

## MOŻLIWOŚCI I PERSPEKTYWY WSPÓŁCZESNEJ GRAFIKI KOMPUTEROWEJ

### POTENTIAL OF MODERN COMPUTER GRAPHICS

Tadeusz Szuba

Katedra Automatyki, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki  
Akademia Górniczo-Hutnicza

**Słowa kluczowe:** Nowoczesna grafika komputerowa, zjawiska fizyczne, moc obliczeniowa, CPU<sup>1</sup>, GPU, GPGPU, obliczenia równoległe

Keywords: Modern computer graphics, physical phenomena, computational power, CPU, GPU, GPGPU, parallel processing

## Wprowadzenie

W rozwoju informatyki można wskazać na klasy komputerów i związane z nimi aplikacje (nazwijmy je KA), które pełniły w danym okresie rolę *dominujących koni pociągowych* ogólnie rozumianej informatyki. Operowanie konkretnymi datami jest ryzykowne, bowiem mieliśmy tu do czynienia z przenikającymi się, rosnącymi w siłę i zanikającymi wpływami poszczególnych elementów na rozwój informatyki. W odniesieniu do dnia dzisiejszego można wręcz mówić o *ekosystemie informatycznym* funkcjonującym na Ziemi, w którym występują (pojawiają się i znikają z rynku) i współpracują ze sobą zarówno mikroskopijne procesory + aplikacje; współistniejące i w różny sposób powiązane<sup>2</sup> z monstrami obliczeniowymi, których główną aplikacją jest np. model pogody, czy model trzęsienia<sup>3</sup> Ziemi. Można historycznie wskazać następujące KA:

1. komputery typu mainframe<sup>4</sup> i aplikacje wojskowe (apogeum lata 80. XX wieku);
2. komputery PC i aplikacje biurowe + proste gry (apogeum lata 90. XX wieku do teraz);
3. serwery sieciowe + Internet (apogeum lata 90. XX wieku do teraz);
4. ?? (grafika komputerowa + gry).

---

<sup>1</sup> CPU – Główny Procesor Komputera (*Central Processing Unit*); GPU – Procesor Graficzny Komputera (*Graphics Processing Unit*); GPGPU – Procesor Graficzny Komputera zdolny także do zastosowań, innych niż graficzne (*General Purpose GPU*).

<sup>2</sup> Powiązanie takie może mieć np. charakter strukturalny: np. superkomputery buduje się z procesorów zaprojektowanych dla komputerów PC (Xeon) lub gier (CELL).

<sup>3</sup> [http://en.wikipedia.org/wiki/Earth\\_Simulator](http://en.wikipedia.org/wiki/Earth_Simulator)

<sup>4</sup> [http://en.wikipedia.org/wiki/Mainframe\\_computer](http://en.wikipedia.org/wiki/Mainframe_computer)

Aktualnie można mówić o zaistnieniu nowej klasy komputerów i związanej z nią aplikacji, która w opinii autora staje się kolejnym *koniem pociągowym* ogólnie rozumianej informatyki. Jest to karta graficzna rozumiana jako komputer + aplikacje umownie nazwane tu „nowoczesna grafika komputerowa”. Na aplikacje te składają się: inżynierska grafika komputerowa typu CAD/CAM/CAE, grafika komputerowa typu filmowego i wreszcie współczesne gry komputerowe – takie jakie są używane dziś w testach wydajności kart graficznych.

Celem tego artykułu jest jasne uświadomienie Czytelnikom tego zjawiska (grafika komputerowa jako *koń pociągowy* informatyki), tak aby z należyтым wyprzedzeniem, w ramach dyscypliny czy obszaru w jakim pracujemy (tu geomatyki), konstruktywnie ustosunkować się do tego zjawiska, wręcz wykorzystać go w długofalowym planowaniu naszych działań. W przypadku geomatyki konsekwencją może być pojawienie się zupełnie nowych obszarów zainteresowania oraz nowych metod, zaadaptowanych z obszarów grafiki komputerowej. Te nowe obszary zastosowań często pojawiają się dlatego, że wreszcie można szybko i w zakresie dostępnego budżetu coś policzyć – najlepszym przykładem są obiekty fraktalne.

