

# **FAMILIA DE SISTEME DE OPERARE WINDOWS**

**Evoluția sistemelor Windows**

**Clasificarea sistemelor Windows**

**Interfața sistemului cu utilizatorul**

**Caracteristici generale ale sistemelor  
Windows**

**Nivelul de abstractizare a hardware**

**Nivelul nucleu**

**Executivul Windows XP**

## Evoluția sistemelor Windows

- Primele versiuni de Windows (1.0- lansată în 1985, 2.0- lansată în 1987, 3.0- lansată în 1990 și cele care au urmat-o 3.1 și 3.11), au fost lansate de Microsoft ca interfețe grafice(GUI), care se utilizau peste sistemul de operare MS-DOS.
- În 1993, apare Windows NT (3.1), prima versiune de sistem de operare Windows, care este un sistem de operare pentru calculatoare pe 32 de biți, care suportă aplicațiile scrise DOS și are aceeași interfață cu Windows 3.1.
- În 1995, sunt lansate versiunile Windows NT 4.0 și Windows 95.
- - Windows 95 a reprezentat o variantă îmbunătățită a Windows 3.1.
- - Windows NT 4.0 are aceeași arhitectura internă ca și versiunile Windows 3.x și furnizează aceeași interfață utilizator ca și Windows 95.
- - Schimbarea arhitecturală majoră este că mai multe componente grafice care se executau în mod utilizator, ca și parte a subsistemului Win32 (în versiunile 3.x), au fost mutate în executivul Windows NT, care se execută în modul nucleu, ceea ce are ca avantaj creșterea vitezei de operare al acestor funcții importante. Un potențial pericol este că aceste funcții grafice se execută accesând serviciile de nivel scăzut ale sistemului, ceea ce poate afecta fiabilitatea sistemului de operare.

- Windows 98 a fost succesorul lui Windows 95.
- - Versiunea 5.0 a NT a fost lansată în anul 2000 și fost redenumită Windows 2000(Win2K).
- - Arhitecturile executivului și nucleului sunt cam aceleași cu cele ale NT 4.0, dar au fost adăugate câteva componente importante.
- - În Windows 2000 s-au adăugat servicii și funcții ca suport al prelucrării distribuite.
- - Componenta esențială introdusă în acest scop este Active Directory; acesta este un serviciu de directoare distribuit (colecție de informații despre obiecte care sunt în legătură unele cu altele într-o anumită privință ), utilizat pentru:
  - ► a organiza și simplifica accesul la resursele unei rețele de calculatoare;
  - ► aplicarea securității pentru a proteja obiectele din cadrul rețelei față de intrușii exteriori sau utilizatorii interni;
  - ► distribuirea resurselor directorului la calculatoarele din rețea;
  - ► replicarea directorului pentru a o face disponibilă la mai mulți utilizatori;
  - ► separarea directorului în mai multe bucăți care sunt stocate pe diferite calculatoare din rețea.
- - Windows 2000 a îmbunătățit facilitățile de “plug-and-play” și “power-management” (importanta pentru laptop-uri).

- - Sub Windows 2000 se face distincție între Windows 2000 Server și Windows 2000 desktop; arhitecturile și serviciile nucleului și executivului sunt aceleași, dar Server include anumite servicii necesare utilizării serverelor de rețea, cum sunt cele de securitate.
- - Au existat patru versiuni ale sistemului Windows 2000:
  - ► versiunea Profesional este destinată calculatoarelor individuale;
  - ► celelalte trei versiuni(Server, Advanced Server și DataCenter Server) sunt destinate calculatoarelor server dintr-o rețea.
  - ► Windows 2000 DataCenter Server este destinat serverelor multi-procesor și poate fi instalat pe sisteme care au până la 32 de procesoare și până la 64 G de memorie.
- Windows 95/98 și Windows ME(Milenium Edition) formează sub-familia sistemelor Windows 9x, care este destinată calculatoarelor individuale, neconectate în rețea.
- Windows ME este o variantă îmbunătățită a lui Windows 98, care la rândul lui perfecționează funcțiile lui Windows 95.
- Windows 9x/ME diferă de Windows NT/2k prin faptul că implementează mai puține funcții Win32API.
- Windows NT/2K conține un model de securitate care nu este întâlnit la sistemele Windows 9x/ME, deoarece Windows NT/2K este destinat calculatoarelor care sunt conectate în rețea.
- Facilitățile de conectare conținute de Windows 9x/ME sunt o sub-mulțime a celor conținute de Windows NT/2K.
- O altă deosebire constă în faptul că Windows NT permite aplicațiilor să manipuleze anumiți parametri, care influențează comportarea administratorului memoriei virtuale.

- Windows CE(Consumer Electronics) este destinat calculatoarelor personale care utilizează anumite echipamente electronice casnice, cum ar fi televizoare; ele pot fi legate la rețelele de televiziune prin cablu. Setul său de funcții Win32API este cel mai redus, fiind orientat mai ales către jocuri.
- În 2001 a fost lansată o nouă distribuție desktop - Windows XP, cu versiuni pe calculatoare pe 32 sau 64 de biți.
- Fiind succesorul lui Windows 2000, el poate fi utilizat ca sistem de operare pentru calculatoarele legate în rețea, dar poate înlocui și sistemele Windows 95/98/ME destinate calculatoarelor individuale.
- Windows XP propune o interfață grafică nouă, care se bazează pe evoluțiile componentelor hardware și urmărește utilizarea facilă a interfeței sistemului de către utilizatori.
- Modelul de securitate adoptat este superior tuturor versiunilor de Windows anterioare.
- De asemenea, sistemul oferă facilități superioare de lucru în rețea.

