Subject: Problem re-compiling an application that was working well
Posted by jibe on Thu, 26 Jun 2014 16:41:19 GMT
View Forum Message <> Reply to Message

Hi,

Don't know if it's the right place for that, but don't know where to put elsewhere...

As I said there, I have a problem with an application that was working well, compiled a little more than one year ago. Nothing was changed in the sources, only the version of upp libraries and theide are different.

I tried to compile it on Linux Mint Maya (where I developped it) and on Scientific Linux 6 (always 32 bit, as it was done last year).

Upp has been updated on Maya, and it's a fresh install on Scientific Linux, where I just imported the sources of this application. In the two cases, I have the same result : some actions done with this application freeze it totally. This happens always with the same actions, and others features of this application are working very well in the 2 new compiled versions.

In the 2 cases, I recompiled all using the bomb icon (rebuild all). The log mentions "Heap leaks detected".

This happens when the application opens a new window to create a new record in a table of a database. The windows opens well, and then the application freezes. I'll try to trace better what happens and more exactly when the heap leaks occurs, but I'm surprised that all was working very well before I recompile. The old version is working well on my Maya computer, and the new version has exactly the same problem on the computer where the old version is installed, when no recent updates have been done.

So, I don't think that it's a problem with some system libraries. Seems more that the problem is with some of upp library... Did one of them had changes that could explain the problem ? Is there some changelog file or anything that could help me to find where is the problem ?

Subject: Re: Problem re-compiling an application that was working well
Posted by jibe on Fri, 27 Jun 2014 10:16:58 GMT
View Forum Message <> Reply to Message

Hi,

This is weird... I have this code :

```
virtual void MyDialog::Paint(Draw& w) {
 if (couverture.IsEmpty()) {
  couverture = StreamRaster::LoadFileAny(ConfigFile("nocover.jpg"));
  cover.Set(couverture);
 }
```

```
 TopWindow::Paint(w);
}
```

It's freezing when the image is not there. Ok. So, I added a test before the call to cover.Set() :

```
virtual void MyDialog::Paint(Draw& w) {
 if (couverture.IsEmpty()) {
  couverture = StreamRaster::LoadFileAny(ConfigFile("nocover.jpg"));
  if (!couverture.IsEmpty()) cover.Set(couverture);
 }
 TopWindow::Paint(w);
}
```

and it works even if the image is missing.

But I don't understand why it is working with the old version even if the image is not there...
