

4.5.4.15. WRAM

Memoria WRAM (*Window RAM*) este o variantă modificată a memoriei VRAM, și asigură atât o creștere a performanțelor, cât și o reducere a costului. Memoria WRAM este de asemenea cu porturi duale, având o rată de transfer cu aproximativ 25% mai mare față de memoria VRAM, și include facilități suplimentare pentru a permite transferuri mai eficiente la operațiile grafice curente, ca umplerea unor blocuri sau desenarea unor texte. Un adaptor prevăzut cu o asemenea memorie și un circuit RAMDAC suficient de rapid asigură rezoluții ridicate, de 1600×1200, în modul “*true color*”.

Observație:

Memoria *Window RAM* nu are legătură cu *Microsoft Windows*.

4.5.5. Acceleratoare video 3D

4.5.5.1. Necesitatea acceleratoarelor 3D

Deoarece ecranul monitorului este bidimensional, imaginile afișate de calculator trebuie să fie de asemenea bidimensionale. Pentru a lucra cu obiecte 3D, este necesar ca ele să fie convertite în imagini 2D. Aceasta necesită prelucrări speciale și o putere de prelucrare care nu a fost disponibilă până acum la calculatoarele din categoria PC.

Motivul pentru care acceleratoarele specializate 3D devin răspândite este că programele încearcă să utilizeze din ce în ce mai multe reprezentări 3D. Pentru a obține mai mult realism, imagini grafice mai detaliate și viteze mai mari în programe ca jocuri de acțiune, simulatoare de zbor, programe grafice și aplicații CAD, trebuie efectuate mai multe prelucrări 3D într-o perioadă mai scurtă de timp.

La începutul anilor ‘90, când sistemele de operare grafice au devenit răspândite, cele mai multe adaptoare video nu aveau funcții de accelerare. Atunci când utilizatorii lansau sistemul *Windows*, unitatea centrală trebuia să efectueze toate operațiile pentru desenarea graficelor pe ecran, ceea ce reducea viteza calculatorului în mod semnificativ. Pentru a rezolva această problemă, au fost proiectate acceleratoare care efectuau multe din aceste operații cu circuite specializate, în loc ca acestea să fie efectuate de procesorul sistemului.

În mod similar, azi nu este necesar de a avea un adaptor grafic 3D pentru a crea grafice 3D, dar cantitatea mare de calcule necesare pentru translatarea imaginilor 3D în imagini 2D într-un mod realist înseamnă că fără circuite specializate care să efectueze aceste operații, ele trebuie efectuate de procesor.

Utilizarea unui accelerator 3D permite programelor afișarea imaginilor virtuale 3D cu un nivel de detaliere care nu este posibil cu un adaptor video standard 2D.

4.5.5.2. Imagini și operații 3D

Imaginile 3D sunt mult mai complexe decât imaginile 2D din cauza cantității mai mari de informații care trebuie utilizate pentru a crea imagini 3D realiste. În plus, trebuie utilizate mai multe operații matematice pentru a converti aceste imagini 3D în imagini care pot fi afișate pe ecran.

Imaginile 3D sunt gestionate de calculator utilizând modele abstracte. De obicei, fiecare obiect 3D este compus din sute sau chiar mii de mici triunghiuri (sau alte poligoane) care descriu structura sa. Grafica 3D animată necesită execuția unei serii continue de calcule geometrice care definesc poziția obiectelor în spațiul 3D. În mod obișnuit, calculele geometrice, care manipulează vârfurile triunghiurilor, sunt executate de UCP, deoarece operațiile în virgulă mobilă necesare pot fi executate în mod eficient de UCP. În același timp, controlerul grafic trebuie să convertească aceste triunghiuri în suprafețe solide, ceea ce implică calcule intense.

În lumea reală, obiectele interacționează unele cu altele. Ele se suprapun, creează umbre, reflectă lumina și apar mai întunecate atunci când sunt la distanță. Există ecuații matematice foarte complexe care se utilizează pentru a determina dacă un obiect este vizibil într-o scenă dintr-un unghi dat, care ar trebui să fie culoarea acestuia etc. Pentru un joc 3D, în scopul obținerii unei animații uniforme, aceste calcule trebuie refăcute de 20-30 ori pe secundă. Acceleratoarele 3D sunt adaptate pentru a executa aceste calcule care solicită în mod intens resursele.

