

# V

## SISTEME DE OPERARE

### V.1 GENERALITĂȚI PRIVIND SISTEMELE DE OPERARE

Orice PC funcționează pe baza unui program (*software*) denumit **sistem de operare** (OS - *Operating System*) care gestionează toate resursele sistemului, funcționând ca interfață între programele de aplicații și componentele fizice (*hardware*).

Sistemul de operare este instalat pe discul-fix (HDD - *Hard-Disk Drive*) al calculatorului, pe așa-numita partiție activă, și este încărcat la fiecare pornire a sistemului (*boot*) din memoria ROM (*Read-Only Memory*) de către BIOS (*Basic Input-Output System*) după testarea sistemului și identificarea unităților de disc (HDD, CDROM, floppy), a memoriei RAM (*Random Access Memory*) disponibile și a monitorului, pe baza informațiilor conținute în memoria CMOS (*Complementary Metal Oxid Semiconductor*).

Un sistem de operare este conceput ca un program care rulează în permanență în calculator, în timpul funcționării acestuia. De multe ori pentru OS se folosește termenul echivalent **kernel**.

Scopul principal al kernelului este acela de a oferi utilizatorului un mod transparent de comunicare cu calculatorul, adică posibilitatea execuției programelor utilitare fără a fi necesară cunoașterea detaliilor privind funcționarea părții hardware sau a nivelelor software inferioare celui de aplicație.

Simplificarea modului de accesare a programelor de aplicație este posibilă prin folosirea **interfețelor grafice de utilizator** (GUI - *Graphic Unit Interface*).

De asemenea, kernelul permite programatorilor de aplicații să lucreze cu un set standard de comenzi adresate acestuia, fără a recurge la comenzile codate specifice limbajelor-mașină, dependente de tipul de hardware sau de software folosit.

Un calculator este un sistem complex alcătuit din cinci subsisteme:

1. procesor (CPU - *Central Processing Unit*)
2. memorii
3. subsistemul de stocare a datelor
4. subsistemul video
5. magistrale (de date, de adrese etc).

La acestea se adaugă echipamentele periferice care reprezintă interfețele dintre utilizator și calculator (tastatură, mouse, monitor).

Frecvența de lucru a procesorului (*clock speed*) influențează direct performanțele calculatorului. Rapiditatea acestuia depinde de cantitatea de memorie de mare viteză disponibilă în cipul procesorului (*on-chip/ primary/Level 1-cache memory*) și în afara acestuia (*off-chip/secondary / Level 2-cache memory*).

În ultimul caz este necesară o magistrală de date (*cache bus*) cu o anumită viteză de transfer și o anumită dimensiune (32; 64 biți), pentru comunicația dintre memoria cache primară și cea secundară.

Comunicația dintre componentele calculatorului se realizează pe diferite magistrale, externe sau interne, de diferite tipuri (ISA, EISA, PCI, AGP, SCSI, PCMCIA).

Kernelul administrează și alocă, la cerere (*system call*), diverselor programe de aplicații, resursele sistemului, precum:

- timp de procesare a unității centrale (CPU - *Central Processing Unit*);
- acces la memoria RAM;
- spațiu de stocare a datelor pe disc simultan cu administrarea sistemului de fișiere;
- resurse de intrare-ieșire.

Comunicarea dintre kernel și componentele hardware ale calculatorului se face prin intermediul unor programe software, de mici dimensiuni, de control a fiecărui echipament în parte (*device driver*), compatibile în general numai cu anumite sisteme de operare și cu un anumit tip de echipament. Funcționarea oricărei componente hardware din calculator este posibilă prin instalarea unui driver corespunzător.

Interacțiunea directă a sistemului de operare cu componentele hardware ale calculatorului se exercită prin intermediul **sistemului de întreruperi** (IRQ - *Interrupt ReQuest*) și prin **excepții**, adică acele evenimente neprevăzute (erori de sistem) pe care kernelul trebuie să le elimine. De exemplu, dacă un program încearcă să scrie în spațiul de memorie alocat altei aplicații, atunci kernelul intervine în soluționarea acestei excepții prin protejarea acelei zone și trimite un mesaj de eroare (*general protection fault*).

În fapt, sistemul de operare destinat clienților urmărește corectitudinea funcționării CPU, administrează spațiile de memorie și sistemul de fișiere (FS - *File System*) asigurând stocarea și recuperarea datelor pe disc, monitorizează procesele de intrare/ieșire și comunicațiile dintre procese, dar răspunde și de securitatea sistemului.

Programul utilizat de un calculator pentru a se conecta la o rețea locală se numește **program-client** (*client software*) iar stația respectivă reprezintă un **calculator-client** (*client-PC*). Clientul, software sau hardware, este cel care adresează o cerere către o componentă a sistemului. Răspunsul la cererea clientului este dat de o altă componentă denumită **server**, care reprezintă un furnizor de servicii. Natura serverului poate fi logică (*software*) sau fizică (o componentă *hardware* din sistem).

Comunicația dintre un PC și rețea este posibilă numai dacă sistemul local este compatibil, hardware și software, cu rețeaua. Din acest motiv, clienții și serverele dintr-o rețea trebuie să folosească același sistem de operare de rețea (NOS - *Network Operating System*), cum sunt NetWare, Windows NT etc. Există versiuni diferite de NOS, ca preț și complexitate, pentru client și pentru server.

În figura V.1, se prezintă interacțiunile dintre componentele unui sistem de tip client, sub forma unui model stratificat.

Între oricare două module software se intercalează interfețe logice cum sunt cele grafice GUI, cele de aplicație API (*Application Programming Interface*) etc.

Arhitectura sistemului de operare poate fi concepută fie în varianta **monolitică**, ca un sistem închis, dificil de modificat sau de îmbunătățit, fie ca sistem deschis, cu numeroase subseturi (**microkernel**) lucrând în **mod privilegiat** (*kernel mode*), cu alocare necondiționată de memorie, și cu alte module care pot lucra în **mod de utilizator** (*user mode*), neprivilegiat.

În prezent, se utilizează **sisteme de operare modulare, bazate pe microkernel-uri**, care au avantaje incontestabile față de cele monolitice în ceea ce privește posibilitățile de expandare, portabilitatea și siguranța funcționării.

O caracteristică avantajoasă a unor sisteme de operare este independența lor de tipul de hardware folosit. De exemplu, Windows NT poate fi utilizat în sisteme Intel, DEC, PowerPC ș.a.

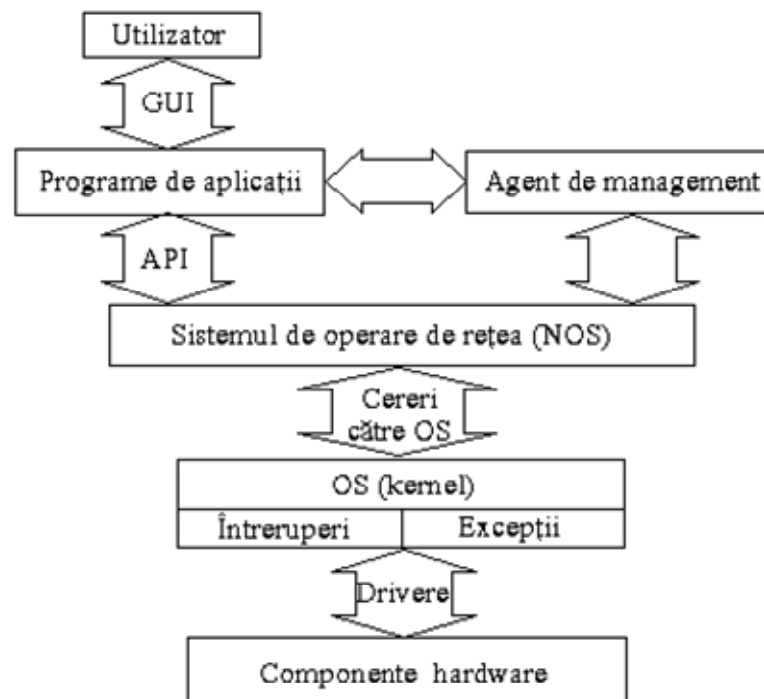


Fig. V.1 Arhitectura sistemului de tip client

Unele sisteme cu microkernel-uri pot fi concepute astfel încât să permită rularea unor aplicații realizate și cu alte kernel-uri, fiind denumite **sisteme de operare cu personalitate multiplă** (*Multiple Personality OS/ Multiple Workplace Environments*). Astfel, Windows NT permite rularea unor aplicații concepute în sisteme OS/2, Windows pe 16 sau 32 de biți.

