

---

Subject: Re: capture mouse and limit all events to window

Posted by [slashupp](#) on Fri, 20 Jan 2017 07:14:31 GMT

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

---

I managed to achieve exclusive grab of mouse by modifying Ctrl:

In CtrlCore.h

Added attribute:

[1049] bool bcaptureLock;

Added parameters:

[1050] bool SetCapture(bool bLock=false);

[1051] bool ReleaseCapture(bool bUnlock=false);

In Ctrl.cpp (ctor)

[596] bcaptureLock=false;

In CtrlMouse.cpp (changed lines marked with: //mais )

[301]

Image Ctrl::MEvent0(int e, Point p, int zd)

```
{  
    GuiLock __;  
    LLOG("MEvent0 " << Name() << " event: " << FormatIntHex(e, 0) << " point:" << p);  
    Ptr<Ctrl> _this = this;  
    mousepos = p;  
    dword mm = 0;  
    if((e & ACTION) == DOUBLE)  
        mm |= K_MOUSEDOUBLE;  
    if((e & ACTION) == TRIPLE)  
        mm |= K_MOUSETRIPLE;  
  
    if (bcaptureLock) return MouseEventH(e, p, zd, GetMouseFlags() | mm); //mais...
```

Rect view = GetView();

if(mouseCtrl != this) {

...

[587]

Image Ctrl::DispatchMouseEvent(int e, Point p, int zd) {

GuiLock \_\_;

#if defined(flagWINGL) || defined(flagLINUXGL)

if(!IsEnabled() && this != (Ctrl\*) &infoPanel)

return Image::Arrow();

#else

if(!IsEnabled())

return Image::Arrow();

#endif

```
if(captureCtrl && captureCtrl->bcaptureLock) ///mais...
{
    if (captureCtrl!=this) return captureCtrl->DispatchMouseEvent(e, p, zd);
    return captureCtrl->MEvent0(e, p, zd);
}
```

```
if(captureCtrl && captureCtrl != this && captureCtrl->IsMouseActive())
...
```

[618]

```
bool Ctrl::SetCapture(bool bLock) { //mais...
    GuiLock __;
    ReleaseCtrlCapture();
    if(!GetTopCtrl()->SetWndCapture()) return false;
    captureCtrl = mouseCtrl = this;
    bcaptureLock=bLock; //mais...
    return true;
}
```

[628]

```
bool Ctrl::ReleaseCapture(bool bUnlock) { //mais...
    GuiLock __;
    if (bcaptureLock) bcaptureLock=!bUnlock; //mais...
    return this == captureCtrl && ReleaseCtrlCapture();
}
```

[634]

```
bool Ctrl::ReleaseCtrlCapture() {
    GuiLock __;
    if (captureCtrl && captureCtrl->bcaptureLock) return false; //mais...
    if(captureCtrl) {
        captureCtrl->CancelMode();
        Ctrl *w = captureCtrl->GetTopCtrl();
        captureCtrl = NULL;
        if(w->HasWndCapture()) {
            w->ReleaseWndCapture();
            return true;
        }
    }
    captureCtrl = NULL;
    return false;
}
```

All mouse events now goes to my control exclusively (for testing I use K\_ESCAPE hotkey to release)

These changes 'should' not affect any other uses of the functions. - mirek, what say you?

I'm looking at 'XWarpPointer()' to force pointer to remain physically within my control's view-area (don't know what MSWindows would require)

Adding test-code:

---

#### File Attachments

---

- 1) [main.cpp](#), downloaded 287 times
-