
Subject: object methods as friends

Posted by [Zorch Tierod](#) on Sun, 18 Apr 2004 00:29:56 GMT

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

Hey there,

I've got an application that uses two types of objects, each with several different method procedures.

What I'm trying to do is to calculate some stuff based on the interaction of one of the objects with an array of the other type of objects. Illustrative example (but not the REAL application!):

```
sb = {bullet, param1:.....}    ;prototype for bullet object
st = {target, param1:.....}    ;prototype for target object
```

```
target_array = obj_arr('target',10)    ;make a bunch of targets
projectile = obj_new('bullet')          ;make a bullet
```

```
jtarget = which_hit(target_array, projectile) ;return which target is
hit
```

My problem seems to be that the function 'which_hit' has no access to the data of either the 'bullet' or the 'target' objects, because it is not a method of either - and I can't make it a method of both...

If I were doing this in C++, I would declare 'which_hit' as a friend of both classes, and this would give me access to the data. Or, living dangerously, I could just declare all data fields as public accessible.

Anyone have advice on this problem? Maybe I'm overlooking something basic (it wouldn't be the first time....). I really hate to drop back to just using structures (equiv of all-public data), because the encapsulation really works well for my particular problem.

Thanks!

--