## V.2 TIPURI DE SISTEME DE OPERARE

Orice sistem de operare funcționează pe baza unor **subrutine** (*thread*) cu instrucțiuni specifice executabile la nivelul procesorului, îndeplinind **sarcini** specifice (*task*) care necesită alocarea unor resurse ale sistemului. Relația sarcină-subrutină este echivalentă cu cea părinte-copil, subrutina neavând dreptul să solicite resurse de sistem pentru execuție.

O aplicație software cu mai multe subrutine (*multithreaded*), odată lansată în execuție, reprezintă o sarcină pentru sistemul hardware. Sarcina respectivă interacționează direct cu sistemul de operare pentru a i se alocă resurse în vederea execuției. Este posibil ca o sarcină să nu poată fi executată imediat, fiind trecută într-o "coadă de așteptare" (*queue*). Aceasta este un **port logic**, în timp ce la nivelul echipamentelor, căile de intrare-ieșire reprezintă **porturi fizice**.

În sistemele hardware cu mai multe procesoare, se folosesc **sisteme de operare client** care admit **procesarea multiplă** (*Multiprocessing OS*), adică execuția diferitelor sarcini simultan, în paralel pe toate procesoarele.

Sistemele de operare care permit ca mai mulți utilizatori să folosească sistemul simultan și să ruleze diferite aplicații în același timp se numesc **sisteme de operare cu utilizatori multipli** (*Multiuser OS*).

Un sistem de operare care poate rula mai multe programe de aplicații simultan este denumit **sistem de operare cu sarcini multiple** (*Multitasking OS*).

Există două categorii de sisteme de operare care admit execuția sarcinilor multiple:

1. **sisteme cooperative** (*Cooperative Multitasking*) în care se alocă unei sarcini toate resursele, celelalte sarcini fiind în așteptare, existând riscul monopolizării sistemului;

2. **sisteme ierarhizate** (*Pre-emptive Multitasking*) care alocă resursele fiecărei sarcini lansate în execuție în funcție de prioritatea acesteia. Un criteriu de stabilire a ierarhiilor de priorități poate fi cel al timpului alocat. În momentul expirării timpului maxim de rulare a unei sarcini, aceasta este trecută în starea de așteptare și se alocă resurse altei sarcini.

**MS-DOS** (*Microsoft Disk Operating System*) a fost inițial proiectat ca sistem de operare cu un singur utilizator, dar în versiunile ulterioare a devenit sistem cu utilizatori multipli.

Acest sistem se instalează pe partiții formate FAT (*File Allocation Table*), pe 16 sau 32 de biți.

DOS a fost conceput ca un program de tip linie-de-comandă, fără o interfață grafică (GUI) ușor de utilizat.

S-au proiectat sisteme de operare pentru LAN bazate pe DOS (de exemplu, WINDOWS și LANtastic), cu caracteristici din ce în ce mai sofisticate.

Începând cu versiunea 5.0, DOS a permis încărcarea programelor și fișierelor de diferite tipuri, inclusiv drivere pentru componentele hardware, în afara memoriei de bază (640 KB).

Totuși pentru a funcționa ca sistem de operare complet într-o rețea bazată pe modelul client/server a fost necesară introducerea unei GUI și a funcționalităților pentru operare în rețea, prin programele MS Windows care oferă GUI iar în versiunile recente, începând cu WIN95, includ și NOS.

Windows 3.1 oferea o GUI "prietenosă", folosea interfețe de aplicații pe 16 biți (Win16 API) și permitea execuția sarcinilor multiple în modul cooperativ. Instalarea sistemului se făcea pe partiții formate FAT 16.

Versiunile Win9x sunt pachete software multistrat (*multilayer client software*) care includ OS, NOS și GUI. Aceste programe utilizează interfețe de aplicații pe 32 de biți (Win32c API) care

admit execuția sarcinilor multiple ierarhizate și protecția zonelor de memorie alocate diferitelor aplicații. De asemenea, este permisă rularea aplicațiilor realizate pentru versiunile anterioare de Windows, pe 16 sau 32 de biți, precum și recunoașterea automată a componentelor hardware, prin autodetecție și autoconfigurare (PnP - *Plug-n-Play*). Instalarea sistemului se face pe partiții formate FAT 32.

Compatibilitatea cu sistemele pe 16 biți creează o relație de subordonare a sistemului de 32 de biți față de acestea. Astfel facilitățile de operare cu sarcini multiple ierarhizate sunt blocate în cazul rulării unei aplicații pe 16 biți, reducându-se astfel performanțele sistemului.

Sistemele Windows NT, Windows 2000, Windows XP , cu versiuni diferite ca preț și complexitate pentru servere și utilizatori particulari (*home users*), oferă facilități sporite privind securitatea și siguranța sistemului.

Win2k poate fi instalat atât pe partiții FAT32 cât și NTFS (*NT File System*).

WinNT și WinXP rulează numai pe partiții non-DOS, formate NTFS, ceea ce face imposibilă rularea aplicațiilor definite exclusiv sub DOS.

WinNT utilizează diferite interfețe de aplicație pe 32 de biți, cu diverse caracteristici:

Win32 API cu facilități de operare numai pe 32 de biți, definită numai pentru aplicații create sub Windows NT;

Win32s API reprezentând un **subset** de interfețe pe 32 de biți, provenite din Win16 API. Acestea sunt definite într-o bibliotecă dinamică (*DLL - Dynamic Link Library*) și permit rularea aplicațiilor pe 16 biți create pentru Win3.1, prin conversia cererilor pe 32 de biți în formatul Win16;

Win32c API, **compatibile** cu Win16 API, prezintă aproape toate facilitățile interfețelor originale pentru WinNT. Sunt utilizate pentru rularea aplicațiilor create pentru Win9x sub WinNT.

Windows NT introduce o facilitate suplimentară față de sistemele Win9x și anume **procedura de apel local** (*LPC - Local Procedure Call*) prin intermediul căreia se alocă, la cerere, resurse hardware tuturor aplicațiilor. Astfel aplicațiile nu mai comunică direct cu sistemul, ceea ce face ca anumite programe, de exemplu jocurile, definite pentru versiunile anterioare de Windows, să nu mai poată fi rulate cu performanțe optime sub WinNT.

Programele de aplicații (*Win32 client*) și subsistemele de securitate lucrează sub WinNT în modul neprivilegiat. Subsistemele de management și kernelul lucrează în mod privilegiat, cu acces direct la resursele hardware ale sistemului.

Rularea sub WinNT a aplicațiilor create pentru alte sisteme de operare (OS/2, Win16) se realizează în subsisteme dedicate acestora, denumite **mașini virtuale** (*virtual machine*), WinNT fiind un sistem cu personalitate multiplă.

Un alt sistem de operare foarte apreciat de utilizatori este sistemul **UNIX**, definit în diferite variante: BSD-UX (*Berkeley Software Distribution Unix*), SVR4-UX (*At&T System V Release 4 Unix*), HP-UX (*Hewlett-Packard Unix*), Ultrix (dezvoltat de DEC), Solaris (definit de Sun Microsystems), Open Server (definit de SCO - *Santa Cruz Operation*) etc.

Indiferent de firma producătoare, Unix este un OS cu utilizatori multipli, care suportă procesarea multiplă simetrică, execuția ierarhică a sarcinilor multiple, rularea simultană a mai multor subrutine și admite modurile de utilizator, neprivilegiat, și kernel, privilegiat, pentru protecția resurselor hardware ale sistemului (*kernel-based, multiuser, multiprocessing, multitasking, multithreaded OS*).

Ca și DOS, Unix este un program de tip linie-de-comandă, criptic, recomandat îndeosebi mediilor științifice.

