);
  }
 }
 else
  for (int i = 0; i < Globals::RoadCategories; i++)
   pens.Add(new Pen(Color_(0,0,0), 3));

}
```

Subject: Re: Array & Gdiplus::Pen problem
Posted by mirek on Wed, 01 Aug 2007 16:01:45 GMT
View Forum Message <> Reply to Message

Obviously, SetCount requires default constructor - you are creating default constructed object in the process...

BTW, this is only required if the new number of elements is greater, but SetCount has to contain the increase variant too.

Anyway, if you are reducing the number of elements, you can use also Trim (without default constructor requirement).

Mirek

Subject: Re: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 17:53:27 GMT
View Forum Message <> Reply to Message

I understand now why , Array gives me reference to inside element so they cannot be null , SetCount causes this to happen

Hoever there is a very wrong thing in MSVC8/NTL don't know in which propably in NTL.

Imagine one cpp file conatining

```
class Pen
{
  Pen(int a){}
}

//somewhere in code compiled first....

Array<Pen> x;
x.Reserve(10);
x.Add(new Pen(0));

//somewhere in code compiled after ....
Array<Pen> x;
x.SetCount(10);
```

Compiler will fail at Reserve
Then when You comment out Reserve, compiler will fail at Add!!!!
This is difficult sometimes to trace where is the deffinition which causes problem ...
And this is why i started posting this topic.

Subject: Re: Array & Gdiplus::Pen problem
Posted by mirek on Wed, 01 Aug 2007 18:45:28 GMT
View Forum Message <> Reply to Message

Well, I think that the trouble is usually caused by the fact that template implementation code is compiled AFTER everything else, way after concrete methods are used.

And, BTW, how could NTL affect error messages?

Mirek

Subject: Re: Array & Gdiplus::Pen problem
Posted by arturbac on Wed, 01 Aug 2007 21:03:03 GMT
View Forum Message <> Reply to Message

I Don't known