De fiecare dată când imaginea de pe ecran este actualizată, trebuie să se recalculeze culoarea și intensitatea fiecărui pixel de pe ecranul 2D. Aceasta se realizează prin aplicarea diferitelor calcule 3D asupra scenei, într-un proces numit interpretare (*rendering*). Prin acest proces se completează toate punctele de pe suprafața obiectului care a fost memorat în prealabil doar ca un set de vârfuri. Astfel, pe ecranul monitorului se va desena un obiect solid cu efecte 3D. Prin eliberarea UCP de operația de interpretare, acceleratorul grafic permite UCP să dedice o putere de calcul mai mare altor funcții, rezultând o frecvență de cadre crescută și o calitate îmbunătățită a imaginii.

Există diferite tipuri de operații care sunt executate în cadrul prelucrărilor 3D. O parte din cele mai obișnuite operații 3D sunt descrise în continuare.

- *Umbrirea Gourad*

Acesta este un algoritm utilizat pentru a realiza o umbrire realistă a suprafețelor 3D. Efectul ajută ca forma obiectului să fie mai bine defini-

tă. Este o operație obișnuită care este utilizată în numeroase aplicații 3D.

- *Decuparea*

Această operație determină partea unui obiect care este vizibilă pe ecran și decupează părțile pe care utilizatorul nu le poate vedea. Aceasta economisește timp, deoarece părțile obiectelor care nu sunt afișate pe ecran sunt ignorate.

- *Illuminarea*

Obiectele din lumea reală sunt modelate de sursele de lumină din cadrul scenei. Efectele de lumină creează nuanțe ale culorilor, reflexii de lumină, umbre și alte efecte care trebuie adăugate obiectelor pe baza poziției acestora și a poziției surselor de lumină.

- *Transparența*

Anumite obiecte din lumea reală sunt transparente sau semi-transparente. Pot fi efectuate calcule speciale pentru a determina obiectele care sunt vizibile printr-un obiect transparent, de exemplu o ușă de sticlă.

- *Maparea texturilor*

În cazul graficii 3D, obiectele pot avea detalii ale suprafețelor, ca lemn granular sau cărămidă, și pot avea desene sau semne pe suprafața lor. Adăugarea acestui nivel de detalii se numește maparea texturilor. Obiectele sau scenele 3D arată mai realist prin aplicarea texturilor bidimensionale asupra poligoanelor 3D. Există diferite tipuri de mapare a texturilor care sunt utilizate de diferitele programe și circuite.

- *Dithering*

Acesta este un efect care este utilizat în diferite aplicații, inclusiv la grafica obișnuită 2D și de asemenea la imprimare. "Dithering" este procedeul de mixare a unui mic număr de culori în modele specifice pentru a crea iluzia existenței unui număr mai mare de culori. De exemplu, imprimantele color cu jet de cerneală utilizează acest procedeu pentru a crea un spectru larg de culori aparente, chiar dacă fiecare punct tipărit are numai una din cele trei (sau patru) culori reale diferite. La grafica 3D, acest procedeu se utilizează pe larg pentru a prezenta culori mai realiste fără a fi necesară creșterea numărului de culori a imaginii (ceea ce ar însemna un timp de calcul mai mare și o memorie mai mare pentru păstrarea graficelor).

- *Încețoșarea*

Este un efect utilizat la scenele de exterior. Acest procedeu are două scopuri, obținute prin estomparea obiectelor care sunt la distanță. În

primul rând, determină ca scena să apară mai realistă. La distanță, obiectele apar neclare datorită umezelii atmosferice. În al doilea rând, permite ca prelucrarea 3D să fie executată mai rapid, deoarece acele obiecte la distanță care sunt încețoșate pot fi prelucrate mai rapid, ele fiind prezentate cu mai puține detalii.

Aspectul cel mai critic al graficii 3D este maparea texturilor, care descrie în detaliu suprafețele obiectelor 3D. Maparea texturilor constă în încărcarea unuia sau a mai multor *texeli* (elemente de textură) dintr-o hartă de biți, combinarea acestora, și apoi scrierea pixelului rezultat în memoria video.

Figura 4.16 prezintă modul în care se efectuează maparea texturilor.