**Novell DOS versiunea 7 (ND7)** este un pachet de programe care include DOS (echivalent MS DOS 6.22), precum și sistemul de operare în rețea (NOS) pentru clienții NetWare cu programe de servicii (e-mail, chat etc) specifice. Ca avantaje ale ND7 se evidențiază faptul că acest sistem de operare (*multitasking kernel*) poate rula aplicații Windows, permite crearea rezervelor de sistem (*backup*), detecția virusilor și managementul rețelei.

**OS/2** este un sistem de operare similar cu WinNT dar incompatibil cu acesta, care include mașini virtuale implementate distinct pentru aplicații proiectate pentru OS/2, DOS sau Windows pe 16 sau 32 de biți.

## V.3 MODELUL STRUCTURAT AL UNUI SISTEM DE OPERARE DE REȚEA

Comunicația dintre un calculator și rețea se realizează fizic prin intermediul plăcii de rețea iar logic (*software*) prin sistemul de operare de rețea (NOS).

Comunicația dintre programele de aplicație și NOS se realizează prin intermediul unei interfețe de aplicații (API). Pentru rețele de PC-uri interfața API standard este **NetBIOS API** (*Network Basic Input Output System API*).

NetBIOS este prin definiție un sistem de comenzi care realizează:

- interpretarea cererilor adresate NOS de către programele de aplicații;
- transferul acestor cereri către programul de comunicație cu rețeaua.

Utilizatorul unui calculator-client din rețea poate solicita un serviciu care implică fie resurse locale, fie resurse din rețea, de pe un server sau alt calculator-client.

**Protocolul NetBIOS de redirectionare** (NetBIOS *Redirector*) transferă cererea în format NCB (*Network Control Block*) fie sistemului de operare local, fie către NetBIOS API care o convertește în formatul acceptat de rețea (SMB - *Server Message Block*), o trimite către NOS iar acesta o comunică plăcii de rețea în vederea transferului în LAN. Serverul răspunde cererii clientului pe baza **protocolului SMB** definit în NetBIOS.

Orice NOS poate fi structurat pe baza modelului OSI.

Pe nivelele inferioare OSI 1 și 2, orice NOS poate lucra pe baza oricărui standard de LAN.

Diferențierile între diverse NOS apar abia pe nivelele OSI superioare, în funcție de protocoalele implementate de fiecare (Fig. V.2).

Comunicația client-server este posibilă numai dacă se folosește același protocol de transport. De aceea, în multe sisteme de operare sunt incluse mai multe protocoale de transport.

În general, TCP/IP este considerată stivă de protocoale universală și este inclusă în majoritatea sistemelor de operare de rețea, pentru clienți și servere.

Modelul OSI	SISTEME DE OPERARE DE REȚEA (NOS)							
	WinNT Win9x		Alte NOS bazate pe DOS		OS/2		Novell Netware	
Nivel aplicație	Redirector NCP	Redirector SMB	Redirector DOS		Redirector NCP	Redirector SMB	Shell NetWare	
Nivel prezentare							Redirector NCP	
Nivel sesiune							Emulator NetBIOS	
Nivel transport	SPX	TCP	NetBIOS		SPX	TCP	SPX	TCP
Nivel rețea	IPX	IP			IPX	IP	IPX	IP
Nivel legătură de date	Standarde LLC							
	Standarde LAN, WAN							
Nivel fizic	Standarde de interfețe fizice							

Fig.V.2 Protocoalele implementate în diferite NOS



**Observații:**

1. Pentru alte NOS, interfețele pentru programele de aplicații (API) sunt denumite:

- **socluri** (*socket*) pentru Berkeley Unix;
- SPX/IPX pentru rețele Novell NetWare.

2. Se folosesc trei tipuri de **interfețe de utilizator**:

- interfețe bazate pe caractere, pentru programele de tip 'linie-de-comnadă';
- interfețe grafice de utilizator (GUI), orientate pe obiect;
- interfețe sociale, care se doresc a fi mult mai dinamice decât cele grafice, mai bine adaptate capacităților de lucru ale utilizatorului, 'simțind' când acesta are nevoie de ajutor și oferindu-i sugestii fie prin intermediul unor meniuri de tip HELP, fie direct pe ecran, prin scurte comentarii.

În prezent, toate firmele producătoare de software dezvoltă astfel de interfețe. Prima interfață socială dezvoltată de Microsoft se numește BOB. Pentru Windows 9x se folosește interfața **Simply Village**, fiind similară în multe aspecte cu sistemul de indicatoare amplasate într-o localitate.

3. Pentru Unix se folosește o interfață grafică cu ferestre multiple (*X Windows*), proiectată pentru mai multe platforme Unix (HP, IBM, Sun, Novell) cunoscută sub abrevierea CDE (*Common Desktop Environment*).

4. WinNT și Unix nu sunt incompatibile. De exemplu, pachetul *eXceed 4 Windows NT* permite sistemului NT să lucreze ca un server X (Unix) pentru clienții X și, în același timp, să ruleze aplicații sub Windows pentru clienții Windows. În general, se folosește un program sau un modul hardware de emulare care implementează **protocolul de prezentare a ferestrelor X** (*X Windows Presentation Protocol*) pentru schimbarea formatului datelor dintr-o interfață în alta.

Totuși trebuie apreciată corect necesitatea utilizării mai multor platforme în aceeași rețea.

5. Pentru a permite furnizarea de servicii în bune condiții în rețea, sistemele de operare pentru servere includ funcții suplimentare celor oferite clienților, mai precis module software dedicate:

- securizării transmisiei;
- autentificării mesajelor;
- criptării datelor;
- gestionării resurselor hardware ale calculatorului-server, având capacitate mare de stocare a informațiilor, memorii RAM rapide, placă de rețea multiport, eventual mai multe procesoare.

## V.4 TIPURI DE DRIVERE PENTRU PLĂCILE DE REȚEA

Întrucât majoritatea sistemelor de operare în rețea (NOS) implementează protocoale multiple, respectiv mai multe suite de protocoale (TCP/IP; IPX/SPX), este necesar ca driverele plăcilor de rețea să fie compatibile cu acestea.

Inițial firmele producătoare de cartele de rețea au creat drivere monolitice, pentru o suită unică de protocoale. De exemplu, firma Novell a dezvoltat drivere pentru suita IPX/SPX, care constau într-un fișier unic `IPX.com`, asociat fișierului `IPX.obj` printr-un proces logic `wsgen`.

Drivererele unice au avantajul că pot fi ușor create și îmbunătățite (*up-date*), dar sunt incompatibile cu alte suite de protocoale.

Ulterior industria de software a adoptat o nouă politică de proiectare a driverelor NIC care trebuie să suporte protocoale multiple. Această tendință este valabilă și pentru driverele altor tipuri de echipamente de rețea.

Firmele Microsoft și 3Com au elaborat în colaborare standardul pentru driverele plăcilor de rețea NDIS (*Network Driver Interface Specification*).

Firmele Novell și Apple au propus dezvoltarea **interfețelor deschise pentru legăturile de date** (*ODI - Open Data-link Interface*).

În prezent, majoritatea driverelor de NIC suportă protocoale multiple, incluzând specificațiile NDIS, cât și cele ODI.

Un driver NDIS stabilește comenzile standard pentru comunicația dintre un NOS, cu diferite suite de protocoale, și NIC, cu driverul asociat (*NDIS MAC Driver*). Gestionarea acestui proces de comunicație se face prin programul de management de protocol.

**Specificațiile NDIS** stabilesc faptul că la pornirea unui calculator bazat pe DOS, se execută fișierul `Config.sys` prin care se lansează programul de management. Acesta accesează fișierul de tip text `protocol.ini` pe baza căruia se inițializează driverele componentelor, inclusiv cele de protocol și NIC, stabilind legăturile dintre acestea (*binding*), după care creează o imagine a acestui fișier în memorie.

Orice nou driver instalat va fi comparat cu parametrii de configurare existenți și adăugat la fișierul `protocol.ini` de către agentul de management.

După stabilirea configurației corecte, se fac legăturile cu protocoalele de nivel superior prin intermediul subnivelului **vector**, inclus în nivelul legătură de date (Fig. V.3) și se salvează configurația în fișierul `protocol.ini`.

