

Temat: Budowa i zasady działania systemu operacyjnego.

1. Budowa systemu operacyjnego
2. Warstwy systemu operacyjnego
3. Główne zadania systemu operacyjnego
4. Najważniejsze cechy systemu operacyjnego

Budowa systemu operacyjnego

System operacyjny jest to oprogramowanie, które pełni w komputerze bardzo istotną rolę. Pod względem informatycznym jego zadaniem jest zarządzanie zasobami komputera, czyli sprzętem, w jaki wyposażony jest pecet oraz aplikacjami uruchamianymi przez użytkownika. To właśnie system operacyjny kontroluje i przypisuje pamięć operacyjną (RAM) dla programów, decyduje o przydziale im procesora, steruje urządzeniami wejścia/wyjścia. Zarządza także plikami, a także powszechnie już ustanawia połączenia sieciowe. Z punktu widzenia użytkownika komputera system operacyjny pomaga mu komunikować się ze sprzętem oraz staje się środowiskiem, w którym uruchamia on potrzebne aplikacje. System operacyjny staje się więc pośrednikiem między użytkownikiem, a sprzętem komputerowym, czyli można powiedzieć, że udostępnia aplikacjom maszynę wirtualną. Od dość dawna ważną cechą systemów operacyjnych jest tzw. graficzny interfejs użytkownika, który poprzez wykorzystanie grafiki ułatwia użytkownikowi korzystanie z komputera.

Przyjmując się podział na trzy główne elementy budowy systemu operacyjnego:

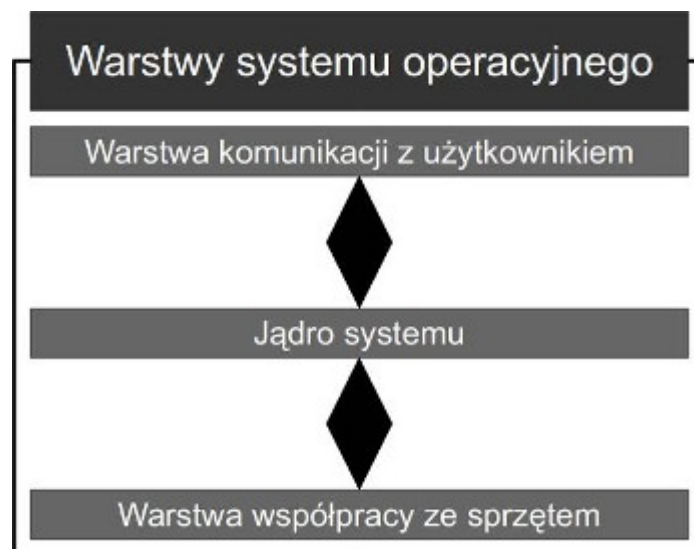
- jądro - jest to warstwa odpowiedzialna za wykonywanie podstawowych zadań systemu operacyjnego;
- powłoka - jest to specjalny program służący do komunikacji użytkownika z systemem operacyjnym;
- system plików - jest to warstwa odpowiedzialna za sposób organizacji danych na nośniku.

Warstwy systemu operacyjnego

W każdym systemie operacyjnym występują mniej lub bardziej wyodrębnione warstwy składające się na architekturę systemu. W ogólnym modelu systemu operacyjnego można wyszczególnić następujące warstwy przypisując im wyszczególnione zadania:

- powłokę, stanowiącą interfejs użytkownika (komunikacja z użytkownikiem) ,
- jądro systemu realizujące jego funkcje (zarządzanie plikami, uruchamianie aplikacji),

- warstwę odpowiedzialną za współpracę ze sprzętem (zarządzanie zasobami maszyny, komunikacja z innymi maszynami).



Główne zadania systemu operacyjnego

Główne zadania stawiane przed systemami operacyjnymi to:

