Subject: Re: Proposed change to U++ to allow owning children.
Posted by mirek on Sat, 19 Mar 2011 09:59:38 GMT
View Forum Message <> Reply to Message

Lance wrote on Fri, 18 March 2011 20:30Thank you for giving it a try.

I was surprised to know there was actually a speed penalty. Maybe that's because of the for loop. Some really smarter compiler will tranlsate the for loop to movsb, movsw, movsdw, etc, then the for loop will no longer pose a speed penalty.

Nope, they are slightly slower than series of assignments.

Actually, smart compiler might unroll that loop into series of assignments anyway.

Quote:
I was thinking set bit fields will be slower. Maybe just simple put the bitfields in a union and the whole flags field to 0.

Now that would be smarter. Still, compiler are good, I would not be surprised if they replaced those assignements anyway.

Anyway, we are speaking here about negligible impacts on result (this area of code is definitely irrelevant optimization-wise).

But my primary complaint is that the loop is poor choice for quality of code, too error-prone.

Quote:
Then here in the Draw.h

```
class Font : AssignValueTypeNo<Font, FONT_V, Moveable<Font> >{
 union {
  int64 data;
  struct {
   word  face;
   word  flags;
   int16 height;
   int16 width;
  } v;
 };
...

 Font()  { data=0; } // that's the whole point why
                     // a union is introduced IMHO
```

Data is used to hash Font value and to compare it too.

Anyway, in any case there are 4 stable fields in union. Such situation is easily manageable.

Ctrl fields are much more numerous and change over time.