

Windows NT 4.0

Windows 2000

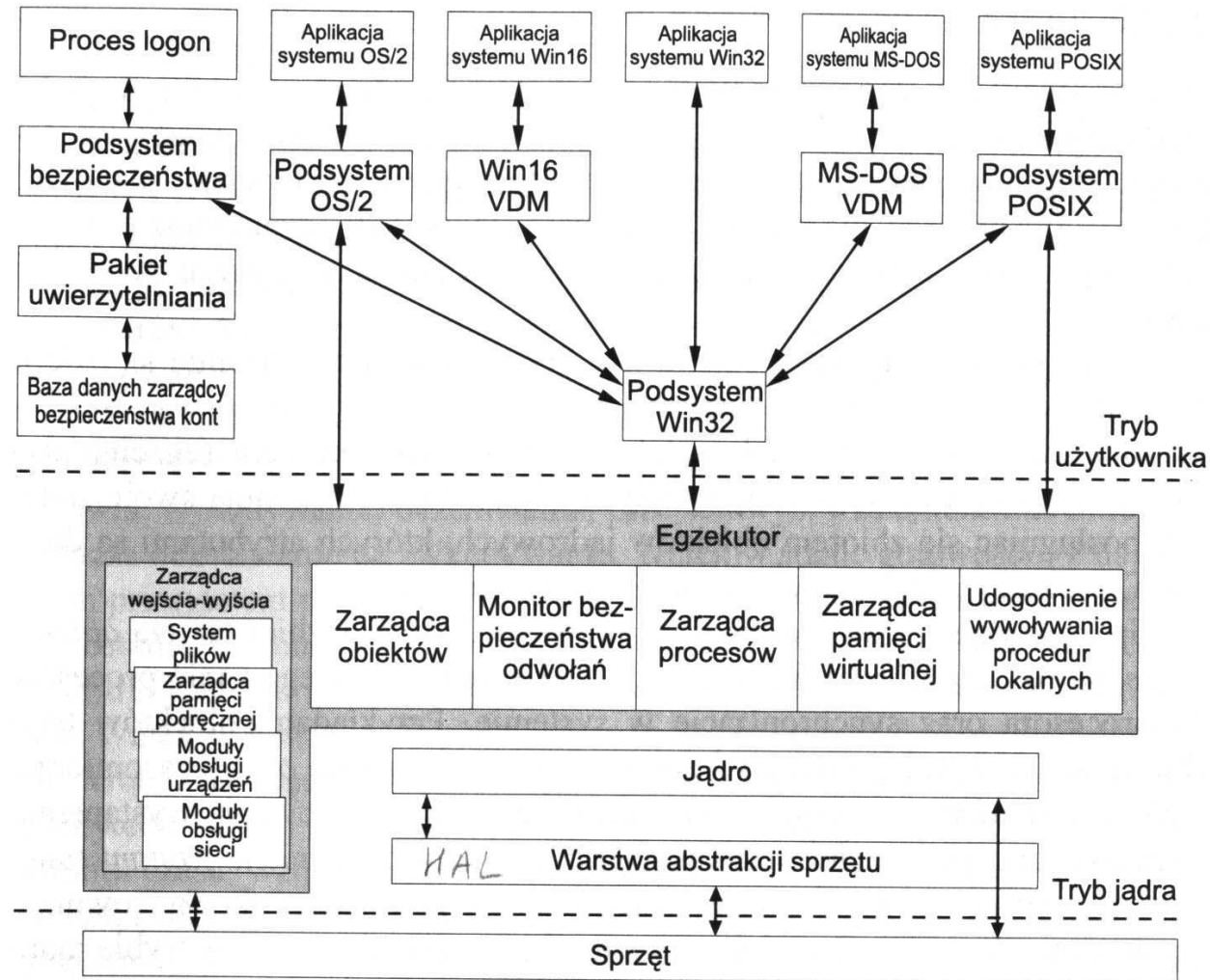
Windows XP

System operacyjny WinNT

- cechy systemu
 - wieloprogramowy, podział czasu, procesy i "wątki poziomu jądra"
 - jedno-dostępny (!?!?!?!?!)
 - "przenośny" na różne komputery (z procesorem Intel-a, DEC Alpha)
 - może działać na komputerach z wieloma procesorami
 - można uruchamiać programy starszych s.o. firmy Microsoft (DOS, WIN16, OS/2)
 - w WinNT 4.0 GUI wzięte z Win95
 - możliwość kompilowania programów POSIX (programy używające tradycyjnych fun. sys. Unix-a !)
 - system plików NTFS, doskonalszy od FAT; odporny na uszkodzenia bo wszelkie operacje na strukturach danych s. plików są robione przy pomocy transakcji (mechanizm z b.d. zapewniający spójność danych nawet po awarii ...)
 - prawa dostępu do plików/katalogów w NTFS na zasadzie ACL (ang. Access Control List) – każdy plik/katalog ma listę użytkowników i ich prawa
 - wbudowana obsługa sieci (tzw sieć Microsoft); możliwość udostępniania zasobów w sieci (zasobów takich jak katalogi, drukarki)

System operacyjny WinNT

- architektura Win NT 4.0 ...



Rys. 23.1 Schemat blokowy systemu Windows NT

System operacyjny WinNT

- architektura Win 2000 ...