- zarządzanie zasobami komputera - systemy operacyjne starują oraz optymalizują wykorzystanie określonych urządzeń, które wchodzi w skład zestawu komputerowego; specjalne moduły zwane sterownikami, które składają się na system operacyjny udostępniają aplikacjom spójne metody programowania urządzeń (interfejsy), co gwarantuje współdziałanie każdego nowego urządzenia z oprogramowaniem (jeżeli producent dostarczy właściwy sterownik)
- gromadzenie oraz zarządzanie danymi - systemy operacyjne wyposażone są w moduły obsługujące system plików, czyli struktury umieszczone na dyskach, pomagające w logiczny sposób uporządkować dane, grupując je w katalogi i pliki
- maszyny wirtualne - systemy operacyjne udostępniają aplikacjom tzw. maszyny wirtualne, czyli uproszczone obrazy maszyn, na których pracują aplikacje; system udostępnia aplikacjom szczegóły dna temat komputera oraz rozszerzenia ułatwiające pracę (np. zasoby udostępniane poprzez sieć aplikacje widzą tak, jakby znajdowały się one na dysku lokalnym; aplikacja, która korzysta z takiego zasobu nie obsługuje pracy sieciowej, więc aby umożliwić jej dostęp do zasobu, system operacyjny symuluje, jest zasób ten jest lokalny udostępniając go dla aplikacji)
- wielozadaniowość - na pojedynczym komputerze funkcjonować może wiele aplikacji w jednym czasie; każda aplikacja otrzymuje swoją maszynę wirtualną i może pracować jakby była pojedynczym programem działającym na maszynie; dzięki takiej

właściwości systemów operacyjnych nie ma potrzeby przystosowywania aplikacji, by mogła dzielić się daną maszyną z innymi aplikacjami

- interakcja z użytkownikiem - rolę tę spełnia najbardziej zewnętrzna warstwa systemu operacyjnego zwana powłoką (*ang. shell*), która pozwala użytkownikowi uruchamiać aplikacje; w systemach graficznych do powłoki zaliczają się także typowe elementy interfejsu, z których korzysta aplikacja, jak kontrolki czy okna dialogowe
- komunikowanie się z innymi komputerami - jest to jeden z najistotniejszych elementów systemów operacyjnych; dzięki modułom obsługi sieci można uzyskać dostęp do Internetu, do dysków komputerów stojących w sąsiednim pomieszczeniu czy sieciowych urządzeń peryferyjnych jak drukarki czy skanery

Najważniejsze cechy systemu operacyjnego

Najważniejszymi cechami, które decydują o użyteczności danego systemu operacyjnego są:

- prostota instalacji oraz użytkowania systemu
- współpraca z innymi systemami, czyli możliwość odczytu i zapisu danych na partycjach z innych systemów a także współpraca oraz wymiana danych między komputerami w sieciach lokalnych i Internecie
- zgodność sprzętowa (instalację na konkretnej maszynie utrudnia czasami brak właściwych sterowników dla określonych urządzeń)
- wymiana danych, czyli możliwość przeglądania oraz wymiany dokumentów pomiędzy różnymi aplikacjami pracującymi pod kontrolą różnych systemów
- możliwość pracy sieciowej (wygoda podczas przeglądania zasobów sieciowych, wymiany protokołów, itp.)
- cena
- liczba aplikacji, które działają w danym systemie (nawet najlepiej pracujący system będzie niemal bezużyteczny, jeżeli oferta oprogramowania, które współpracuje z tym systemem będzie niewielka)
- lokalizacja (możliwość komunikacji użytkownika z systemem w ojczystym języku)

Zasada działania systemu operacyjnego

System operacyjny jest programem, jednak jego działanie jest dość specyficzne, gdyż musi on nadzorować (monitorować) pracę komputera nawet wówczas, gdy wykonywany jest jakiś program aplikacyjny. System operacyjny musi reagować na błędy w programach

aplikacyjnych, porządkować system komputerowy po awariach, z kolei błędy w kodzie jądra systemu operacyjnego mogą zdestabilizować funkcjonowanie całego systemu komputerowego.

Działanie współczesnych systemów operacyjnych jest rezultatem ewolucji w architekturze sprzętowo-programowej, w której potrzeby w zakresie implementacji pewnych mechanizmów systemu operacyjnego wymuszały wprowadzanie stosownych rozwiązań na poziomie architektury komputera (procesora, jednostki zarządzania pamięcią, układu bezpośredniego dostępu do pamięci, procesorów wejścia-wyjścia itp.). Rozwiązania na poziomie architektury komputera otwierały z kolei drogę do dalszego rozwoju oprogramowania systemowego.

Cykl rozkazowy

Działania procesora, zmierzające do wykonania rozkazu, powtarzają się cyklicznie, w związku z czym określa się je jako cykl rozkazowy. Realizacja cyklu rozkazowego wymaga na ogół kilku interakcji procesora z pamięcią. Każdą taką interakcję określa się mianem cyklu maszynowego. W każdym cyklu rozkazowym występuje cykl maszynowy pobrania kodu rozkazu (fetch). W zależności od trybu dostępności operandów mogą też wystąpić cykle pobrania operandu z pamięci (albo rejestrów wejścia-wyjścia) lub składowania operandu w pamięci (albo rejestrach wejścia-wyjścia). (Operandy są argumentami operacji wykonywanej w ramach rozkazu.) Każdy cykl maszynowy oznacza zatem zapis lub odczyt pamięci, przy czym cykl pobrania kodu rozkazu oznacza zawsze odczyt.

