

## C++FA 2.1 Event Example

This simple example shows how to create an event in C++. This example fits in section C++FA 2.1 of *C++ with Financial Applications* by Ben Van Vliet, available at [www.benvanvliet.com](http://www.benvanvliet.com).

### Event.h

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

template< class RetType, class ArgType >
class Event
{
    ///////////////////////////////////////////////////
private:
    class DelegateBase
    {
    public:
        virtual ~DelegateBase() {}
        virtual RetType operator()( ArgType ) = 0;
    };
    ///////////////////////////////////////////////////
public:
    template< class ClassType >
    class Delegate : public DelegateBase
    {
        typedef RetType ( ClassType::*Function )( ArgType );
    protected:
        ClassType *m_Obj;
        Function m_Func;
    public:
        Delegate( ClassType *a, Function f ) : m_Obj( a ), m_Func( f ) {}
        RetType operator()( ArgType arg )
        {
            return ( m_Obj->*m_Func )( arg );
        }
    };
    ///////////////////////////////////////////////////

private:
    DelegateBase *m_Del;
public:
    ~Event()
    {
        delete m_Del;
    }
    RetType operator()( ArgType arg )
    {
        return ( *m_Del )( arg );
    }
    Event &operator+=( DelegateBase *d )
    {
        m_Del = d;
    }
};
```

```

        return *this;
    }
};

////////////////////////////////////

typedef Event< void, int > EventHandler;

```

## MyClass.h

```

#include <iostream>
#include "MyOther.h"
using namespace std;

class MyClass
{
private:
    MyOther *m_Other;

public:
    // Create an instance MyOther class and
    // subscribe to the event using += notation.
    MyClass()
    {
        m_Other = new MyOther;
        m_Other->OnUpdate += new EventHandler::Delegate< MyClass >(
this, &MyClass::Print );
        m_Other->RaiseEvent();
    }
    ~MyClass()
    {
        delete m_Other;
    }

    // Event handler function definition.
    void Print( int x )
    {
        cout << x << endl;
    }
};

```

## MyOther.h

```

#include <iostream>
#include "Event.h"
using namespace std;

class MyOther
{
public:
    // This class contains the event here.

```

```
    EventHandler OnUpdate;

    void RaiseEvent()
    {
        OnUpdate( 4 );
    }
};
```

### **Main.cpp**

```
#include "MyClass.h"

int main()
{
    MyClass *m_Obj = new MyClass;
    delete m_Obj;

    return 0;
}
```