

Operačný systém

História

Prvé počítače vznikli počas druhej svetovej vojny pre vojenské potreby. Sprvoti počítače nemali operačný systém. Programátor sám vložil program do počítača (napríklad u počítača ENIAC páčkami krok po kroku naprogramoval počítač). Údaje vkladal podobným spôsobom. Neskôr boli používané dierne štítky, ktoré si programátor vyhotovil v dierkovačke, čím sa skrátilo blokovanie počítača pre iných používateľov počas vkladania. Potom prišli magnetické pásky, na ktoré sa program nahrával. Túto nudnú činnosť vykonával **operátor**. Nevýhoda spočívala v odstránení interaktivity. Programátor zadal výpočtovému pracovisku úlohu (job) a po dlhom čase dostal výsledky v podobe diernych štítkov či výstupu z tlačiarne alebo v prípade neúspešného behu programu výpis pamäte (core dump) pre ladenie. Pre urýchlenie celého procesu sa programy často zgrupovali – najprv sa vykonalo niekoľko prekladov programov vo Fortrane, potom v Cobole apod. Ušetril sa tak čas potrebný na neustálu výmenu pásov (napríklad pri nahrávaní prekladačov jazykov).

Vznik prvého operačného systému umožnil, že pokým sa spracovával program jedného z používateľov, ostatní mohli zadávať vstupy pre svoje programy. Princíp sa nazýva **spooling** – (Simultaneous Peripheral Operation On-Line) – pokým počítač pracuje na nejakej úlohe, výsledky predošlej úlohy sa tlačia na papier a ešte sa aj nahráva nový program na spustenie. Spooling bol umožnený až príchodom diskov s tzv. náhodným prístupom. Tak sa funkcia operátora, ktorý vymieňal pásky a obsluhoval tlačiarne a čítačky diernych štítkov, zmenila na akéhosi opravára počítačov večne pobiehajúceho so spájkovačkou ☺.

Neskôr prišlo na scénu **multiprogramovanie**. Myšlienkou bolo držať v pamäti niekoľko rozpracovaných úloh a činnosť procesora medzi nimi prepínať. Počítač pracoval na úlohách tak, že chvíľku (niekoľko milisekúnd) pracoval na jednej úlohe, chvíľku na ďalšej atď. pričom prácu na úlohách (programoch) neustále striedal, čo bolo pre používateľa skryté a navodzovalo dojem, že rôzne rýchlo (podľa počtu aktuálne spracovávaných úloh) počítač pracuje výlučne pre neho. Nazývame to zdieľanie počítačového (resp. škaredšie strojového) času¹. Podobným spôsobom dodnes pracujú všetky „multitaskové“ (vykonávanému programu hovoríme proces; niekedy ho nazývame aj úlohou – anglicky task) operačné systémy – **OS/2, Be-OS, Windows** (Windows 95/98/ME a Windows novej technológie – tzv. NT – t.j. Windows 2000/XP/2003) a klony **Unixu** (AIX, ATS, A/UX, BSD, Dynix, HP/UX, Irix, Linux (GNU), Minix, OSF/1, Solaris, SunOS, Ultrix, Unicos, Unix, Xenix). Multitaskové operačné systémy opätovne pripustili používateľa k počítaču. Využívajú obrovský rozdiel medzi rýchlosťou používateľa a rýchlosťou počítača.

Multitasking sa delí na preemptívny (časovačom vynútený) a na nepreemptívny (pseudomultitasking), ktorý využíva starší systém Windows (okrem Windows NT). Preemptívny multitasking vnucuje procesom násilné prepínanie. Počítač vďaka nemu nemožno zahltiť jedinou aplikáciou, ktorá by skonsumovala väčšinu strojového času a nepripustila ostatné procesy k činnosti. Pri programovaní vlákien procesov sa môžete stretnúť s možnosťou povolenia prepnutia úlohy na inú úlohu aj skôr, než vyprší jej pridelený časový limit. Neskôr sa však prepnúť nedá. Pri nepreemptívnom multitaskingu sa procesor medzi úlohami neprepína na základe údajov časovača, ale v okamihu, keď to práve vykonávaná úloha dovolí. Jeho nevýhodou je možné zrušenie sa počítača v prípade zacyklenia (sa) niektorej z úloh. Výhodou je niekedy rýchlejší prechod úlohy počítačom – v priemere trvá počítaču zvládnutie jednej úlohy menej času.

