Adapter 패턴

타입

Structural Pattern

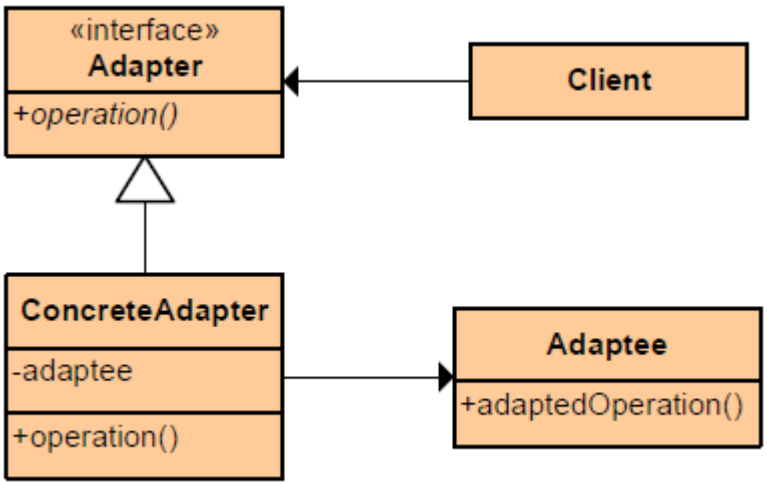
문제

기존에 개발되어 있는 모듈을 우리 시스템에서 사용하려 했더니 인터페이스가 달라 사용하기 힘들다.

해결

- 다중 상속을 이용한다. (클래스 어댑터)
- 클래스 내부에 기존에 개발되어 있는 모듈을 위한 객체를 멤버 변수로 선언한다. (객체 어댑터)

클래스 다이어그램



예제

class_adapter.cpp

```
#include <iostream>
using namespace std;

struct POINT
{
    int x,y;
```

```
};

// 도형
class CShape
{
public:
    virtual void GetShapeInfo(POINT& ptTopLeft, POINT& ptBottomRight)
{}
};

// 사각형 보기
class CRectView
{
public:
    void GetOrigin(int x, int y)
    {
        cout << x << " " << y << endl;
    }
    void GetSize(int nWidth, int nHeight)
    {
        cout << nWidth << " " << nHeight << endl;
    }
};

// 사각형
class CRectangle : public CShape, public CRectView
{
public:
    virtual void GetShapeInfo(POINT& ptTopLeft, POINT& ptBottomRight)
    {
        GetOrigin(ptTopLeft.x, ptTopLeft.y);
        GetSize(ptBottomRight.x - ptTopLeft.x, ptBottomRight.y -
ptTopLeft.y);
    }
};

int main(void)
{
    CShape* rect = new CRectangle;

    POINT pt1 = {3, 4};
    POINT pt2 = {7, 8};

    rect->GetShapeInfo(pt1, pt2);

    delete rect;

    return 0;
}
```

object_adapter.cpp

```
class CRectangle : public CShape
{
private:
    CRectView pView;
public:
    virtual void GetShapeInfo(POINT& ptTopLeft, POINT& ptBottomRight)
    {
        pView.GetOrigin(ptTopLeft.x, ptTopLeft.y);
        pView.GetSize(ptBottomRight.x - ptTopLeft.x, ptBottomRight.y -
ptTopLeft.y);
    }
};
```

나머지 부분은 클래스 어댑터와 동일

참고

http://en.wikibooks.org/wiki/C%2B%2B_Programming/Code/Design_Patterns#Adapter
http://showmiso.tistory.com/120