W tym punkcie artykułu należy już powiedzieć, co jest tą *istotą rzeczy*, w grafice komputerowej i kartach graficznych, co pozwala na stwierdzenie, że jest to następny dominujący *koń pociągowy informatyki*. Rzeczą tą jest konieczność modelowania np. dla potrzeb gier i animacji filmowych zjawisk fizycznych (Erleben, et al., 2005) oraz ogólnie rozumianych istot – nie dość tego – tworzone są w ramach współczesnej grafiki komputerowej nowe jakościowo wirtualne byty, odchodzące już dość daleko od realnych pierwowzorów. Powoduje to, że złożoność obliczeniowa wynikających stąd problemów jest trudna do oszacowania w oparciu o „podręcznikową teorię złożoności obliczeń”. Tym samym zmuszeni jesteśmy do daleko idących uproszczeń albo do tworzenia autorskich modeli rzeczywistości, Wirtualnej Rzeczywistości, gdzie np. woda nie jest wodą ale ją wizualnie przypomina. Z drugiej strony kolosalny rynek gier komputerowych wymusza prace nad budową i wdrożeniem nowych systemów obliczeniowych (architektur komputerowych). Pamiętajmy, że karta graficzna/procesor graficzny (GPU) ma już w założeniu inną organizację obliczeń (dominacja obliczeń równoległych) niż uniwersalny w założeniu procesor (CPU) w zwykłym komputerze PC, czy też w stacji roboczej. Stąd też wynika trudność z przeniesieniem procesu obliczeniowego zaprojektowanego dla CPU na procesor GPU.

W kolejnych rozdziałach artykułu zostanie podjęta próba przybliżenia podjętego zagadnienia, w kontekście oczekiwanych przyszłych możliwości wykorzystania osiągnięć współczesnej grafiki komputerowej na polu np. geomatyki.

## Karta graficzna jako układ obliczeniowy

Popatrzmy na to co musi zrobić dzisiejszy PC/stacja robocza<sup>5</sup> (rys. 1) podczas pracy nad symulacją i wizualizacją obiektu lub procesu przemysłowego (np. wirtualny prototyp koparki na placu budowy), podczas badania zjawiska fizycznego (np. opływ cieczy w wymienniku ciepła) lub analizy zjawiska socjologicznego dla celów np. architektury (np. symulacja przemieszczania się ludzi w np. centrum handlowym), czy też dla celów tworzenia animacji reklamowych (np. zbójnik Janosik reklamujący proszek do prania). Należy zwrócić uwagę na ok. 5-krotnie większą moc obliczeniową procesora graficznego GPU (np. proce-

<sup>5</sup> „Wypasiony” PC z procesorem Intel Core i7, mocną kartą graficzną i 30” monitorem o rozdzielczości 2560x1600 w praktyce jest już stacją roboczą.

sor CELL o mocy ok. 256 GFlops @ 4 Ghz – 9 rdzeni: 1 + 4 IU + 4 FPU; wprowadzenie na rynek rok 2007) nad nowszym o rok procesorem CPU (np. Intel Core i7 965 o mocy ok. 51.2 Gflops – 4 rdzenie; wprowadzenie na rynek rok 2008). Rzecz w tym, że obliczeniowe zadania graficzne (etapy przetwarzania) dla procesora GPU są aktualnie jasno zdefiniowane, tak jak pokazuje to w uproszczeniu rysunek 2. Są to w zasadzie zadania polegające na policzeniu i wyświetleniu na ekranie wyglądu „nieruchomej skorupy (powierzchni) obiektu”. Przewaga procesora GPU już na wstępie wynika stąd, że w grafice komputerowej równoległość obliczeń pojawia się w sposób naturalny, nie musi stanowić przedmiotu dodatkowych prac przygotowawczych (człowiek) czy obliczeń (zrównoleglający kompilator); co rujnuje w praktyce zysk ze zrównoleglania obliczeń w CPU.

W konsekwencji obliczenia w GPU implikują drzewiastą architekturę procesora, z zainstalowaną na poszczególnych etapach mocą obliczeniową rosnącą prawie wykładniczo w miarę zbliżania się do umownego ekranu. Dzisiejsze karty graficzne instalowane na łączu PCI-E komputera PC wyglądają w sensie architektury komputera (rys. 3) prawie tak samo jak super-stacje robocze<sup>6</sup> SGI (Silicon Graphics Inc.) z roku 1991 np. SGI 480 VGX. Wtedy takie komputery miały wielkość sporej szafy.