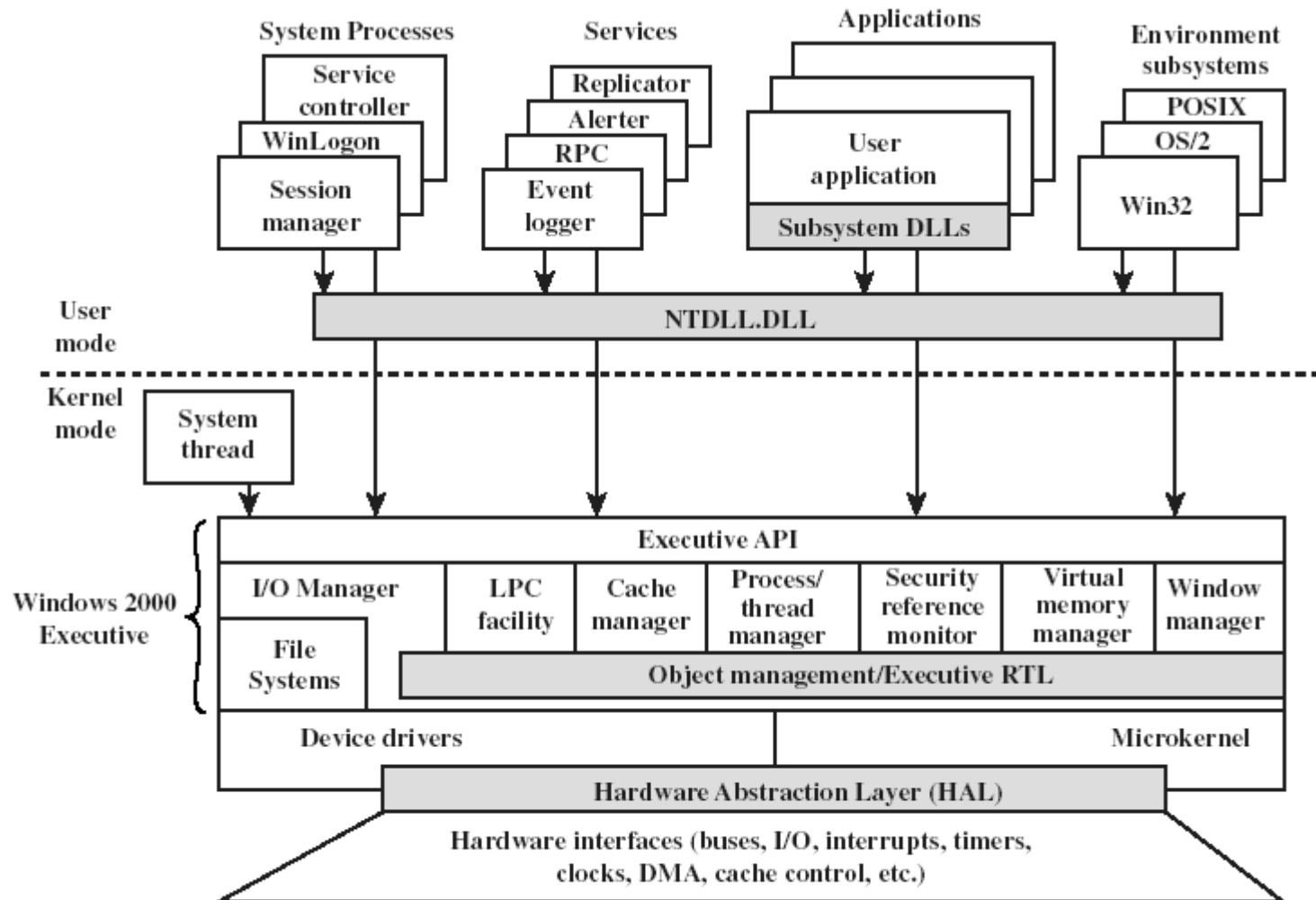


Figure 2.13 Windows 2000 Architecture

System operacyjny WinNT

- architektura
 - zasadnicza cecha: oparta na mikrojądrze
(ale "zmodyfikowana" – procesy usługowe (=procesy serwery) działają w trybie jądra a nie w trybie użytkownika jak w prawdziwej "arch z mikrojądrem")
 - mikrojądro zajmuje się:
 - przełączaniem procesora między wątkami i procesami (+planowanie)
 - obsługą wyjątków i przerwań sprzętowych
 - jedynie mikrojądro NIE działa jako proces/ wątek
 - dzięki arch z mikrojądrem łatwo wprowadzać nowe podsystemy/ procesy usługowe bez "rewolucji" w reszcie systemu ...
 - specyficzne nazewnictwo:
 - (ang. executive) egzekutor lub "centrum wykonawcze" (= jądro)
 - procesy usługowe (ang. executive services) – w nich jest realizowana większość usług s.o.
 - podsystemy środowiskowe (ang. environment subsystems) – w nich stwarza się środowisko różnych s.o. (DOS/WIN16(?), POSIX, Win32)
 - Win32 to główny ("wrodzony") podsystem środowiskowy
 - podsystemy środowiskowe i procesy usługowe są "normalnymi" procesami (???) różnica: podsystemy działają w trybie użytkownika, proc usługowe w trybie jądra – tj są bardziej uprzywilejowane

System operacyjny WinNT

- architektura c.d.
 - są ładowalne moduły obsługi urządzeń (ang. drivers); dzięki temu system może zostać uzupełniony o:
 - nowe urządzenia we/wy (np. nowa karta graficzna)
 - nowe systemy plików (np. NT4.0 nie obsługuje FAT32 ale można dodać driver)
 - nowe rodzaje sieci
 - NT napisano głównie w języku C/C++ (jednak kod źródłowy nie jest znany publicznie jak w przypadku Linux-a)
