
Subject: Re: RPC_METHOD how to "define"
Posted by [dolik.rce](#) on Tue, 31 Jan 2017 21:10:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi NilaT,

NilaT wrote on Tue, 31 January 2017 16:56 How do i extract that weird define?
`#define RPC_METHOD(name) void name(RpcData& rpc); INITBLOCK { Register(#name, name);
} void name(RpcData& rpc)`

So in particular, how can I use the example method `getClientList` as a member function?
And whats `INITBLOCK` and `#name`?

First a little hint: In TheIDE open a file where `RPC_METHOD` is used and hit Build -> Preprocess (or Alt+F7). It will show you the file with all includes included and macros expanded.

Now, to decipher the macro:

The `INITBLOCK` is another U++ macro, which allows you to add a code block that will be executed when the program starts, before `main()` is executed. It is often used to register methods or plugins.

`#name` creates a string literal of the value passed to the `name` parameter. So the `"RPC_METHOD(MyMethod){ some_code; }"` expands to this:
`void MyMethod(RpcData& rpc);
INITBLOCK {
 Register("MyMethod", MyMethod);
}`

```
void name(RpcData& rpc) {  
    some_code;  
}
```

Note that I added some newlines for better readability. So what the macro does, step by step, is:

- 1) The function is declared.
- 2) It is registered, so that the RPC server knows that the method exists.
- 3) The function is defined. Here the code block that follows macro handily becomes the function body.

Does this explanation make it clearer?

To create the methods as class members, you'll have to do the same steps as the macro does, but with a member functions. That's is declare and/or define the member function and register it before it can be used. It would be probably ok to call `Register` in the class constructor. So it could look somewhat like this:

```
class MyRpcClass {  
public:  
    typedef MyRpcClass CLASSNAME;  
    void MyMethod(RpcData& data);  
    MyRpcClass() {  
        Register("MyMethod", THISBACK(MyMethod));  
    }  
}
```

```
}
```

Warning: I haven't tested the code (and I'm not even familiar with how the RpcServer works internally), so it will probably need a bit more work. But in general, it should be possible to do this.

Best regards,
Honza
