

Nyílt forrású 3D grafikus motor

Howard bemutatja a kereskedelmi 3D grafikus motorok nyílt forrású választási lehetőségét, a Crystal Space-t.

Szeretnél 3D-s játékot vagy alkalmazást készíteni? Legelőször 3D grafikai motorra lesz szükséged, amelylyel mindezt felépítheted. Hagyományosan két választásod van: vagy kódolsz magadnak egy teljesen új motort, vagy magas használati díj fejében valamelyik ismert cég motorját használod. Ez utóbbi megoldás behatárolja végleges termék kereskedelmi lehetőségeit.

Létezik azonban egy harmadik lehetőség is: a Crystal Space, egy 3D grafikus motor, amelyet *Jorrit Tyberghein*, 31 éves belga programozó írt.

A Crystal Space-t Tyberghein 1997-ben készítette el, miután megismerte az olyan játékokat, mint a Doom és a Quake, és kíváncsi volt rá, hogyan is állítják elő őket. Nem lévén tapasztalata a grafikus motorok kódolásában, az Interneten keresett adatokat a 3D-s programozással kapcsolatban, majd mindössze két hónap alatt el is készítette a Crystal Space első változatát. Tyberghein a kódot nyílt forrásban elérhetővé tette elérhetővé, így kisvártatva megszületett a Crystal Space fejlesztői közösség. A grafikus motort az eredeti Linux-rendszeről azóta már Unix, 32-bites Windows, Windows NT és egyéb operációs rendszerek alá is átültették.

Jelenleg a Crystal Space-szel írt alkalmazások többsége szerepjáték. Mivel a motor egy vagy több kameraállás megvalósítását teszi lehetővé, valamint beépített hálózati támogatást tartalmaz, és nagymértékben megkönnyíti a többfelhasználós játékok – például a szerepjátékok – fejlesztését. Felhasználják még repülőszimulátorok, valós idejű stratégiai játékok és a Doomhoz vagy a Quake-hez hasonló saját nézetű lövöldözős játékok fejlesztése során is.

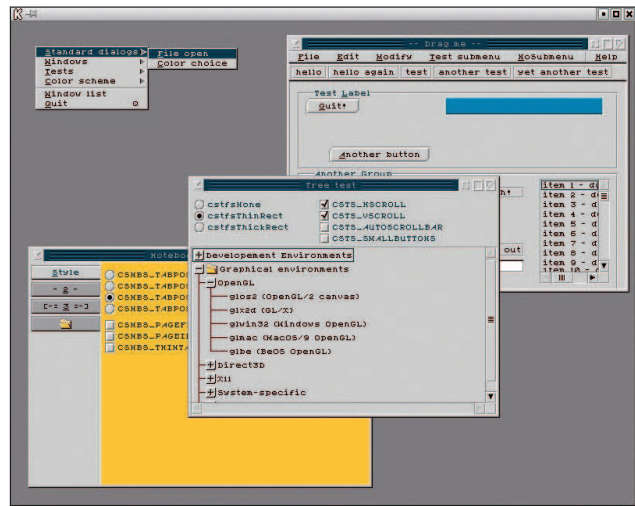
Ha azt vizsgáljuk, hogyan állja meg a Crystal Space a helyét a kereskedelmi 3D grafikus motorokkal szemben, főként a Quake III- és az Unreal Tournament-motorjával szemben, azt mondhatjuk, hogy bizonyos grafikai jellemzők terén összemérhető velük, de a kód kiforrottsága és sebessége tekintetében alattuk marad. Általánosságban a Crystal Space sokoldalúbb, mivel nem csupán játékok készítésére alkalmas. Gyakorlatilag egy olyan általános célú grafikus API, amit egy multimédiás böngésző, egy hangszerkesztő és egy képnézegető program készítéséhez is felhasználtak.

„A motornak számos olyan jellemzője van, amire a Quake-nek vagy az Unrealnek sosem lesz szüksége, mivel az ilyen típusú játékok a grafikai szolgáltatásoknak csupán szűk körét használják” – állítja *Andrew Zabolotny*, 28 éves szentpétervári programozó, aki a Crystal Space számára készít magas szintű kódokat.

A Crystal Space jellemzői: a Jó és a Rossz

A Crystal Space fejlesztői a motor megbízhatóságát (szinte sosem omlik össze) és a 3D-s leképezés (rendering) teljes megvalósítását emlegetik a fejlesztésben való részvétel és a használat melletti okokért.