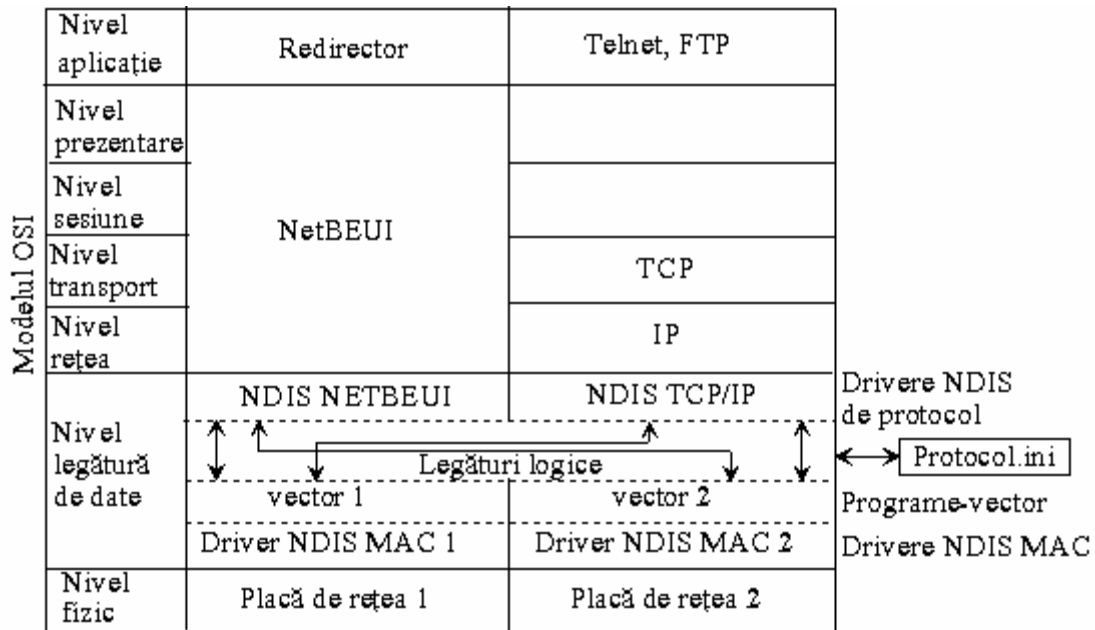


Fig. V.3 Arhitectura NDIS

Conform **specificațiilor ODI**, informațiile de configurare referitoare la drivere și protocoalele implementate în NOS sunt stocate în fișierul `net.cfg`, fiind gestionate de programul `LSL.com` corespunzător subnivelului `LSL (Link Support Layer)`, din cadrul nivelului OSI legătură de date (Fig. V.4).

Conform ODI, driverele plăcilor de rețea se numesc **drivere pentru interfețe cu legături multiple** (MLID - *MultiLink Interface Driver*). Orice driver MLID comunică prin intermediul unei interfețe cu legături multiple cu subnivelul de suport a legăturii care la rândul său comunică prin intermediul interfeței multiprotocol cu orice suită de protocoale de pe nivelele OSI superioare.

Driverele plăcilor de rețea pentru magistrale PCMCIA (*PC cards*) includ interfețele cu sistemul de operare local (OS) și cu cel de rețea (NOS) implementate pe baza specificațiilor NDIS și ODI, precum și interfața cu controlerul PCMCIA, realizată conform **specificațiilor CSS** (*Card and Socket Specification*), care prevăd două subnivele logice:

subnivelul `CS (Card Services)`, independent de caracteristicile hardware, realizează interfațarea cu OS și NOS, fiind responsabil de transmiterea mesajelor de eroare, de gestionarea resurselor și de configurarea plăcii de rețea;

subnivelul `SS (Socket Services)`, dependent de partea hardware și evident de firma producătoare, furnizează informațiile referitoare la introducerea și extragerea 'la cald', în timpul funcționării, a cartelei de rețea din portul fizic (*hot-swapability*), fără a fi necesară reconfigurarea lor.

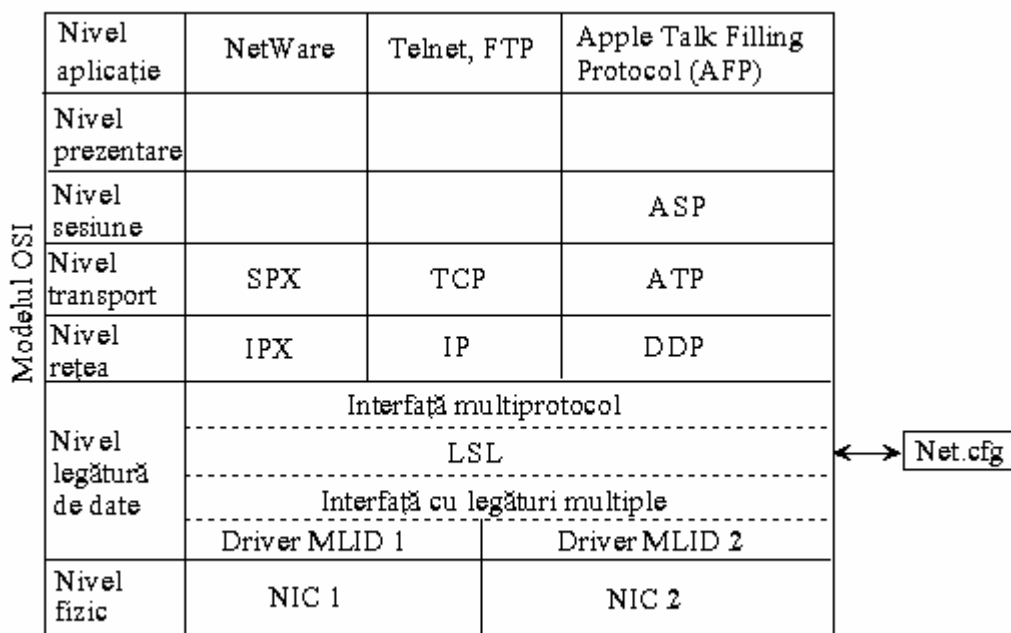


Fig. V.4 Arhitectura ODI

**Observații:**

1. Majoritatea echipamentelor de comunicații se achiziționează cu un set extins de drivere proiectate pentru diverse sisteme de operare. De asemenea, sistemele de operare au incluse arhive cu driverele pentru echipamentele diferitelor firme astfel încât în cazul celor PnP se încarcă automat driverul corespunzător.

2. În afara standardelor amintite mai-sus, există standarde și pentru driverele plăcilor de rețea utilizate în alte rețele (ATM, ISDN etc).

3. Calculatoarele-server pot fi dotate cu **plăci de rețea multiple** sau **multiport**, compatibile cu mai multe suite de protocole (MNP - *Multiple Network Protocol*). Se pune problema în acest caz ca sistemul de operare în rețea instalat pe server să poată lucra simultan cu diversele suite de protocole, adică să admită, pe principiul sistemelor deschise, comunicația simultană cu clienți având sisteme de operare de rețea distincte. În acest caz, serverul devine **router multiprotocol** sau **poartă de rețea** (*gateway*).

4. Windows NT nu admite comunicația directă dintre aplicații și componentele hardware prin intermediul driverelor clasice care operează în mod real (*real-mode driver*). Prin introducerea interfețelor de aplicație (API) pentru comunicația cu sistemul de operare și a procedurii de apel local (LPC) ca subnivel de control al accesului spre nivelul fizic, sistemul devine extrem de stabil iar driverele folosite apar ca **drivere virtuale** (VxD - *Virtual Device Driver*) față de sistemul de operare.

## V.5 WINDOWS NT

Windows NT este un sistem de operare **multistrat** scris în limbajul C, incluzând OS, NOS și GUI, **portabil** în sensul că poate fi executat cu diverse procesoare produse de diferite firme (Intel, AMD, RISC, DEC Alpha etc) și **scalabil** întrucât suportă procesarea multiplă simetrică (SMP - *Symmetric Multiprocessing*). Totodată, WinNT admite accesul simultan al mai multor utilizatori (*multiuser*), rularea în paralel a mai multor aplicații (*preemptive multitasking*), a mai multor subrutine (*multithreading*), fiind un sistem stabil, sigur, cu capacități de procesare deosebite referitor la operații complexe de gestionare a bazelor de date, a sistemului de fișiere și a altor servicii.

