

Hi folks,

Newbie here.

I got this "call to implicitly-deleted copy constructor" error in this very simple program.

The error changes to "no matching constructor for initialization of xxx" after I added a copy constructor into the class. Adding a default constructor explicitly into the code didn't get rid of this new error.

The code is attached with the copy constructor and default constructor in comment. If the .push\_back statement is commented, no error will show up. I guess it has something to do with the containers, which i am not familiar at all.

Please help me. Thank you very much!

Best Regards

David

=====

```
#include <Core/Core.h>
#include <deque>
```

```
using namespace Upp;
using namespace std;
```

```
struct AMessage
{
    virtual void Serialize(Stream& s) = 0;
    virtual ~AMessage() {}
};
```

```
struct SignalStatus : Moveable<SignalStatus> {
    int sg_ID;
    int sg_Color;
    int sg_Elapsed;
    virtual void Serialize(Stream& s) {
        s % sg_ID % sg_Color % sg_Elapsed;
    }
};
```

```
struct SignalStatusMessage : AMessage {
```

```

int nTime;
Vector<SignalStatus> status;
/*
SignalStatusMessage(){};
SignalStatusMessage(SignalStatusMessage& temp)
{
    nTime = temp.nTime;
    status.clear();
    for(int i=0; i<temp.status.size(); i++)
    {
        SignalStatus sgStatus;
        sgStatus.sg_ID = temp.status[i].sg_ID;
        sgStatus.sg_Color = temp.status[i].sg_Color;
        sgStatus.sg_Elapsed = temp.status[i].sg_Elapsed;
        status.push_back(sgStatus);
    }
}
*/
virtual void Serialize(Stream& s) {
    s % nTime % status;
}
};

```

```

CONSOLE_APP_MAIN
{
    //deque<String> deqAbs;
    SignalStatusMessage sigStatus;
    deque<SignalStatusMessage> deqSigStatus;

    deqSigStatus.push_back(sigStatus);
}

```