Zajmijmy się teraz tym co aktualnie liczy karta graficzna, a co będzie musiała liczyć w najbliższej przyszłości.

## Procesy obliczeniowe w nowoczesnej grafice komputerowej

W chwili obecnej można powiedzieć, że karta graficzna w komputerze jest zoptymalizowana na obliczanie wyglądu statycznego obiektu (rozumianego trochę jak rzeźba). Tyle, że sytuacja w tym zakresie doszła już do pewnego punktu granicznego. Jak bowiem policzyć wygląd leżącej się wody „zamkniętej w kadrze”, tj. dla danego momentu czasu; bez policzenia jak ta woda łała się w poprzedzających momentach? Podobnie nie da się wyliczyć jak ma się ułożyć realistyczna tkanina na ciele wirtualnej (nieruchomej !) modelki bez zasymulowania procesu dynamicznego układania się tej tkaniny na jej ciele – a co dopiero można powiedzieć o momencie, gdy ta modelka zacznie się poruszać w stronę wybiegu aby pokazać się publiczności na pokazie mody. Jak ogromne jest to zagadnienie i jaki ma ono kontekst finansowo-biznesowy widać w pracach MIRALab<sup>7</sup> – Geneva University. Badania tam prowadzone są ukierunkowane na zbudowanie systemów dla celów komputerowego projektowania kolekcji mody dla wielkich projektantów, angażujących aktualnie ogromne fundusze i zasoby ludzkie, konieczne dla wyprodukowania i pokazania wiosenno/letnio/jesienno/zimowych kolekcji. Takie przykłady można mnożyć bez końca.

Doświadczenia autora z pracami dyplomowymi studentów pokazują, że obecnie mając do dyspozycji mocny komputer i profesjonalne oprogramowanie dla grafiki komputerowej<sup>8</sup> np. Softimage (Rossano, 2005) nie jest problemem zrobienie ładnej (w ostateczności ciekawej artystycznie) jaskółki, ale problemem jest aby ta jaskółka poprawnie latała. Jeszcze gorzej jest w przypadku np. konia arabskiego w biegu. Do pewnego stopnia daje się „obejść

<sup>6</sup> Autor pracował w latach 1991–1997 na takim komputerze.

<sup>7</sup> <http://www.miralab.unige.ch/>

<sup>8</sup> Obecnie firma Autodesk jest w praktyce monopolistą jako dostawca takiego oprogramowania. Jest właścicielem wszystkich najważniejszych systemów programowania w grafice 3D: Softimage, 3D Studio, Maya, itd.

problem” przy pomocy tzw. systemów przechwytywania ruchu (Motion Capture<sup>9</sup>). W systemach takich ciało człowieka zostaje oklejone specjalnymi znacznikami (markerami), których późniejszy ruch wynikający z ruchu samego człowieka śledzą specjalne kamery i komputery zdolne do określenia położenia tych markerów w przestrzeni. W ten sposób można bardzo wiernie zarejestrować jak przestrzennie wygląda ruch ciała np. tancerki, tenisisty, itp. Metoda ta pozwala na relatywnie tanie „ożywienie” postaci w grafice komputerowej, poprzez przeniesienie ruchu aktora na ruch obiektu graficznego. Zachodzi pytanie jak to zrobić w przypadku jaskółki, pająka, itp. w ich naturalnym środowisku, a nie w sali z kamerami śledzącymi markery – jeśli odpowiadające obiekty graficzne muszą zagrać swoją rolę w jakimś filmie? Musimy tu pamiętać, że oko i mózg ludzki są ewolucyjnie bardzo wyczulone na wykrywanie defektów ruchowych u obserwowanej osoby, zwierzęcia, obiektu, itp. bo to często decydowało o przeżyciu lub zdobyciu np. pożywienia.