- Versiunea XP propune două variante:
- - Windows XP Net Server este utilizat de calculatoarele server dintr-o rețea; această variantă oferă o serie de facilități și mecanisme noi, printre care: lansarea simultană de mai multe sesiuni de pe calculatoarele din rețea, posibilitatea ca utilizatorul să poată comuta rapid printre serviciile oferite de sistem.
- - Windows XP Professional este destinat calculatoarelor individuale precum și calculatoarelor client legate în rețea.
- În 2003, a apărut o nouă versiune server - Windows Server 2003, care suportă procesoare atât pe 32 de biți, cât și pe 64 de biți.
- În 2007, a fost lansat Windows Vista, care suporta arhitecturile procesoarelor existente. S-au adus îmbunătățiri ale modelului de securitate.
- Windows Server 2008 este varianta coresp. care se instalează pe servere.

- In 2007 apare Windows Vista.
- **Securitatea** reprezinta telul Microsoft pentru acest nou sistem de operare.
- - noi facilitati de securitate care fac mai dificila expunerea sistemului de operare la atacurile informatice. Acestea avertizeaza utilizatorul asupra oricarei activitati suspecte si cer confirmarea inainte de acordarea accesului la sistem.
- - sistemul contine o aplicatie care va proteja computerul fata de amenintarile informatice care pot determina publicarea de informatii confidentiale in exterior cu sau fara voia utilizatorului; va trebui procurata separat o aplicatie antivirus, pentru ca noul sistem de operare nu contine o protectie si la aceste atacuri, de exemplu, „One Care”, bazata pe tehnologie romaneasca, urmare a achizitiei companiei romanesti Gecad de catre Microsoft in 2003.
- - Protectia stabilitatii sistemului este imbunatatita, astfel incat noul sistem de operare are disponibile facilitati ce permit imbunatatirea continuitatii operarii prin restaurarea starii sistemului la un punct anterior in timp, astfel incat dupa instalarea software-ului pentru o componenta hardware sau a unei aplicatii care duce la blocarea sau instabilitatea sistemului se poate reveni la starea initiala.
- - Protectia informatiilor stocate a facut un pas inainte:
- ► sist. este capabil sa recreeze varianta anterioara pentru anumite fisiere si directoare;
- ► software-ul pentru realizarea de copii de siguranta a informatiilor stocate a fost imbunatatit, putand realiza salvari periodice mai usor si cu mai multe facilitati.

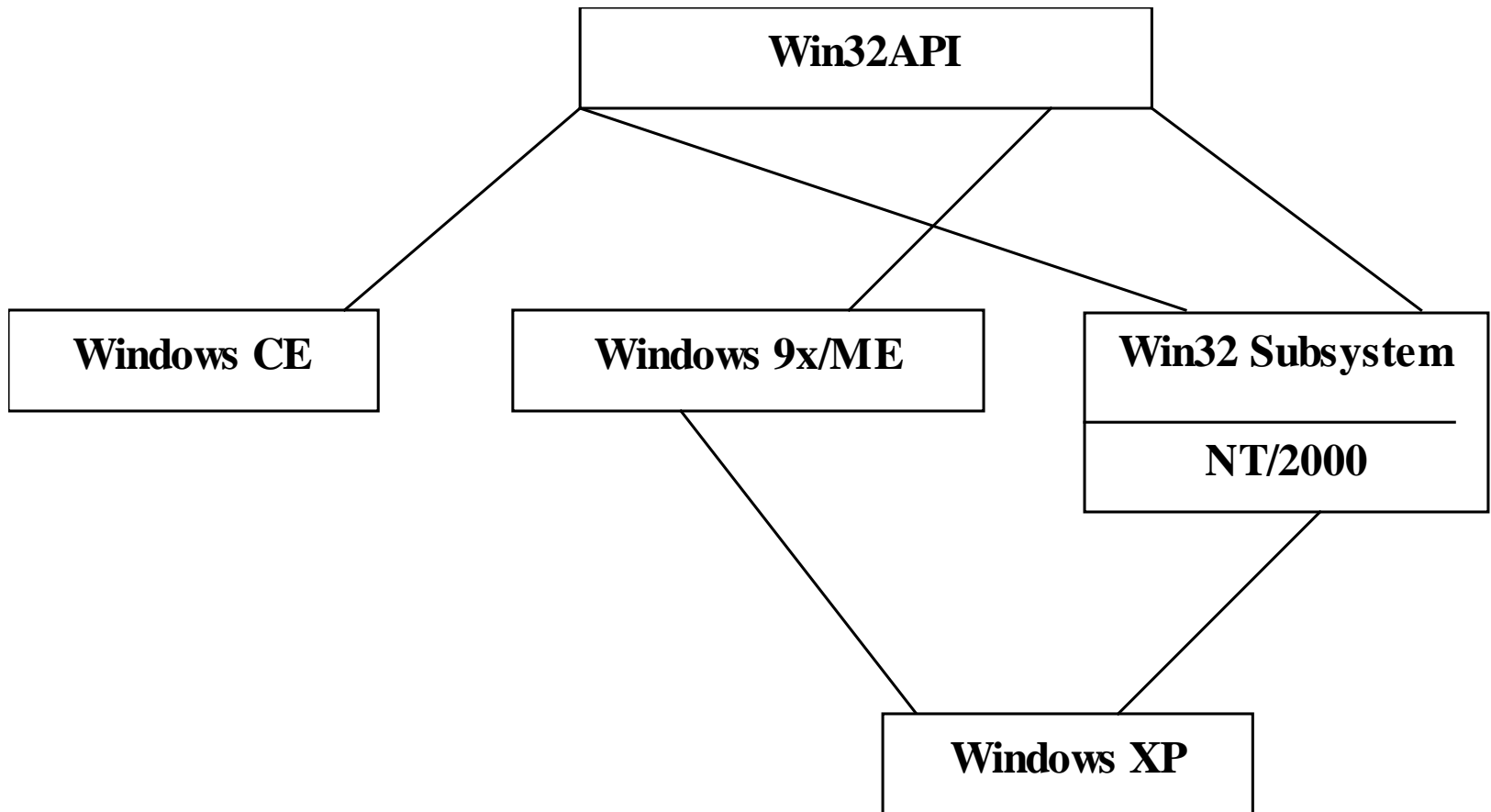
- **Performanta.** Din acest punct de vedere, testele efectuate au aratat ca Windows Vista ruleaza la fel sau mai lent comparativ cu Windows XP. Noile optiuni adaugate din punct de vedere al functionalitatii sau al securitatii necesita mai multa putere de calcul si memorie mai multa.
- **Comunicatii.** La acest capitol, lucrurile au fost simplificate si imbunatatite. Este mult mai simplu sa conectezi un computer la retea, locala sau Internet, indiferent de tipul conexiunii, prin cablu sau fara fir si este mult mai sigur; Microsoft a adaugat standarde mult mai sigure pentru transferurile de informatie.
- **Interfata grafica 3D.** Aceasta foloseste la maximum noile placi video - fara de care este inutila - si ofera o noua imagine asupra traditionalului desktop, in 3 dimensiuni.  
**Aplicatii noi. Exemple** „Sidebar”, care, instalata pe desktop, furnizeaza informatii despre ora exacta, vreme sau poate afisa ultimele stiri, sau „Gallery”, o aplicatie de afisare si gestionare a fotografiilor digitale.



