

3D grafika Linuxon

A játékfejlesztők legújabb fegyvertársai már Linux alatt is elérhetők.

Legszívesebben megváltozott a játékkörültek Linuxról alkotott véleménye az elmúlt évben nagymértékben megváltozott. Mára a Linux-alapú gépekre nem csak a hálózati játékok kiszolgálójaként tekintenek: egyre többen használják játékaikra is kedvenc operációs rendszerüket. A Linux beállítása már önmagában is nehéz feladat, a játékosok dolgát pedig a rengeteg rövidítés nehezíti meg. OpenGL, DRI, DGA: tudja valaki, mit jelentenek ezek? Írásunkban megkísérelünk egy kis fejtágítást tartani e témakörben.

Mi is az OpenGL?

A körképet kezdjük az OpenGL-lel, ezzel biztosan találkozott már mindenki a játékokban vagy a csúcsmínőségű grafikai alkalmazásokban.

A Silicon Graphics (SGI) által fejlesztett, eredetileg IRIS GL néven futó OpenGL (Open Graphics Library) egy programkönyvtár, mely megkönnyíti a síkbeli és térbeli grafikai programozást. A csomag hamar a grafikai alkalmazás programozási felület (API-k) szabványává nőtte ki magát. A Linux Windows és Macintosh rendszereken egyaránt elérhető. Míg a programozók számára az OpenGL fontos munkaeszköz, addig a megrögzött játékosok leginkább az id Software Quake sorozatából ismerhetik. Az OpenGL az első rész óta a Quake leképezőmotorjának része. Itt fényesen bizonyította, hogy elképesztő játékokat is írhatunk segítségével. A szabvány egyik fő elve a nyitottság: az OpenGL minden operációs rendszertől és grafikus felületről független, innen ered a neve is (nyílt grafikus könyvtár).

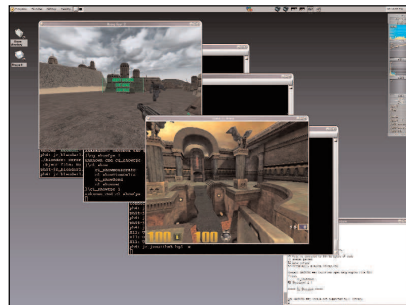
Ha az OpenGL-t bármilyen operációs rendszerben használni kívánjuk, valakinek meg kell írnia a szükséges eljárásokat, melyeket aztán az OpenGL program meghívhat. Ha azonban az OpenGL nevet szeretnénk ráakasztani a programunkra, a név használatának jogát is meg kell vásárolnunk az SGI-től vagy a Microsofttól. A megvásárolható csomaggal egy SI (Sample Implementation) nevű mintaalkalmazást kapunk, C nyelven írt forráskódjával együtt. Ennek alapján egy felkészült programozó elkészítheti saját felületén futó OpenGL könyvtárát. Mivel az OpenGL nyitott szabvány, az SI beszerzésére nincs is szükség. Hasonlóan gondolkodtak a Mesa fejlesztői is, ugyanis terméküket

teljesen az alapokról indulva, az OpenGL szolgáltatásait lemásolva készítették: megírásához egyetlen SGI-től kapott programsort sem használtak fel. Az OpenGL védjegy használatához természetesen ekkor is szükség van az SGI-vel kötött szerződésre, s az új változatot az OpenGL szabványra felügyelő szakmai testületnek, az ARB-nek is be kell mutatni. Az ARB (Architecture Research Board) eredetileg számítógépeket gyártó cégekből és az SGI-ből állt, mára azonban néhány olyan cég is tagja, amelyek játékokhoz szükséges részcégeket gyártanak. Érdekes megjegyeznünk, hogy az OpenGL-t az SGI fejlesztette ugyan ki, de a felügyelő bizottságban már csupán egy tag a többi között.