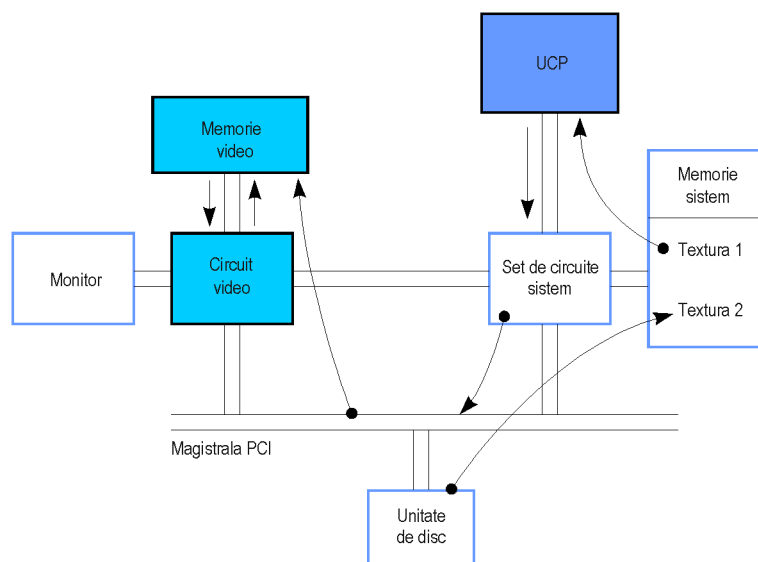


Figura 4.16. Operațiile executate pentru maparea texturilor.

Există cinci etape de bază implicate în prelucrarea texturilor.

1. Înainte de utilizarea lor, texturile sunt citite de pe unitatea de disc și sunt încărcate în memoria sistem.
2. Atunci când o scenă trebuie să utilizeze o textură, aceasta este citită din memoria sistem de către UCP.
3. Asupra texturii se aplică apoi transformări de iluminare și de perspectivă. Rezultatele acestor operații sunt ulterior scrise înapoi în memoria sistem.

4. Controlerul grafic citește apoi texturile transformate din memoria sistem și le scrie în memoria video. Aceste date trebuie să treacă prin magistrala PCI pentru a ajunge la controlerul grafic.
5. Controlerul grafic citește în continuare texturile și informațiile 2D de culoare din memoria sa video. Aceste date sunt utilizate pentru interpretarea unui cadru care se poate afișa pe ecranul monitorului. Rezultatul este scris înapoi în memoria video. Circuitul RAMDAC al adaptorului video citește cadrul și îl convertește în semnale analogice care controlează monitorul.

Există mai multe probleme legate de modul în care sunt gestionate texturile:

- În primul rând, texturile trebuie păstrate atât în memoria sistem, cât și în memoria video. Aceste copii redundante determină o utilizare ineficientă a resurselor de memorie.
- În al doilea rând, păstrarea, chiar temporară, a texturilor în memoria video limitează dimensiunea acestora. Există o cerere pentru texturi din ce în ce mai detaliate, ceea ce obligă producătorii să echipeze adaptoarele lor cu o memorie video de dimensiune mai mare. Însă, deoarece acest tip de memorie este destul de costisitoare, aceasta nu este o soluție optimă.
- În sfârșit, rata de transfer a magistralei PCI limitează rata cu care texturile pot fi transferate către subsistemul grafic. Mai mult, în sistemele tipice mai multe dispozitive de I/E conectate la magistrala PCI trebuie să partajeze rata de transfer disponibilă. Introducerea altor dispozitive de viteză ridicată, ca unitățile de discuri *Ultra DMA* și adaptoarele de rețea de 100 MB/s, creează o congestie pe magistrala PCI care poate limita performanțele graficii 3D.

Aplicațiile utilizează diferite strategii pentru a compensa limitările calculatoarelor actuale. Se utilizează algoritmi de *swapping* pentru a decide care texturi trebuie memorate în memoria video și care în memoria sistem. În mod tipic, aplicațiile dedică o parte a memoriei video ca spațiu de *swapping* al texturilor, în timp ce memoria video rămasă conține texturi utilizate în mod frecvent. Depinzând de numărul de texturi utilizate într-un cadru, algoritmul poate varia proporția memoriei alocate pentru schimbarea texturilor și pentru texturile fixe.

Dacă circuitele pot executa maparea texturilor numai din memoria video, algoritmul încearcă de obicei să încarce în prealabil în memoria video texturile necesare pentru fiecare cadru sau scenă. Fără această încărcare prealabilă, utilizatorii vor observa o pauză în timp ce se încarcă textura necesară din memoria sistem în memoria video, sau de pe disc în memoria sistem și în memoria video.