- Printre **imbunatatirile aduse de Microsoft in viitoarea versiune Windows 7** (lansat in 2009) se numara:
  - un taskbar imbunatatit;
  - noi moduri de manipulare si organizare a ferestrelor;
  - un nou Internet Explorer, ajuns acum la versiunea 8;
  - o integrare mai buna cu serviciile online Windows Live;
  - un management al perifericelor si al dispozitivelor conectabile la PC mult mai bine pus la punct;
  - wizard de conectare/creare a unei retele mult mai intuitiv;
  - o performanta mai buna comparativ cu Windows Vista;
  - optiuni noi de optimizare a consumului bateriei la [laptopuri](#);
  - mai multe optiuni de personalizare;
  - functii noi pentru sistemele dotate cu ecrane sensibile la atingere si capacitati de recunoastere a scrisului.

## Interfața sistemului cu utilizatorul

- Fiecare dintre sistemele de operare Windows importă o submulțime a unei interfețe unice, denumită Win32API. Această API (**A**pplication **P**rogramming **I**nterface) este cuprinzătoare și dinamică.
- În anii 2000, Win32API conținea aproximativ 2000 de funcții, de la cele folosite pentru crearea de procese, până la cele pentru verificarea unui criteriu de performanță.
- Diferitele versiuni Windows implementează submulțimi ale Win32API.
- Rațiunea pentru care a fost creată o singură API, este leagată de necesitatea portabilității între diferite sisteme Windows.
- Mulțimea de funcții implementate de către Windows 95/98/ME include pe cele utilizate de către Windows CE.
- Windows NT, Windows 2000 și Windows XP implementează o mulțime de funcții care le include pe cele implementate de către Windows 95/98/ME (vezi figura urm.).
- Nucleul lui Win2K și subsistemele Win32 implementează toate funcțiile Win32API, fiind cel mai complex dintre membrii familiei apărute până la acea dată.
- Win2K folosește un model de calcul bazat pe procese și fire de execuție și se sprijină pe conceptele proiectării și programării orientate pe obiecte.



## Caracteristici generale ale sistemelor Windows

- Windows înlocuiește modul de **lucru text** cu cel **grafic** și introduce, conceptual următoarele elemente:
  - permite afișarea într-o mulțime de **ferestre** organizate ierarhic;
  - textele pot fi afișate într-o mare gamă de fonturi scalabile;
  - permite interceptarea de evenimente; un eveniment este interceptat atunci când se apasă o anumită tastă sau un buton de la **mouse** (evenimentul este interceptat diferit, în funcție de poziția cursorului în momentul evenimentului);
  - oferă o serie de mecanisme de captare a unor informații: "butoane" cu răspuns unic sau "ferestre" de alegere a unuia sau mai multor răspunsuri posibile, dintr-o listă predefinită;
  - oferă posibilitatea de comunicare între programele utilizator și mediul Windows; în legătură cu această ultimă posibilitate, se introduce o nouă fază în dezvoltarea de programe și anume adăugarea de resurse Windows în aplicații. Astfel, după link-editarea programului, acesta este completat cu o serie de module, care permit programului să manevreze cu elemente Windows: **ferestre, butoane, evenimente** etc.

- **Execuția simultană a mai multor lucrări.**
- Sub Windows multi-tasking-ul este oarecum “aproximativ”, în sensul următor: O aplicație lansată poate fi suspendată temporar. După suspendare, poate fi lansată o alta sau poate fi relansată una dintre cele suspendate. Singurele aplicații efectiv executabile în paralel sunt cele oferite de DOS, cum ar fi de exemplu listarea la imprimantă în paralel cu execuția altor programe și supravegherea ceasului sistem în vederea lansării în execuție a unei lucrări la o oră fixată.
- Odată cu versiunile Windows 95/98, a apărut un mod mai ușor de a comuta între programele lansate, și anume **bara de aplicații**. Când lansați un program, Windows adaugă un buton pentru acel program la bara de aplicații. Atunci când lansați mai multe programe, Windows rearanjează automat butoanele în așa fel încât să puteți vedea tot timpul acele butoane. Butonul asociat programului activ are o luminozitate deosebită față de celelalte programe de pe bara de aplicații.
- **Schimbul de date între aplicații.** Windows gestionează o aplicație numită **ClipBoard**. Prin intermediul ei, utilizatorul poate depozita temporar, din orice aplicație, informații în zona rezervată Clipboard, care pot fi utilizate de către oricare altă aplicație.

