

ARCHITEKTURY SYSTEMÓW OPERACYJNYCH

dr hab. inż. Krzysztof Patan, prof. PWSZ

Instytut Politechniczny
Państwowa Wyższa Szkoła Zawodowa w Głogowie
k.patan@issi.uz.zgora.pl

- Systemy operacyjne nie są jednym spójnym tworem, lecz są zbudowane z wielu pomniejszych składowych
- Każda ze składowych powinna być dobrze określonym fragmentem systemu ze starannie zdefiniowanym wejściem, wyjściem i działaniem
- Każdy system operacyjny jest inaczej zbudowany, ale większość posiada podobne składowe

Główne składowe systemów operacyjnych

- ◇ moduł zarządzania procesami
- ◇ moduł zarządzania pamięcią operacyjną
- ◇ moduł zarządzania plikami
- ◇ moduł zarządzania systemem wejścia–wyjścia
- ◇ moduł zarządzanie pamięcią pomocniczą
- ◇ praca sieciowa
- ◇ system ochrony
- ◇ system interpretacji poleceń

Zarządzanie procesami

- **Proces** – program, który jest wykonywany, np. zadanie systemowe, program wsadowy, program użytkownika, zadanie do drukowania
- Proces musi korzystać z pewnych zasobów systemu komputerowego, np. czasu CPU, pamięci, plików, urządzeń peryferyjnych
- Zasoby mogą być przydzielone procesowi w chwili jego utworzenia, lub podczas jego wykonywania
- Po zakończeniu działania procesu, system komputerowy odzyskuje wszelkie zasoby przez niego wykorzystywane i nadające się do powtórnego użytku

- ☛ Proces – jednostka pracy w systemie; system składa się ze zbioru procesów, część z nich to procesy systemu operacyjnego, pozostałe – procesy użytkownika
- ☛ Program – jest **elementem pasywnym** i sam w sobie nie jest procesem
- ☛ Proces – **element aktywny**, w którym licznik rozkazów wskazuje następną instrukcję do wykonania:
 - proces jest wykonywany sekwencyjnie
 - CPU wykonuje instrukcje jedna po drugiej
 - na żądanie procesu może być jednocześnie wykonana co najwyżej jedna instrukcja
 - z jednym programem może być powiązanych wiele procesów, zawsze rozważa się je jako oddzielne ciągi instrukcji

◇ W odniesieniu do zarządzania procesami system operacyjny odpowiada za:

- 1 tworzenie i usuwanie procesów użytkowych i systemowych
- 2 wstrzymywanie i wznowianie procesów
- 3 dostarczanie mechanizmów synchronizacji procesów
- 4 dostarczanie mechanizmów komunikacji procesów
- 5 dostarczanie mechanizmów obsługi zakleszczeń

Zarządzanie pamięcią operacyjną

- ☛ Pamięć operacyjna – kluczowy element każdego systemu komputerowego
- ☛ Pamięć to "magazyn" szybko dostępnych danych wykorzystywanych przez CPU i urządzenia IO
- ☛ Przykładowe wykonywanie cyklu rozkazów
 - 1 czytanie → pamięć operacyjna → wykonywanie
 - 2 pobieranie danych z dysku → pamięć operacyjna → czytanie danych przez proces → wykonywanie
- ☛ W celu wykonania programu, należy go zaadresować oraz załadować do pamięci
- ☛ Podczas wykonywania programu, rozkazy i dane są pobierane z pamięci za pomocą generowania potrzebnych adresów

- ◇ Lepsze wykorzystanie CPU oraz szybsza odpowiedź komputera – przechowywanie w pamięci kilku programów jednocześnie
- ◇ W odniesieniu do zarządzania pamięcią operacyjną system operacyjny odpowiada za:
 - ❶ utrzymywanie ewidencji aktualnie zajętych części pamięci wraz z informacją w czym są władaniu
 - ❷ decydowanie, które procesy mają być załadowane do zwolnionych obszarów pamięci
 - ❸ przydzielanie i zwalnianie obszarów pamięci stosownie do potrzeb