4.5.5.3. Factori care afectează performanțele 3D

Diferitele adaptoare 3D implementează diferite tipuri de facilități de accelerare 3D. Cu cât se adaugă mai multe efecte unei imagini, cu atât ea apare mai realistă. Totuși, mai multe efecte necesită mai multă memorie și timp de prelucrare. Atunci când timpul necesar interpretării unei imagini crește, frecvența cadrelor scade. *Frecvența cadrelor* este numărul de noi imagini 3D care pot fi calculate pe secundă. Aceasta nu este aceeași cu *frecvența de reîmprospătare*, care indică de câte ori pe secundă este transmisă imaginea curentă de la adaptorul video la monitor.

Dacă frecvența cadrelor devine prea mică, mișcarea într-un program 3D devine neuniformă, ceea ce anulează mare parte din realismul obținut prin efectele 3D. Din acest motiv, anumite adaptoare video nu implementează în mod intenționat anumite funcții 3D, pentru a putea calcula imaginea suficient de rapid în scopul furnizării unei frecvențe rezonabile a cadrelor.

Performanțele 3D sunt afectate de un număr de factori. Acestea cuprind tipul memoriei acceleratorului grafic și implementarea prin hardware a unor facilități ca maparea corectă în perspectivă a texturilor, umbrirea *Gourad*, *mastering* de tip “*scatter-gather*” pe magistrala PCI, și buffer Z de 32 biți.

Tipul memoriei

În timp ce memoria cu port unic SGRAM este adecvată pentru accelerarea jocurilor 3D, pentru grafica complexă, profesională 3D sunt necesare tipuri de memorii cu porturi duale, ca VRAM sau WRAM. Posibilitatea acestor memorii de a citi informația simultan cu scrierea acesteia este necesară pentru a menține performanțele 3D ridicate la rezoluții înalte, în modul “*true-color*” și la frecvențe de reîmprospătare ridicate.

Maparea corectă în perspectivă a texturilor

În lumea reală perspectiva unui privitor față de un obiect se modifică pe măsură ce poziția privitorului se schimbă. Pentru a se crea aceleași efecte optice într-o aplicație 3D, texturile trebuie corectate continuu pentru a adapta perspectiva privitorului relativ la obiect. Dacă maparea texturilor nu este corectă în perspectivă, imaginea nu va fi realistă.

Anumite acceleratoare 3D simulează prin hardware maparea corectă în perspectivă a texturilor utilizând un artificiu numit aproximare cuadratică. Aceasta însă poate avea drept rezultat imagini 3D distorsionate și erori inacceptabile de interpretare a imaginilor.

Mastering pe magistrala PCI

Implementarea acestei facilități în cadrul circuitului video este esențială pentru obținerea unei frecvențe ridicate de cadre. Există două tipuri de *mastering* pe magistrala PCI.

- *Mastering de bază*

Acceleratorul 3D poate funcționa independent de UCP, dar numai pentru perioade scurte de timp, după care trebuie să întrerupă UCP și să solicite comenzi. În cazul operațiilor intensive ca volum de date, cum sunt operațiile 3D, aceasta poate minimiza avantajele obținute prin *mastering*.

- *Mastering de tip "scatter-gather"*

Acceleratorul 3D poate funcționa aproape independent de UCP, obținând astfel avantajele performanțelor ridicate. Anumite adaptoare pot atinge o îmbunătățire de până la 50% față de adaptoarele fără *mastering* de magistrală și 30% față de adaptoarele cu *mastering* de bază.

Buffer Z de 32 biți

Performanțele graficii 3D depind de un buffer Z puternic, care testează valoarea Z, sau adâncimea, fiecărui pixel care se va reprezenta în fața altor pixeli. Aceasta determină pixelii sau poligoanele care vor fi desenate și cele care vor fi ascunse. Bufferul Z este o parte a memoriei video rezervată pentru a păstra valoarea Z a fiecărui pixel. Implementarea unui buffer Z de 32 biți de către acceleratorul grafic este o cerință importantă pentru multe aplicații de proiectare.

Bufferul Z este important atât pentru jocuri, cât și pentru aplicații profesionale. Pentru aplicații 3D pretențioase ca 3D *Studio Max* și aplicații *Open GL*, un buffer Z de 32 biți este obligatoriu.

