Subject: How to monitor key release event?
Posted by Klugier on Sat, 26 Apr 2014 23:01:13 GMT
View Forum Message <> Reply to Message

Hello,

I am writing small platformer game with map editor and I would like to handle key release events. Is there any way to do this in current version of U++?

What I have spied is that Ctrl::Key(dword key, int count) returns two events(X11):

Key: 130897, 1 // <- Key press event.
Key: 1179475, 1 // <- Key release event.

Of course I can handle it manually (by absolute key value), but it isn't elegant solution and probably it isn't platform independent.

Sincerely,
Klugier

Subject: Re: How to monitor key release event?
Posted by dolik.rce on Sun, 27 Apr 2014 08:14:20 GMT
View Forum Message <> Reply to Message

Hi Klugier,

If I remember correctly, the K_KEYUP value should be what you're looking for. Just check if (key & K_KEYUP) is true or not. This an other flags used in key-codes are defined directly in CtrlCore.h.

Best regards,
Honza

Subject: Re: How to monitor key release event?
Posted by Klugier on Sun, 27 Apr 2014 13:33:02 GMT
View Forum Message <> Reply to Message

Hello Honza,

I have solved this problem. Thanks for tip Honza!

This is my code:

```
bool GameCtrl::Key(dword key, int count)
{
 if (key == K_RIGHT) {
  moveRight = true;
  return true;
 }
 else if (key == K_LEFT) {
   moveLeft = true;
  return true;
 }
  else if (key == K_SPACE) {
  jump = true;
  return true;
 }

 if (key & K_KEYUP) {
  dword keyUp = (key & ~K_KEYUP);

  if (keyUp == K_RIGHT) {
   moveRight = false;
   return true;
  }
  else if (keyUp == K_LEFT) {
   moveLeft = false;
   return true;
  }
  else if (keyUp == K_SPACE) {
   jump = false;
   return true;
  }
 }

 return false;
}
```

BTW, I think that U++ should offer following method "Ctrl::KeyUp(dword key, int count)". This will facilitate a few things...

Sincerely,
Klugier

Subject: Re: How to monitor key release event?
Posted by ManfredHerr on Wed, 03 Dec 2014 15:07:54 GMT
View Forum Message <> Reply to Message

My findings with KEY_UP are that it is NOT platform independent at all. First of all, it depends which key is pressed. The arrow keys are generating press and release events for you. But if you wait for key release events for aplha keys, e.g. 'a', then you wait forever in X11. But if you use GTK then a key release event comes up: KEY_UP + 'A'. This means that the key symbols for press and release differ. I tried it with a "keyhook" in order to be at a lower level. No change.

The keyhooks are called in Ctrl::DispatchKey. But this obscure selection of keys, having a release event or not, already takes place in the platform dependent event processing.
I miss a hook to monitor every key on the keyboard with press and release.

Subject: Re: How to monitor key release event?
Posted by slashupp on Wed, 11 Feb 2015 08:19:58 GMT
View Forum Message <> Reply to Message

(delete this pls)

Subject: Re: How to monitor key release event?
Posted by mirek on Wed, 11 Feb 2015 08:30:18 GMT
View Forum Message <> Reply to Message

ManfredHerr wrote on Wed, 03 December 2014 16:07My findings with KEY_UP are that it is NOT platform independent at all. First of all, it depends which key is pressed. The arrow keys are generating press and release events for you. But if you wait for key release events for aplha keys, e.g. 'a', then you wait forever in X11. But if you use GTK then a key release event comes up: KEY_UP + 'A'. This means that the key symbols for press and release differ. I tried it with a "keyhook" in order to be at a lower level. No change.

You should not use ASCII (like 'A' + KEY_UP), but K_A|K_KEYUP.

Subject: Re: How to monitor key release event?
Posted by ManfredHerr on Thu, 12 Feb 2015 19:58:39 GMT
View Forum Message <> Reply to Message

Hi,
thank you for responding, I hardly can believe it!
To make it clear: I don't want to have a key-logger! I just want to monitor the keyboard events then when my application has the input focus, with key press and key release info in the proper time. It seems that this is possible in a portable way despite of security issues. But how you do it in UPP?
In the meantime, I decided to go back to FLTK that provides better support on that, and claims to be portable too. Sorry.
My intent is, to assign a particular function to a set of keys of the keyboard, that can be invoked

independent and in parallel, as long as the key is pressed. This function has nothing to do with text. To give an example: In former days there were games for two players. Each of the two controlled an object (his object), one of them with the cursor keys and the other with the keys 'A','W','D','X'. I see, that nowadays both players have a computer of their own and play over the network. So, that every one has his keyboard and cursor keys available. However, it's an example only.

Regards
Manfred