Zarządzanie plikami

- ☛ Najbardziej widoczna część systemu operacyjnego
- ☛ Dane można przechowywać na różnego rodzaju nośnikach: magnetycznych, optycznych, itp.
- ☛ Dla wygody użytkownika system operacyjny tworzy jednolity, logiczny obraz magazynowanej informacji:
 - definiuje on pliki niezależnie od fizycznych właściwości używanych urządzeń przechowywania informacji
 - system operacyjny odwzorowuje pliki na fizyczne nośniki informacji
 - umożliwia do nich dostęp za pomocą urządzeń pamięci

- ☞ Jeśli wielu użytkowników ma dostęp do tych samych plików to jest pożądane sprawowanie pieczy nad tym kto i w jaki sposób korzysta z tego dostępu
- ☞ W odniesieniu do zarządzania plikami system operacyjny odpowiada za:
 - 1 tworzenie i usuwanie plików
 - 2 tworzenie i usuwanie katalogów
 - 3 dostarczenie elementarnych operacji do manipulowania plikami i katalogami
 - 4 odwzorowanie plików na obszary pamięci pomocniczej
 - 5 składowanie plików na trwałych nośnikach pamięci

Zarządzanie urządzeniami wejścia–wyjścia

- Jednym z celów systemu operacyjnego jest ukrywanie przed użytkownikiem szczegółów dotyczących specyfiki urządzeń sprzętowych
- Przykładem poprawnie zaprojektowanego systemu jest system UNIX, gdzie osobliwości urządzeń I/O są ukryte przed większością samego systemu przez tzw. podsystem wejścia–wyjścia
- Podsystem wejścia–wyjścia składa się z:
 - części zarządzającej pamięcią (buforowanie, pamięć podręczna, spooling)
 - ogólnego interfejsu do modułów sterujących urządzeniami
 - modułów sterujących (programów obsługi) poszczególnych urządzeń sprzętowych
- Osobliwości poszczególnego urządzenia wejścia–wyjścia zna tylko odpowiadający mu moduł sterujący

Zarządzanie pamięcią pomocniczą

- ☛ Pamięć operacyjna jest za mała aby pomieścić wszystkie dane i programy
- ☛ Zawarte w niej dane giną po odcięciu zasilania
- ☛ System komputerowy powinien posiadać zatem pamięć pomocniczą będącą zapleczem dla pamięci operacyjnej
- ☛ Pamięcią dyskową jako podstawowym środkiem magazynowania zarówno danych jak i programów
- ☛ W odniesieniu do zarządzania pamięcią operacyjną, system operacyjny odpowiada za:
 - 1 zarządzanie obszarami wolnymi
 - 2 przydzielaniem pamięci
 - 3 planowaniem przydziału obszarów pamięci dyskowej

System ochrony

- W systemie z wieloma użytkownikami i współbieżnie wykonywanymi wieloma procesami poszczególne procesy należy chronić przed wzajemnym oddziaływaniem
- Należy opracować mechanizmy gwarantujące, że pliki, segmenty pamięci, procesor i inne zasoby będą użytkowane tylko przez te procesy, które zostały przez system operacyjny odpowiednio uprawnione
- **Ochrona** – mechanizm nadzorowania dostępu programów, procesów lub użytkowników do zasobów zdefiniowanych przez system komputerowy
- Mechanizmy ochrony zawierają sposoby określenia, co i w jakiej formie ma podlegać ochronie, jak również środki do wymuszania zaprowadzonych ustaleń

- ☛ Za pomocą działań ochronnych można polepszać niezawodność systemu poprzez poszukiwanie błędów ukrytych w interfejsach między składowymi podsystemami
- ☛ Wczesne wykrywanie błędów w interfejsach może zapobiec zanieczyszczeniu zdrowego podsystemu przez podsystem uszkodzony
- ☛ Zasoby niechronione nie mogą obronić się przed użyciem lub nadużyciem przez nieupoważnionego lub niekompetentnego użytkownika
- ☛ System ochrony dostarcza środków do rozróżniania między prawomocnym i nieprawomocnym użyciem