Arhitectura sistemului WinNT (fig.V.5) include **executorul NT** (NT *Executive*), lucrând în modul kernel, și setul de **aplicații de tip client/server**, care se execută în modul de utilizator.

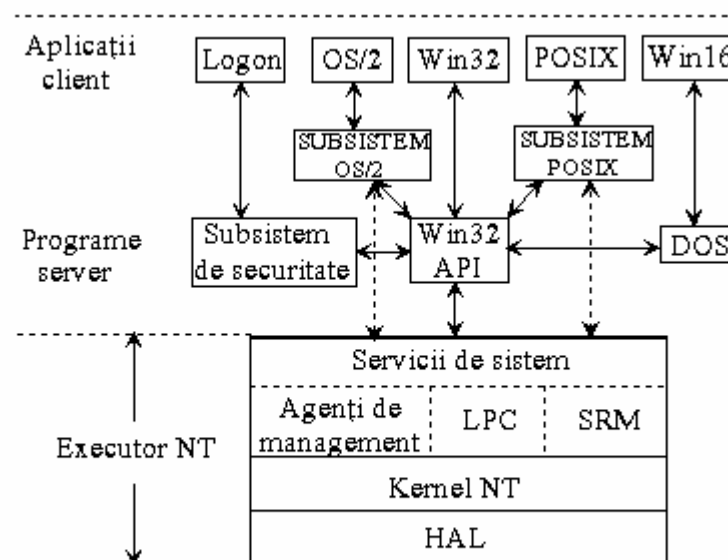


Fig. V.5 Arhitectura sistemului Windows NT

**Executorul NT** include kernelul NT și subsistemele de servicii de sistem:

- agenții de gestionare a obiectelor, proceselor, memoriei virtuale și sistemului de intrări-ieșiri (incluzând driverele pentru memoria cache, sistemul de fișiere și alte componente);
- procedura de apel local (LPC - *Local Procedure Call*);
- subsistemul de monitorizare a securității (SRM - *Security Reference Monitor*).

Un rol important în creșterea performanțelor Windows-ului NT față de alte OS îl are sistemul de management al memoriei cache, care minimizează numărul de operații de citire/scriere din și în memorie.

Pentru a permite aplicațiilor să ruleze cu o capacitate mai mare de memorie decât cea instalată, WinNT poate aloca, prin intermediul sistemului de gestiune a memoriei virtuale, zone de memorie de pe HDD (*swap memory*) de până la 4GB pentru fiecare aplicație. WinNT protejează zonele de memorie alocate diverselor aplicații.

WinNT suportă diverse sisteme de fișiere, cu drivere specifice:

- NTFS - exclusiv pentru aplicații NT;
- FAT - numai pentru diskete;
- CDFS (*Compact Disk File System*) - pentru CD-uri;
- HPFS (*High-Performance File System*) - pentru aplicații OS/2.

Windows NT folosește pentru fiecare fișier, sistemul de fișiere adecvat.

Sistemul FAT creat pentru DOS utilizează convenția 8.3 pentru numele de fișiere.

În format NTFS, lungimea maximă a numelor de fișiere este de 255 de caractere. De aceea, transferul fișierelor din sistem NTFS în sistem FAT/DOS impune conversia formatului numelui (de exemplu, un fișier denumit abcdefghij.doc va fi numit pe partiția FAT: abcdef~1.doc). Pot fi create partiții FAT și NTFS pe același HDD. Sistemul FAT pentru Windows NT admite stocarea fișierelor pe diskete cu nume de lungime maximă de 255 de caractere.

HPFS suportă nume de fișiere cu cel mult 255 de caractere, folosește o structură de directoare în formă de 'arbore' și minimizează fragmentarea fișierelor stocate pe HDD.

NTFS preia atributele sistemelor FAT și HPFS dar introduce și caracteristici noi:

- controlul accesului se poate face atât la nivel de director, cât și la nivelul fișierelor;
- utilizează tabele 'master' de fișiere (MFT - *Master File Table*), cu caracter redundant față de NTFS, în care sunt stocate atributele tuturor fișierelor iar fișierele de mici dimensiuni (sub 1500 de octeți) sunt în întregime stocate în MFT;
- admite compatibilitatea cu sistemul POSIX pentru UNIX prin respectarea caracteristicilor fișierelor create în UNIX: diferențierea literelor mari de cele mici (*case-sensitive*), stabilirea atributelor temporale de creare și accesare a fișierelor etc.
- permite detecția sectoarelor de disc defecte (*bad sectors*), marcarea lor automată și mutarea fișierelor în alte zone ale memoriei, fără intervenția utilizatorului;
- refacerea rapidă a sistemului de operare în cazul unei funcționări neadecvate.

**Kernelul NT**, prin mecanismul de întreruperi și excepții, realizează transferul mesajelor dintre sistemul de operare și partea hardware, prin intermediul **nivelului de abstractizare fizică**

(HAL - *Hardware Abstraction Layer*), echivalent sistemului BIOS utilizat de DOS. Refacerea sistemului și securizarea cad de asemenea în sarcina kernelului NT.

Pe nivelul superior, apar procese de comunicație între **aplicațiile-client** și cele de tip **server**:

- procesul de accesare a sistemului de către un utilizator NT (*logon process*) și subsistemul de securitate;
- aplicații proiectate pentru alte sisteme de operare (OS/2, Win32, Win16, Posix) și interfețele de aplicații specifice.

**Portabilitatea** Windows-ului NT se realizează prin compilarea codului-sursă cu ajutorul compilatoarelor C din fiecare sistem, compatibile între ele, în vederea execuției cu diverse procesoare având nivele HAL adaptate acestora.

Tot în limbaj C sunt scrise și driverele pentru componentele hardware (mouse, tastatură, imprimantă etc) și cele pentru protocoalele de comunicație în rețea (NetBIOS, redirectoare NCP și SMB, TCP/IP, NetBEUI, IPX/SPX). Specificațiile NDIS pentru driverele plăcilor de rețea simple și multiport prevăd posibilitatea utilizării protocoalelor multiple.

**Accesul rapid** la fișiere și **siguranța** de funcționare (*reliability*) a sistemului WinNT sunt date de redundanța tabelelor de fișiere MFT, primar și secundar sau 'oglină' (*MFT Mirror*), stocate în sectorul de BOOT primar, respectiv în sectorul de BOOT de rezervă (*Duplicate BOOT Sector*).

Un avantaj major al NTFS este dat de rapiditatea de refacere a sistemului de fișiere (*File System Recoverability*) în cazul unor defecte majore (*Crash*). Orice tranzacție cu fișiere este monitorizată și periodic se notează starea acesteia, în așa-numitele 'puncte de verificare' (*checkpoints*). În cazul distrugerii sistemului de fișiere, NTFS reia tranzacțiile din aceste puncte, fie continuându-le, fie anulând efectele acestora (*undo*), astfel ca starea finală a sistemului să fie corectă, fără erori. Recuperarea NTFS se face în doar câteva secunde.

## V.5.1 Domenii NT

Informațiile despre utilizatorii și resursele sistemului WinNT nu sunt incluse într-o bază de date unică, ci în mai multe baze de date distribuite în rețea, guvernate însă de un singur sistem de securitate, pentru autorizarea accesului din orice nod al rețelei.

Rețelele NT sunt structurate pe **domenii** incluzând mai multe servere Windows NT subordonate subsistemului de securitate care controlează accesul la un domeniu.

Informațiile referitoare la fiecare domeniu sunt stocate pe serverul NT care oferă **serviciile de domeniu**.

Un singur server din rețeaua NT devine **controlerul primar al domeniului** (*Active/Primary Domain Controller*). Celelalte servere din domeniu pot să reprezinte doar rezerve ale acestuia sau să aibă cu totul alte funcții (servere de aplicații; servere care oferă servicii de rețea etc).

Calculatoarele de tip client WinNT pot forma **grupuri de lucru** (*workgroups*) care diferă de domeniile NT prin aceea că nu dispun de un sistem unic de securitate, fiecare calculator fiind responsabil de aplicarea politicilor de securitate la nivelul său.

Într-un domeniu oricare controler, primar sau secundar, poate autoriza accesul unui utilizator în rețea, în timp ce în grupurile de lucru, pe fiecare calculator se stabilesc și se păstrează **listele de control al accesului** (ACL - *Access Control List*), cu conturile de utilizator create pentru acea stație.

