

---

Subject: Re: Some Shocking Facts You Probably Didn't Want to Know about U++ Framework...

Posted by [Klugier](#) on Mon, 16 Nov 2020 23:30:09 GMT

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

---

Hello Mirek,

I see that your comment on the reddit caused controversy :) We collected a lot of valuable data. From my point of view - the three main causes why they do not want to use our technology are:

- impossible to write in other IDE's than TheIDE.
- not easy to integrate with other build systems. If I want to add GUI part for existing code (front-end) with U++, this is not possible. Everything needs to be integrated with ecosystem. I know that many people like to work with CMake as a build system, so why not let them work with Upp :)
- custom allocator is not selling point - we should sell it that our GUI is super responsive thanks to that :) Anyway, we should think about it and what we would like to do with that.

In my opinion we should focus on U++ as library (only core libraries Console + GUI) and provide it as collection of .so and .dlls. Thanks to that our libraries can be integrated with other build system easily. In this case we should do not bundle allocator - it can works within library, but externally it should be hidden. So, all binaries linking to upp libraries should be in USEMALLOC mode.

I am thinking about creating tool called udeploy that will compile selected uppsrc libraries and produce folder with shared libraries (.dlls) and headers. udeploy tool should copy headers from packages. In Upp not as a platform mode we should have includes like this:

```
#include <Upp/CtrlLib/CtrlLib.h>
```

With the change described above we should get out of the niche, however the investment in this case is huge. This will also allow Upp to be used in bigger projects.

Should, we do something with aggressive keyword in README.md? I see that there are some remarks about it.

Klugier

---