- **Interfața grafică**(GUI-Graphics User Interface). Comunicarea între utilizator și sistem se poate face prin intermediul tastaturii sau a mouse-ului. Prin intermediul tastaturii, se poate selecta o tastă sau o combinație de taste, corespunzătoare unei anumite acțiuni iar prin intermediul mouse-ului se pot selecta unul sau mai multe „butone”, din anumite entități grafice, pe care le vom descrie pe scurt în cele ce urmează. Elementul esențial de afișare în această interfață este **fereastra**, care reprezintă o porțiune dreptunghiulară afișată într-o zonă a ecranului, destinată comunicării dintre utilizator și sistem. Sub Windows, orice fereastră conține câteva elemente comune, și anume:
  - ► **Bara de titlu** (Caption) din partea cea mai de sus a ferestrei afișează numele aplicației.
  - ► Imediat sub bara de titlu este **bara de meniuri** (Menu Bar), care afișează meniurile disponibile ( opțiunile specifice aplicației care rulează în momentul respectiv).
  - ► **Bara de unelte** (Toolbar) conține butoane și alte elemente care vă permite să lansați comenzi. Conținutul barei de unelte variază de la fereastră la fereastră, depinzând de fereastră sau aplicație, dar poziția ei este constantă.

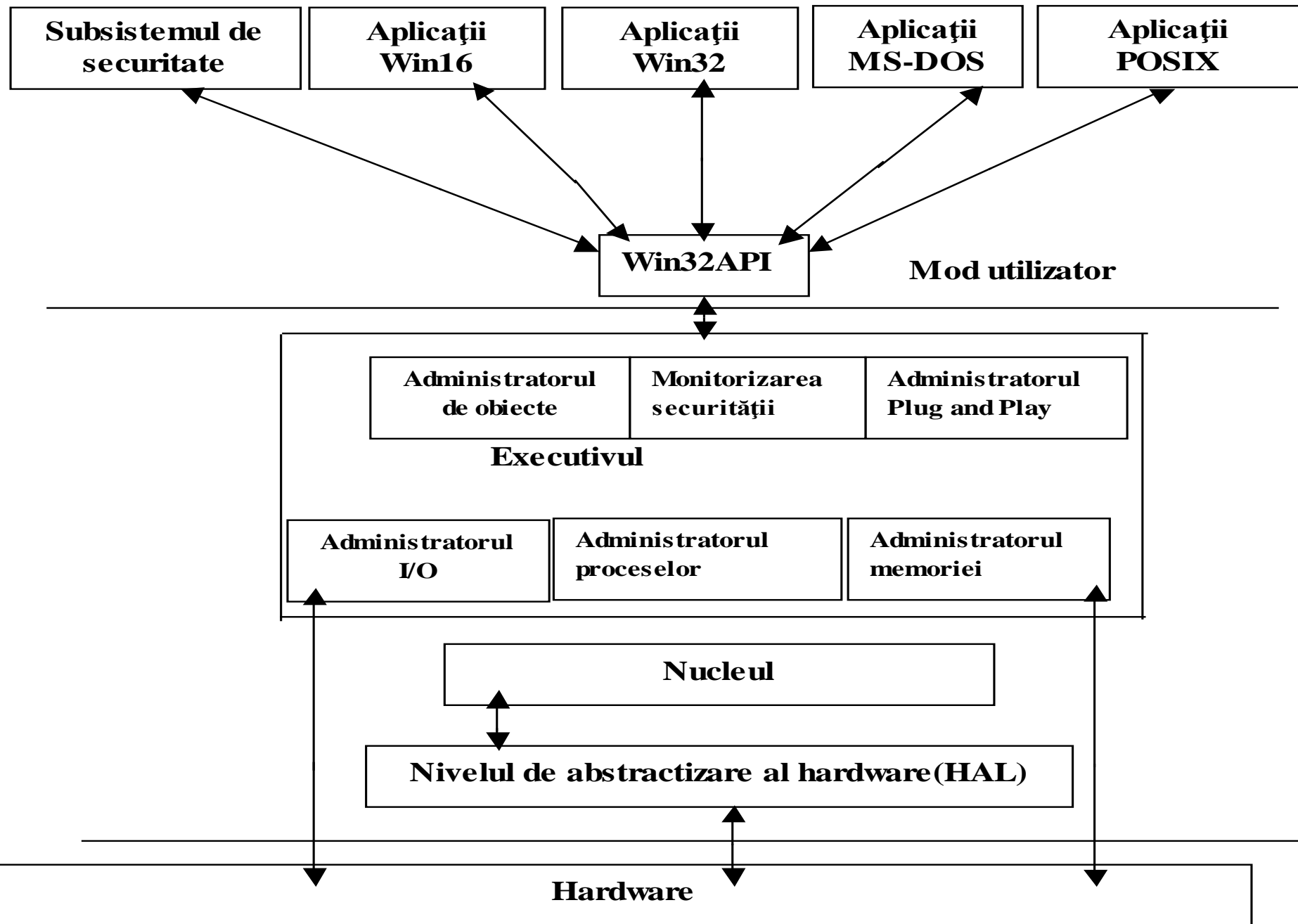
- ► În fereastra dreptunghiulară se află **icon-uri**, mici imagini sau pictograme, fiecare reprezentând părți din sistemul de calcul și aplicațiile care controlează sistemului.
- ► **Butoanele de minimizare** (Minimize), de maximizare (Maximize) și de închidere (Close) apar în colțul din dreapta-sus al ferestrei pe aceeași linie ca și bara de titlu. Butonul de minimizare reduce fereastra la un buton pe bara de aplicații, iar butonul de maximizare mărește fereastra la dimensiunea întregului ecran. Butonul de închidere reprezintă o modalitate rapidă de a închide fereastra. După folosirea butonului de maximizare, acesta se transformă în **butonul de restaurare** (Restore button). Butonul de restaurare, permite aducerea ferestrei la mărimea ei anterioară (mai mică).
- ► **Marginile** ( borders) sunt cele patru linii care definesc limitele unei ferestre.
- ► **Bara de informații** ( status bar) furnizează informații. Pe măsură ce se aleg opțiuni din meniu, se selectează obiecte din ferestre sau se lansează comenzi, acțiunea este descrisă pe bara de informații.