Într-o rețea NT, alcătuită din mai multe domenii, trebuie stabilit modul de autorizare a accesului dintr-un domeniu în altul, pentru realizarea comunicației între domenii, pe baza unor **relații de încredere**:

1. **exclusivă** (*Trust Relationship*), numai între două domenii, pe baza procedurii de extindere a autentificării (*passthrough authentication*) dintr-un domeniu într-altul, astfel încât un utilizator admis într-un domeniu poate accesa și resursele celuilalt domeniu. Aceste relații sunt restrictive în cazul a trei sau mai multe domenii întrucât dacă între domeniile A și B s-a stabilit o relație de încredere exclusivă, și similar între domeniile B și C, aceasta nu permite utilizatorilor din domeniul A să acceseze resursele domeniului C.

2. **totală** (*Interdomain trust accounts*), între toate domeniile definite în rețea. În acest caz, un controler de domeniu poate permite accesul unui utilizator în oricare alt domeniu.

3. **între stații și servere** (*Workstation trust accounts*), astfel ca autentificarea în vederea accesului într-un domeniu, să se poată realiza și la nivelul stației, fără intervenția serverului cu rol de controler de domeniu.

4. **între servere** (*Server trust accounts*), pe baza acestora realizându-se copiile de siguranță ale bazelor de date stocate pe controlerul primar de domeniu pe serverele secundare (*Backup Domain Controller*).

Arhitectura rețelei NT poate fi proiectată folosind diverse **modele de domenii** (Fig.V.6):

- **domeniul unic** - această arhitectură admite cel mult 10000 de utilizatori și un singur sistem de securitate (SAM - *Security Account Manager*).
- **domeniile multiple independente** - resursele domeniilor sunt total separate și protejate, gestionate independent, între domenii nefiind stabilite relații de încredere.
- **domeniul "master" unic** - permite implementarea ierarhică a rețelei NT, cu un domeniu principal (*master*) și mai multe domenii subordonate. Conturile de utilizator



(maxim 10 000) sunt definite în domeniul principal și gestionarea resurselor se face centralizat, cu un singur agent de management. Accesul din domeniul master în domeniile subordonate este permis pe baza relațiilor de încredere dar nu este admis accesul dinspre subdomenii în domeniul principal.

- **domenii "master" multiple** - această arhitectură extinsă admite până la 10 000 de utilizatori pentru fiecare domeniu master, care deține încrederea domeniilor subordonate lui.
- **domeniile multiple dependente** - reprezintă o arhitectură ideală, cu relații de încredere totală stabilite între oricare două domenii.

În funcție de natura activităților desfășurate în rețea, se alege arhitectura NT potrivită.

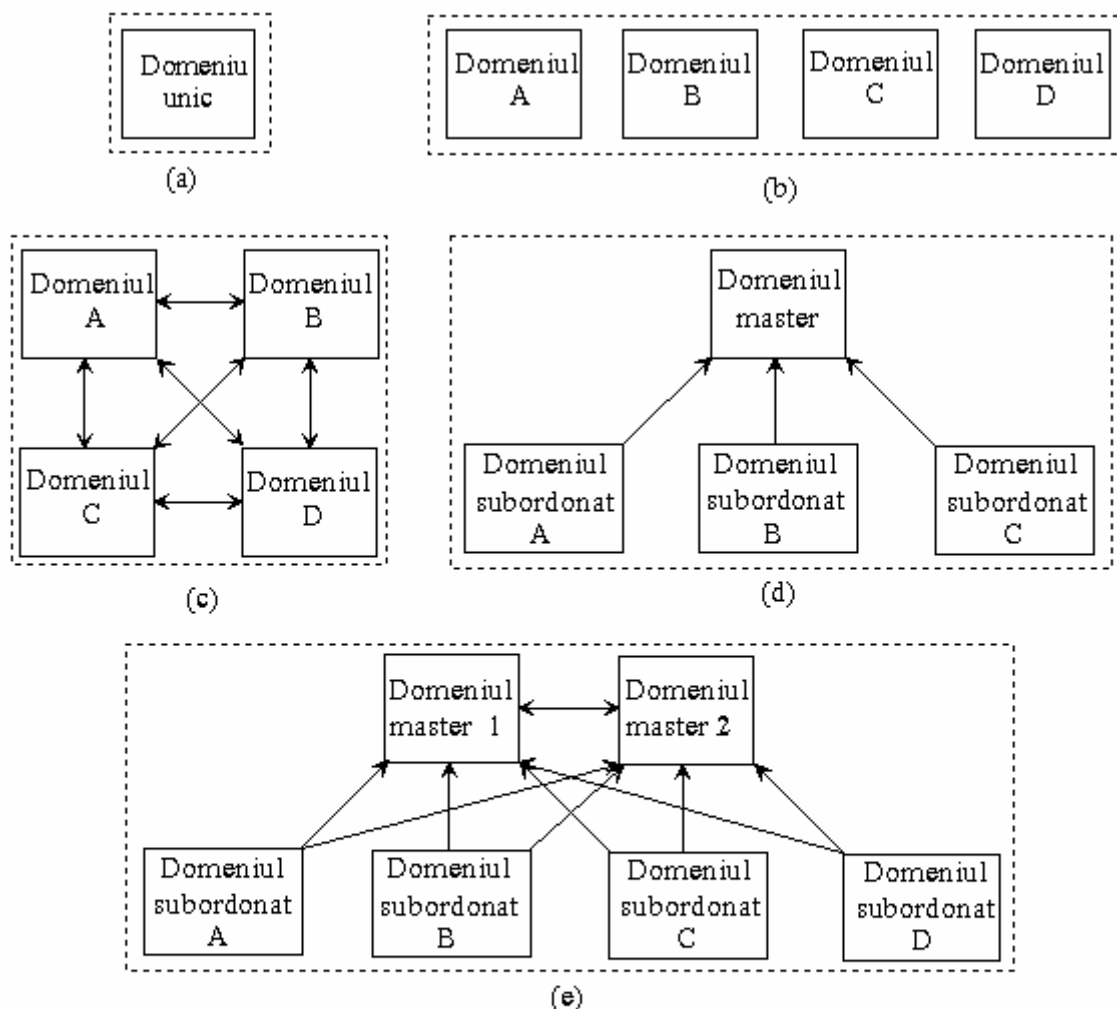


Fig. V.6 Arhitecturi NT: (a) cu domeniu unic;  
 (b) cu domenii multiple independente;  
 (c) cu domenii multiple dependente;  
 (d) cu domeniu master unic;  
 (e) cu mai multe domenii master.

## V.5.2 Servicii NT

Windows NT furnizează diverse servicii de rețea, pe baza unei arhitecturi de protocoale stratificate conform modelului de rețea OSI (Fig. V.7).

Placa de rețea cu driver de tip NDIS poate comunica cu mai multe suite de protocoale: TCP/IP, IPX/SPX, NetBEUI, AppleTalk, DLC (*Data Link Control*).

