
Subject: VectorMap inside VectorMap

Posted by [White_Owl](#) on Mon, 02 Feb 2009 22:08:39 GMT

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

What is wrong with this code?

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

```
using namespace Upp;
```

```
class mycontainer: Moveable<mycontainer> {  
public:  
    VectorMap<String, String> data;  
};
```

```
VectorMap<String, mycontainer> vm;
```

```
CONSOLE_APP_MAIN
```

```
{  
    mycontainer ca = vm.GetAdd("First");  
    ca.data.FindPut("first-first", "aa");  
    ca.data.FindPut("first-second", "bb");  
  
    mycontainer cb = vm.GetAdd("First");  
    cb.data.FindPut("first-first", "cc"); // run-time error  
    cb.data.FindPut("first-second", "dd");  
}Why do I get a run-time error in debug mode?
```

```
-----  
Fatal error
```

```
-----  
Assertion failed in C:/upp/uppsrc/Core/Vcont.h, line 18  
Broken pick semantics
```

```
-----  
OK  
-----
```

Subject: Re: VectorMap inside VectorMap

Posted by [kodos](#) on Tue, 03 Feb 2009 00:17:47 GMT

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

```
mycontainer ca = vm.GetAdd("First");
```

With this line you "pick" the content of the returned mycontainer. You probably want to use a reference:

```
mycontainer &ca = vm.GetAdd("First");
```

Subject: Re: VectorMap inside VectorMap
Posted by [White_Owl](#) on Tue, 03 Feb 2009 16:24:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

I thought for "pick" you have to use GetAddPick() method?

Anyway, I am looking for a way to redefine items in my dictionaries. And both dictionaries (inner and outer) should have unique sets of keys.

Subject: Re: VectorMap inside VectorMap
Posted by [White_Owl](#) on Tue, 03 Feb 2009 16:58:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ok, here is the code which does exactly what I want:

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

```
using namespace Upp;
```

```
class mycontainer: Moveable<mycontainer> {  
public:  
    VectorMap<String, String> data;  
};
```

```
VectorMap<String, mycontainer> vm;
```

```
void define_word(const String& outer_key, const String& inner_key, const String& value) {  
    mycontainer *ca = vm.FindPtr(outer_key);  
    if(!ca)  
        ca = &(vm.Add(outer_key));  
  
    if(ca->data.Find(inner_key)>=0)  
        ca->data.RemoveKey(inner_key);  
    ca->data.Add(inner_key, value);  
}
```

```
CONSOLE_APP_MAIN
```

```
{  
    define_word("First", "first-first", "aa");  
    puts(vm.Get("First").data.Get("first-first"));  
    puts("Sizeof(vm)=" + FormatInt(vm.GetCount()));  
    puts("Sizeof(vm{First}.data)=" + FormatInt(vm.Get("First").data.GetCount()));
```

```
    define_word("First", "first-first", "bb");  
    puts(vm.Get("First").data.Get("first-first"));  
    puts("Sizeof(vm)=" + FormatInt(vm.GetCount()));  
    puts("Sizeof(vm{First}.data)=" + FormatInt(vm.Get("First").data.GetCount()));  
}
```

I think GetAdd()/FindPut and other "double" methods already have all this checking and removing

old values. But I can not find which one of them actually does it without invalidating objects inside the container.

Subject: Re: VectorMap inside VectorMap
Posted by [kodos](#) on Tue, 03 Feb 2009 17:12:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

White_Owl wrote on Tue, 03 February 2009 17:24I thought for "pick" you have to use GetAddPick() method?

The methods with "pick" in the name, pick the argument that you pass the method. But in your first example you use the assignment operator which "picks" the value from the right to the left variable.

I think you want something like this:

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

using namespace Upp;

class mycontainer: Moveable<mycontainer> {
public:
    VectorMap<String, String> data;
};

VectorMap<String, mycontainer> vm;

CONSOLE_APP_MAIN
{
    mycontainer &ca = vm.GetAdd("First");
    ca.data.FindPut("first-first", "aa");
    ca.data.FindPut("first-second", "bb");

    mycontainer &cb = vm.GetAdd("First");
    cb.data.FindPut("first-first", "cc"); // run-time error
    cb.data.FindPut("first-second", "dd");
}
```

Like that you don't call the assignment operator but you just use the reference that get's returned from GetAdd. So there is no picking.

Subject: Re: VectorMap inside VectorMap
Posted by [White_Owl](#) on Tue, 03 Feb 2009 18:33:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ok. Right now I have this:

```
void define_word(const String& outer_key, const String& inner_key, const String& value) {  
    mycontainer &ca = vm.GetAdd(outer_key);  
    ca.data.GetAdd(inner_key) = value;  
}
```

That works exactly as I want.

I'm considering this problem solved.

Subject: Re: VectorMap inside VectorMap

Posted by [mirek](#) on Sat, 07 Feb 2009 14:03:53 GMT

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

You can save one line:

```
vm.GetAdd(outer_key).data.GetAdd(inner_key) = value;
```

Mirek
