

Curs 6 : Baze de date Oracle: Generalitati

1. **Notiuni generale**
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

1. Notiuni generale

- Baze de date; depozit de date
- *SGBD*, dictionarul datelor
- sistem de baze de date
- administratorul de date
- administratorul bazei de date
- structura unui SGBD.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Baza de date

= un ansamblu structurat de date coerente, fără redundanță inutilă, astfel încât acestea pot fi prelucrate eficient de mai mulți utilizatori într-un mod concurent

= o colecție de date persistente, care sunt folosite de către sistemele de aplicații ale unei anumite „întreprinderi“

- două clase mari de date:
 - **BD operationale** → date operaționale (necesare pt. aplicațiile operaționale (care reflectă funcționalitatea zilnică a întreprinderii)
 - **depozitul de date** → informații de sinteză (necesare pt. susținerea procesului decizional).

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Datele din baza de date

- **persistente**: după ce au fost acceptate de către SGBD pentru introducerea în BD, ele pot fi șterse din BD numai printr-o cerere explicită adresată SGBD
- **integrate**: BD consta din unificarea mai multor fisiere
- **partajate**: BD poate fi utilizata concurent de mai multi utilizatori
- **suporta prelucrari frecvente, de complexitate redusa =>**

Aplicațiile de baze de date:

- **in principal**: operații de memorare și regăsire efectuate asupra unor volume mari de date, in scopul obtinerii de informatii din BD
- **secundar**: operații de prelucrare a datelor: introducere, actualizare, stergere.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Sistem de gestiune a bazelor de date (SGBD – *Data Base Management System*)

= un produs *software* care asigură interacțiunea cu o BD, permițând definirea, consultarea și actualizarea datelor din BD

Dicționarul datelor = catalog de sistem =

= conține „date despre date”,

= furnizează descrierea tuturor obiectelor unei BD (starea obiectelor, constrângeri de securitate și de integritate, informații despre utilizatori)

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Sistem de baze de date

= consta din:

1. BD propriu-zisă (în care se memorează datele)
2. SGBD (gestionarea și prelucrarea complexă a datelor)
3. dicționarul BD (metabaza de date: informații despre date, structura acestora, statistici, documentație)
4. mijloace *hardware* (comune sau specializate);
5. reglementări administrative destinate bunei funcționări a sistemului
6. personalul implicat:
 - administratori de date și baze de date,
 - proiectanți (designeri) de baze de date,
 - programatori de aplicații,
 - utilizatori finali.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Administratorul de date (DA – *Data Administrator*)

= un manager care stabileste:

- care sunt datele care trebuie stocate în baza de date;
- regulile de întreținere și de tratare a acestor date (ex. o anumită politică de securitate a datelor)

Administratorul bazei de date (DBA–*Database Administr.*)

= o persoană sau un grup de persoane ce răspund de ansamblul activităților legate de BD (creează baza de date reală, implementează elementele tehnice de control, asigura funcționarea sistemului la performanțe adecvate, monitorizeaza performanțele BD, furnizează diverse servicii tehnice etc.)

= este responsabil cu implementarea deciziilor DA și cu controlul general al sistemului, la nivel tehnic

=> are 4 mari categorii de atribuții:

- de proiectare,
- administrative,
- operative,
- de coordonare.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Proiectanții de BD:

- (i) cei care abordează nivelul logic:
 - proiectează conceptual baza de date (independent de programele de aplicatii si limbajele de programare)
- (ii) cei care abordează nivelul fizic:
 - aleg modul de implementare fizica a modelului conceptual

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Programatorii de aplicații:

- scriu programele aplicație ce conferă funcționalitatea cerută de utilizatorii finali
- utilizeaza limbaje de programare de nivel inalt (C++, *Java*, *PL/SQL* etc.).

