

Inicializace počítače

Při zapnutí počítače do sítě nebo při stisknutí tlačítka RESET proběhne studený start operačního systému v těchto krocích:

1. ROM-BIOS zjišťuje konfiguraci počítače, provádí test operační paměti a činnosti řadičů.
2. ROM-BIOS provádí základní nastavení počítače, načítá zavaděč a předává řízení zavaděči. Zavaděč hledá soubory IO.SYS a MSDOS.SYS. Čtení zavaděče a hledání operačního systému provádí v tomto pořadí:
 - (a) Čtení diskety A: a mohou nastat tyto případy:
 - i. V mechanice A: je systémová disketa, potom se operační systém zavede – již se nepokračuje bodem (b)
 - ii. V mechanice A: je jiná disketa než systémová, v tom případě se zavádění přeruší a uživatel je vyzván k nápravě situace
 - iii. V mechanice A: není žádná disketa (nebo není uzavřena páčka), potom ROM-BIOS pokračuje bodem (b)
 - (b) Čtení pevného disku C:
 - i. Je-li disk C: systémový, potom se operační systém zavede
 - ii. Není-li disk C: systémový nebo je v poruše, potom se počítač zastaví

Pro úspěšné zavedení systému z pevného disku C: je nutné při inicializaci počítače otevřít páčku na mechanice A: (je-li v ní disketa).

3. Jsou zavedeny soubory IO.SYS a MSDOS.SYS.
4. Systém se nastavuje podle obsahu CONFIG.SYS, vyhrazují se vyrovnávací paměti, instalují se ovladače zařízení.
5. Startuje se interpret příkazů, např. COMMAND.COM.
6. COMMAND.COM předává řízení dávce příkazů AUTOEXEC.BAT.
7. COMMAND.COM vypisuje prompt, běžný disk je systémový disk (pokud nebyla změna běžného disku v AUTOEXEC.BAT).

Teplý start se vyvolává stiskem kombinace kláves ALT+CTRL+DEL. Při teplém startu proběhne vše jako při startu studeném, kromě bodu 1. Bod 1 se neprovede.

Charakteristika operačního systému CP/M (respektive MICROS)

1 Úvod

Operační systém CP/M vznikl pro mikropočítače s mikroprocesorem 8080 a s pružnými disky (disketami). Je konstruován jako přenosný operační systém, tj. lze jej poměrně snadno implementovat na jiných počítačích. Jsou známy např. implementace pro počítače s procesory 8085, 8086, Z80, s většími disky apod. Jde o systém jednoduchý a přitom mocný, který je v mnoha zemích velmi oblíben a rozšířen.

2 Řídící jazyk CP/M (r-jazyk CP/M)

Výraz (program) v řídicím jazyce je tvořen posloupností příkazů. Prvním (nepísaným) příkazem je inicializace, která zpravidla spočívá v zavedení CP/M ze systémové diskové jednotky spuštěním zaváděcího programu. Další příkazy se čtou ze souboru pojmenovaného SSS. SUB na systémovém disku (existuje-li takový soubor) a poté z klávesnice systémové konzoly. Lze užívat dávkový i konverzační režim práce.

Příkazy CP/M mají tvar

[jednotka:] [jméno] [parametry],

kde [X] značí nepovinný výskyt X. Jednotka určuje písmenem A až P aktuální diskovou jednotku (určení platí až do změny, na počátku je aktuální jednotkou systémová disková jednotka). Jméno určuje činnost interpretu řídicího jazyka (nazývaného CCP - Console Command Processor) nebo název programu, který má být proveden. Parametry jsou znaky předávané interpretu nebo programu, které zpravidla upřesňují požadovanou činnost.

Jména příkazů pro CCP jsou následující:

DIR	zobrazení adresáře souborů (DIRectory),
TYPE	zobrazení souboru znakově,
ERA	zrušení souboru (ERase),
REN	přejmenování souboru (REName),
SAVE	uložení obsahu paměti na disk,
USER	nastavení aktuálního uživatele.

