
Subject: Linux dual monitor: Not showing a TopWindow on the task bar

Posted by [Alboni](#) on Mon, 24 Apr 2017 15:39:09 GMT

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

My point of sale software Kassa uses a second monitor to display a client receipt screen.

I am using two main screens for that. The client screen is full screen.

(linux Mint 17.3 64 bit KDE flavour)

```
void DoMain()
{
    KassaAppWindow app(&session);
    One<KassaClientScreen> clientscreen;
#ifdef PLATFORM_LINUX
    app.OpenMain();
#endif
    if (KassaSettings::UseClientScreen())
    {
        clientscreen.Create();
#ifdef PLATFORM_LINUX
        clientscreen->OpenMain();
#endif
    }
    Ctrl::EventLoop();
}
```

The above works both on Linux and Windows, however on Linux the client screen also shows on the task bar. That creates confusion and not useful for a full screen display. Is there a way to not show it on the task bar? Note that on Linux I need to Call OpenMain() on the client screen or it won't show at all. On Windows calling OpenMain will freeze the program.

Subject: Re: Linux dual monitor: Not showing a TopWindow on the task bar

Posted by [mirek](#) on Wed, 31 Oct 2018 13:53:10 GMT

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

Alboni wrote on Mon, 24 April 2017 17:39My point of sale software Kassa uses a second monitor to display a client receipt screen.

I am using two main screens for that. The client screen is full screen.

(linux Mint 17.3 64 bit KDE flavour)

```
void DoMain()
{
    KassaAppWindow app(&session);
    One<KassaClientScreen> clientscreen;
#ifdef PLATFORM_LINUX
    app.OpenMain();
#endif
```

```

if (KassaSettings::UseClientScreen())
{
    clientscreen.Create();
#ifdef PLATFORM_LINUX
    clientscreen->OpenMain();
#endif
}
Ctrl::EventLoop();
}

```

The above works both on Linux and Windows, however on Linux the client screen also shows on the task bar. That creates confusion and not useful for a full screen display. Is there a way to not show it on the task bar? Note that on Linux I need to Call OpenMain() on the client screen or it won't show at all. On Windows calling OpenMain will freeze the program.

Is this related to the other taskbar issue?

Subject: Re: Linux dual monitor: Not showing a TopWindow on the task bar
 Posted by [Alboni](#) on Fri, 02 Nov 2018 14:14:06 GMT

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

Yes, same project, but not exactly the same. This is about an application with 2 main windows of which I don't want one of them on the task bar. The other is about popups that make dialog windows appear on the task bar.

Alternatively I've been thinking of making the second monitor window a separate program, but that one shouldn't appear on the taskbar at all. Maybe only as tray icon with a fullscreen display on the second monitor.

Subject: Re: Linux dual monitor: Not showing a TopWindow on the task bar
 Posted by [mirek](#) on Sat, 03 Nov 2018 16:17:17 GMT

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

Alboni wrote on Mon, 24 April 2017 17:39My point of sale software Kassa uses a second monitor to display a client receipt screen.

I am using two main screens for that. The client screen is full screen.

(linux Mint 17.3 64 bit KDE flavour)

```

void DoMain()
{
    KassaAppWindow app(&session);
    One<KassaClientScreen> clientscreen;
#ifdef PLATFORM_LINUX
    app.OpenMain();
#endif
}

```

```
if (KassaSettings::UseClientScreen())
{
    clientscreen.Create();
#ifdef PLATFORM_LINUX
    clientscreen->OpenMain();
#endif
}
Ctrl::EventLoop();
}
```

The above works both on Linux and Windows, however on Linux the client screen also shows on the task bar. That creates confusion and not useful for a full screen display. Is there a way to not show it on the task bar? Note that on Linux I need to Call OpenMain() on the client screen or it won't show at all. On Windows calling OpenMain will freeze the program.

What about clientscreen->Open(&app) ?