Utilizatorii finali:

- accesează interactiv baza de date
- pot fi:
 - ✓ utilizatori simpli
 - ✓ utilizatori sofisticăți

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Structura unui SGBD:

- complexa; dinamica; minimum 5 clase de module:
 1. **programe de gestiune a bazei de date (PGBD):**
realizează accesul fizic la date ca urmare a unei comenzi;
 2. **module pentru tratarea LDD**
permit traducerea unor informații despre date în obiecte ce pot fi apoi exploatate în manieră procedurală / neprocedurală;
 3. **module pentru tratarea LMD** (interpretativ, compilativ, generare de programe)
permit utilizatorilor inserarea, ștergerea, reactualizarea sau consultarea informației dintr-o bază de date;
 4. **module utilitare**
asigură întreținerea, prelucrarea, exploatarea corectă și ușoară a bazei de date;
 5. **module de control**
 - permit controlul programelor de aplicație,
 - asigurarea confidențialității și integrității datelor,
 - rezolvarea unor probleme de concurență, r
 - ecuperarea informației în cazul unor avarii sau defecțiuni hardware sau software etc.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Modul de lucru al SGBD la nivel conceptual:

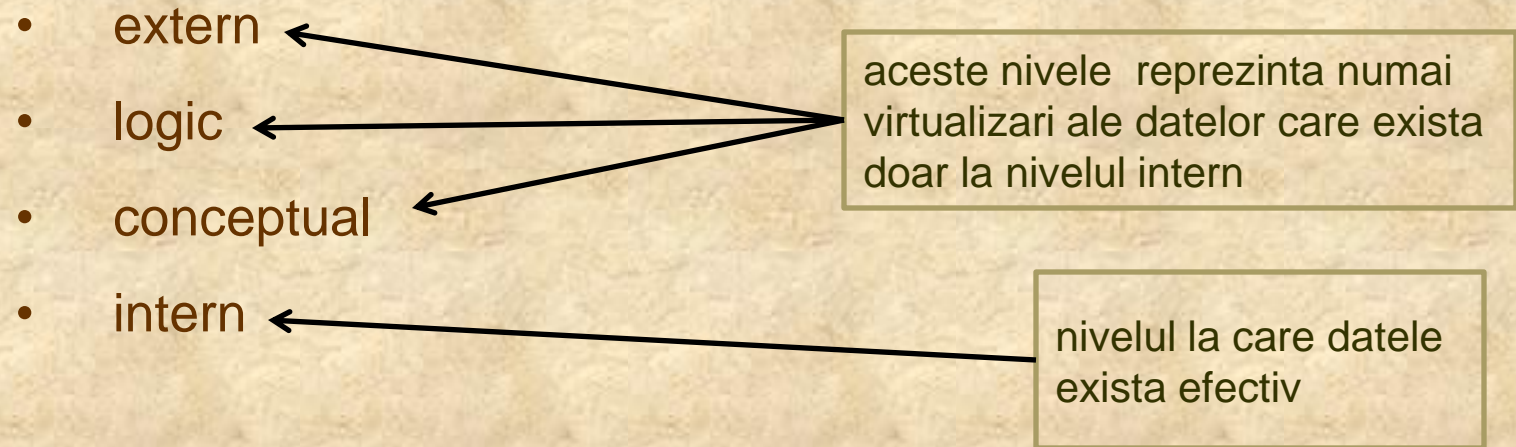
1. utilizatorul lansează o cerere de acces
2. SGBD acceptă cererea și o analizează
3. SGBD inspectează pe rând
 - schema internă corespunzătoare utilizatorului
 - schema conceptuală
 - definiția structurii de stocare
 - corespondențele
4. SGBD execută operațiile necesare în baza de date stocată, i.e. modulele PGBD asigură accesul fizic la date:
 - găsesc descrierea datelor implicate în comandă
 - identifică datele și tipul acestora
 - identifică informații ce permit accesul la structurile fizice de stocare (fișiere, volume etc.)
 - verifică dacă datele sunt disponibile
 - extrag datele și fac conversiile
 - plasează datele în spațiul de memorie al utilizatorului
 - transmit informații de control necesare execuției comenzii, în spațiul de memorie al utilizatorului
 - transferă controlul programului de aplicație.