System interpretacji poleceń

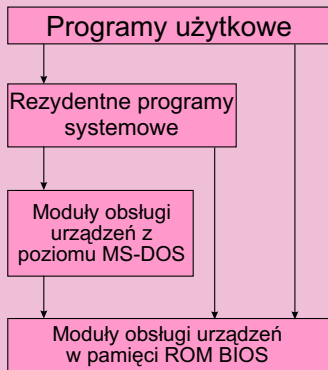
- ☛ Interpreter poleceń – jeden z najważniejszych programów w systemie operacyjnym
- ☛ **Interpreter poleceń** – interfejs pomiędzy użytkownikiem, a systemem operacyjnym
 - niektóre systemy zawierają interpreter poleceń w swoim jądrze
 - w innych systemach (MS-DOS, UNIX) interpreter poleceń jest specjalnym programem – powłoką (interpreterem wiersza poleceń)

- ☛ Systemy operacyjne często różnią się od siebie warstwą powłoki
 - przyjazny interpreter – systemy Apple Macintosh i Microsoft Windows – operowanie za pomocą systemu okien i menu oraz myszki
 - interpretery trudniejsze do opanowania lecz o większych możliwościach – systemy MS-DOS, UNIX – polecenia pisane na klawiaturze są wyświetlane na monitorze lub drukowane
- ☛ Polecenia rozpoznawane przez interpreter dotyczą:
 - 1 tworzenia procesów i zarządzanie nimi
 - 2 obsługi wejścia-wyjścia
 - 3 administrowania pamięcią pomocniczą i operacyjną
 - 4 ochrony i pracy sieciowej

Struktury systemów operacyjnych

- Wiele systemów nie ma ściśle określonej struktury
- Małe systemy często rozszerzano przekraczając pierwotne założenia

System MS-DOS



- ◇ wyraźnie rozdzielone poziomy funkcjonalne
- ◇ z poziomu programów użytkowych można korzystać z podstawowych procedur IO, np. pisanie na ekran czy dyski
- ◇ swoboda tego rodzaju powoduje, że system nie jest odporny na błędnie działające programy użytkowe
- ◇ możliwość zawieszenia się systemu lub uszkodzenia podzespołów

Systemy z jądrem

System UNIX

UŻYTKOWNICY		
Powłoki i polecenia Kompilatory, interpretery Biblioteki systemowe		
Interfejs funkcji systemowych jądra		
System plików, planowanie przydziału czasu CPU pamięć wirtualna, system wejścia-wyjścia moduły sterujące dysków i taśm moduły sterujące terminali		
Interfejs między jądrem a sprzętem		
sterowniki terminali Terminale	sterowniki urządzeń Dyski i taśmy	sterowniki pamięci RAM

- ◇ przykład strukturalizacji systemu
- ◇ system składa się z dwóch części: jądra i programów systemowych
- ◇ jądro dzieli się na ciąg interfejsów i programów obsługi urządzeń
- ◇ za pośrednictwem funkcji systemowych jądro udostępnia system plików, zarządzanie pamięcią operacyjną, planowanie przydziału CPU, itd.
- ◇ programy systemowe korzystają z udostępnianych przez jądro funkcji systemowych w celu wykonywania użytecznych działań
- ◇ wadą systemu UNIX w wersji podstawowej jest to, że posiada tylko dwie "warstwy", które zawierają bardzo dużo elementów
- ◇ można podzielić system operacyjny na mniejsze, lepiej dobrane elementy – większa kontrola nad komputerem i programami użytkowymi

Systemy o strukturze warstwowej

- W podejściu warstwowym dzieli się system na tzw. warstwy (poziomy)
- Poziom najniższy (warstwa 0) – sprzęt
- Poziom najwyższy (warstwa N) – interfejs z użytkownikiem
- Poziom M -ty – struktury danych i procedury wywoływane z wyższych warstw
- Warstwa M -ta może wywoływać operacje dotyczące niższych warstw