- ► **Tab-ul de redimensionare** (resize tab) oferă o suprafață mare care se poate „agăța” cu mouse-ul atunci când se dorește modificarea mărimii unei ferestre.
- ► Atunci când o fereastră nu este suficient de mare pentru a afișa tot conținutul ei, apare butonul de derulare. Trăgând de **glisor** (butonul pătrat de pe bara de derulare) se mută conținutul ascuns al ferestrei în partea vizibilă a acesteia. De asemenea, se pot folosi săgețile de la capetele barelor de derulare pentru a obține același rezultat.
- ► **Desktop(Biroul electronic Windows)**. După instalarea completă a sistemului apare pe ecran imaginea similară unui birou. Acesta poate conține un ceas, un calculator de birou, un loc de depozitare a fișierelor și dosarelor, o listă completă a tuturor instrumentelor și dosarelor etc. **Bara de aplicații** conține butonul **Start**, care reprezintă cheia către instrumentele, accesoriile și caracteristicile Windows. Când se apasă butonul Start, Windows afișează o listă de opțiuni ( numită pop-up) din care se poate face o alegere. Pentru fiecare aplicație lansată, pe bara de aplicații apare un buton cu numele aplicației.



## Organizarea Sistemelor Windows

- **Structura sistemului** este descrisă în figura urm.
- Observăm organizarea modulară și pe niveluri a sistemului.
- Principalele niveluri sunt cel de **abstractizare a hardwarelui** (HAL), **nucleul** și **executivul**. Toate aceste componente sunt executate în mod protejat.
- Subsistemele se împart în două categorii: **subsistemele de mediu**, care emulează diferite sisteme de operare și **subsistemul de protecție(securitate)**.



- **Nivelul de abstractizare a hardware**
- Unul dintre obiectivele Windows a fost portabilitatea (sistemul sa functioneze pe o clasa cit mai mare de calc., indiferent de producator).
- **Portabilitatea** este asigurată de faptul că majoritatea componentelor sistemului sunt scrise în C sau C++ .
- De asemenea, tot codul care este dependent de un anumit procesor, se află în fișierele DLL(**D**ynamic **L**ink **L**ibrary), care se găsesc în componenta **de abstractizare a hardware ( HAL)** a sistemului.
- Un fișier DLL realizează maparea spațiului de adrese al procesului, astfel încât orice funcție implementată de către DLL apare ca o componentă a procesului.
- HAL manipulează direct componenta hardware a sistemului, astfel că restul sistemului Windows funcționează independent de aceasta.
- HAL reprezintă cel mai scăzut nivel al sistemului de operare ce ascunde diferențele hardware dintre diferitele clase de sisteme de calcul, ceea ce face din Windows un sistem de operare portabil.
- HAL exportă o interfață virtuală care este folosită de către nucleu, executiv și driverele de unitate. Un avantaj al acestei abordări, este că este necesară numai o versiune a fiecărui driver, care poate fi executată indiferent de platforma hardware, deci driverele nu trebuie să mai fie adaptate de la un sistem la altul.
- De asemenea, HAL furnizează suport pentru sisteme cu multiprocesare simetrică. Din motive de performanță, driverele de I/O și cele grafice pot accesa direct componenta hardware.

- **Nivelul nucleu.**
- Deasupra HAL se afla nucleul si drivererele de dispozitiv; el stă la baza executivului și a subsistenelor.
- Cea mai importanta sarcina a nucleului este de a permite aplicatiilor sa fie executate.si sa le dea accesul catre componenta hard.
- Ptr. a executa programul, ii fixeaza un spatiu de adrese,incarca codul programului in memorie si ii furnizeaza o stiva.
- Nucleul administreaza resursele calculatorului (procesoare, memorii, periferice)
- Nucleul este incarcat odata cu sist. de operare.
- Nucleul lanseaza in executie toate componentele necesare functionarii windows.
- Nucleul urmareste toate componentele hardware ale sistemului si le pune la dispozitia proceselor.
- Nucleul face restul sistemului de operare complet independent de hard (portabil). Nucleul continua de acolo de unde se opreste HAL.

- Nucleul accesează hard-ul prin intermediul HAL și realizează abstractizări de nivel mai înalt decât HAL.
- **Exemplu:** HAL conține apeluri pentru asocierea procedurilor de tratare a întreruperilor cu întreruperile. În schimb, nucleul realizează schimbarea de context.
- Planificarea firelor de execuție: atunci când este momentul pentru a verifica dacă un fir de execuție nou se poate executa, de exemplu după ce a expirat o cantitate de timp sau după terminarea unei întreruperi de I/O, nucleul alege firul de execuție și face schimbarea de context necesară rularii lui.
- Nucleul sistemului este orientat pe obiecte. El execută o anumită sarcină, folosind o mulțime de obiecte ale sale ale căror atribute conțin date și ale căror metode execută activități specifice lui.
- Nucleul asigură suport pentru utilizarea a două clase de obiecte de nivel scăzut: **Obiecte de control și Obiecte dispecer** .

- Obiectele de control sunt acele obiecte care controleaza sistemul.
  - **apeluri de proceduri intirziate** (DPC-Deferred Procedure Call);
    - ▶ un obiect DPC este folosit pentru a separa partea care nu este critica de partea critica ( din punct de vedere al timpului) dintr-o procedura de tratare a intreruperii.
    - ▶ In general, o procedura de tratare a intreruperii salveaza citiva registri hard volatili asociati cu dispozitivele de I/O care au generat intreruperea, astfel incit acestia sa nu fie stersi si reactiveaza echipamentul, dar lasa majoritatea prelucrarilor pt. mai tirziu.
  - Exemplu.** Dupa ce a fost apasata o tasta, procedura de tratare a intreruperilor tastaturii citeste codul tastei dintr-un registru si reactiveaza intreruperea tastaturii, dar nu are voie sa prelucreze tasta imediat, daca o alta activitate, cu o prioritate mai inalta trebuie executata.
  - ▶ De asemenea, DPC-urile sunt folositoare pentru expirarea contoarelor si alte activitati a caror procesare efectiva nu trebuie sa fie instantanee.
  - ▶ Coada DPC este mecanismul prin care se memoreaza ceea ce trebuie facut mai tirziu.

