

The results are in! [See the 2018 Developer Survey results.](#) »



C++ std::function bind callback between libraries without exposing method API

I have looked at quite a few links before asking this question and read quite a lot around std::function, std::bind and callbacks in C++. As I understand, the concept is mostly for event handlers to notify listeners after a certain event has happened. I am struggling to find the right implementation when it applies across library boundaries as I am a newbie into this style of programming.

Here is the situation or the design I need to implement:

I have a library A which has a private function in a class which accepts certain data in and does some processing. There is a libraryB which provides the same data which the function in library A needs, but libraryB exposes a std::function to receive that data. Other library needs to bind its function to its callback to receive the data. So, I need to bind libraryA's func to libraryB's std::function in my Application class.

```
Inside libraryA {

Class A {

private:
    AA objAA;
}

Class AA {

private:
    void func(int x) {
        //this is the function I want to tie to callback in library libB without
        exposing the method api
        //How can I expose a public api to return the address of this function so
        that the app can bind it to libraryB's callback ?
    }
}

}

Inside libraryB {

Class B {
public:
    void registerCallback(std::function<void (int)> callmePls) {
        m_callback = callmePls;
    }
private:
    typedef std::function<void (int)> theCallback;
    theCallback m_callback;
}

}

Inside my Application which uses both libraryA & libraryB {
//How can I bind/assign "func" in libraryA to the callback in libraryB ?
}
```


How can I expose a public api from libraryA to return the address of the interested function so that the app can bind it to libraryB's callback ? How can I bind/assign "func" in libraryA to the callback in libraryB ?

[c++](#) [c++11](#) [callback](#) [std-function](#) [stdbind](#)

edited Aug 5 '16 at 13:20

 **Yakk**
156k 17 150 319

asked Jul 22 '16 at 17:43

 **Game_Of_Threads**
1,155 7 30

1 Answer

with the bodies implemented in a .cpp file (not in the header):

```
class A {
```

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH

 Google

Facebook