Mennyire van létjogosultsága jelenleg egy 3D grafikus



1. kép Egy próbaalkalmazás, amely a Crystal Space ablakozórendszer számos jellemzőjét mutatja

motoroknak, amikor a fejlett és erőteljes 3D grafikus alkatrészek már szinte alapvető tartozékai a legtöbb PC-nek? Az egyszerű válasz az, hogy még a legjobb 3D-s technológia mellett is szükségünk van egy programból megvalósított grafikus motorra, amely bizonyos mértékben megszabja az alkatrész működését.

„Mindig megvan a határa annak, hogy az alkatrész mire képes” – mondja Tyberghein, a Crystal Space atyja. „Még a legfejlettebb alkatrész is nehezen boldogulna, ha sokszögek millióit kellene leképeznie. A grafikus motor feladata az hatékonyabbá tétel, vagyis behatárolja, hogy a grafikus kártyához egyáltalán mi jut el.”

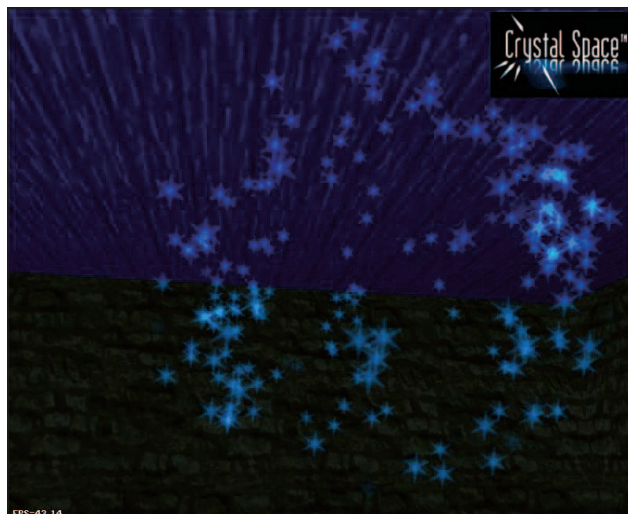
Például programból megvalósított grafikus motorra van szükség annak meghatározására, hogy a játék virtuális környezetnek mely részei láthatók egy adott pillanatban a felhasználó szemszögéből. Így a 3D-s alkatrész teljesítménye hatékonyabb lesz, hiszen nem kell erőforrásokat pazarolnia arra, hogy olyan objektumot képezzen le, amely ténylegesen nem is jelenik meg a képernyőn, még ha elméletileg létezik is (csak a játékos szemszögéből nem látható) a virtuális környezetben belül.

A Crystal Space az LGPL alatt ingyenesen terjeszthető. Ez a C++ nyelven írt 3D-s fejlesztőkörnyezet számos grafikus szolgáltatást és látványos jellemzőket támogat:

- tényleges hat szabadságfok;
- színes megvilágítás;
- mipmapping;
- kapuk;
- tükrök;
- alfaátlátszóság;



2. kép A Crystal Space által leképezett belső környezet



3. kép Csillagok, melyeket a Crystal Space részecskeanimáló rendszere képezett le

- tükrözőfelületek;
 - keret alapú és vázanimált 3D-s szerkezetek;
 - eljárás alapú textúrák;
 - radiosity;
 - részecske rendszerek;
 - volumetriás kód;
 - parancsállományok készítése Python vagy egyéb nyelveken;
 - 8-, 16- és 32-bites megjelenítés támogatása;
 - Direct3D,
 - OpenGL és programból megvalósított grafikaleképezés;
 - betűkészlet-támogatás;
 - hierarchikus transzformációk,
- hogyan csak néhányat emeljük ki.

Bár a teljesítmény terén nem veheti fel a versenyt a Quake III vagy az Unreal Tournament motorjával, a Crystal Space-nek mégis megvannak a maga előnyei: több rendszeren is használható, vagyis olyan kódot készíthetünk, amely ugyanolyan jól fut Linux, Unix, 32-bites Windows, Windows NT

és hét egyéb operációs rendszeren, amelyre a grafikus motort átültették.