Podstawy działania systemu operacyjnego

Ogólnie, sterowanie przekazywane jest do jądra systemu operacyjnego poprzez przerwania. Program jądra jest więc zbiorem procedur obsługi przerwania i wywoływanych przez nie innych podprogramów. Przerwania, wspomniane na poprzednim slajdzie, pochodzą z układów na zewnątrz procesora, czyli od urządzeń wejścia-wyjścia, czasomierzy, układu bezpośredniego dostępu do pamięci itp. Inny rodzaj to przerwania zgłaszane wewnętrznie przez procesor, będące następstwem wykrycia jakiegoś stanu wyjątkowego. Jeszcze inny rodzaj to przerwania programowe, wynikające z wykonania specjalnej instrukcji procesora, umożliwiające programom użytkownika dostęp do wybranych funkcji jądra systemu operacyjnego.

Stabilność pracy systemu wymaga ochrony przynajmniej jądra systemu operacyjnego przed niekontrolowaną ingerencją użytkowników. Wymaga to monitorowania odniesień do pamięci i weryfikowania poprawności adresów. Ze względów wydajnościowych zadanie to

realizowane jest sprzętowo, ale odpowiednie dane na potrzeby weryfikacji musi dostarczyć system. W celu zabezpieczenia tych (i innych) newralgicznych danych wyróżnione są pewne instrukcje uprzywilejowane, niedostępne dla programów aplikacyjnych. Powstaje jednak problem odróżnienia programów systemowych od aplikacyjnych, którego rozwiązaniem jest wyodrębnienie dwóch (w niektórych procesorach nawet większej liczby) poziomów pracy (trybów pracy). Możliwe staje się narzucenie sprzętowych restrykcji odnośnie wykonywania niektórych instrukcji na odpowiednich poziomach. Proces użytkownika uruchamiany jest w trybie nieuprzywilejowanym, w związku z czym nie może wykonać pewnych instrukcji, dostępnych tylko w trybie uprzywilejowanym, tym samym ma ograniczoną możliwość swobodnego ingerowania w „obszary” zastrzeżone dla jądra systemu operacyjnego.

Przerwania w systemie komputerowym

System przerwania umożliwia niesekwencyjne (współbieżne) wykonywanie programów. Zmiana sekwencji wykonywania instrukcji polega na tym, że w reakcji na przerwanie następuje zapamiętanie bieżącego stanu przetwarzania (najważniejszych rejestrów procesora), przekazanie sterowania do ustalonej procedury obsługi i rozpoczęcie wykonywania instrukcji tej procedury. W szczególności może to prowadzić do przełączenia kontekstu, czyli przekazania sterowania po zakończeniu procedury obsługi przerwania do innego przetwarzania, niż to które zostało przerwane.

Zasady ochrony pamięci

Kluczowa dla stabilnej pracy i bezpieczeństwa systemu komputerowego jest ochrona pamięci jądra systemu operacyjnego przed ingerencją programów wykonywanych w trybie użytkownika – przede wszystkim przed modyfikacją kodu lub danych, ale ze względów bezpieczeństwa również przed odczytem (np. hasła). W systemach wielozadaniowych ważna jest również ochrona pamięci jednego zadania (procesu) przed ingerencją innego zadania.

W najprostszym przypadku ochrona pamięci polega na ograniczeniu zakresu dostępnych adresów do pewnego podzbioru, opisanego przez podanie najniższego i najwyższego dopuszczalnego adresu. W przedstawionym schemacie najniższy dopuszczalny adres przechowywany jest w rejestrze bazowym. Rejestr graniczny z kolei określa wielkość dostępnego obszaru pamięci. Dopuszczalne adresy należą do zatem do przedziału W środowisku wielozadaniowym z każdym przetwarzaniem mogą być związane inne ograniczenia na dostępność obszarów pamięci. Przełączanie między zadaniami wymaga zatem zmiany zawartości rejestrów, ograniczających zakres dostępności pamięci. W

rozważaniach pominięte zostały też kwestie ograniczonego dostępu do określonych obszarów (np. tylko do odczytu). Dostępność obszaru w takim przypadku zależy od rodzaju cyklu maszynowego. Tego typu zagadnienia będą rozważane przy omawianiu zarządzania pamięcią.