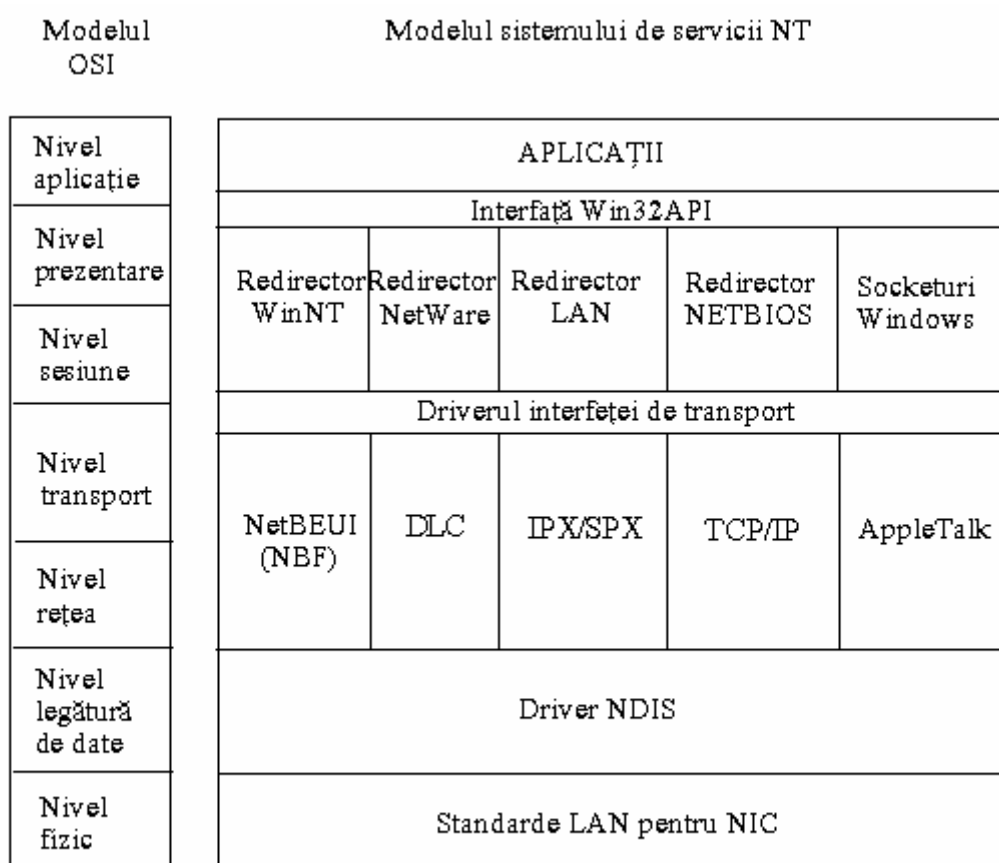


Fig.V.7 Sistemul serviciilor de rețea NT

Pe baza modelului client/server, Windows NT poate lucra simultan cu protocoale definite pentru diverse NOS:

- NBF (*NetBEUI Frame*) este versiunea NT a NetBEUI pentru sisteme OS/2 și Microsoft LAN Manager;
- DLC pentru comunicația cu imprimante de rețea (de exemplu, Hewlett-Packard LaserJet), calculatoare de tip "mainframe" IBM și servere-gateway Microsoft SNA (*Service Network Architecture*) pentru Windows NT;

- IPX/SPX sau NWLink pentru clienți și servere NetWare;
- TCP/IP pentru rețele de PC-uri bazate pe WinNT, cu facilități de configurare pentru DHCP și servere WINS;
- AppleTalk pentru clienți și servere Macintosh.

**Interfața de transport** asigură transparența comunicării între diversele protocoale de transport și redirectoarele admise de WinNT, cu diferite **opțiuni de comunicare între procese**:

- **WinSock** (rulat din fișierul WinSock.DLL) este cel mai flexibil mecanism de comunicare interprocese pentru aplicații distribuite între clienți și servere sub Windows NT. Versiunea WinSock2 are suport multiprotocol (IPX/SPX, AppleTalk, DECnet, OSI) și lucrează cu două interfețe, una pentru aplicații, WinSock API, și alta pentru selectarea protocolului de transport, WinSock SPI (*Service Provider Interface*).

- **NetBIOS** este interfața API (implementată în fișierul NetBIOS.DLL) pentru comunicații între procese, în cazul aplicațiilor distribuite, cu redirector NetBIOS și suita de protocoale NBF. Mesajele dintre clienți și servere se realizează în format SMB.

- **RPC** (*Remote Procedure Call*) este un mecanism superior de comunicare între procese, definit de DCE (*Distributed Computing Environment*), pentru rularea aplicațiilor distribuite în rețea, pe mai multe calculatoare. RPC poate utiliza pentru anumite aplicații interfețele WinSock și NetBIOS. Serviciul RPC este oferit de modulul RPC inclus în sistemul de operare.

Pentru compatibilitatea cu sistemul OS/2, Windows NT include și alte mecanisme de comunicare între procese (*Named Pipes, Mail Slot*).

Comunicațiile între procese, indiferent de tipul de interfață utilizat, se pot face fie în mod sincron, orientat pe conexiune, fie asincron fără conexiune.

Orice comunicație între procese se realizează cu permisiunea sistemului de securitate NT, care pe de o parte autorizează accesul utilizatorilor la resursele locale și de rețea, în funcție de drepturile acordate fiecăruia în mod diferențiat, și pe de altă parte supervizează schimburile de informații dintre procese pentru a nu se încălca în nici un fel politica de securitate.

Windows NT folosește un sistem de monitorizare a aplicării regulilor de securitate (SRM), care lucrează în modul kernel. Acesta urmărește respectarea nivelelor de acces ale utilizatorilor, cu diverse permisiuni: de citire, scriere sau ștergere de fișiere, execuție de programe, schimbarea drepturilor sau crearea conturilor de utilizator, interzicerea parțială sau totală a accesului la resursele sistemului. Conturile de utilizator și parametrii de configurare a acestora sunt stocate într-o bază de date care se păstrează în controlerul primar de domeniu. Copii ale acestora pot exista pe serverele NT de backup.

**Observații:**

1. Interoperabilitatea dintre Windows NT și alte sisteme de operare implică soluționarea problemelor de compatibilitate pe toate nivelele modelului OSI. La nivele inferioare, această problemă este soluționată pe baza standardelor pentru plăcile de rețea și driverele acestora, specificațiile NDIS și ODI permițând utilizarea mai multor suite de protocoale pe nivelele de rețea și de transport. Pe nivelele OSI superioare, în funcție de sistemul de fișiere utilizat de client și de server (NTFS, NetWare etc), compatibilitatea poate fi realizată fie direct cu redirecționare NCP sau SMB care stabilesc formatul acceptat pentru comunicația între procese, fie indirect cu module software (*shell*) Netware create special pentru acest tip de rețea, redirector NCP și emulator NetBIOS pentru conversia de format. În cazul comunicației NT/Unix, standardul X Windows face posibilă comunicația client-server având sisteme de operare distincte, folosind protocolul Telnet.

2. Accesul de la distanță într-un sistem NT (RAS - *Remote Access Service*) reprezintă un serviciu NT furnizat prin module software specializate, de tip client și de tip server.

Modulul RAS server se execută pe serverul NT și este responsabil de **autentificarea** accesului de la distanță a utilizatorului, eventual cu parolă criptată (*encrypted authentication*), și supravegherea sesiunilor de comunicație create de acesta. Acest modul permite maximum 256 de conexiuni simultane stabilite de la distanță. Protocoalele de autentificare utilizate de RAS sunt:

- CHAP (*Challenge Handshake Authentication Protocol*) cel mai sigur protocol, care criptează atât parola cât și datele transmise într-o sesiune.
- SPAP (*Shiva Password Authentication Protocol*) criptează numai parola de acces a utilizatorului, fiind obligatorie aplicarea lui atât la client cât și pe server.
- PAP (*Password Authentication Protocol*) autorizează accesul pe baza unei parole de utilizator, transmisă necodat (*clear-text*).

Accesul la serverul NT se poate face fie printr-o legătură telefonică (*dial-up*), fie în sistem ISDN sau X.25.

Modulul RAS client poate opera sub DOS, Windows NT precum și alte versiuni Windows, cu diferite suite de protocoale (TCP/IP, IPX/SPX, NBF) și diverse protocoale pe nivelul legătură de date (NDIS, PPP).

## V.6 UNIX

UNIX este un sistem de operare modular, structurat pe două nivele (Fig.V.8):

1. **kernelul** UNIX include modulele de gestionare a resurselor (CPU, HDD, memorie instalată, memorie virtuală), sistemul de fișiere, driverele de componente, modulul de securitate,

sistemul de intrări/ieșiri (I/O - *Input/Output*), modulul de control a proceselor, sistemul de întreruperi;

2. **programele** de sistem UNIX pot fi: compilatoare, interpretoare, aplicații, programe utilitare, biblioteci de program, procese ale utilizatorului (*shell*).

