

Games & GNU/Linux



Michal Turek

<http://woq.nipax.cz/>

(sorry my English, please...)

Games & GNU/Linux



Content

- Importance of Linux games
- Commercial and free games for Linux
- Game development
- Libraries, IDEs, manuals
- Demonstration of some Linux games

Games & GNU/Linux



Importance of Games

- Who doesn't play games in these days?
- Base of users

games + many children = A LOT OF USERS in future



Games & GNU/Linux



Commercial Games for Linux

- Much less than for MS Windows® :-)
- Unreal Tournament, Quake, Doom
- Ports - Loki, Hyperion, LinuxGamePublishing, EON Games
- Emulators (Wine, ...)



Games & GNU/Linux



Free Games for Linux

- Mostly they are not as high-quality as commercial
- Tux Racer, Frozen Buble, Pingus, Armagetron ...

- <http://happypenguin.org/>
- <http://games.linux.sk/>
- <http://www.libsdl.org/games.php>
- <http://www.root.cz/r/hry/>
- <http://hippo.nipax.cz/>
- <http://zimtech.ceskehry.cz/>



Games & GNU/Linux



Graphics Cards

- ATI
 - Drivers of some old cards is in XFree/X.Org
 - Linux was unsupported for a long time but new cards has ATI's Linux drivers now
 - Quality of this drivers?
- nVIDIA
 - Linux si supported for a long time
 - Installation in text mode only



Games & GNU/Linux



What is a game?

- For example this is a game

```
// Copyright © 2005 by Bzenda programmer  
// GNU GPL licensed
```

```
int main(int argc, char *argv[])  
{  
    Init();  
    while(done)  
    {  
        ProcessEvent();  
        Update();  
        Draw();  
    }  
    Destroy();  
    return 0;  
}
```

Games & GNU/Linux



Game Development

- Game development != MS DirectX®
- Is programming for ONLY one operating system enough?



Games & GNU/Linux



Libraries

- ~~DirectX®~~ (does not support GNU/Linux :-)
- SDL, Allegro, ClanLib, GLUT, ALUT
- OpenGL
- OpenAL, FMOD, SDL_mixer
- + gettext, doxygen...



Games & GNU/Linux



IDEs

- Anjuta, KDevelop, Eclipse...
- Borland C++ Builder, Borland Kylix
- Text editor + Makefile + gdb
- Auto* tools

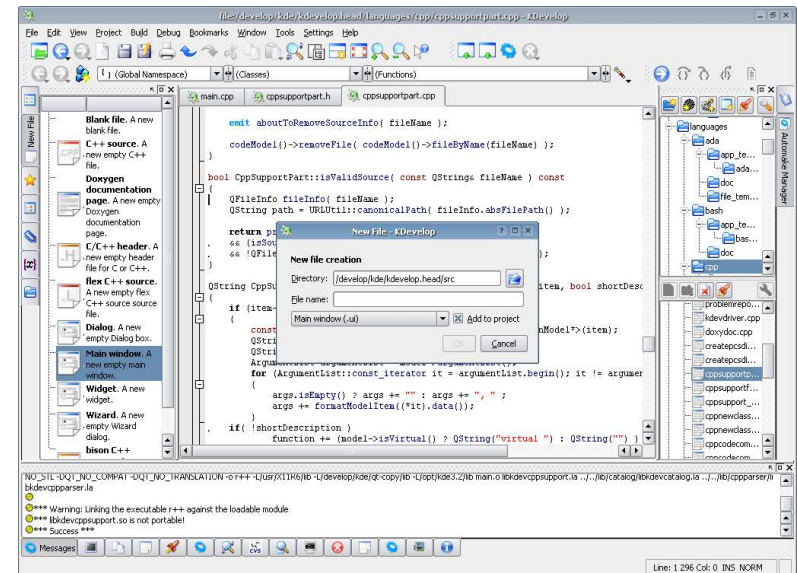
```
qfirstapp.cpp (/home/wog/libquark) - VIM
for(i = 0; i < NUM_OBJ_TEXS; i++)
{
    stringstream << "/data/objects/player" << i << ".png";
    img.Init(ssstream.str());
    ssstream.str(""); // Delete content

    m_textures[i] = img.CreateGLTexture();
}

QVector<Float> vector;

for(int i = 0; i < NUM_STATIC_OBJECTS; i++)
{
    vector.Set(
        (Float)((rand() % (m_hmap.GetWidth()-2))
        - ((m_hmap.GetWidth()-1) / 2)),
        0.0f,
        (Float)((rand() % (m_hmap.GetDepth()-2))
        - ((m_hmap.GetDepth()-1) / 2))
    );

    vector.SetY(m_hmap.GetHeightGL((int)vector.GetX(),
    (int)vector.GetZ()));
}
```





```
QStaticObject::Begin();
for(int i = 0; i < NUM_STAT
    m_obj[i].Draw();
QStaticObject::End();
```

```
if(m_which_menu != MENU_NO)
{
    m_menu.Draw(350, 350);
}
```

```
glColor3ub(255, 255, 255);
GetFont().Begin();
    GetFont().Draw(710,
    if(m_which_menu ==
        GetFont().D
GetFont().End();
```

```
QApplicationEx::Draw();
```

```
void QFirstApp::Update()
```

```
{
    if(m_which_menu != MENU_NO)
        return;
```

```
    SDL_PumpEvents();
```

```
    Uint8* keys;
```

```
    keys = SDL_GetKeyState(NULL);
```

```
    if(keys[SDLK_UP] == SDL_PRESSED || keys[SDLK_w] == SDL_PRESSED)
```

```
    {
        m_cam.GoFront(GetFPS());
```

```
    if(keys[SDLK_DOWN] == SDL_PRESSED || keys[SDLK_s] == SDL_PRESSED)
```

```
    {
        m_cam.GoBack(GetFPS());
```

```
    if(keys[SDLK_LEFT] == SDL_PRESSED || keys[SDLK_a] == SDL_PRESSED)
```

```
    {
        m_cam.GoLeft(GetFPS());
```

```
    if(keys[SDLK_RIGHT] == SDL_PRESSED || keys[SDLK_d] == SDL_PRESSED)
```

```
    {
        m_cam.GoRight(GetFPS());
```

```
    m_cam.ChangePosToQuadArea((m_hmap.GetWidth() >> 1)-1,
```

```
    (m_hmap.GetDepth() >> 1)-1);
```

```
    m_cam.SetYPos(m_hmap.GetHeightGL((int)m_cam.GetXPos(),
```

```
    (int)m_cam.GetZPos()));
```



```
g++ -c qvector.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qcheat.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qsceneobject.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qstaticobject.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qdynamicobject.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qplayer.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qteam.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qmenu.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -c qsimplemenu.cpp -Wall -I/usr/include/SDL -D_REENTRANT
g++ -o libquark main.o qapplication.o qfirstapp.o qgrid.o qheightmap.o qini.o qa
ppllicationex.o qfont.o qcamera.o qimage.o qvector.o qcheat.o qsceneobject.o qsta
ticobject.o qdynamicobject.o qplayer.o qteam.o qmenu.o qsimplemenu.o -L/usr/X11R
6/lib -L/usr/lib -lSDL -lpthread -lSDL_image -lGL -lGLU -lm
woq@woq:~/DATA/libquark$ ./libquark
woq@woq:~/DATA/libquark$ ./libquark
woq@woq:~/DATA/libquark$ ./libquark
woq@woq:~/DATA/libquark$ ./libquark
```