ZALETY – modularność

- ◇ każda z warstw korzysta z usług tylko niżej położonych warstw
- ◇ łatwe wyszukiwanie błędów i weryfikacja systemu
- ◇ pierwsza warstwa może być poprawiana bez troski o resztę systemu
- ◇ po uruchomieniu warstwy można przystąpić do realizacji

System THE

- ◇ pierwszy system warstwowy – THE (Technische Hogeschool Eindhoven)

5	PROGRAMY UŻYTKOWE
4	BUFOROWANIE URZĄDZEŃ WEJŚCIA–WYJŚCIA
3	PROGRAM OBSŁUGI KONSOLI OPERATORA
2	ZARZĄDZANIE PAMIĘCIĄ
1	PLANOWANIE PRZYDZIAŁU PROCESORA
0	SPRZĘT

- ◇ schemat zarządzania pamięcią – pamięć wirtualna
- ◇ program obsługi konsoli operatora i buforowanie urządzeń wejścia–wyjścia były zlokalizowane wyżej od zarządzania pamięcią, bufony urządzeń mogły być umieszczone w pamięci wirtualnej
- ◇ buforowanie urządzeń wejścia–wyjścia znajdowało się powyżej konsoli operatora, dlatego informacje o błędach

System VENUS

6	PROGRAMY UŻYTKOWE
5	PROGRAMY OBSŁUGI I PLANOWANIA PRZYDZIAŁU URZĄDZEŃ
4	PAMIĘĆ WIRTUALNA
3	KANAŁ WEJŚCIA-WYJŚCIA
2	PLANOWANIE PRZYDZIAŁU PROCESORA
1	INTERPRETER POLECEŃ
0	SPRZĘT

- ◇ poziomy niższe 0–4 – planowanie przydziału czasu procesora i zarządzanie pamięcią
- ◇ warstwy 0–4 zostały napisane jako mikroprogramy – szybsze działanie i przejrzysty interfejs między warstwami mikroprogramowanymi, a warstwami wyższymi

Właściwości systemów warstwowych

- ✗ główna wada – trudności w odpowiednim zdefiniowaniu zawartości poszczególnych warstw
 - ◇ program obsługi pamięci pomocniczej powinien być poniżej procedury zarządzania pamięcią
 - ◇ wiele wymagań nie jest jawnie określonych
- ✗ realizacje warstwowe – mniej wydajne od innych
- ✓ główna zaleta – łatwość w lokalizacji błędów i weryfikacji systemu

Podsumowanie

- problemy z efektywnością spowodowały nieznaczne odchodzenie od modeli warstwowych
- dąży się do systemów złożonych z mniejszej liczby, ale bardziej funkcjonalnych warstw

Systemy o strukturze modularnej

System OS/2

Aplikacja	Aplikacja	Aplikacja
Interfejs programowania aplikacji		Rozszerzenie API
Podsystem	Podsystem	Podsystem
JĄDRO SYSTEMU <ul style="list-style-type: none">• zarządzanie pamięcią• ekspediowanie zadań• zarządzanie urządzeniami		
Moduł sterujący	Moduł sterujący	Moduł sterujący