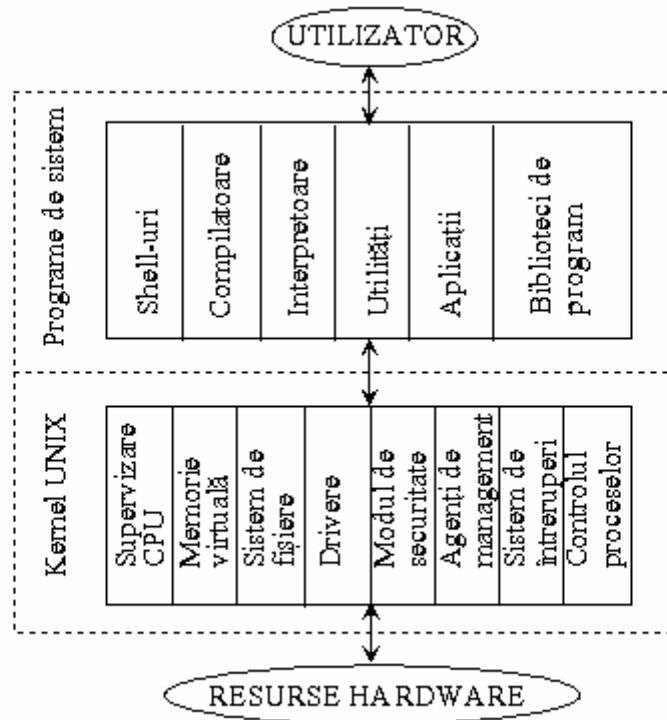


Fig.V.8 Arhitectura sistemului de operare UNIX

Sistemul UNIX nu conține modulele software pentru comunicația în rețea, pentru aceasta fiind utilizată suita de protocoale TCP/IP combinată cu sistemul de fișiere în rețea NFS (*Network File System*), împreună alcătuind sistemul de operare de rețea NOS (Fig. V.9).

S-au dezvoltat diferite variante de UNIX, în diverse limbaje de programare (în principal, C), implementate pentru diferite sisteme hardware, în prezent fiind dezvoltat un standard unic (*Spec 1170 APIs*), reprezentând o colecție de peste 1000 de interfețe de aplicații (API), care poate să suporte toate versiunile UNIX. În paralel, s-a proiectat o interfață grafică de utilizator CDE unică pentru aplicațiile dezvoltate în orice sistem UNIX.

UNIX a fost conceput ca **sistem modular**, în care se pot introduce noi module software, fie în kernel, fie în grupul fișierelor de sistem.

Compilarea codului-sursă la nivelul fiecărui sistem hardware conferă UNIX-ului caracteristica de **portabilitate**, între diverse echipamente hardware precum și între diferitele versiuni de UNIX.

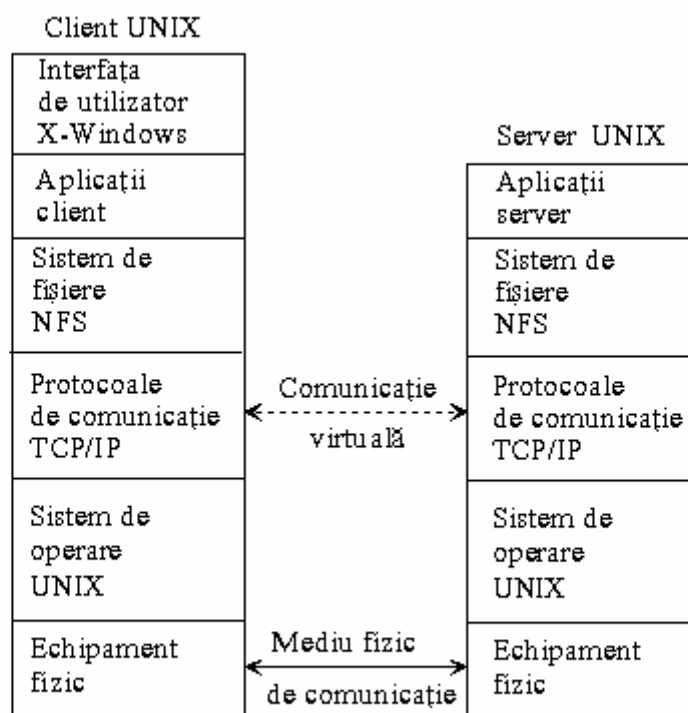


Fig. V.9 Modelele structurate ale sistemelor client și server UNIX

Aplicațiile-client sunt cele care comunică direct cu utilizatorul (*Front-End Client Application*). Cele de tip server rulează în mod transparent față de clienți (*Back-End Server Application*).

Asemenea WinNT, UNIX admite utilizatori multipli, rularea simultană a mai multor aplicații în modul ierarhic, instalarea în sistem cu mai multe procesoare (*multiuser, pre-emptive multitasking, multithreaded, multiprocessing OS*).

Accesul la un sistem UNIX se face pe baza numelui și parolei de utilizator, eventual a numelui stației dacă sistemul este accesat de la distanță (*remote access*).

Spre deosebire de alte sisteme de operare, UNIX este **dinamic** în sensul că orice utilizator poate să intervină în structura sa și să-și creeze propriile aplicații.

Pentru a crea noi aplicații UNIX, fiecare utilizator poate să scrie module software (*shell script*) în diverse limbaje de programare: C, Turbo-C, PERL (*Practical Extraction and Reporting Language*) etc. Interfața utilizatorului cu sistemul este realizată prin programele de interpretare a comenzilor.

Există mai multe tipuri de shell-uri UNIX (cu prompteri diferiți), cele mai utilizate fiind:

- Shell-uri Bourne (*bash*)
- Shell-uri C (*cs*h)
- Shell-uri Korn (*ksh* - combinație între *cs*h și *bash*).

Editarea acestor shelluri (*scripting*) se poate face cu diverse editoare de text pentru UNIX, de exemplu, TCL&TK. TCL (*Tool Command Language*) este un limbaj de scriere și interpretare a comenzilor, utilizat pentru realizarea comunicației între aplicații dezvoltate în diverse medii.

Sistemul de fișiere utilizat de UNIX este structurat ierarhic sub forma unui 'arbore', cu mai multe nivele și un nod 'rădăcină' (Fig.V.10). Fișierele sunt interpretate ca șiruri de octeți și sunt grupate în directoare, directorul 'rădăcină' fiind notat simplu cu caracterul '/'.

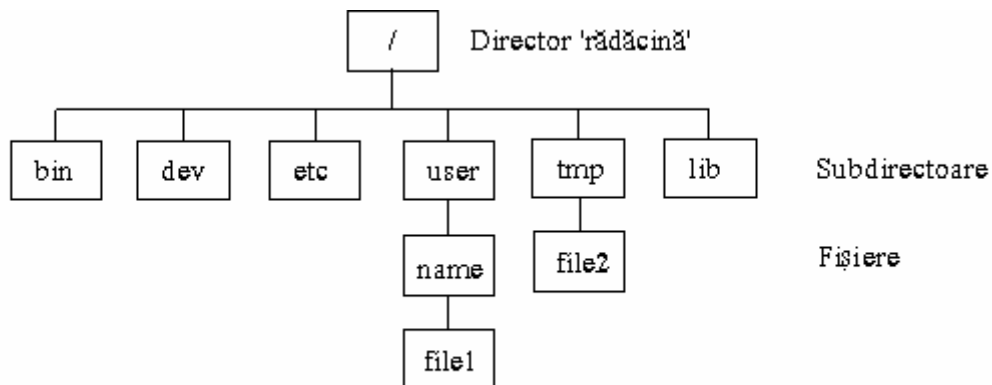


Fig.V.10 Structura sistemului de fișiere UNIX

Accesarea fișierelor și directoarelor se poate face exprimând fie calea absolută pornind din nodul 'rădăcină', fie calea relativă la un alt nod (Anexa B).

De exemplu, calea absolută care identifică fișierul `file1` este: `/user/name/file1` iar calea relativă la directorul curent este `name/file1`.

UNIX suportă existența mai multor sisteme de fișiere pe același disc, permite alocarea de memorie virtuală aplicațiilor (*swap memory*) și stocarea driverelor în afara kernelului.

Ca mecanism de comunicare între procese, UNIX folosește așa-numitele **socluri** (*socket*), asociate cu adresele zonelor de memorie alocate diverselor procese. TCP folosește socketuri de tip "șir" (*stream socket*), UDP lucrează cu socketuri de tip "datagramă" iar în cazul comunicației între două procese de pe nivele OSI diferite se implementează socketuri "brute" (*raw socket*) care trebuie să realizeze și funcțiile altor nivele OSI, necesare în procesul de comunicație.