 - warstwa abstrakcji sprzętu HAL (ang. Hardware Abstraction Layer); tworzy "logiczną maszynę"; maszyny z procesorami Intel'a i DEC Alpha wyglądają tak samo dzięki temu
 - niektóre procesy usługowe pomijają HAL i odwołują się bezpośrednio do sprzętu (np. moduł graficzny; chodzi o szybkość operacji graficznych)
 - niezawodność:
 - ochrona sprzętowa pamięci (wirtualnej prz. adr. procesu)
 - NTFS – odporność na uszkodzenia dzięki mechanizmowi transakcji
 - LPC/RPC – podsystemy i procesy usługowe jądra komunikują się przy pomocy mechanizmu *zdalnego wywołania procedury* (mechanizm LPC dotyczy wywołań w na jednej maszynie, RPC między różnymi maszynami; RPC jest używane do realizacji przetwarzania rozproszonego ...)

System operacyjny WinNT

- architektura c.d.
 - **rodzaje aplikacji Win32**
 - GUI = Graphical User Interface - otwierają okna, używają grafiki
 - CUI = Console User Interface – wyglądają jak programy dos-owe (ale nimi nie są! są programami używającymi Win32)
 - **obiekty jądra**
 - każdy element s.o. (proces, wątek, semafor, otwarty plik, łącze komunikacyjne, skrzynka pocztowa) jest reprezentowany przez "obiekt jądra" ...
 - obiekty są dostępne poprzez "uchwyty"; poprzez uchwyt do obiektu można na tym obiekcie wykonywać różne operacje
 - programy użytkowników też mogą tworzyć/otwierać obiekty jądra przy pomocy fun. sys. Create*() o Open*()
 - każdy proces posiada "tablice obiektów" (z uchwytami do tych obiektów)
 - przykłady programów Win32:
 pliki np??.cpp + np??.exe
(omówić ...)