A Crystal Space rugalmasan bővíthető, azaz egyetlen futtatható állomány is olyan különböző leképezőkkel képes működni, mint amilyen az OpenGL, Direct3D és a Glide. A grafikus motor első kiadása óta az OpenGL-leképezőt újraírták, így előző változatainál gyorsabban fut. Lényeges jellemző, hogy a Crystal Space sok 3D grafikus állomány-formátumot képes magától felismerni és olvasni. Lehetőség nyílik egyéb 3D-formátumok importálására is (például 3DS, OBJ, MDL, MD2, LWO és ASE). Megfordítva: a motor olyan



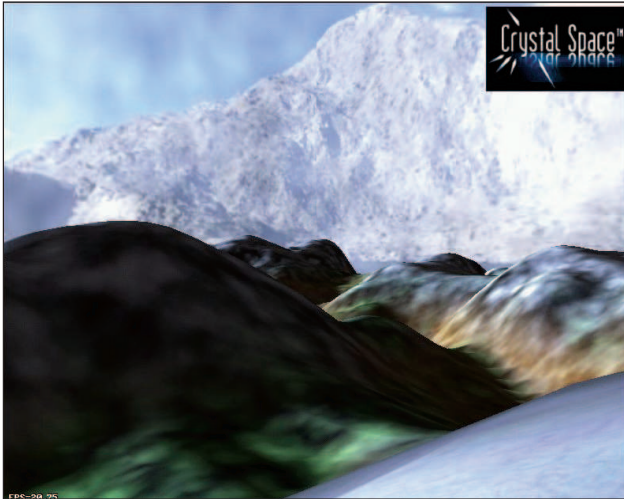
4. kép Egy Föld felé tartó űrhajó, amelyet a Crystal Space valós időben képezett le

Python-programokkal rendelkezik, amelyekkel a környezet és a modellek a Blender programból exportálhatók a Crystal Space-be. Bár a motor elsődleges célja 3D-s grafikák létrehozása, emellett még egy 2D-s API-val is rendelkezik. „Megírtam egy teljes körű ablakozó GUI-t, ami a Crystal Space alacsony szintű API-ján alapul” – mondta Zabolotny.

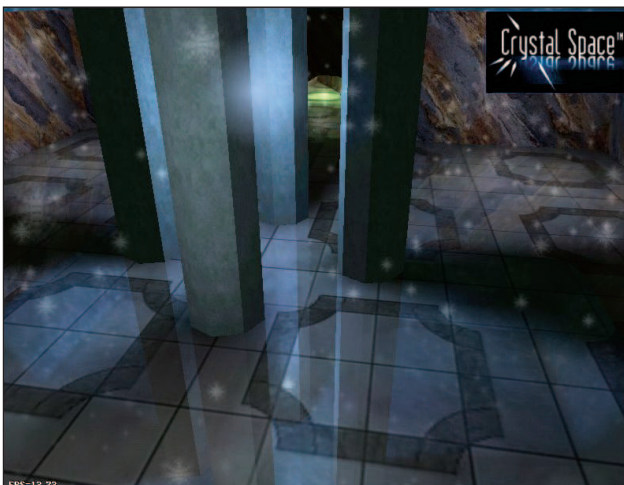
Különálló kódrészleteket használhatunk a Crystal Space-környezetben kívül is olyan munkafolyamatokban, amelyek nem játék- vagy multimédiafejlesztésekkel kapcsolatosak. Ezek között található egy csIniFile-osztály (.ini állományok kezeléséhez), egy SCF (osztott osztály lehetőség) alrendszer, egy csArchive-osztály (.zip állományok kezeléséhez) és egy VFS (virtuális állományrendszer) alrendszer.

A Crystal Space legnagyobb gyengesége, hogy hiányzik belőle az ütközések felismerését megvalósító programrész. **Thomas Hieber**, aki a Crystal Space segítségével egy Crystal Shooter nevű, saját szemszögű lövöldözős játékot fejleszt, ideje nagy részét azzal tölti, hogy a motor ezen képességét igyekszik fejleszteni. „Bizonyos mértékű támogatás ugyan létezik a motorban, de ez nem túl hasznos a játékok számára” – mondja a 30 éves német programfejlesztő. „Csak a statikus objektumok ütközésének ellenőrzése támogatott, ami arról nyújt információkat, hogy hol történt az ütközés – ha történt egyáltalán. De a gyorsan mozgó tárgyak esetén ez nem elégséges.”