Jako jména programu lze uvést jména systémových programů:

STAT	zobrazení a modifikace stavu systému (STATus),
PIP	kopírovací program (Pheripheral Interchange Program),
DUMP	zobrazení souboru hexadecimálně,
SUBMIT	program pro zpracování dávky příkazů uložených na disku (volání procedury řídicího jazyka),
ED	textový editor,
ASM	Asembler pro 8080 (ASembler)
LOAD	příprava spuštění,

Charakteristika operačního systému UNIX

Příznačným rysem operačního systému Unix je jeho „vrstevná“ struktura, přičemž každá z vrstev si v rámci systému uchovává své relativně samostatné postavení.

Vnitřní uspořádání operačního systému UNIX si lze představit jako souhrn tří programových celků. Jsou jimi: jádro operačního systému „obalené“ vrstvou systémových volání a knihovnicích funkcí, nad nímiž je vrstva systémových programů.

Do všech tří programových celků se promítají tři základní problémové oblasti operačního systému, kterými jsou procesy, souborový systém a mechanismus vstupu a výstupu. Základní představu o vnitřním uspořádání Unixu poskytuje schéma 1.1. Zastavme se blíže u jednotlivých programových celků operačního systému Unix.

1.1 Jádro operačního systému

Jádro operačního systému (kernel, exekutiva) — lze z obsahového hlediska charakterizovat jako souhrn systémových rutin, systémových záznamů a struktur a systémových vyrovnávacích pamětí.

Systémové rutiny obsahují instrukční část systému, vznikají v procesu generace z několika desítek programů ve zdrojovém tvaru. Tyto rutiny jsou sestaveny do jediného souboru v proveditelném tvaru, obsahující zaveditelný obraz jádra, jenž nese zpravidla označení `unix` a je umístěný v kořenovém adresáři.

V okamžiku svého zavedení do operační paměti si systém vytváří prostor nejen pro svůj instrukční kód, ale je alokován rozsáhlý prostor pro **systémové záznamy** definované pomocí systémových struktur.

Poznamenejme v této souvislosti, že struktura je chápána v souladu s definicemi jazyka C jako vzor záznamu. Struktura naplněná daty tvoří záznam, který může být buď jednoduchý nebo vytvářet pole záznamů — tabulku. Záznamy jsou zpravidla vnitřně členěny do položek, které mohou obsahovat proměnné různého typu.

Systémové záznamy vytvářejí určitou datovou základnu systému, která bývá označována jako systém tabulek jádra. Operační systém Unix si udržuje několik desítek systémových záznamů a tabulek pro zabezpečení veškerých funkcí, které plní (např. tabulka otevřených souborů, tabulka procesů aj.).

Třetí obsahovou složkou jádra operačního systému jsou **systémové vyrovnávací paměti**. Tyto vyrovnávací paměti slouží zejména k usnadnění komunikace s perifériemi s přímým přístupem.

Vlastní instrukční kód operačního systému — systémové rutiny — lze při určité schematičnosti rozdělit na dvě skupiny, a to na systémové rutiny mající zcela interní charakter a na rutiny poskytující systémové služby.

První skupina systémových rutin — **interní rutiny jádra** — realizuje aktivity systému, jež jsou spojeny se správou a přidělováním systémových zdrojů a zajišťuje dále vazbu na technické prostředky (ovladače periferních zařízení, zpracování přerušení aj.).

Zastavme se v této souvislosti u krátkého zhodnocení dvou základních mechanismů, které operační systém Unix používá pro správu a přidělování systémových zdrojů. Jedná se o sdílení času a swapování.

Sdílení času (time sharing) přiděluje rovnoměrně procesor mezi jednotlivé základní aktivity počítačového systému — procesy. Základní kritérium, které slouží pro stanovení přístupu určitého procesu k procesoru, tvoří priorita. Priorita je charakterizována celým číslem; čím je toto číslo vyšší, tím priorita klesá a daný proces je vzhledem k ostatním znevýhodňován. Číselná hodnota priority odráží kvantum času procesoru, které daný proces po dobu své aktivity spotřeboval, blíže je tato otázka diskutována v oddíle 2,3.1 v souvislosti s popisem struktury proc.

