Subject: Trouble with reference variable in THISBACK1
Posted by Tom1 on Thu, 04 Aug 2016 07:17:45 GMT
View Forum Message <> Reply to Message

Hi,

I'm working with U++ 9251 on Windows 10 64-bit using MSC9, MSC10 and MSC15 in 32 bit mode. I just can't figure out what's wrong with my code. It seems the reference variables do not work as parameters in THISBACKx and the actual vector items being referenced do not get updated in the process:

```
#include <Core/Core.h>

using namespace Upp;

class c_t: Moveable<c_t> {
public:
  int x;
  int y;

  c_t(){
   x=99;
   y=99;
  }
};

class RefTest{
typedef RefTest CLASSNAME;

 Vector<c_t> cv;
public:
        RefTest(){}

        void prepare_c(c_t &c){
         c.x=1;
         c.y=2;
     }

  void Run(){
  CoWork cw;
  for(int i=0;i<1;i++){
   c_t &ref=cv.Add();
   cw.Do(THISBACK1(prepare_c,ref));
  }
  cw.Finish();

  for(int i=0;i<cv.GetCount();i++){
```

```
   printf("%d %d\r\n",cv[i].x,cv[i].y);
  }
 }
};


CONSOLE_APP_MAIN{
 RefTest t;
 t.Run();
}
```

It just prints "99 99" whereas I'm expecting "1 2".

Help, anybody?

Best regards,

Tom

Subject: Re: Trouble with reference variable in THISBACK1
Posted by Tom1 on Thu, 04 Aug 2016 07:54:29 GMT
View Forum Message <> Reply to Message

Replying to myself, it seems using a pointer instead of a reference variable can workaround this problem. Still, it would be nice to know what's the actual problem with the reference variable.

Tom

Subject: Re: Trouble with reference variable in THISBACK1
Posted by mirek on Fri, 12 Aug 2016 06:33:06 GMT
View Forum Message <> Reply to Message

BEWARE: Pointers might compile, but do not work either!

Trouble is: Vector::Add invalidates existing references to elements. As the time when the 'job' is actually performed is undefined, it is likely that at that time the pointer is already invalid.

(Of course, with loop to 1, it works...)

Solution is either to SetCount before loop (if it is known) or to use Array.

Anyway, THISBACK1 indeed does not work with pointers. Solution is simple: Forget THISBACK1 and be modern, use MSC15, trunk U++ and C++11:

```
cw & [=] { prepare_c(cv.Add()); };
```

Subject: Re: Trouble with reference variable in THISBACK1
Posted by Tom1 on Fri, 12 Aug 2016 07:10:47 GMT
View Forum Message <> Reply to Message

Hi,

Thanks for pointing out that Vector::Add invalidates pointers, although I had already switched to Array for other reasons.

As for MSC15, I'm still waiting for MSC15 / MSC15x64 Protect to emerge...

It seems to me that THISBACK1 does work with pointers but not references. Well, anyway, I guess I'll need to do this in the old way and get prepared for being modern as soon as Protect gets modern too. :) In any case the new way looks neat.

Thanks,

Tom