Inne źródło informacji o Win NT

- książka:
"Microsoft Windows 2000 od środka"
David A. Solomon, Mark E. Russinovich
- "procesy usługowe" nie są normalnymi procesami bo nie mają własnej przestrzeni adresowej pamięci (działają w jednej prz. adr. jądra);
dlatego NT ma architekturę monolityczną !!!
- podsystemy środowiskowe to: POSIX, OS/2, Win32; programy Win16 i DOS działają na "wirtualnej maszynie DOSowej", która nie jest zaliczana do podsystemów środowiskowych

Podsystem środow. Win32

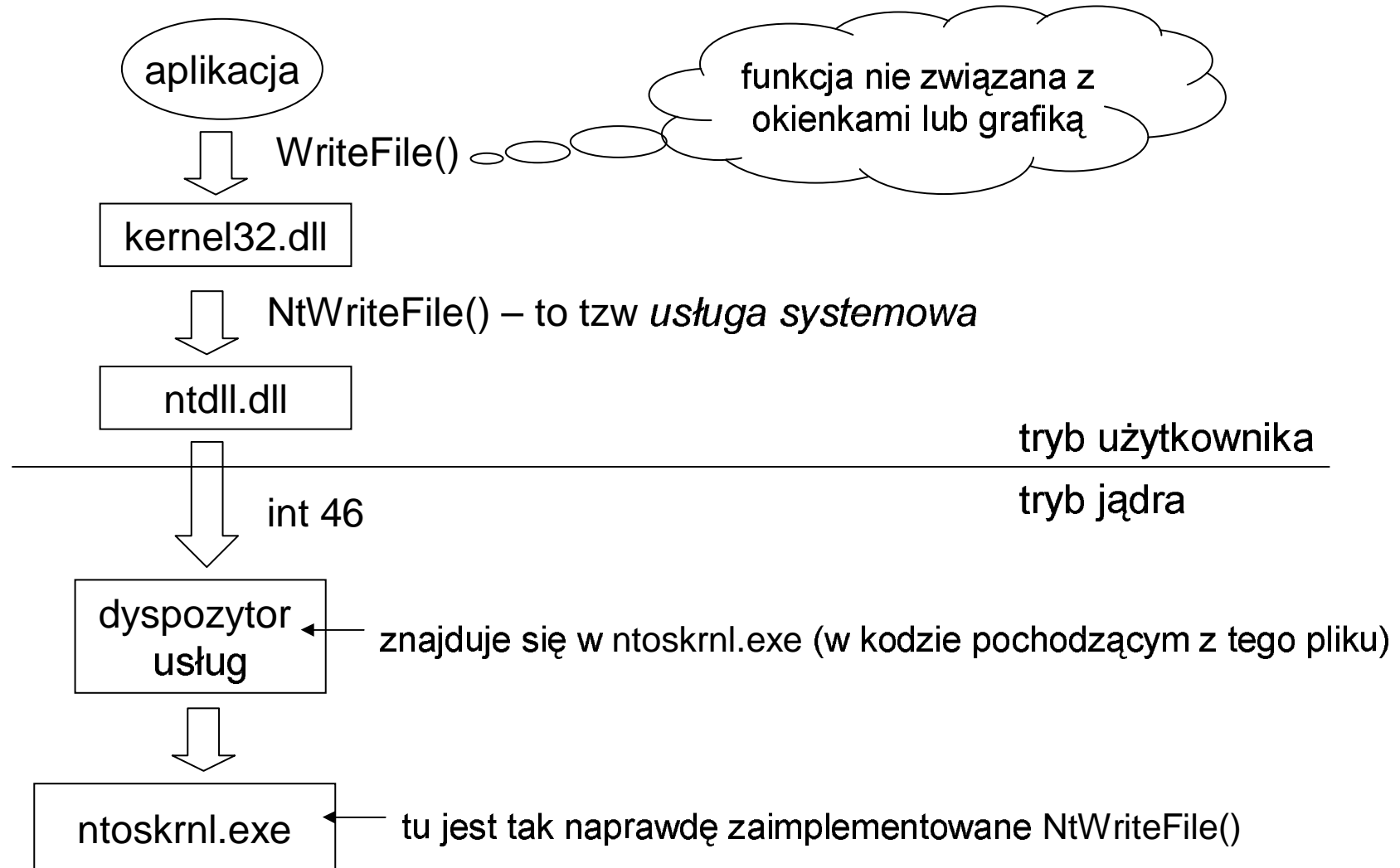
- Z czego składa się podsystem środowiskowy Win32 ???
 - proces trybu użytkownika "csrss.exe"
 - pracuje w trybie użytkownika
 - obsługuje konsolę tekstową
 - tworzy i niszczy wątki oraz procesy
 - i ma wiele innych funkcji
 - sterownik win32k.sys
 - pracuje w trybie jądra
 - stanowi część centrum wykonawczego
 - biblioteki *32.dll
 - np. kernel32.dll, user32.dll, gdi32.dll
 - sterowniki urządzeń graficznych

Inne podsystemy środow.

