Subject: two small bugs to check for GLCtrl
Posted by luoganda on Fri, 20 Feb 2026 19:32:04 GMT
View Forum Message <> Reply to Message

this was on older versions and also in v18458.
Tested only on Linux !!

edit: (!!confirmed on linux - with+withoutGTK and GLCtrl/GLDraw no '-' drawn, on win11 ok)
When using w.DrawText with '-number'(eg -24.5) in it
that - before number is not drawn at all, is this the intent?
This is in GLCtrl/GLDraw ver, but i think in regular Draw ver works ok.
For workarround i now use + prepended in front of numbers, so when
- is in front of number, no + is drawn.
======

Not sure about this but worth checking.
In GLCtrl there is

```
private:
#ifdef PLATFORM_WIN32
...
public:
 GLPane() { NoWantFocus(); }
...
#endif
```

for winVer - allOk.
But for linux/gtk - no public(upper private still holds here) - is this the intent?
I make Sync public for myself becasue i need it sometimes - but would be worth
checking if for example one would subClass GLCtrl and would need some of the vars/funcs.

```
#ifdef PLATFORM_POSIX // we assume X11 or GTK
```

In linux ver - there are no public(or protected) methods whatsoever.
======

edit: (!!confirmed on win11 using both case K_A:case 'a' will fire twice, on linux it fires once - only 'a' which is probably how it should work,
altough this is not highest priority since one must check if <65536 anyway. One thing on win11 is plus - that one can catch eg K_A when released - by current implem. not possible on linux debian)
Not sure about this either - but worth checking:
On win11 same app worked, but on linux it doesn't

```
bool Key(dword key,int c){
  switch(key){
    case K_A:
      int abc=0;
      return true;
   }
  return false;
}
```

For workarround on linux - i now use "case K_A:case 'a':".
If correct would be "case 'a':case 'A':"(because it's character code and not key-code),
then case K_A should probably not work on winVer too.
Which is 'the way to go ver', case 'a'?
probably yes - if one reads the docs - in that case i guess K_A is only defined so
that it can be later used as a helper to eg K_CTRL_A or K_SHIFT_A etc.
====

By the way - i noticed this in newer uppVers:
using TcpSocket = Socket <= for backward compatibility,
but it should probably not be marked as just 'for backward compatibility',
since there are also UdpSockets - are they avail in Upp?