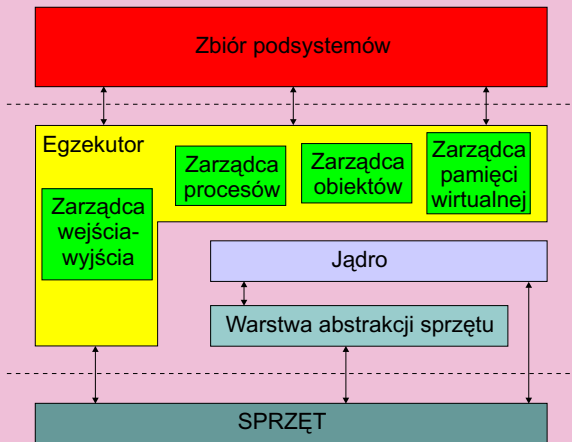
Właściwości

- system OS/2 – następca MS-DOS
- system wielozadaniowy, podwójny tryb operacji
- zaprojektowany z uwzględnieniem warstwowości
- użytkownik nie może korzystać bezpośrednio z udogodnień niskiego poziomu
- zwiększona kontrola nad sprzętem i lepsze rozeznanie co do zasobów wykorzystywanych przez programy użytkowników

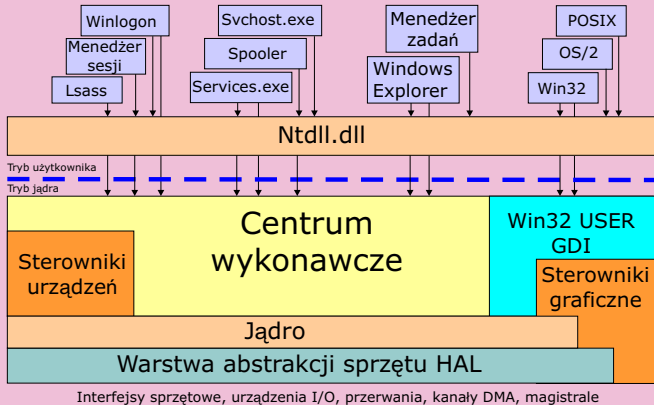
System Windows NT

- pierwsza wersja systemu – struktura warstwowa
- niska wydajność w stosunku do systemu Windows 95
- w wersji Windows NT 4.0 poprawiono niedogodności przesuając warstwy z przestrzeni użytkownika do przestrzeni jądra ściśle je integrując
- architektura systemu NT wersji 4 – warstwowy układ modułów
 - ◇ warstwa abstrakcji sprzętu
 - ◇ jądro
 - ◇ egzekutor
 - ◇ zbiór podsystemów działających w trybie użytkownika

Struktura systemu Windows NT



System Windows 2000



Założenia projektowe

- **Specyfikacja systemu** – wybór sprzętu i typ systemu
- **Wymagania użytkowników** – system operacyjny powinien być wygodny w użyciu, łatwy do nauki, niezawodny, bezpieczny i szybki
- **Cele projektantów** – system operacyjny powinien być łatwy do zaprojektowania, implementacji i konserwacji, elastyczny, niezawodny, pozbawiony błędów oraz efektywny
- problem zdefiniowania wymagań względem systemu operacyjnego nie ma jednoznacznego rozwiązania
- **Ogólne zasady projektowania** – *Inżynieria oprogramowania*

Języki programowania

- obecnie do projektowania systemów operacyjnych wykorzystuje się języki wysokiego poziomu
- polepsza to implementację, pielęgnację oraz przenaszalność systemu
- wady stosowania języków wysokiego poziomu
 - ① spowolnienie działania systemu
 - ② większe zapotrzebowanie na pamięć
- pomimo wielkich rozmiarów systemów operacyjnych na ich efektywność ma wpływ niewielka ilość kodu: zarządca pamięci i planista przydziału CPU
- procedury będące wąskimi gardłami można zastąpić odpowiednikami napisanymi w asemblerze

Funkcja systemowe

Funkcje systemowe (wywołania systemowe – *ang.* system calls) stanowią interfejs pomiędzy wykonywanym programem, a systemem operacyjnym

Podział funkcji systemowych:

- nadzorowanie procesów, np. `end`, `abort`, `wait for time`
- operacje na plikach, np. `open`, `close`, `read`, `write`, `get file attributes`
- operacje na urządzeniach, np. `request device`, `release device`, `read`, `write`, `reposition`
- otrzymywanie informacji, np. `get time`, `get date`, `get device attributes`
- komunikacja, np. `create connection`, `delete connection`, `send message`