- **apeluri asincrone de procedură**(APC), sunt asemănătoare DPC-urilor, cu excepția faptului că ele se execută în contextul unui proces specific.

**Exemplu.** Atunci când se prelucrează apăsarea unei taste, nu contează în ce context va rula DPC-ul deoarece singurul lucru care se va întâmpla este că codul tastei va fi pus într-un tampon. În schimb, dacă întreruperea necesită copierea unui tampon din zona nucleu în zona utilizator, atunci procedura de copiere trebuie să ruleze în contextul unui proces destinat. Contextul destinat este necesar pt. ca tabela de pagini să conțină ambele tampoane.

- **obiectele întrerupere**, asociază sursa întreruperii cu o anumită clasă de întreruperi;
- **obiecte de urmărire a tensiunii de alimentare**, pentru a se apela automat o anumită rutină când apare o cădere de tensiune;
- **obiectul proces** reprezintă spațiul de adrese virtuale și informațiile de control necesare execuției firelor de control ale procesului respectiv;
- **obiectul profil** este utilizat pentru a măsura cantitatea de timp necesar unui bloc de cod.

- **Obiecte ale dispecerului.**
- **Exemple** de astfel de obiecte sunt:
  - **obiectele eveniment** sunt folosite pentru a înregistra apariția evenimentelor și pentru a se realiza sincronizarea lor cu anumite acțiuni;
  - **obiectele de comutare** sunt utilizate pentru comutarea sistemului din modul nucleu(protejat) în cel utilizator și invers;
  - **obiectele semafor** sunt folosite ca porți de intrare pentru a controla numărul firelor de execuție care accesează anumite resurse; cele cu **excludere mutuală(mutex)** sunt utilizate numai în modul nucleu, pentru a se rezolva problema interblocării;
  - **obiectele „thread”** sunt entități care sunt executate de către nucleu, care aparțin unor clase proces;
  - **obiectele ceas** sunt folosite pentru măsurarea intervalelor de timp și pentru a semnala depășirile de timp(timeout), când timpul de efectuare a operațiilor este mai lung decât cel așteptat și acestea trebuie să fie întrerupte.



- **Manipularea întreruperilor.**
- Windows definește mai multe situații de excepție software și hardware, independente de arhitectura sistemului de calcul, cum ar fi:
  - - violarea drepturilor de acces la memorie,
  - - împărțire la zero,
  - - instrucțiune ilegală,
  - - depășire a domeniului de valori pentru o expresie,
  - - eroare la citirea unei pagini.
- Aceste situații conduc la abandonarea fluxului de control pe care procesorul îl urmează în mod normal și sunt rezolvate prin intermediul **întreruperilor**.
- Manipularea excepțiilor se poate realiza în **mod nucleu** sau **utilizator**.
- **Dispecerul** situațiilor de excepție creează o înregistrare care conține motivul acelei excepții și caută manipulatorul excepției respective, care poate prelucra acea înregistrare.
- Tipul și numărul întreruperilor depinde de tipul procesorului.
- Din motive de portabilitate, dispecerul de întreruperi realizează o corespondență între întreruperile hardware și o mulțime standard. Fiecare întrerupere are un număr de prioritate, în funcție de care se realizează servirea.
- **Exemplu.** Sub Windows XP există 32 nivele de întreruperi (IRQL - Interrupt **Re**Quest **L**evel); opt dintre ele sunt rezervate nucleului, iar celelalte 24 reprezintă întreruperi hardware care se realizează prin intermediul HAL.
- Nucleul folosește o tabelă prin care se asociază la fiecare nivel de întrerupere o anumită rutină de serviciu.

- **Întreruperi DPC.**
- O procedură DPC (Deferred Procedure Call) îndeplinește funcții de sistem și este executată în mod supervizor.
- Atributul „amânat” (deferred) se referă la faptul că execuția ei este amânată, până când IRQL-ul ei coboară la un nivel destul de scăzut.
- Acest tip de procedură este utilizat de către nucleu, pentru procesarea evenimentelor de expirare a ceasurilor de timp real și de către drivere în prelucrarea cererilor de I/O.
- DPC-urile sunt puse în coadă la nivelul 2, de către rutine care se execută la niveluri de prioritate superioare.
- Odată coborât IRQL, DPC-urile din coadă vor fi lansate în execuție, folosind resursele firului care se află în momentul curent în execuție, fără ca acesta să fie informat în vreun fel.

- **Sincronizarea proceselor la nivel scăzut.**
- Se realizează prin intermediul procedurilor APC(**A**synchronous **P**rocedure **C**all), care sunt executate în contextul unui anumit fir.
- Aceste proceduri sunt păstrate în liste asociate firelor de execuție, spre deosebire de procedurile DPC care sunt păstrate într-o listă unică de sistem și sunt lansate atunci când firul corespunzător se află în execuție.
- Procedurile APC sunt destinate situațiilor care necesită operare într-un anumit spațiu de adrese, cum este, de **exemplu**, copierea datelor transferate de către un fir care a inițiat o cerere de I/O în/din bufferul unui controller.
- **Refacerea după o cădere de tensiune.**
- Întreruperea datorată unei pene de curent atenționează sistemul de operare, atunci când se petrece un astfel de eveniment.
- Obiectul folosit pentru notificarea căderii de tensiune furnizează informațiile necesare driverului de unitate de a apela o rutină care este lansată în momentul în care există din nou tensiune de alimentare și care va „reinițializa” unitatea respectivă.

## Executivul Windows

- Oferă o mulțime de servicii, pe care le poate folosi oricare subsistem de mediu:
- - administrarea obiectelor;
- - administrarea memoriei;
- - administrarea proceselor;
- - administrarea I/O;
- - lucrul cu apeluri de proceduri locale;
- - monitorizarea securității;
- - mecanismul „plug-and-play”;
- - autoîncărcarea sistemului.
- **Administrarea obiectelor.**
- La nivel de executiv sunt utilizate o serie de clase de obiecte (director, semafor, eveniment, proces sau thread, port, fișier).

