
Subject: ArrayMap / Polymorphism question
Posted by [Kozaluss](#) on Fri, 16 Feb 2018 21:03:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello everyone!

I've discovered U++ yesterday and would really like to use it's potential.
But these were years since I've last used C++ - now I mainly work in Clarion - job reasons.

I am trying to write an application - it is supposed to have many different classes (machines) derived from one base class (base machine).

Then on top of that I need an array of pairs int and someclass from the set above.
I understand, that I should declare this like this:

```
ArrayMap<int,BaseMachine> ML;
```

where BaseMachine is the base class for all other machine classes.
So when I add some derived type machine, I do it this way:

```
ML.Add( NUID, MasterMachine() ); // NUID is generated int
```

Then I have a virtual method like this declared in every class:

```
virtual String Typ();
```

which now simply returns a predefined string.

And finally I have an ArrayCtrl "TabelaMaszyn" to display the contents of ML with this code:

```
for(int i=0;i<ML.GetCount();i++){  
    TabelaMaszyn->Add( ML.GetKey(i), ML[i].Typ(), ML[i].Nazwa );  
}
```

And the problem is, that ML[i].Typ() always returns the value generated by function of the base class, instead of being polymorphic, returning the value generated in derived class, which was passed during creation of ML[i] element.

So... what am I doing wrong here?
Please be polite :)
Thanks in advance.