Curs 6 : Baze de date Oracle: Generalitati

1. **Notiuni generale**
2. **Arhitectura SGBD**
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Cele 4 niveluri de abstractizare și de percepție a datelor într-o BD:



=> arhitectura pe 3 niveluri a BD si existenta unor corespondente intre acestea.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Nivelul extern (modelul extern, subschema, vizualizarea)

- reprezintă viziunea utilizatorului final asupra datelor
- permite asigurarea unui nivel de securitate a datelor: un utilizator va accesa doar datele descrise în schema sa externă

Nivelul logic (una din schemele logice posibile ale datelor)

- reprezintă viziunea programatorului de aplicație asupra datelor;

Nivelul conceptual (schema conceptuală a datelor: articol, înregistrare, zonă)

- este nivelul central
- reprezintă viziunea programatorilor de sistem asupra datelor
- corespunde structurii semantice a datelor fără implementarea pe calculator

Nivelul intern (schema fizică a datelor: bit, octet, adresă)

- permite descrierea datelor unei BD sub forma în care sunt stocate în memoria calculatorului
- sunt definite fișierele care conțin aceste date, articolele din fișiere, căile de acces la aceste articole etc.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-



Observatie

- La nivel conceptual sau intern:
 - schemele respective descriu in mod unic o bază de date
- La nivel extern:
 - schemele reprezintă o descriere a unei părți a bazei de date ce corespunde viziunii unui program sau unui utilizator

=>Pentru o BD particulară există:

- 1! schemă internă,
- 1! schemă conceptuală
- mai multe scheme externe.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

Avantajele oferite de arhitectura pe nivele a sistemelor de BD:

- **independența datelor:**

- ✓ **fizică:**

- modificarea schemei fizice a datelor nu implica modificarea schemei conceptuale si a schemei logice și nici a programelor de aplicație

- ✓ **logică:**

- modificarea schemei conceptuale a datelor nu implica modificarea schemei logice și a programelor de aplicație

⇒se creaza fiecărui utilizator iluzia că este singurul beneficiar al unor date pe care, în realitate, le folosește în comun cu alți utilizatori;

- **independența față de strategiile de acces:**

permite programului să precizeze

- data pe care dorește să o acceseze,
- dar nu modul cum accesează această dată

SGBD-ul va stabili drumul optim de acces la date.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

⇒ in sistemele de baze de date, funcțiile de declarare și de prelucrare a datelor sunt realizate cu ajutorul unor limbaje diferite, numite **limbaje pentru baze de date**:

- LDD
- LMD
- LCT
- LCD;

Limbajele universale si gestionarea unei BD:

Oracle este dotat cu precompilatoare (*C/C++*, *Pascal*, *ADA*, *Cobol*, *PL/1*, *Fortran*)

-> incorporarea de instrucțiuni *SQL* sau blocuri *PL/SQL* în programe scrise în alte limbaje, de nivel înalt, numite limbaje gazdă.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Arhitectura funcțională de referință

- propusă de grupul de lucru ANSI/X3/SPARC
- este axată pe dicționarul datelor și cuprinde două părți:
 1. prima parte: permite descrierea datelor (compoziția dicționarului datelor)
 2. a doua parte: permite prelucrarea datelor (interogarea și reactualizarea bazei de date)
- in fiecare parte se regăsesc cele trei niveluri: intern, conceptual și extern (nu sunt neapărat distincte);
- interfețele numerotate din Figura 1.1, ce descriu arhitectura de referință a unui SGBD, corespund următoarelor transformări (->S27):

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

SGBD si arhitectura ANSI/SPARC

SGBD dispune de urmatoarele procesoare:

dicționarul de date:

- prelucreaza comenzile de definire a schemei și comenzile de definire a vizualizărilor;

analizorul:

- analizeaza sintactic si semantic comenzile de definire a schemei și a vizualizărilor precum și cererile de prelucrare,
- o cerere în format intern care face referință la o vizualizare este tradusă în una sau mai multe cereri care fac referință la obiecte ce există în baza de date (modificarea cererilor);

translatorul:

- modifica cererile,
- asigură controlul drepturilor de acces,
- asigura controlul integrității în cazul reactualizărilor;

optimizorul:

- elaborează un plan de acces optim pentru a trata cererea,
- descompune cererea în operații de acces elementare și alege o ordine de execuție optimă,
- evaluează costul planului de acces înaintea execuției sale;

executorul:

- executa planul de acces elaborat de optimizor,
- asigura controlul concurenței.