¹ Podľa predstavy, že počítač je stroj, ktorý počíta. ☺

Operačný systém

Operačný systém (OS) je program fungujúci ako prostredník medzi používateľmi a hardvérom (technickým vybavením) počítačového systému. Vytvára prostredie pre spúšťanie programov používateľov. Hlavným cieľom je zabezpečenie pohodlného prístupu používateľov k programom a efektívne využívanie hardvéru. Podľa širšej definície je operačný systém všetko, čo dostanete v krabici s OS – kompilátory, editory, GUI nadstavby... Podľa odbornej definície je OS iba program, ktorý neustále beží na počítači – tzv. **výkonné jadro** (kernel) **operačného systému**.

Počítačový systém sa dá rozdeliť na štyri časti:

1. **hardvér**: CPU, pamäť (primárna a sekundárna) a V/V (vstupno-výstupné) zariadenia systému;
2. **operačný systém**: koordinuje celý počítačový systém, prideluje zdroje, kontroluje ich použitie;
3. **aplikačné programy**: prekladače, databázy, hry (špeciálne sú tzv. nástroje na správu systému – programy na kontrolu disku, formátovanie diskiet, zavádzanie používateľov apod.);
4. **používatelia**. (Je trochu nesympatické považovať používateľov systému za jeho súčasť).

OS pre personálne počítače

Špeciálnym typom operačných systémov sú OS pre osobné počítače (personal computer – PC). Objavili sa v 70-tych rokoch a kladú väčší dôraz na pohodlie než na využitie zdrojov počítača či na bezpečnosť. Pre PC ide o jednoúlohový MS DOS a viacúlohové (ale stále jednopoužívateľské) Windows a OS/2. Existuje aj viacpoužívateľská verzia Windows NT.

Zložité počítačové systémy

Paralelné systémy

Multiprocessorový počítačový systém obsahuje niekoľko ($p > 1$) procesorov zdieľajúcich spoločnú pamäť, zbernicu, hodiny a periférie. Paralelné systémy delíme na **symetrické** a **asymetrické**.

Symetrický multiprocessing (SMP) používa rovnocenné univerzálne procesory, na každom beží časť operačného systému. Príkladom takéhoto operačného systému je IRIX pre počítače firmy Silicon Graphics. Procesy sa procesorom pridelujú ako celky. Niekedy sa procesy delia na relatívne samostatné časti – tzv. vlákna (thread). Potom aj vlákna môžu byť pridelované jednotlivým procesorom. Delenie procesov na vlákna (vytvára sa už pri preklade programu do strojového kódu, čo je postupnosť strojových inštrukcií, ktoré dokáže procesor vykonať) vedie na viacprocesorových počítačoch k urýchleniu výpočtu. Ako príklad by sme možno mali uviesť aj Windows NT, to však multiprocessorové servery neobsluhuje tak dobre ako napríklad multiprocessorový Linux, ktorý procesory dokáže zrovnoprávniť. U počítačov Silicon Graphic býva zvykom spájať v rámci jedného počítača do 16 procesorov, u firmy Cray (už neexistuje, jej technológiu skúpila Silicon Graphics) niekoľko desiatok až stovák, IBM vraj pokusne spojila ohromné množstvo (až 50000) procesorov Pentium.

Asymetrický multiprocessing uplatňuje master-slave vzťah medzi procesormi, ktoré ani nemusia byť rovnakého typu. Typickým prípadom je „vektorový“ počítač Convex so 128 procesormi, ktorého výkon vedel využiť iba špecializovaný softvér. (Tento počítač bol pre bežné použitie relatívne málo výkonný.) Asymetrický multiprocessing však funguje i u počítačov triedy PC pre špecializované (pomocné) procesory – napríklad FPU (pracuje s

číslami s pohyblivou rádovou čiarkou), procesory grafickej karty, zvukovej karty apod. „Šéfom“ je tu CPU (napr. Pentium IV), preto je jeho názov **centrálny (univerzálny) (mikro)procesor**.

Distribuované systémy

Klasické distribuované systémy

V distribuovanom počítačovom systéme sú procesory menej tesne zviazané – ide de facto o samostatné počítače (uzly) spojené pomocou komunikačnej siete. Dôvody pre budovanie distribuovaných systémov sú:

- *zdieľanie zdrojov* (tlačiarňí, diskov, výpočtových zdrojov),
- *urýchlenie výpočtov* rozdelením úloh medzi uzlami,
- *zvýšenie spoľahlivosti* v prípade, že niekoľko počítačových systémov vykonáva tú istú prácu a navzájom si porovnáva výsledky (v prípade potreby jeden počítačový systém kompletne zastúpi iný) ale aj tým, že napríklad časť súborov sa nachádza na pamäťových jednotkách jedného počítačového systému, zvyšok na pamäťových jednotkách ďalších počítačových systémov a v prípade výpadku jedného z počítačových systémov sa z pamäťových médií distribuovaného výpočtového informačného systému nestratí všetko.