Az SGI nemrégiben adta ki az SI-t, ennek segítségével bármilyen számítógépre elkészíthető a grafikus könyvtár. E változat szintje teljesen megegyezik a grafikus kártyákat fejlesztő cégek számára átadott forráskóddal. Bár az SI állítólag nyílt forráskódú, ez nem teljesen igaz. Cikkünk megírásakor az SI szerződése még jó néhány olyan elemet tartalmazott, amelyek megakadályozták, hogy a csomag teljesen nyíltá váljon. Sokak számára az OpenGL egyet jelent a grafikai gyorsítással. Bár tényleg ragyogóan használja a komolyabb kártyákat, az OpenGL-re írt programok futtatásához nincsen szükség gyorsító kártyára. Ugyanis ezek a programok a kártyák nélkül is ugyanazt a képet rajzolják ki, természetesen jóval lassabban. Ha az adott gépen elérhető a grafikai gyorsítás (tehát egy 3D kártya és a hozzá tartozó meghajtó), akkor az OpenGL alkalmazások gyorsan és simán futnak, hiszen a szükséges számításokat ekkor a kártya végzi el. Az OpenGL szolgáltatásait nem minden grafikus kártya támogatja százszázalékosan. Ha a grafikus könyvtár egy, a kártya által nem támogatott művelet észlel, akkor azt programból, a központi processzort terhelve végzi el. Például a Transform & Lighting (T&L) nevű szolgáltatást a legtöbb közismert gyorsító kártya sem ismeri, ilyenkor az OpenGL háromszögeléssel és különböző eljárásokkal oldja meg a helyzetet. Születése óta az OpenGL-en számos bővítést hajtottak végre, jelenleg az 1.2 változatnál tart. A Linux alatt a leggyakrabban használt OpenGL-változat a *Brian Paul* vezetésével kifejlesztett Mesa.

Mesa: a nyílt forráskódú OpenGL

A Mesa az OpenGL szerződés nélkül készített változata, ezt Linux, Windows és Macintosh operációs rendszerben egyaránt használhatjuk. 1995-ös indulása óta egy kutatócsoport fejlesztí, melyet Brian Paul vezet. A Mesa ingyenes és nyílt forráskódú, azaz bárki, akinek kedve van hozzá, átültetheti egy másik operációs rendszerre. A legfontosabb, hogy a Mesa használatával



A Heavy Gear II és a Quake III egyszerre fut

bármilyen OpenGL alkalmazást futtathatunk. A Mesával az eszközökben rejlő gyorsítás is kihasználható, például egy 3dfx kártya tulajdonosa letöltheti és telepítheti a Glide SDK-t a 3dfx honlapjáról. A Mesa újrafordításakor érzékeli a telepített Glide fejlécállományokat és könyvtárakat, a 3dfx kártya ezután használhatja az OpenGL gyorsítási lehetőségeit (a Glide 2.x segítségével). A Mesát is számtalanszor bővítették már, akárcsak az OpenGL-t, így a Mesa 2.x az OpenGL 1.1 szabványt, a Mesa 3.x pedig az OpenGL 1.2-t, ezenkívül a GLUT és a GLU szabványokat is támogatja. Van tehát az OpenGL, melynek egy nyílt változata a Mesa. Most nézzük meg, hogy mi kapcsolja össze az OpenGL-t az X Windows rendszerrel.

GLX: az OpenGL és az X Windows

Az OpenGL géptől, operációs rendszertől és grafikus felületről is független. Használatához tehát valamilyen módon egy grafikus felülethez kell kapcsolnunk. Ez a kapcsolat lehetővé teszi az olyan feladatok elvégzését, mint például egy ablak helyének megtalálása vagy a bemenő adatok feldolgozása. A Linux- és Unix-rendszerek esetében ezt a GLX nevű könyvtár teszi lehetővé (Windows alatt a WGL használatos).

Vagy inkább úgy fogalmaznánk, a GLX segítségével megoldható, hogy az OpenGL egy tetszőleges X képernyőt használjon. A Linuxhoz használatos GLX az SGI által 1999. februárjában kiadott forráskódon alapul. Néhányan talán már a Utah-GLX kifejezésbe is belebotlottak, így most essék szó erről kicsit bővebben.

