Subject: How does widget/variable destruction in U++ work?
Posted by crydev on Wed, 12 Dec 2012 09:53:28 GMT

View Forum Message <> Reply to Message

I am working on a program to manage movies on my computer. I have already built a lot, and I am testing all the time, also on memory usage. Now a question came to me. I use task manager to find out which actions take up memory, but I notice that when I execute a block as following:

```
void MovieManager::Options()
{
 SettingsDialog().Execute();
}
```

Which creates an instance of my settings dialog form, the memory usage increases by 600 KB. This is not weird, since I have a lot of controls, loading code and images inside there. However, when I close the dialog again by using:

```
Close();
```

The 600 KB memory that has been eaten is not being freed. Logically using C++ this way, the SettingsDialog variable that I created inside the function is destroyed as soon as the function finishes, and the used memory is freed.

Now I know that task manager is not the greatest tool to monitor memory usage as it is not quite as accurate as it should be. But this situation certainly brought up the question to me as it were: How does variable destruction and memory releasing work in U++?