[**Error C2662， cannot convert ‘this’ pointer from ‘const class ’ to ‘class &’**](http://blog.csdn.net/guosong421/article/details/5505583)

分类： [Error](http://blog.csdn.net/guosong421/article/category/535240)2010-04-20 09:581723人阅读[评论](http://blog.csdn.net/guosong421/article/details/5505583#comments)(0)[收藏](javascript:void(0);)[举报](http://blog.csdn.net/guosong421/article/details/5505583#report)

**[c-sharp]** [view plaincopyprint?](http://blog.csdn.net/guosong421/article/details/5505583)

1. class Point3d
2. {
3. public:
4. Point3d(float x=0.0,float y=0.0,float z=0.0)
5. :\_x(x),\_y(y),\_z(z)
6. {
7. }
8. float GetX() {return \_x;}
9. float GetY() {return \_y;}
10. float GetZ() {return \_z;}
11. private:
12. float \_x,\_y,\_z;
13. };
14. inline ostream& operator<<(ostream&out,const Point3d& pd)
15. {
16. out<<pd.GetX()<<" "<<pd.GetY()<<" "<<pd.GetZ()<<endl;
17. return out;
18. }



上述的代码是导致错误的例子。错误主要的原因是const类型的对调用非const类型的方法导致的。

由于const对象在调用成员函数时，会将this指针强制转换成const this指针，它调用成员函数时会去找对应的const Get\*函数，而编译器无法将非const类型的Get\*函数转换成const类型的Get\*函数，因此出现编译错误。

解决方法就是将Get\*函数转化为const类型的函数

在对应函数后面加上const关键字

**[c-sharp]** [view plaincopyprint?](http://blog.csdn.net/guosong421/article/details/5505583)

1. class Point3d
2. {
3. public:
4. Point3d(float x=0.0,float y=0.0,float z=0.0)
5. :\_x(x),\_y(y),\_z(z)
6. {
7. }
8. float GetX() const{return \_x;}
9. float GetY() const{return \_y;}
10. float GetZ() const{return \_z;}
11. private:
12. float \_x,\_y,\_z;
13. };
14. inline ostream& operator<<(ostream&out,const Point3d& pd)
15. {
16. out<<pd.GetX()<<" "<<pd.GetY()<<" "<<pd.GetZ()<<endl;
17. return out;
18. }



因此书《深度探索C++对象模型》第一章关于对象模型 P2页举得例子有问题的。它的代码是上面出错的代码。