Utah-GLX

A Utah-GLX célja, hogy az XFree86 3.3.x-et, az OpenGL-t és az újabb grafikus kártyákat (Matrox G400/G200, ATI Rage Pro/Rage 128) együtt használhassuk. A Utah-GLX meghajtója a gyorsítást legtöbbször közvetett leképezéssel éri el, azonban ez a módszer – ahogy a későbbiekben látni fogjuk – bizonyos esetekben csökkentheti a teljesítményt. A Utah-GLX-nek köszönhetően valósították meg a grafikus gyorsítást a PPC processzoros Linux gépeken, valamint a laptopokon egyaránt. Mindkét esetben az ATI Rage Pro grafikus kártyát használták.

Az XFree86 4.0 megjelenésével remélhetjük, hogy a Utah-GLX-et hamarosan közvetlen leképezéshez (DRI) is felhasználhatjuk. A közvetlen és közvetett leképezés közötti különbségek tisztázására most ismét következzenek egy kis magyarázat.

Közvetett és közvetlen leképezés

A két módszer között az adatfeldolgozás menetében van különbség. Minél kevesebbszer kell a grafikus adatokhoz nyúlni, annál gyorsabban jelenhet meg a kép.

A közvetett leképezésnél az adatok először az alkalmazásból az X kiszolgálóhoz kerülnek, mely ezután a kártyához továbbítja azokat. Ez természetesen idővesztéssel jár, hiszen az X kiszolgálóra küldés előtt az adatokat a megfelelő formátumra kell alakítani, majd az X kiszolgálónak is átalakítást kell végeznie, mielőtt a kép a monitorra kerülne. Síkbeli grafikák esetében ez a sebességsökkenés elhanyagolható, de a manapság gya-

kori 3D-s alkalmazások jelentősen lelassulnak a közvetett leképezés következtében. A közvetlen leképezés az alkalmazás számára lehetővé teszi a grafikus kártya közvetlen elérését, ennek köszönhetően az X kiszolgálónak ilyenkor sokkal kevesebb munkát kell végeznie. E módszert – mely a DRI (Direct Rendering Infrastructure) néven ismeretes – a Precision Insight fejlesztette ki, és az XFree86-ban is helyet kapott.

Most végre rátérhetünk a titokzatos DRI tárgyalására: megvizsgáljuk, milyen előnyökkel jár e módszer használata a Linux-felhasználók számára.

DRI: biztonság, megbízhatóság, gyorsaság

A DRI lehetővé teszi, hogy az alkalmazás biztonságosan és hatékonyan érje el a kártyát. Az Utah-GLX-szel ellentétben használatához nincs szükség magas szintű felhasználói jogokra: a munka nagy részét maga az alkalmazás, és nem az X kiszolgáló végzi.

A DRI három része: egy különleges X kiszolgáló egy, a közvetlen leképezést végrehajtó ügyfél és egy eszközmeghajtó, mely a rendszeremagba épül be. A leképezőmotor további elemekből épül fel: valamely OpenGL-változatból (általában Mesa), a GLX-ből és Linux-közeli, de gépfüggetlen meghajtókönyvtárakból. A rendszeremagba épülő meghajtó közvetlen memóriáhozáférést (DMA) enged a grafikus kártya számára, s úgy készítették el, hogy a továbbiakban a rendszeremag módosítására csak nagyon különleges esetben legyen szükség. Egyszóval: a DRI a grafikus gyorsítás első működőképes megvalósítása. Használatához nincs szükség különleges felhasználói jogokra, biztonsága pedig jóval meghaladja az eddigi megoldásoknál tapasztalt szintet. Mindemellett a DRI az XFree86 része, így számos próbasorozatban ellenőrizték működését. Ezenkívül magában hordozza az XFree86-nál megszokott üzembiztonságot is. A DRI lehetővé teszi, hogy nagy erőforrás-igényű alkalmazásokat (magas szintű grafikai tervezőrendszerek, 3D-s játékok) futtassunk Linux alatt.