Swapování, jako druhý z klíčových mechanismů OS Unix, představuje odsouvání aktivních programů — procesů z operační paměti na tzv. swapovací zařízení a naopak. Swapovací zařízení představuje samostatnou část diskové kapacity, jež je spravována jádrem systému a je uživatelsky nepřístupná. Kritérium pro rozhodnutí o odsunutí určitého procesu na swapovací zařízení je v tomto případě opět čas, tentokrát však čas, který strávil určitý proces v operační paměti.

Druhou skupinu systémových rutin tvoří rutiny mající vazbu na uživatelské prostředí — tzv. **systémové služby**, které realizují určitou aktivitu operačního systému na základě požadavku z vnějšku.

1.2 Vrstva systémových volání a knihovních funkcí jazyka C

Systémová volání představují určitý uživatelský programový požadavek na provedení systémové služby. Systémová volání operačního systému Unix lze při určité schematičnosti rozdělit na tři skupiny:

- Systémová volání pro práci s procesy — jedná se o získávání číselných charakteristik procesů, volání zajišťující mechanismus vytváření, pozastavování a ukončování procesů a volání zabezpečující datový přenos mezi procesy a jejich synchronisaci.
- Systémová volání pro V/V operace — slouží k otevírání a uzavírání souborů, zápis do souboru, čtení ze souboru, práce s ukazovátkem do souboru aj.
- Systémová volání pro práci se souborovým systémem — vytváření souborů, změna a nastavování přístupových práv, začleňování dalších souborových systémů aj.

Rozhodující část systémových volání je v systémech BSD obsažena, společně se základními funkcemi jazyka C, ve standardní knihovně jazyka C — libc. a. Doplňková část systémových volání je zahrnuta v knihovním souboru libjobs. a.

Knihovní funkce představují knihovní podprogramy (moduly) některého z kompilátorů vyšších programovacích jazyků. Knihovní funkce navazují na systémová volání, rozšiřují je a přibližují je požadavkům aplikace. Operační systém standardně vyčleňuje pro knihovní soubory dva adresáře — /lib a /usr/lib.

Vzhledem ke klíčovému postavení jazyka C v operačních systémech Unix hrají zde důležitou roli i jeho standardní knihovny. Systém BSD obsahuje proti předchozím verzím operačního systému Unix významné zdokonalení — podporu překryvného formátu instrukční části programů. — Veškeré knihovny jazyka C (platí to i pro Fortran 77) jsou vytvořeny ve dvou tvarech — překryvném i nepřekryvném. Formát knihovního souboru je zřejmý i z jeho jména — knihovny pro překryvný formát mají ve svém názvu obsaženu příponu „ov“ (overlay).

Knihovní funkce jazyka C, případně i jiných kompilátorů, je možné, obdobně jako v případě systémových volání, rozdělit podle účelu nasazení na knihovní funkce pro práci s procesy, se souborovým systémem, pro V/V operace a na ostatní.

V této souvislosti je nutné upozornit na mimořádný význam té skupiny knihovních funkcí jazyka C, jež se vztahuje k vstupním a výstupním operacím. Význam vyplývá z charakteru jazyka C, který v sobě nemá zabudován uvedený typ operací ve formě příkazů jazyka. V důsledku toho jsou veškeré požadavky na V/V operace řešeny voláním externích funkcí.

Závěrem popisu této vrstvy operačního systému Unix se zastavme u krátkého porovnání systémových volání a knihovních funkcí. Jak je zřejmé, knihovní funkce i systémová volání mají v operačním systému Unix mnoho společného — jsou obsažena ve stejném knihovním souboru — libc.a a práce s nimi je řešena externím voláním. Knihovní funkce i systémová volání provádějí i shodné nebo obdobné činnosti, avšak rozdílným způsobem. Systémová volání komunikují se systémovými službami jádra, knihovní funkce provádějí požadované činnosti samostatně nebo pomocí odkazů na systémová volání. Knihovní funkce umožňují využívat systémových vyrovnávacích pamětí, což výrazně zrychluje V/V operace. Některé z knihovních funkcí (např. printf, scanf, aj.) umožňují provádět přímou konverzi formátů. Knihovní funkce tedy poskytují rozsáhlejší programátorský komfort.