4.5.5.4. Tipuri de adaptoare video 3D

Există mai multe abordări diferite pentru a asigura unui sistem posibilități 3D. Cu toate că adaptoarele și tehnologiile disponibile se schimbă cu rapiditate, în general adaptoarele disponibile se pot împărți după cum urmează:

- *Adaptoare convenționale (Numai 2D)*

Acestea sunt adaptoare video obișnuite care nu au integrate funcții speciale de accelerare 3D. De obicei acestea sunt fie adaptoare mai vechi, fie adaptoare mai noi care sunt optimizate pentru funcții 2D.

- *Adaptoare 3D dedicate*
 Acestea sunt acceleratoare care sunt proiectate numai pentru funcții hardware 3D. Deoarece nu au funcții de accelerare convențională 2D, în cele mai multe cazuri ele trebuie să lucreze cu un adaptor 2D pentru a asigura performanțe bune 2D+3D. Cele mai multe adaptoare 3D de calitate mai înaltă sunt de acest tip. Acestea utilizează de obicei un conector AVFC sau VMC pentru conectarea directă la adaptorul 2D. Aceasta permite adaptorului 3D să execute funcțiile sale de accelerare fără a avea nevoie de un circuit RAMDAC sau o logică proprie de control a magistralei. În general, aceasta este soluția cea mai bună pentru grafica pretențioasă, dar ea implică costul a două adaptoare video.
- *Adaptoare combinate 2D+3D*
 Cu scopul de a evita utilizarea unui adaptor suplimentar separat pentru accelerarea 3D, au fost dezvoltate adaptoare care execută atât funcții 2D, cât și 3D. Pentru mulți utilizatori, acesta este un compromis acceptabil. Cele mai multe din aceste adaptoare asigură performanțe 2D moderate sau bune, și dispun de unele din funcțiile de accelerare 3D. Totuși, aceste adaptoare nu asigură de obicei nivelul de performanță sau facilitățile adaptoarelor dedicate 3D.

4.5.5.5. Aspecte software

Pentru a beneficia de facilitățile de accelerare 3D, este necesar ca programele să cunoască aceste facilități și să le utilizeze. Din păcate, grafica 3D fiind un domeniu relativ nou, au apărut numeroase adaptoare diferite și incompatibile. Este necesar ca un program să fie compatibil cu un anumit adaptor pentru a beneficia de avantajele oferite de circuitele 3D.

Multe din facilitățile circuitelor 3D sunt utilizate de jocuri. Pe lângă o versiune obișnuită a unui anumit joc, pot fi create versiuni speciale care sunt adaptate pentru diferite seturi de circuite 3D. O asemenea versiune va avea fie o grafică superioară, fie o viteză superioară, sau ambele. Totuși, aceasta nu va funcționa cu un tip diferit de adaptor 3D.

Pentru a rezolva această problemă, au fost dezvoltate mai multe interfețe API (*Application Program Interface*) și biblioteci standard. Aceste interfețe și biblioteci, ca *Direct3D* și *OpenGL*, sunt proiectate pentru a permite ca programele să fie scrise în mod generic, fără a fi adaptate pentru fiecare set de circuite 3D. Figura 4.17 prezintă aplicațiile 3D și unele din interfețele acceptate [14].

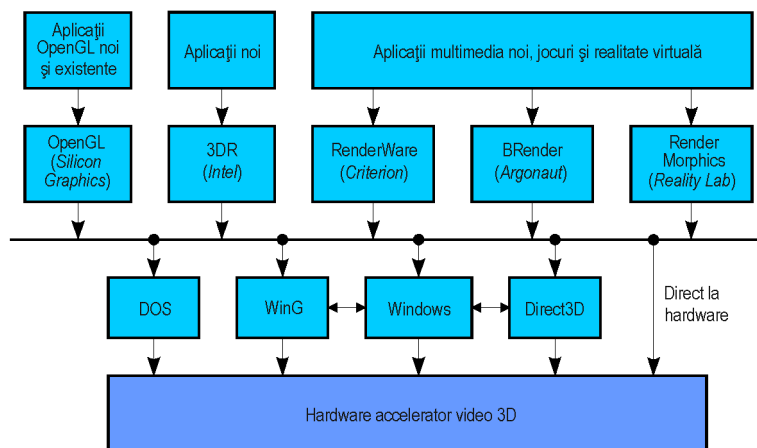


Figura 4.17. Aplicații și interfețe 3D.