- **Implementarea obiectelor.**
- Obiectele asigura o interfata uniforma si consistenta catre toate resursele sistemului si structurile de date coresp. lor (procese, fisiere, semafoare etc.).
- Aceasta uniformitate are diverse aspecte:
  - Toate obiectele sunt referite si accesate in acelasi fel, prin referinte catre ele.
  - Deoarece toate accesarile trec prin manag. de obiecte, verificarile de protectie si securitate se fac in acelasi loc si, astfel nici un proces nu le poate ocoli.
  - Partajarea resurselor intre obiecte se face intr-un mod uniform.
  - Este usor de urmarit care obiecte sunt active si care pot fi sterse.
  - Usureaza administrarea cotelor de resurse care se aloca unui obiect.

- **Structura obiectelor.**
- Fiecare obiect contine:
  - un antet, cu informatii comune tuturor obiectelor, de toate tipurile. Cimpurile din acest antet includ: numele obiectului, informatii despre spatiul de adrese in care el exista, informatii de protectie si securitate (ce poate folosi obiectul), costurile legate de folosirea obiectului, lista proceselor care au referinte deschise catre obiect, indicator catre tipul de obiect ( Numele obiectului, tipuri de acces, drepturi de acces, cote de incarcare, daca este sincronizabil, daca este paginabil).
  - Date si metode specifice (de deschidere, de inchidere, de stergere, de cerere de nume, de analiza sintactica, de securitate) obiectului.
- Atunci cind un obiect este creat, managerul de obiecte alocă un bloc de memorie virtuala din spatiul de adrese al nucleului; reciproc, cind un obiect este dealocat, spatiul resp. este eliberat.
- Rolul administratorului de obiecte este de a superviza utilizarea obiectelor.
- Când un thread dorește să folosească un obiect, el apelează metoda **open**, pentru a obține accesul la manipulatorul acelui obiect, care este o interfață standard către obiect.
- Deoarece administratorul de obiecte este singura entitate care poate genera un manipulator de obiect, el face parte din modulul de securitate al sistemului.
- De **exemplu**, administratorul obiectelor verifică dacă un proces are drepturile de acces la un anumit obiect, în momentul în care încearcă să deschidă acel obiect.

- Administratorul obiectelor poate urmări care dintre procese folosește un anumit obiect.
- Fiecare antet(header) de obiect, conține un **contor** al numărului de procese care au accesat obiectul respectiv. Acest contor este incrementat, respectiv decrementat, când un proces cere, respectiv nu mai utilizează acel obiect. Dacă obiectul respectiv este temporar, când valoarea contorului a ajuns la 0, obiectul respectiv este șters din spațiul de nume.
- Orice obiect are un **nume**; spațiul de nume al proceselor are un caracter global, deci obiectele create de un proces pot fi partajate împreună cu alte procese.
- Numele unui obiect poate fi permanent sau temporar. Un nume permanent reprezintă o entitate cu caracter permanent, chiar dacă nu este utilizată la un moment dat, cum este cazul unui driver de disc. Numele temporare există numai atâta timp cât anumite procese păstrează manipuloare către aceste obiecte.
- Numele de obiecte sunt structurate la fel ca o structură de directoare arborescentă. Directoarelor le corespund **obiectele director**, care conțin numele tuturor obiectelor din acel director; în plus, acestea pot conține **domenii de obiecte** care sunt mulțimi de obiecte. Astfel, există posibilitatea construcției ierarhice a unei structuri de obiecte.
- Un **exemplu** de domeniu de obiecte sunt driverele de discuri. În momentul când un disc este introdus în sistem, spațiul de nume al acestuia este adăugat celui deja existent.

- Adm. de obiecte creează manipulatori de obiecte, care constau din informații de control al accesului și un pointer la obiect.
- Un proces poate obține un manipulator de obiecte prin:
  - - crearea obiectului de către el însuși;
  - - primirea unui duplicat al unui manipulator de la un alt obiect;
  - - prin moștenirea unui manipulator de la un proces părinte.
- **Tabela de obiecte** a procesului conține toate aceste manipuloare.
- O intrare în tabela de obiecte conține drepturile de acces și stabilește dacă manipulatorul poate fi moștenit de către procesele fiu.
- Când un proces se termină, sistemul închide toate manipuloarele de obiecte deschise de către acesta.
- Când un utilizator se loghează la un sistem și trece de faza de autentificare, îi este atașat un obiect care conține o serie de atribute, printre care: **identificatorul de securitate, grupurile din care face parte, grupul primar, lista control a accesului**. Aceste atribute determină care dintre serviciile și obiectele sistemului pot fi folosite de către un utilizator dat.
- Fiecare obiect este protejat de către o listă de control a accesului, care conține identificatorii de securitate și drepturile acordate fiecărui proces.
- Când un proces încearcă să deschidă un obiect, sistemul compară identificatorul lui de securitate, corespunzător utilizatorului căruia îi aparține procesul respectiv, cu lista de control a accesului la obiect, pentru a determina dacă accesul este permis.



- **Memoria virtuală sub Windows.**
- Vom discuta gestionarea memoriei pentru sistemele de calcul cu procesor pe 32 de biți, în cazul celor pe 64 de biți, modalitatea de administrare fiind asemănătoare.
- Fiecărui proces sub **Windows**, îi este acordat un spațiu de adrese virtuale de dimensiune fixă (4 Go), care este mult mai mare decât dimensiunea memoriei primare a unui calculator.
- Procesul nu folosește tot spațiul de adrese virtuale, ci numai ceea ce îi este necesar, de obicei mult mai puțin din ceea ce îi este alocat.
- O parte a spațiului de memorie virtuală, de obicei 2 Go, este folosit pentru a permite firului de execuție referențierea unor obiecte din spațiul de memorie primară, iar restul este folosit pentru a referenția adrese folosite de către SO (spațiu utilizat în mod supervizor).
- Dimensiunile spațiilor de adrese virtuale utilizate de proces, respectiv de nucleul SO diferă de la o variantă la alta de sistem Windows.
- Chiar dacă partea de adrese supervizor există în spațiul de adrese al procesului, memoria poate fi referită de un fir de execuție al procesului, numai dacă acesta se găsește în mod supervizor.