Počítačové clustre umožňujú efektívny spôsob spájania výpočtového výkonu viacerých počítačov, nielen ich procesorov. Počítačový cluster tvoria blízko seba umiestnené samostatné počítačové (po overenej inštalácii bez monitorov či klávesníc, tie má len jeden z nich). Sú vzájomne prepojené vysokorýchlostnou počítačovou sieťou. Úlohy, ktoré treba riešiť, sú „rozhadzované“ na jednotlivé počítače tak, aby dohromady podali čo možno najväčší výkon. Počítačové clustre za nižšiu cenu podávajú výkon superpočítača. V klasifikácii sa nachádzajú medzi paralelnými systémami a klasickými distribuovanými systémami.

Agentové systémy

Agentové systémy sú viazané voľnejšie než klasické distribuované systémy. Agenti sa poväčšine skladajú z jadra (programu) a obálky umožňujúcej komunikáciu medzi agentmi. Agent je samostatný systém (človek, program, zviera) schopný spolupracovať s inými systémami. Dobre navrhnutý agentový systém je prispôsobivý čo do schopnosti spracovávanía rôznorodých úloh. Na rozdiel od presne organizovaného distribuovaného systému sú agentové systémy schopné v prípade potreby priberať ďalších agentov k spolupráci.

Agentové systémy využíva napríklad organizácia pre výskum mimozemských civilizácií (SETI) pre distribuované výpočty na miliónoch počítačov Internetu. Bližšie viď <http://www.seti.org>.

Architektúra operačného systému

Veľký a zložitý systém, akým je OS, sa dá vytvoriť a spravovať, ak pozostáva z menších častí s dobre definovaným rozhraním a chovaním. Rozdelenie komponent je vo všeobecnosti takéto:

Správa procesov

Proces je programový kód zavedený do pamäte na vykonanie. Procesy môžu spúšťať iné procesy – svojich potomkov, ktoré sa môžu vykonávať súbežne s rodičom alebo rodič môže čakať na ich výsledky (volaním služby **wait** výkonného jadra OS). Okrem procesov spustených používateľom beží v modernom OS aj niekoľko systémových procesov. V Unixe ich nazývame démonmi, pretože existujú mimo vedomia používateľov a mimo dosahu ich terminálov. Existuje démon pre tlač súborov (`lpd`), démon pre swapovanie (démon pre

odkladanie stránok na disk a pre ich spätné nahrávanie), pre prihlasovanie vzdialených používateľov cez ssh (sshd), démon pre umožnenie sťahovania súborov vzdialeným používateľom po sieti protokolom FTP (ftpd) apod. V modernejšej terminológii sa termín démon čoraz častejšie zamieňa termínom agent.

Operačný systém by mal umožniť

- ❖ vytváranie a ukončovanie procesov,
- ❖ pozastavenie a reaktiváciu procesov,
- ❖ synchronizáciu procesov a
- ❖ komunikáciu medzi procesmi (aj po sieti).

Správa (primárnej) pamäte

- ❖ odkladanie stránok (odkladanie nepotrebných stránok na disk a ich spätné nahrávanie do pamäte),
- ❖ simulovaná vyrovnávacia pamäť disku,
- ❖ prehľad o obsadenosti úsekov pamäte,
- ❖ pridelovanie pamäte novým procesom a
- ❖ pridelovanie pamäte procesom žiadajúcim o ďalšiu pamäť (a jej navracanie).

Správa sekundárnej pamäte

Patrí sem správa diskovej pamäte, pamäte na magnetických páskach, disketách, dátových CD nosičoch atď. Vo viacúlohovom operačnom systéme môže existovať viacero **súčasných** požiadaviek na prístup k disku (či dokonca k rovnakému súboru) a operačný systém počítača si s tým musí poradiť. V záujme koordinácie prístupov na sekundárnu pamäť musí operačný systém počítača:

- ❖ spravovať voľný priestor, mať o ňom prehľad a pridelovať ho,
- ❖ riadiť prístup k sekundárnej pamäti.