Dochodzimy zatem do punktu, w którym stwierdzamy, że karta graficzna musi policzyć następujące elementy:

- 1) **zjawiska fizyczne sterujące zachowaniem się obiektu graficznego**, na dwu poziomach
  - **wewnętrznej dynamiki obiektu**. W przypadku obiektów ożywionych może dochodzić tutaj dodatkowo warstwa psychiczna (np. kot musi zachowywać się „po kociemu”). Innymi słowy nie wystarcza zbudowanie wirtualnej graficznej maszyny: szkielet + mięśnie + skóra, która będzie udawała kota. Musi powstać jeszcze autonomiczny moduł sterujący (Dave, 2009).
  - **interakcji z otoczeniem**. Podobnie jak poprzednio, w przypadku obiektów ożywionych oprócz oddziaływania ze środowiskiem (np. skradający się między krzakami kot dostosowuje swoje ruchy do otaczającego środowiska), mogą pojawić się np. zjawiska stadne. Pięknym przykładem jest scena stampede<sup>10</sup> z filmu Lion King<sup>11</sup>.
- 2) **realistyczny wygląd obiektu** (po wyliczeniu aktualnej dla danej chwili konfiguracji obiektu graficznego). Mimo postępu na tym polu, dalej jest to ogromne zagadnienie obliczeniowe, zwłaszcza gdy mamy do czynienia z obiektami, które są w jakimś stopniu przezroczyste: np. zarośla, włosy, itp. Dodatkowo wejście standardu FULL HD (1920x1080) zwiększyło znacznie ilość pikseli jakie trzeba policzyć np. stosując RayTracing<sup>12</sup>. Powoli także wchodzi do użytku w grafice komputerowej monitory 30” o rozdzielczości 2560x1600, tzw. WQXGA (Wide Quad-XGA).

Jeśli chodzi o zjawiska fizyczne dla potrzeb grafiki komputerowej (Erleben, et al., 2005), to trzeba pamiętać tutaj o dwu zasadniczych elementach. Pierwszy z nich jest oczywisty, ale drugi już nie.

1. Przypuszczalnie istnieje nieskończenie wiele możliwych zjawisk fizycznych, psychiczno-umysłowych i innych, które mogą stać się obiektem zainteresowania grafiki komputerowej. Jak wiemy, nasza wiedza o realnej rzeczywistości, mimo kolosalnego postępu, jest co najmniej znikoma. To implikuje, że oprogramowanie dla potrzeb grafiki komputerowej będzie gwałtownie rosło jeśli chodzi o rozmiar i trudno wskazać inny dział informatyki, w którym zjawisko wzrostu rozmiaru oprogramowania występuje na taką skalę. Już dziś pakiety graficzne z wbudowanymi modułami symulacji zjawisk fizycznych są dużo większe niż inne pakiety oprogramowania (aplikacje).

<sup>9</sup>[http://en.wikipedia.org/wiki/Motion\\_capture](http://en.wikipedia.org/wiki/Motion_capture)

<sup>10</sup><http://en.wikipedia.org/wiki/Stampede>

<sup>11</sup> Walt Disney Pictures.

<sup>12</sup>[http://pl.wikipedia.org/wiki/Ray\\_tracing](http://pl.wikipedia.org/wiki/Ray_tracing)

2. Okazuje się, że zjawiska fizyczne i psychiczno-umysłowe, które potrzebuje grafika komputerowa, nie muszą, a nawet nie powinny dokładnie odzwierciedlać realnej rzeczywistości. Muszą one w sposób kreatywny przypominać realną rzeczywistość, ale jednocześnie muszą umożliwiać programiście swobodne edytowanie danego zjawiska, np. muszą pozwolić na wytworzenie fal oceanu, które nigdy by w rzeczywistości nie powstały, zmiany grawitacje, itp. Oznacza to dla np. fizyka budującego dany plug-in dla np. systemu programowania grafiki 3D Softimage konieczność najpierw dogłębnego poznania danego realnego zjawiska (w sensie parametrycznym), a potem stworzenia meta-zjawiska, w którym dane realne zjawisko jest tylko jednym z wariantów. Mamy tu jak widać prawie Boski akt stworzenia.

Jeszcze raz należy podkreślić fakt, że nie należy oczekiwać, że procesor główny (CPU) przejmie choćby w części obowiązki związane z liczeniem zjawisk fizycznych dla potrzeb karty graficznej, czy też GPU. Wynika to stąd, że obowiązki obsługi komputera jako całości (w tym kontakt z człowiekiem), nie pozwalają na to. Wystarczy wspomnieć o coraz bardziej złożonych systemach operacyjnych i o konieczności ciągłego monitorowania bezpieczeństwa komputera w sposób ciągły podłączonego do sieci (ochrona przed atakami wrogiego oprogramowania).