Jocurile care sunt create prin utilizarea acestor biblioteci pot evita necesitatea de a fi adaptate pentru diferite circuite. Totuși, aceste sisteme sunt încă în stadiile lor de început și au propriile lor probleme, inclusiv probleme de performanță. Programele nu rulează cu viteza care s-ar obține dacă circuitele ar fi programate direct, ceea ce reprezintă compromisul în obținerea universalității bibliotecilor.

De exemplu, performanțele grafice ale aplicațiilor sunt reduse dacă ele utilizează interfața standard *Screen API Windows*. Aplicațiile nu pot scrie direct în memoria video. *Microsoft* a creat o extensie a interfeței pentru *Windows*, numită *WinG (Windows for Games)*. Aceasta nu este un pachet 3D complet, dar permite aplicațiilor să creeze imagini grafice în memoria sistem sau în partea neutilizată a memoriei video, fără a utiliza interfața *API Windows*. După ce imaginea este gata pentru a fi afișată, *WinG* o transferă în partea activă a memoriei video, utilizând interfața *API standard*.

Direct3D, introdus în 1996, este o componentă importantă a setului de interfețe *API DirectX*, care este setul unificat de interfețe interactive multimedia al *Microsoft*. *DirectX* cuprinde de asemenea următoarele interfețe *API*:

- *DirectDraw*, pentru acceleratoare grafice 2D
- *DirectSound*, pentru sunet pozițional 3D
- *DirectPlay*, pentru jocuri *on-line* cu până la 256 de jucători

Direct3D cuprinde noi caracteristici destinate necesităților dezvoltării jocurilor 3D. Un exemplu este *DrawPrimitive*, un set compact de funcții esențiale

ale de nivel inferior pentru desenarea triunghiurilor, a liniilor și punctelor 3D pe suprafețele bazate pe biblioteca *DirectDraw*. Prin utilizarea bibliotecii *Direct3D*, programatorii pot exploata performanțele ridicate ale tehnologiei *Intel MMX* și noua interfață *AGP*. În plus, arhitectura scalabilă a *Direct3D* permite programatorilor să integreze grafica 3D cu alte tipuri de multimedia, ca sunete 3D și secvențe video *MPEG*.

4.5.6. Interfața AGP

4.5.6.1. Principiul AGP

AGP (Accelerated Graphics Port) este o nouă interconexiune pentru acceleratoarele grafice din sistemele bazate pe procesorul *Pentium II* și *Pentium III*, utilizate pentru grafică 3D și redarea secvențelor video. Utilizatorii calculatoarelor PC pot beneficia acum de tipul de grafică 3D și video disponibile în prealabil numai pe stațiile de lucru cu prețuri mari (20.000 \$ sau peste).

Procesorul *Pentium II* constă dintr-un nucleu încapsulat cu o memorie *cache* integrată de nivel 2 (*L2*). Acest procesor dispune de asemenea de o arhitectură *Dual Independent Bus (DIB)*, în care două magistrale independente conectează nucleul cu memoria *cache L2* și cu magistrala sistem a calculatorului. Faptul că ambele magistrale pot funcționa în același timp îmbunătățește semnificativ performanțele procesorului, deoarece procesorul poate executa instrucțiuni din memoria *cache L2* și simultan poate comunica cu dispozitive externe.

Noile aplicații grafice 3D impun cerințe riguroase calculatoarelor PC, cuprinzând calcule geometrice mai rapide, o interpretare grafică mai sofisticată, și texturi mai detaliate. Cu toate că procesorul *Pentium II* este adaptat pentru a executa calcule geometrice sporite (cu o rată mai mare de triunghiuri pe secundă), iar generația viitoare de controlere grafice poate implementa o mare varietate de efecte grafice, dimensiunea crescută a texturilor a devenit o problemă importantă.

O problemă o reprezintă dimensiunea memoriei video utilizată de controlerele grafice. În mod tipic, această memorie are o dimensiune de 2-4 MB. Cu toate acestea, au început să apară aplicații grafice care utilizează peste 20 MB pentru o singură textură. Memoria video poate fi extinsă pentru a satisface aceste cerințe, dar o asemenea soluție este foarte costisitoare.

A doua problemă este rata de transfer permisă de magistrala *PCI*. Controlerele grafice trebuie să încarce în prealabil texturile din memoria sistem în memoria lor *RAM* locală. Deoarece dimensiunea texturilor a crescut, magistrala *PCI* a început să devină congestionată. Problema este chiar mai acută în cazul aplicațiilor care implică redarea secvențelor video.