1.3 Vrstva systémových programů

Programy jsou v operačním systému Unix jedním z typů souborů v proveditelném tvaru.

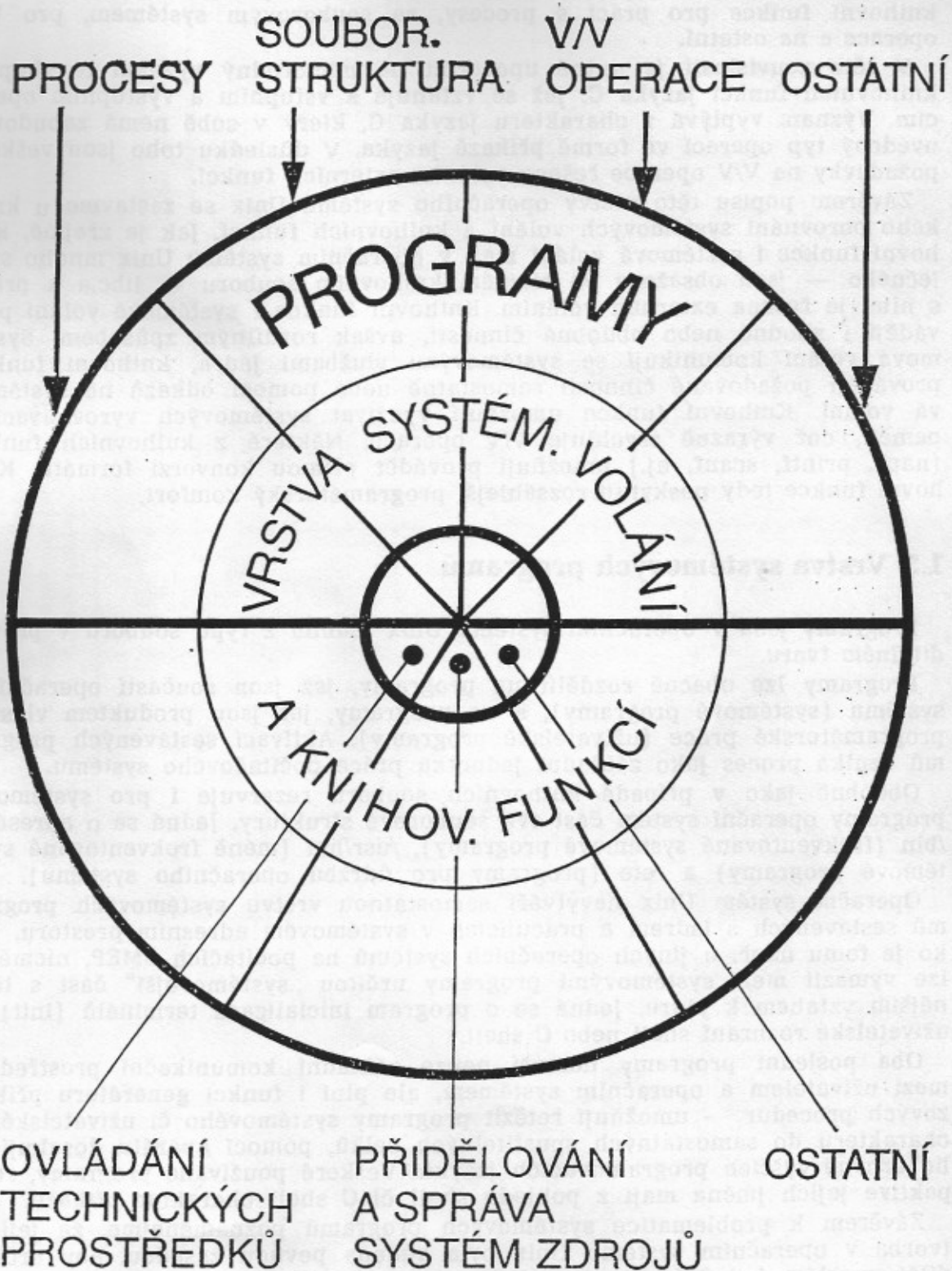
Programy lze obecně rozdělit na programy, jež jsou součástí operačního systému (systémové programy), a na programy, jež jsou produktem vlastní programátorské práce (uživatelské programy). Aktivací sestavených programů vzniká proces jako základní jednotka práce počítačového systému.

