

Not sure if you are directly using the BarcodeTest program, anyway, the following line generates the problem you referred to

```
GUI_APP_MAIN
{
    auto a=MyApp();           <===== This line
    a.Sizeable().MaximizeBox();
    a.Maximize();
    a.Run();
}
```

while MyApp has this definition

```
struct MyApp : WithBarcodeTestLayout<TopWindow>
{
    MyApp(){
        CtrlLayout(*this, "Barcode Test");
        input<=<=THISBACK(Updated);
        print.SetImage(CtrlImg::print())<=<=THISBACK(Print);
        top<=<=2;
        left<=<=2;
    }
}
```

```
virtual void Paint(Draw& w);
```

```
void GenBarcode();
void Print();
```

```
void Updated()
{
    Refresh();
    Code128 c(String().Cat()<<~input);
    EAN ean(AsString(~input));
    PDF417 pdf(AsString(~input));
    richview.SetQTF(String("[ ").Cat()
    <<c.DisplayText("Hi, U++ user!")
    .Color(Red()).BarRatio(36)
    <<"&&" // new line
    <<ean.Type(EAN::EAN8UPCAEAN13)
```

```

        .DisplayText("Best Seller")
        <<"&&"
        <<pdf.FixedAspectRatio(3,2)
    );
}

```

```
typedef MyApp CLASSNAME;
```

```
};
```

There are not even any data members (except ones inherits from its ancestors) in the MyApp struct definition, so it could not be because of anything from my barcode library.

I am also puzzled on that line. Logically local variable a should be default constructed once, instead of a default construct of a temporary and then an assignment, so it should be equivalent to

```
MyApp a;
```

Apparently I was wrong. Anyway, change the line to above fixed the problem. I was wondering why should I use the more cumbersome form at first place. So if you are using the BarcodeTest program directly, change the GUI_MAIN to the following form will fix your problem:

```

GUI_APP_MAIN
{
    MyApp a;
    a.Sizeable().MaximizeBox();
    a.Maximize();
    a.Run();
}

```

HTH