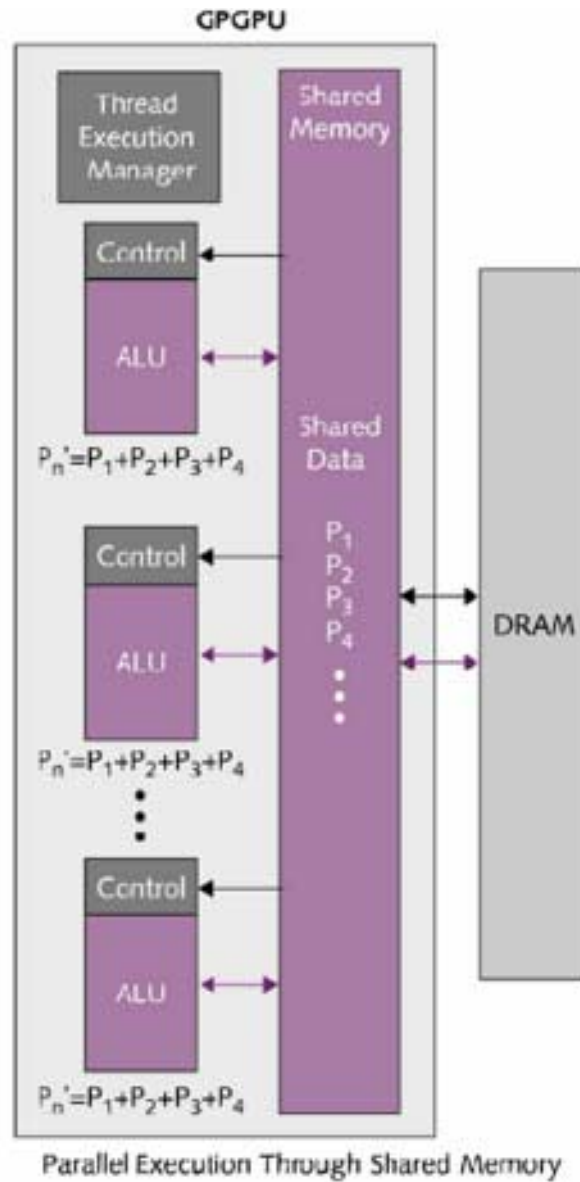
Ta mnogość zjawisk, które musi umieć zasymulować karta graficzna, już w najbliższej przyszłości implikuje, że będzie ona musiała na powrót stać się uniwersalnym komputerem. Już obserwujemy symptom tego zwrotu – jest nią koncepcja CUDA realizowana przez firmę NVIDIA, jednego z dwu czołowych producentów kart graficznych.

W koncepcji tej procesor GPU (*Graphics Processing Unit*) staje się GPGPU (*General Purpose GPU*) i struktura karty graficznej zostaje tak przekonstruowana, aby umożliwić wpisanie i realizację wątku obliczeń<sup>13</sup> (niekoniecznie już tylko graficznych) do czegoś w rodzaju pod-procesora (realizującego zadania FPU); do tzw. *programmable pixel shader*. Istotne jest to, że firma NVIDIA zapewnia kompilator C++, który jest w stanie „rozpisać” wątki obliczeń na wewnętrzne procesory karty graficznej, przy stosunkowo małym zaangażowaniu programisty. Oczywiście musi on zdawać sobie sprawę dla jakiego środowiska buduje proces obliczeniowy, i o ile się da, powinien on rozdzielać wątki i zasoby<sup>14</sup> tak, aby kompilator sobie dalej już sam poradził.

W rezultacie powstaje znakomite środowisko obliczeniowe (rys. 4) dla obliczeń zjawisk fizycznych, zachowania się wirtualnych istot, itp. Swoistą ciekawostką jest to, że tylko nieliczne uniwersytety (te na najwyższym poziomie) uczą studentów praktyki programowania równoległego – bo potrzebne jest do tego celu drogie i wyspecjalizowane laboratorium komputerowe. Tymczasem w oparciu o technologie CUDA, w zasadzie każdy kto już ma laboratorium grafiki wyposażone w sensowne komputery z kartami graficznymi o strukturze odpowiadającej CUDA, może zacząć uczyć tam programowania równoległego, np. dla obliczeniowych zagadnień w fizyce, mechanice górotworu, itd. Jest to kolejny przykład dlaczego grafikę komputerową należy zacząć uważać za *konia pociągowego* współczesnej informatyki.