Obdobně jako v případě knihovních souborů rezervuje i pro systémové programy operační systém část své souborové struktury. Jedná se o adresáře /bin (frekventované systémové programy), /usr/bin (méně frekventované systémové programy) a /etc (programy pro údržbu operačního systému).

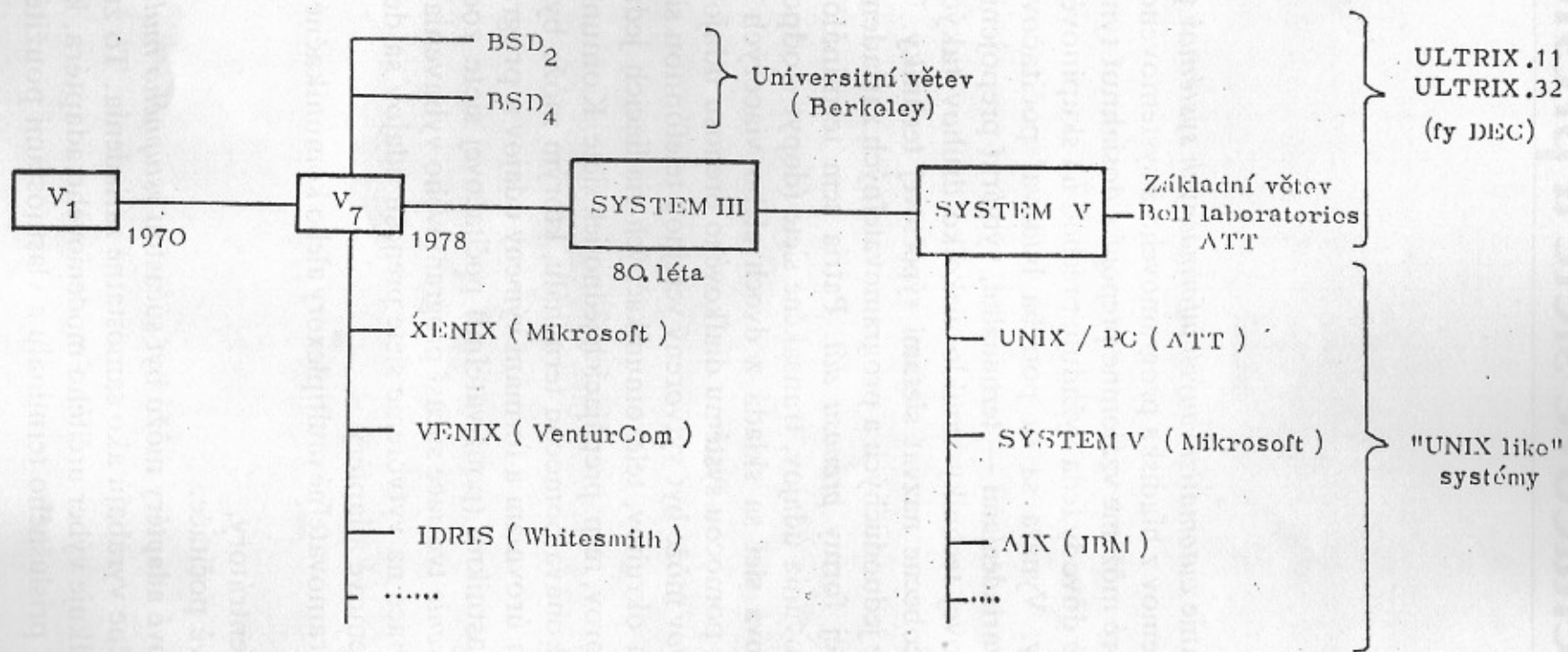
Operační systém Unix nevytváří samostatnou vrstvu systémových programů sestavených s jádrem a pracujících v systémovém adresním prostoru, jako je tomu např. u jiných operačních systémů na počítačích SMEP, nicméně lze vymezit mezi systémovými programy určitou „systémovější“ část s těsnějším vztahem k jádru. Jedná se o program inicializace terminálů (init) a uživatelské rozhraní shell nebo C shell.