Správa periférnych zariadení

Operačný systém počítača by mal skrývať (pred aplikáciami) hardvérové špecifiká rovnakých typov V/V zariadení – o tie sa starajú ovládače daných zariadení. Operačný systém počítača však procesom poskytuje aplikačné rozhranie virtuálneho ovládača V/V zariadení, ktoré procesom zabezpečuje rovnaký prístup k rovnakej skupine zariadení.

Správa súborov

Stromovo hierarchizované adresáre a súbory sú prirodzenou štrukturalizáciou dát uložených v počítačovom systéme. V Unixe sa dokonca ku každému zariadeniu pristupuje ako k súboru. OS zabezpečuje:

- ❖ vytváranie a mazanie súborov a adresárov (adresár je súbor obsahujúci zoznam iných súborov či adresárov),
- ❖ ochranu dát systému a jednotlivých používateľov proti náhodnému alebo inak mienenému poškodeniu,
- ❖ mapovanie súborového systému na sekundárnu pamäť (513. až 1024. byte súboru je uložený v 23236. sektore disku...),
- ❖ zálohovanie súborov (väčšinou sa vykonáva pomocnými systémovými programami).

Komunikačný systém (sieťová podpora)

OS musí zabezpečiť komunikáciu medzi aplikáciami. Jednou z úloh operačného systému je tiež poskytovanie a práca so sieťovými službami. Ak je systém prevažne zameraným na poskytovanie sieťových služieb, hovoríme o **serverovom** operačnom systéme. Ak je

zameraný prevažne na využívanie sieťových služieb, hovoríme o **desktopovom** operačnom systéme.

Jednotlivé systémy musia vedieť spolu spolupracovať. Cieľom takejto spolupráce je zabezpečiť

- **Distribuované spracovanie dát** - systémy často vytvárajú architektúru typu klient-server, časť úlohy sa spracováva na serveri a časť na lokálnom počítači, z používateľského hľadiska však ide o jednu úlohu a používateľ väčšinou nerozozná čo sa kde spracováva.
- **Zdieľanie dát a zariadení** - operačný systém zabezpečuje prístup k centrálne spravovaným údajom (napr. na aplikačnom, databázovom alebo súborovom serveri) odosiela požiadavky ostatným zariadeniam siete (napr. tlačovému serveru) ale umožňuje zdieľanie aj lokálnych údajov a zariadení pre ostatných používateľov v sieti. Niektoré systémy umožňujú prostredníctvom siete zdieľať aj výpočtový výkon prostredníctvom tzv. cloud computingu.
- **Vzájomnú komunikáciu používateľov** - systém zabezpečuje komunikáciu medzi používateľmi (napr. pomocou mailu, textového rozhovoru, videohovoru a podobne)

Používateľské rozhranie

Používateľské rozhranie je prostredníkom medzi používateľom a počítačovým systémom (či konkrétne medzi používateľom a procesmi). Rozhranie je v skutočnosti proces, ktorý zabezpečuje, aby ostatné programy mohli byť používateľom spúšťané, a ovládané. Odovzdáva im vstup (z klávesnice či myši) od používateľa a zobrazuje výstup programu.

Rozoznávame používateľské rozhrania textové a grafické. Či už ide o textové či o grafické rozhranie, ich základný účel je rovnaký: Získať (od používateľa) príkaz, vykonať ho (väčšinou spustením nejakého programu s parametrami) a oznámiť, ako jeho vykonávanie dopadlo (to je väčšinou vecou samotného príkazu).

Textové používateľské rozhranie používa pre vstup údajov klávesnicu a pre výstup monitor terminálu. Textové rozhranie zabezpečuje **interpretátor** príkazov. U MS-DOSu je ním program Command.com, u Unixu si môžeme vybrať z niekoľkých tzv. shellov. Interpretátor príkazov zabezpečuje prepojenie vstupov a výstupov používateľa so spúšťanými programami. Textové rozhrania umožňujú používanie scenárov (keďže ide o interpretre príkazov). Ide o tzv. „batch“ súbory v MS-DOSe, v ktorých sa jednoducho nachádza niekoľko riadkov s príkazmi v poradí, v akom majú byť vykonávané za sebou. V Unixe sa scenáre nazývajú skriptami. Unixové shelly sú omnoho silnejšie ako príkazový interpretátor MS-DOSu – v skutočnosti ide o jednoduché interpretované programy. (Podobne je tomu v OS/2.)

Grafické používateľské rozhrania (GUI – Graphical User Interface) umožňujú ovládať počítač a jeho V/V zariadenia pomocou grafických ovládacích prvkov, ktoré tiež voláme **widgety**.