- Sistemul de operare are nevoie de anumite mijloace pentru a determina cantitatea de spațiu de adrese necesare procesului. Editorul de legături construiește imaginea statică a fișierului executabil, care definește spațiul de adrese virtuale.
- Anumite componente alocate dinamic(biblioteci ale sistemului), sunt adăugate spațiului de adrese virtuale în faza de execuție a procesului. Există două faze de adăugare dinamică de adrese la spațiul de adrese virtuale:
  - rezervarea unei porțiuni a spațiului de adrese virtuale, numită regiune;
  - cererea unui bloc de de pagini, dintr-o regiune a spațiului de adrese.
- Fiecare procesor are o anumită unitate de alocare, adică dimensiunea minimă a blocului de adrese care va fi rezervat. De obicei, aceasta este de 64 Ko.
- De asemenea fiecare procesor are o anumită dimensiune a paginii, de obicei 4 sau 8 Ko. Memoria este cerută în bucăți formate din pagini, deci porțiunea cerută la un moment dat, este mult mai mică decât cea rezervată anterior.

- Un fir de execuție din cadrul unui proces poate, în mod dinamic să rezerve o regiune a spațiului de adrese virtuale, fără a scrie nimic în fișierul de pagini din memoria secundară.
- De asemenea, el poate elibera o regiune a spațiului de adrese virtuale rezervată anterior.
- A doua fază este de a cere adrese, care anterior au fost rezervate, spațiu care este alocat în fișierul de pagini.
- Dacă firul de execuție al procesului, referențiază memoria alocată, pagina care conține adresa referențiată va fi încărcată din fișierul de pagini, în memoria primară.
- **Traducerea adreselor** se face cu ajutorul unor componente hardware, pentru a detecta rapid paginile de memorie virtuală ale procesului care lipsesc din memoria internă și pentru a le pune în corespondență cât mai rapid pagini fizice.
- O adresă virtuală este memorată pe 32 de biți și este generată de către procesor.
- Win2K folosește o schemă de adresare structurată pe două niveluri. Cei mai puțin semnificativi K1 biți din cadrul adresei, reprezintă deplasamentul. În Windows XP restul de biți se numește **numărul de pagină virtuală** și este separat în două părți, denumite **indexul tabelii de pagini** (K2 biți) și **indicele directorului de pagini** (K3 biți). Valorile lui K1, K2, K3 sunt specifice fiecărui procesor în parte. Val. cele mai utilizate sunt de 12 biți pentru K1, resp. de câte 10 biți pentru K2, K3.

- Translatarea adreselor folosește aceste trei câmpuri astfel:
- Descriptorul de proces conține un pointer la începutul **directorului de pagini**, care are 1024 de intrări, pentru procesul dat. Indicele din dir., este un deplasament unde este localizată o intrare în descriptorul de pagini(PDE- **P**age **D**escritor **E**ntry). Fiecare proces poate avea mai multe tabele de pagini. PDE referențiază o anumită tabelă de pagini ce va fi folosită pentru această referință la memorie.
- Intrarea în tabelele de pagini (PTE – **P**age **T**able **E**ntry), care la rândul lor au câte 1024 de intrări este găsită folosind indicele tabeli de pagini din adresă, ca un index în tabela de pagini. Dacă pagina căutată este încărcată în memoria internă în pagina fizică j, atunci PTE punctează către pagina fizică. Dacă ea nu este încărcată, adm. Mem. virtuale trebuie să localizeze pagina în fișierul de pagini, să caute o pagină fizică, să o aloce procesului, și apoi să încarce pagina virtuală respectivă în pagina fizică. În final, deplasamentul este adăugat adresei paginii fizice, pentru a obține adresa baitului căutat din memoria primară.
- Teoretic, directorul de pagini se poate memora în oricare dintre locațiile de memorie primară, dar de obicei el se află într-o locație fixă, specifică fiecărui procesor, a cărei adresă este salvată într-un registru. Conținutul acestui registru face parte din contextul firului de exec.resp. și atunci când acesta nu mai beneficiază de serviciile CPU, conținutul registrului este salvat.

- Windows folosește tabele de pagini multiple, pentru a face distincție între diverse utilizări ale spațiului de adrese.
- Cea mai evidentă diferențiere este că anumte pagini fac parte din spațiul utilizator, iar altele din spațiul supervisor.
- Obs. că prin plasarea nucleului într-o tabelă de pagini separată, procese diferite au punct de intrare(PTE) către nucleu.
- Fiecare PTE referențiază un număr de pagină fizică, atunci când pagina coresp. este încărcată.
- De asemenea, există o colecție de fanioane care descriu modul cum pagina poate fi referențiată(pagina resp. este validă sau nu, este rezervată sau nu, daca conținutul ei a fost modif. din mom. încărcării etc.).
- Windows folosește paginarea la cerere, ceea ce înseamnă că paginile sunt încărcate în memoria primară când sunt cerute. În plus, PTE-urile nu sunt create până când pagina coresp. nu este încărcată în memorie, în ideea că procesele nu trebuie să rezerve spații de memorie primară, pe care să nu le folosească. Din acest motiv, SO trebuie să păstreze alte structuri de date care să reliefeze operațiile de rezervare și de cerere de alocare.
- Descriptorul de adrese virtuale(VAD – **V**irtual **A**ddress **D**escriptor) este creat atunci când un proces face o rezervare de adrese de mem. virtuală sau o cerere de astfel de spațiu. Când un fir de execuție referențiază pentru prima dată adrese din VAD, este creat PTE-ul și astfel translatarea adreselor se desfășoară normal.