Egyéb finomságok

A DRI, az OpenGL, valamint a Mesa kifejezéseken kívül olvasóink bizonyára más titokzatos rövidítésekkel is találkoztak eddigi linuxos pályafutásuk során. Ne hagyjuk ki ezeket sem!

- *GLUT* – A GLUT alapját a *Mark Kilgard* által kifejlesztett aux könyvtár képezi, mely eredetileg az általa írt *OpenGL Programming for the X Window System* című könyv példáinak futtatásához készült.
- *DGA* – A DGA (Direct Graphics Architecture) nem gyorsítás, csupán az X kiszolgáló egyszerű bővítése, mely lehe-

tővé teszi a képernyőmemória közvetlen elérését az alkalmazások számára. Használatához magas szintű jogosultságokra van szükség. Az XFree86 4.0 a DGA 2.0-s változatát tartalmazza, ez már elérhetővé tesz némi gépszintű gyorsítást, de csak a 2D-s grafikai alkalmazások esetében. A DGA ezenkívül az egér és a billentyűzet jeleinek közvetlen olvasásában is az alkalmazás segítségével lehet. Alapesetben ugyanis ezek a jelek először az X kiszolgálóhoz érkezik, s csak innen kerülnek továbbításra a program felé.

- *VidMode* – A VidMode bővítés segítségével az alkalmazás bármikor lekérdezheti az X kiszolgáló által támogatott képernyőmódokat (frekvencia, felbontás, színmélység stb.), és ezek között váltani is tud. Általában olyan játékok használják, amelyek futásához egy adott képernyőmódra van szükség, az X kiszolgáló éppen használt módjától függetlenül.
- *Glade* – A Glade egy, a 3dfx által, saját Voodoo sorozatú kártyáihoz kifejlesztett, teljes képernyős, gépszintű gyorsítást nyújtó API. A régebbi, gyorsítást használó játékok alkalmazták, ilyen például a Tomb Raider is. Az OpenGL általános sikerével a Glade jelentősége egyre csökken, ennek hatására a 3dfx nemrégiben nyíltta tette a forráskódot.

Akkor foglalkozzunk össze...

Az OpenGL tehát egy grafikus könyvtár, olyan eljárásgyűjtemény, mely a térbeli grafikák megjelenítését segíti elő. Linux alatt a Mesa teszi lehetővé az OpenGL alkalmazások futtatását, ezeket pedig a GLX hangolja össze az X kiszolgálóval. A még mindig XFree86 3.3.x-szel dolgozó felhasználók a Utah-GLX használatával helyettesíthetik az XFree86 4.0-t és a DRI-t. Az XFree86 4.0 azonban jóval nagyobb teljesítményt és megbízhatóságot jelent.

Reméljük, ezen összefoglaló cikk segített eloszlani a Linux 3D gyorsítással kapcsolatos sötét foltokat. Jó játékos és eredményes munkát mindenkinek!

Daryl Strauss 15 éve dolgozik Unix-rendszerekkel, 1995 óta pedig a Linuxszal. Ezen idő alatt különleges filmes hatások készítésével és nyílt forráskódú 3D-alkalmazásokkal foglalkozott. Jelenlegi munkaadója a VA Linux Systems.

Matt Matthews az NCSU PhD ösztöndíjas hallgatója. 1999 nyara óta tartozik a Linux-felhasználók népes táborába. Szívesen próbál ki új játékokat és grafikus kártyákat; az eredmények a <http://www.linuxgames.com> honlapon jelennek meg.

Kapcsolódó címek

XFree86:

➔ <http://www.xfree86.org/>

Mesa:

➔ <http://mesa3d.sourceforge.net/>

Precision Insight:

➔ <http://www.precisioninsight.com/>

DRI Development:

➔ <http://dri.sourceforge.net/>

Utah-GLX:

➔ <http://utah-glx.sourceforge.net/>

SGI Open Source:

➔ <http://oss.sgi.com/>