- **Ikony** – piktogramy – malé obrázky reprezentujúce aplikácie, súčasti počítača, sieťové zdroje a podobne.
- **kurzor** – malý pohyblivý symbol
- **Ponuky** (napr. ponuka Štart alebo ponuka Aplikácie)
- **Okná**, v ktorých sa zobrazujú práve spustené aplikácie alebo otvorené súbory.

- **Ostatné grafické objekty**, napr. tlačidlá, prepínače, zaškrŕavacie polia, vstupné textové polia, výberové zoznamy, dialógové okná pre výber súboru a podobne.

Používajú pre vstup myš a klávesnicu. Hovoríme tiež o grafickom používateľskom prostredí (operačného systému Windows či Unixovej nadstavby X-Window). Výstup programov zobrazujú v grafickej podobe. Sú intuitívnejšie a vyžadujú menšiu námahu pri učení sa. **Je to ich veľkou prednosťou.** Podmienkou je, aby bol program čo najviac intuitívny. Má to dve výhody. Za prvé, aj laik sa rýchlo naučí so systémom pracovať. To je mimoriadne dôležité pre všetkých. Za druhé, ak používateľ so systémom dlhšiu dobu nepracuje, ľahšie si naň opäť zvyká. Nevýhodou grafiky býva nemožnosť vytvárať scenáre. (Tie sa síce vytvárať dajú, no už je to v interpretácii príkazov. Nevýhoda býva vyvažovaná aspoň možnosťou tvorby tzv. makier.)

Používateľa nič z toho netrápi – proste komunikuje so systémom pomocou GUI resp. shellu a pomocou rozhrania samotných programov. Ani programátora štruktúra OS veľmi nezaujíma – ak potrebuje komunikovať so systémom či používateľom, používa iba API (Applications Programming Interface) tvorené tzv. volaniami služieb výkonného jadra operačného systému.

Komponenty opísané vyššie sa musia dať dohromady, aby vznikol funkčný OS. V skutočnosti mnohé systémy nemajú skoro žiadnu štruktúru, vznikli malé a neštruktúrované a postupne sa rozrastali. Dôsledkom je, že jednotlivé komponenty nie sú dobre oddelené a nerešpektujú modularitu.

Vlastnosti moderných operačných systémov:

- viacúlohové - umožňujú zdanlivo vykonávať viacero aplikácií naraz
- viacpoužívateľské - umožňujú pracovať viacerým používateľom súčasne
- distribuované - umožňujú rozdeliť spracovávanie úloh na viacero systémov

Operačné systémy môžeme rozdeliť podľa niekoľkých kritérií:

Podľa určenia

- desktopové - určené pre stolné počítače v domácnosti
- serverové - určené pre počítače, ktoré poskytujú sieťové služby
- pre konkrétne zariadenie (netbooky, mobily, PDA)

Podľa dostupnosti

- Komerčné - Windows, MAC OS, UNIX
- Slobodné - Ubuntu, Debian, OpenSuse, Fedora, Mandriva, Sabayon, Slackware, Gentoo, Knoppix, FreeBSD, OpenSolaris, FreeDos ...

Podľa použitého jadra

- Windows
- Linux - Ubuntu, Debian, OpenSuse, Fedora, Mandriva, Sabayon, Slackware, Gentoo, Knoppix ...

- BSD - Mac OS, PC-BSD, FreeBSD, OpenBSD, NetBSD...
- Solaris - OpenSolaris, Oracle Solaris
- FreeDOS
- ...

Bezpečnosť

Vzhľadom na základné požiadavky na operačný systém t.j. potreba spúšťania viacerých úloh naraz, potreba práce viacerých používateľov naraz a distribúcie úloh medzi systémami, je potrebné zabezpečiť, aby:

- jeden program nemohol narušiť chod iného programu alebo samotného operačného systému,
- aby používateľ svojím konaním nemohol narušiť chod programov a samotného operačného systému,
- aby používateľ nemohol pracovať s údajmi iných používateľov
- aby so systémom nemohli pracovať alebo narušiť neoprávené osoby
- aby boli sledované zaznamenávané kritické akcie programov a používateľov
- aby sa prostriedky (čas procesora, pamäť a V/V) boli pridelované spravodlivo
- aby nemohla byť narušená komunikácia medzi systémami
- aby sa stav systému a údajov dal obnoviť zo zálohy

Zdroj: M. Schmotzer: Operačné systémy
<http://maturitazinf.mrazovci.eu>