
Subject: Re: Know what you're using. Size of some common types.

Posted by [Lance](#) on Tue, 28 Dec 2021 19:27:43 GMT

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

Novo:

By moving similarly aligned items together (combined two separated bitfield section, and move int8 item together with them)

```
int8  push;
int8  light;
bool  horz:1;
bool  jump:1;
bool  track:1;
bool  autohide:1;
bool  autodisable:1;
bool  is_active:1;
```

We can save like 16 bytes on 64 bit platform. If we try harder, like declaring linesize, etc as int8, we can save some more bytes. But these are all marginal.

What I have in mind is to get rid of the 4 Buttons completely. That way we can save around 1K in each ScrollBar object. ScrollBar is definitely a class that is worth rewritten. The rewriting work might even not be too difficult.

Here is a function from ScrollBar.cpp

```
int ScrollBar::GetMousePart()
{
    int q = -1;
    for(int i = 2; i >= 0; i--)
        if(HasMouseIn(GetPartRect(i))) {
            q = i;
            break;
        }
    return q;
}
```

The slider area is divided into 3 parts, the upper blank area, the slider button, the bottom blank area. We can divide it into 5(or seven), with addition to accomodate prev,next (and even prev2, next2: anybody can ecudate me on what these two buttons are doing? I don't see it on the GUI at all)

These kind of refinement do not add functionalities but still contribute to a better U++ experience.