Oba poslední programy netvoří pouze základní komunikační prostředek mezi uživatelem a operačním systémem, ale plní i funkci generátoru příkazových procedur — umožňují řetězit programy systémového či uživatelského charakteru do samostatných spustitelných celků, pomocí aparátu dosahujícího úrovně vyšších programovacích jazyků. Veškeré používané programy, respektive jejich jména mají z pohledu shell či C shell charakter příkazů.

Závěrem k problematice systémových programů poznamenejme, že jejich tvorba v operačním systému Unix byla vedena pevnou zásadou, aby určitý dílčí problém byl řešen samostatným, jednoúčelovým a pokud možno krátkým programem. Takto koncipované systémové programy lze jednoduchým způsobem spojovat a řetězit např. výše uvedenými příkazovými procedurami uživatelského rozhraní. Zvláštní význam v této souvislosti zaujímají tzv. filtry, kdy vstupní data program bere ze standardního vstupu (klávesnice terminálu) a po zpracování ukládají výsledky na standardní výstup (obrazovku terminálu).



Vnitřní struktura OS Unix



obr. 0.1 Vývoj OS Unix

5. Počítačové siete a prenos údajov

Budovanie *automatizovaných informačných systémov* predpokladá použitie nových systémov z hľadiska programového a systémového navzájom kompatibilných, ktoré môžeme vzájomne prepojiť a dosiahnuť tým vyššiu využiteľnosť. Ekonomické dôvody teda vyžadujú prechod na skupinové využívanie výpočtovej techniky. Vynára sa tu potreba budovať počítačové systémy vybavené **koncovými zariadeniami — terminálmi**, vytvárať prepojenia počítačov. S vývojom výpočtovej techniky vzniklo niekoľko druhov takýchto systémov, ktoré môžeme všeobecne nazvať **sietami výpočtovej techniky**. Tieto tvoria rozličné konfigurácie jednoduchých a programovateľných zariadení prepojených pomocou niektorej formy *prenosu dát*. Patria sem terminálové siete, siete zberu a predspracovania údajov, transakčné siete (dopyt—odpoveď) a pod.

Počítačová sieť sa skladá z dvoch alebo viacerých počítačov vzájomne prepojených pomocou systému diaľkového prenosu údajov. Systém *diaľkového prenosu údajov* môže byť vytvorený verejnou telefónnou sieťou alebo pomocou prenosových okruhov, telekomunikačných riadiacich jednotiek alebo špeciálnych procesorov, resp. prepájacich jednotiek siete. Komunikácia používateľa so sieťou sa vykonáva pomocou terminálu, ktorým môže byť aj osobný počítač.

Rôznym úrovňam a formám výmeny údajov a programov medzi počítačmi, resp. účastníkmi (používateľmi) počítačovej siete zodpovedá aj príslušné *sieťové vybavenie* tvoriace súčasť programového vybavenia siete. **Technické prostriedky** slúžiace na vytvorenie siete prenosu údajov sa delia na:

- a) modemové adaptéry,
- b) programovateľné multiplexory alebo komunikačné predradené procesory,
- c) koncentrátory,
- d) uzlové počítače.

Modemové adaptéry môžu byť súčasťou *vstupného multiplexora*, ale dnes sa takmer výlučne vyrábajú ako samostatné zariadenia. To znamená, že multiplexor sprostredkuje výber určitého modemového adaptéra, ktorého typ je určený vlastnosťami príslušného terminálu a vlastnosťami použitého systému prenosu údajov.