<sup>13</sup> Tzw. thread.

<sup>14</sup> Autor w latach 1995–1996 programował obliczenia równoległe na 8-procesorowej maszynie SGI 480 VGX, pracującej pod systemem operacyjnym IRIX, stosując kompilator Parallel C. Było to bardzo czasochłonne i stresujące zadanie programistyczne, dające więcej satysfakcji naukowej niż rezultatów w sensie przyspieszenia obliczeń.



**Rys. 4.** Organizacja współpracy 8 jednostek typu FPU (*Floating Point Unit*) pracujących jako *programmable pixel shaders* w ramach jednostki obliczeniowej karty graficznej Nvidia GeForce 8xx (źródło Halfhill, 2008)

## Podsumowanie

Tezą artykułu było stwierdzenie, że współczesna grafika komputerowa staje się *koniem pociagowym* ogólnie rozumianej informatyki, i to przypuszczalnie ona właśnie będzie narzucała przyszłe trendy w zakresie oprogramowania (*software*), architektury komputera i procesorów (*hardware*) i wreszcie zastosowań. Mając tego świadomość, należy tak orientować rozwój aplikacji jakie stosujemy i problemy jakie przy ich pomocy rozwiązujemy, aby odnieść maksymalną korzyść z trafnego odgadywania trendów w informatyce. Pamiętajmy, że opanowanie danej platformy oprogramowania i aplikacji to przynajmniej 2–3-letni proces i błędne decyzje mogą w przyszłości być bardzo kosztowne, w zależności od tego, jak daleko zabrnęliśmy w przysłowiową ślepią uliczkę.

Na potrzeby geomatyki, jak się wydaje, należy zaproponować przeniesienie obliczeń na karty graficzne jako płaszczyznę obliczeniową. Niezbędne jest dokonanie uważnej analizy osiągnięć grafiki komputerowej, a zwłaszcza obliczeń związanych ze środowiskiem gier komputerowych, celem zaadaptowania wybranych elementów (oczywiście po odpowiednim rozwinięciu) do celów geomatyki. W grach komputerowych w bardzo szybkim tempie pojawiają się różnorodne elementy związane z problematyką geomatyki, które nie występują jeszcze w profesjonalnych aplikacjach, bo są zbyt złożone obliczeniowo. Tymczasem w grach, przy „swobodnym potraktowaniu zagadnienia” już występują, są wykorzystywane, a co najważniejsze są dla nich zaproponowane metody obliczeniowe. Przy przeniesieniu obliczeń na platformę CUDA, już na poziomie komputera za 15 000–20 000 zł, możliwe jest podjęcie prób wdrożenia takich aplikacji w ujęciu profesjonalnym, z wykorzystaniem programowania równoległego.

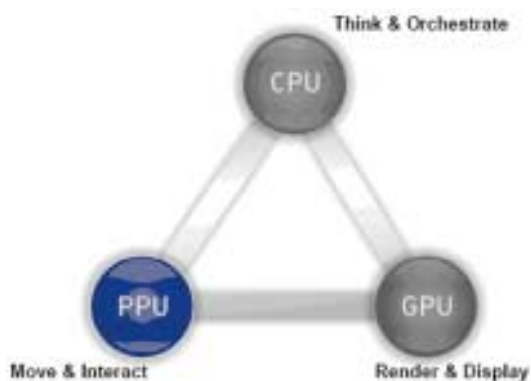
## Literatura

- Dave M., 2009: Behavioral Mathematics for Game AI. Course Technology PTR. 1-ed. 2009.  
Halfhill T.R., 2008: Parallel processing with CUDA. Microprocessor Repport. January 2008.  
Rossano A., 2005: SOFTIMAGE | XSI 5: The Official Guide. Course Technology PTR. 1-ed. 2005  
Erleben K., Sporning J., Henriksen K., Dohlmann H., 2005: Physics Based Animation (Graphics Series). Charles River Media.

