

---

Subject: Re: A little theming

Posted by [Didier](#) on Sun, 24 May 2009 09:33:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

The compile problem is not only on MINGW, there is the same problem on linux ==> the "problem" comes from the compiler GCC.

Here is a valid correction but it needs to be optimised (on point 2):

In Theme.cpp

-----

1 - I removed the second template parameter: V is supposed to be a Value type since you use ' v.Is<Upp::String>' just let C++ inheritance do it's work.

```
//template <typename T, typename V>
template <typename T>
inline void SetIfNotNull(T& t, Value v)
{
    if ( v.Is<Upp::String>() && ((String)v == "null") )
        t = Null;
    else
        if (!IsNull(v))
            t = v;
}
```

2 - You have to add a temporary object: Image img because problems when using temporary references: It is not excluded that ImageBuffer img( img ); will use the address of the passed object which would be completely false for a temporary reference. Even though the problem is still here after correction (img is a temporary to the scope) at least it is visible while reading the code and not left to compiler will.

```
Value Theme::StringToObject(const String& s, const String& def) {
    Vector<String> v = Split(s, ' ');
    if (v.GetCount() == 0)
        return Null;
    if (v[0] == "png") {
        Image img = PNGRaster().LoadFileAny(AppendFileName(dir, def));
        ImageBuffer img( img );
        if (img.IsEmpty())
```

In Theme.h

-----

3 - remove the Theme:: from the following line

```
void Theme::LoadEditField(EditField::Style& d, const VectorMap<String, String>& set, const String& dir, const String& file);
```

Question:

-----

- 1 - Is there some documentation for the 'theme.ini' file syntax ?
- 2 - Can I use several themes in one application (like have two tabs not using same theme) or add in 'theme.ini' several different versions of skin for a ctrl ?

Skulpture looks good on linux