- Co się dzieje podczas uruchamiania programu?
 - Win32 (.exe)
program .exe jest uruchamiany bezpośrednio
 - DOS (.exe, .com)/ Win16 (exe, pif)
jest uruchamiany pomocniczy program Win32 o nazwie "ntvdm.exe",
który "tworzy" wirtualną maszynę DOSową;
jeden proces ntvdm obsługuje wszystkie programy dosowe
 - .bat, .cmd
cmd.exe
 - POSIX
posix.exe
 - OS/2
os2.exe

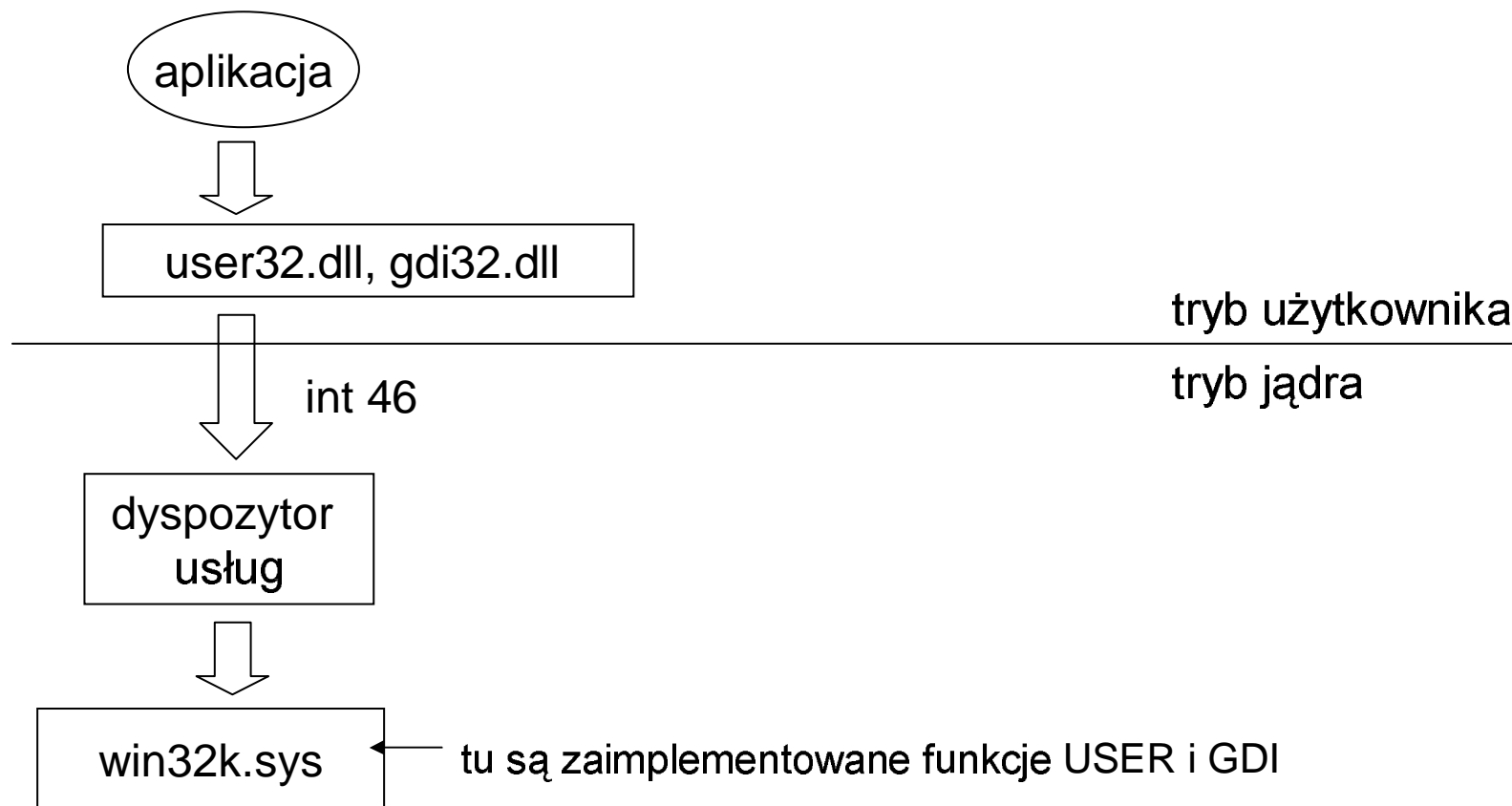
Wywołanie funkcji Win32

- Funkcje Win32, są dla aplikacji Win32 "fun. sys."
- Co się dzieje gdy aplikacja Win32 wywołuje fun. WriteFile()



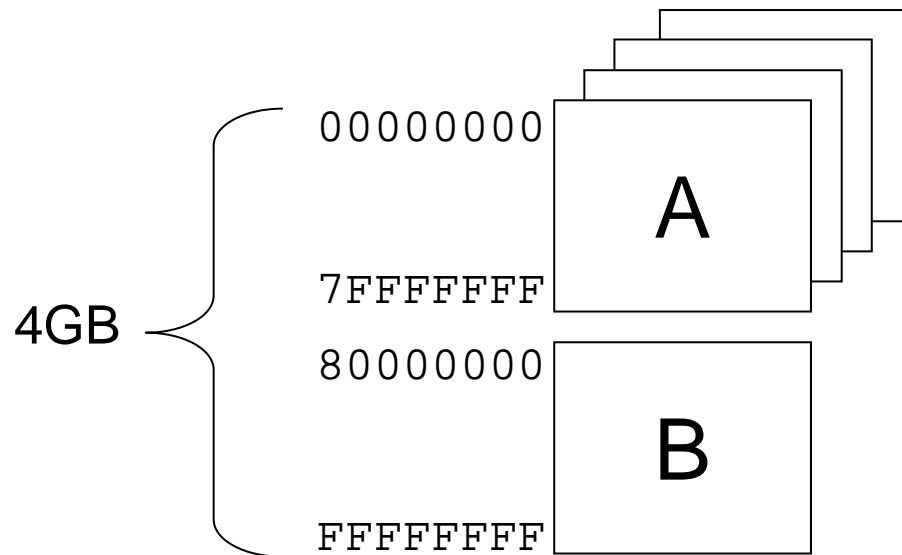
Wywołanie funkcji Win32

- Co się dzieje gdy aplikacja Win32 wywołuje funkcje modułów USER (=zarządzanie oknami) i GDI (=zarządzanie grafiką)



Obraz pamięci procesu w NT

- proces używa pojedynczego 4GB segmentu
- przestrzeń adr. 4GB jest podzielona na 2 części:
 - A: prywatna należąca do procesu (2GB lub 3GB)
 - B: systemową (2GB lub 1GB); obejmuje kod i dane jądra
- w trybie użytkownika jest dostępna tylko część prywatna, w trybie jądra są dostępne obie
- zarówno w trybie jądra jak i użytkownika używa się wyłącznie adresów logicznych!



Różne ciekawostki ...

Zdalna praca z Unix-em

- pokażemy jak pracować "zdalnie" z Unixem/Linuxem siedząc przy maszynie z MS Windows
 - z programami "tekstowymi" nie ma problemu: wystarczy się wtelnetować do maszyny z linux-em ...
 - aby używać programów linuxowych/Xwindowsowych trzeba zainstalować na lokalnej maszynie XSerwer !
np. program WinaXe v6.0
 - następnie telnetujemy się do linuxa; do zmiennej DISPLAY wpisujemy adres IP naszej lokalnej maszyny (odczytamy go poleceniem ipconfig)

```
export DISPLAY=lokalny_adr_IP:0.0
```
 - i możemy włączać programy linuxowe/Xwindowsowe, które będą otwierały swoje okna na naszej lokalnej maszynie (będziemy widzieć zarówno okna zwykłych lokalnie uruchomionych programów Windows ORAZ okna programów uruchomionych na linuxie ...)
 - następny slajd:
równoczesne używanie MS Word-a i programu OOWriter z Linuxa !!!

Zdalna praca z Unix-em