## Abstract

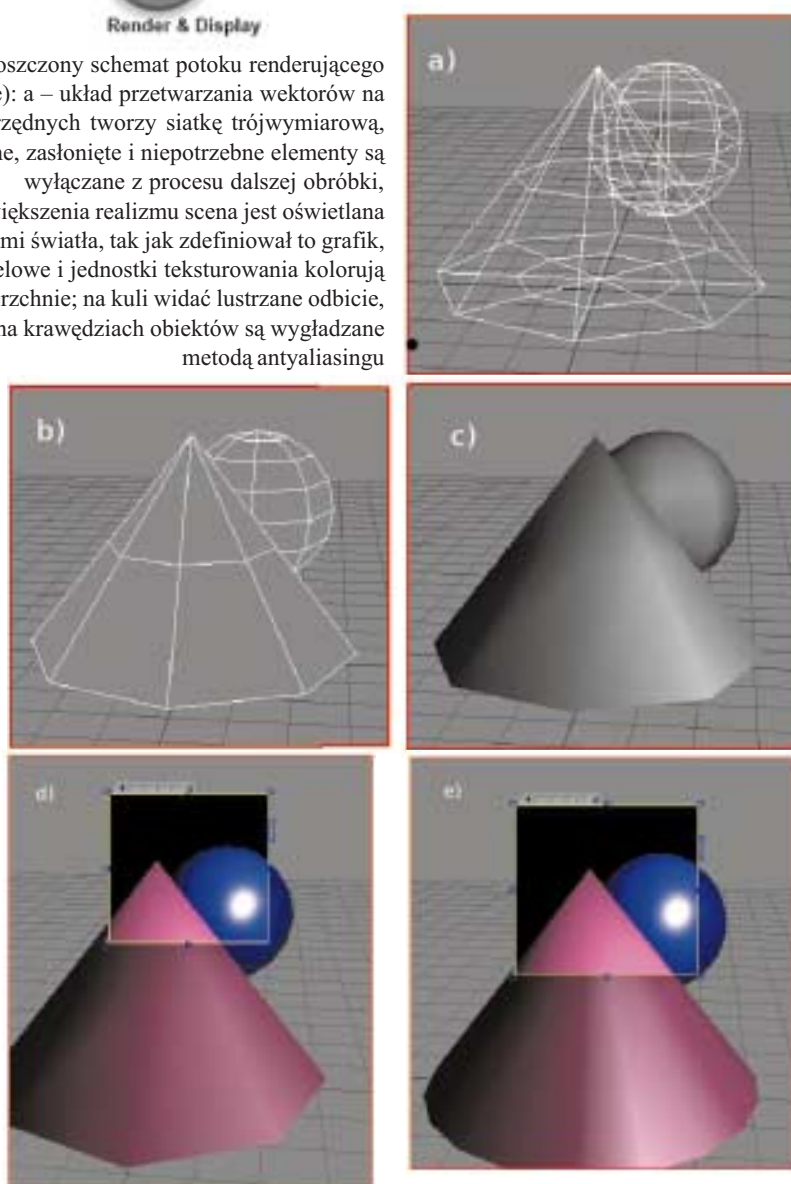
*The paper deals with the problem: what is modern computer graphics now and what is its potential. If we think in terms of "the centre of gravity", modern computer graphics is moving from the art towards capturing the essence of an object or a being to be modeled. In other words, key problems for the computer graphics are physical phenomena (e.g. liquids), mechanical properties (e.g. textile, hairs) or even mental properties of virtual beings. Therefore, modern computer graphics requires extremely high computational abilities. Advanced computer games demonstrate this very well. Having all this in mind, many researchers think that modern computer graphics is the main leading force in the development of modern computer science. On the basis of above remarks, the paper tries to resume the application areas of modern computer graphics now and in the near future.*