woq

So 14. kvě
18:24 19
100%

Procesor
91 proc
7 users

Procesy

Disk
1,0M

RAM

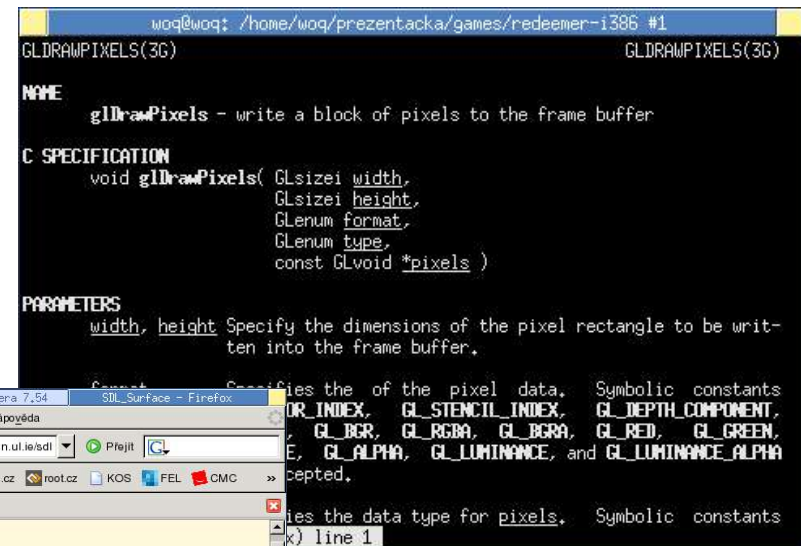
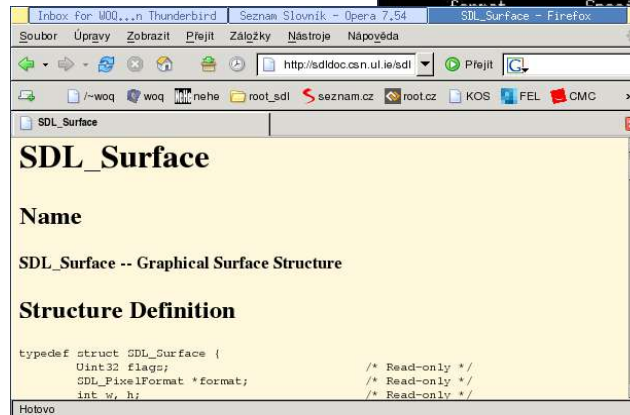
Lock Shoot
Od 3:47

Games & GNU/Linux



Manuals

- Man & info pages
- HTML documentation
- No *.CHM style documentation like MSDN



Games & GNU/Linux



Questions?



Games & GNU/Linux



Note...

- This presentation is GNU FDL licensed and can be downloaded from

http://woq.nipax.cz/o_prednasky.php