Egy másik kényes terület a Crystal Space megvilágításkezelése. Bár lágy árnyékokkal támogatja a színes statikus és dinamikus megvilágítást, nagyon nehéz rávenni, hogy mindezeket kellő sebességgel valósítsa meg a legkülönfélébb játékhelyzetekben. Ezen a területen a grafikus motor még kihívásokkal küszködik.



5. kép Környezet, melyet a Crystal Space tájleképezője hozott létre



6. kép A fényes felületre hulló hó a Crystal Space visszatükörlő képességét szemlélteti

A Crystal Space jövője

A Crystal Space-közösség számára nyilvánvalóan elkél a programozók segítsége. Tyberghein olyanokat keres, akik képzettek a grafikus motorok magjának programozásában és járatosak az algoritmikus gondolkodásban – főleg olyanokat vár, akik segítenek a motormag teljesítményének finomhangolásában. „Sokan segítenek a Crystal Space egyéb területein (például: OpenGL- és Direct3D-programozás, windowsos és linuxos átültetések), de csak nagyon kevesen képesek magának a grafikus motornak a fejlesztésében közreműködni” – mondja. „Ha több jó programozónk lenne, akkor sokkal több lennénk képesek” – állítja Zabolotny. „Elsősorban olyanokra lenne szükségünk, akik járatosak a keresztrendszeres C/C++ programozásban.”

E cikk írásakor a Crystal Space-csapat elődleges célja az API kellő üzembiztonságának megvalósítása. „A fejlesztői példány már tőrhetően üzembiztos, ám akad még teendőnk bőven” – mondja Tyberghein. A Crystal Space jelenlegi 0.90-es kiadása a régóta várt 1.0-s kiadás elődje. A 0.90-es és az 1.0-s kiadás között az API számottevően nem fog változni, viszont a 0.90-es kiadás célkitűzése, hogy felgyorsítsa a hibakeresést és a leíráskészítést.

A 0.90-es kiadásban kipróbált egyik újdonság a megújult tájleképező motor, amelyet szorosabban és jobban bevontak a Crystal Space-kódjába, mint az előző kiadásokban. A grafikus motor számos új különleges hatást (effect) képes előállítani, mint például ködök, lencsék, továbbá részecskeleképező rendszerrel is felruházták. Technikai szempontból nézve a Crystal Space-segédesszközök modulárisabbak és egyszerűbben elérhetőek lettek. Több bővítmény és kódrészlet, amely a korábbi kiadásokban külön függvénykönyvtárakban foglalt helyet, most egyesítve lett.

A végső kérdés, hogy a Crystal Space eljuthat-e arra pontra, amikor már mindazon képességekkel rendelkezik, amelyek lehetővé teszik a kereskedelmi célú játékfejlesztést, valamint elterjedhet-e olyan mértékben, mint a neves 3D grafikus motorok? Maga Tyberghein is kétségekkel küszködik:



7. kép A Crystal Space honlapja

„Ha megfizeted a Quake III-motor használati díját, egy olyan minőségi termékhez jutsz, ami biztosan működik. Ha termék-támogatásra vágysz, ne használj szabad forrású motorokat. Ha azonban úgy érzed, hogy megvagy a terméktámogatás nélkül is, vagy ha drágállod a használati díjat, jól járhatasz egy nyílt forrású motorral.”

Hieber elismeri, hogy „a Crystal Space mérföldekre van a Quake III-tól”, de nem hiszi, hogy ez bárkit is eltántorítana attól, hogy nagyszerű játékokat készítsen motor segítségével, hiszen a Crystal Space jól átgondolt és tervezett motor, bár nem szükségképpen áll mögötte olyan erőteljes technológia, amely igazán jó minőségű megjelenítést tenne lehetővé. „Nézd meg a Tomb Raidert vagy a Half-Life-ot” – mondja. „Egyikük sem rendelkezik kiemelkedően jó 3D-s motorral, mégis mindkettő sikeres, egyszerűen azért, mert színvonalas játékok.”

A Crystal Space a <http://crystal.sourceforge.net> weboldalon kereshető fel.

Linux Journal május, 97. szám



Howard Wen

tíz éve foglalkozik a videojátékiparral. Írásai különböző kiadványokban és olyan webhelyeken jelennek meg, mint a Wired, a Salon.com, a Playboy.com, a GameSpot.com, O'Reilly Network és a Dallas Observer. Kapcsolatba léphetünk vele a honlapján: <http://www.howardwen.com>