dr hab. inż. Tadeusz Szuba  
szuba@ia.agh.edu.pl  
tel. +48 12 617 3409

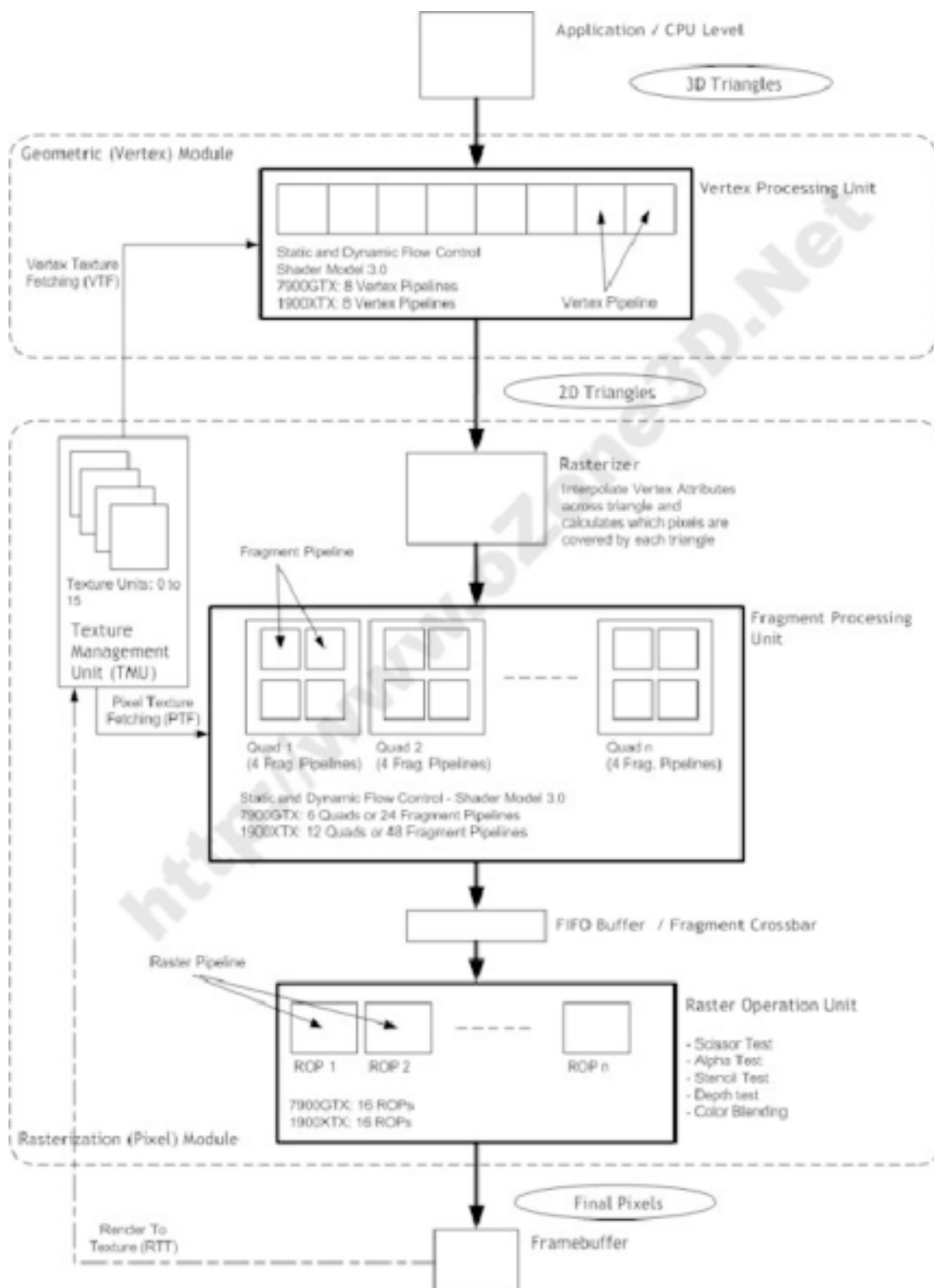


**Rys. 1.** Rozdział czynności we współczesnym PC/stacji roboczej  
(źródło: AGEIA Physx Processor Website)

**Rys. 2.** Uproszczony schemat potoku renderującego (źródło własne): a – układ przetwarzania wektorów na podstawie współrzędnych tworzy siatkę trójwymiarową, b – niewidoczne, zasłonięte i niepotrzebne elementy są wyłączane z procesu dalszej obróbki, c – dla zwiększenia realizmu scena jest oświetlana źródłami światła, tak jak zdefiniował to grafik, d – shadery pikselowe i jednostki teksturowania kolorują powierzchnie; na kuli widać lustrzane odbicie, e – nierówności na krawędziach obiektów są wygładzane metodą antyaliasingu







Rys. 3. Ogólna architektura karty graficznej  
 (źródło [http://www.ozone3d.net/tutorials/gpu\\_sm3\\_dx9\\_3d\\_pipeline\\_p02.php](http://www.ozone3d.net/tutorials/gpu_sm3_dx9_3d_pipeline_p02.php))