Curs 6 : Baze de date Oracle: Generalitati

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

- Baze de date prerelationale:
 - ierarhice
 - retea
- Baze de date relationale
- Baze de date orientate obiect
- Baze de date relationale orientate obiect
- Baze de date deductive
- Baze de date distribuite
- Calculatoare și mașini baze de date
- Multibaze de date
- Baze de date cu suport decizional
- Data warehouse.

Curs 6 : Baze de date Oracle: Generalitati

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Criterii de clasificare a sistemelor de bazelor de date:

1. modelul de date
2. numarul de utilizatori
3. numarul de calculatoare pe care sunt stocate BD și SGBD
4. modul de functionare
5. implementarea sistemului de baze de date.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

1. Clasificarea dupa modelul de date:

- Baze de date prerelationale:
 - ierarhice
 - retea,
- Baze de date relationale,
- Baze de date orientate obiect,
- Baze de date relationale orientate obiect.

1. Notiuni generale	3. Evolutia bazelor de date	5. Arhitectura sistem. Oracle
2. Arhitectura SGBD	4. Clasificarea bazelor de date	6. Grid computing și SOA in Oracle.

2. Clasificarea dupa numarul de utilizatori:

- sisteme *multiutilizator*:
 - cele mai frecvente
 - permit accesul concurent al mai multor utilizatori la aceeasi BD;

sisteme monoutilizator:

- destul de rare
- permit accesul unui singur utilizator.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

3. Clasificarea dupa numarul de statii pe care este stocata BD:

- centralizate:
 - datele si SGBD sunt stocate pe o singura statie (calculator)
- distribuite:
 - și datele si SGBD sunt distribuite pe mai multe calculatoare interconectate printr-o retea de comunicatie.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

4. Clasificarea dupa modul de functionare:

- **teleprocesarea**
 - arhitectura tradițională: 1! calculator cu 1! unitate CPU și
 - un numar de terminale, incapabile să funcționeze singure;
- **arhitectura fișier-server**
 - procesarea este distribuită în rețea (de obicei LAN)
 - arhitectura cuprinde fișierele cerute de aplicații și SGBD-ul
 - aplicațiile și funcțiile SGBD sunt executate pe fiecare stație de lucru, solicitând atunci când este nevoie fișiere de pe serverul de fișiere;
- **arhitectura client-server**
există
 - un proces client, care necesită resurse și
 - un proces server, care oferă resurse.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Arhitectura client-server:

1. clientul (*front-end*) emite o cerere SQL →
2. cererea este executata pe server (*back-end*) →
3. serverul trimite clientului ca răspuns ansamblul înregistrărilor rezultat;

Clientul:

- administrează interfața cu utilizatorul și logica aplicației,
- acceptă și verifică sintaxa intrărilor utilizatorilor,
- procesează aplicațiile,
- generează cererile pentru BD și le trimite serverului,
- transmite răspunsul înapoi la utilizator.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Arhitectura client-server: (cont.)

Serverul:

- primește și procesează cererile clienților adresate BD,
 - verifică autorizarea,
 - efectuează procesarea interogare-reactualizare,
 - garantează respectarea constrângerilor de integritate,
 - realizează optimizarea interogărilor,
 - asigură controlul concurenței dintre mai multi clienți care se ignoră (mecanisme de blocare),
 - trimite clientului răspunsul,
- întreține dicționarul datelor,
- oferă acces simultan la baza de date,
- asigură robustețea în cazul defecțiunilor etc.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Arhitectura client-server: (cont.)

- Arhitectura tradițională client-server pe „două etaje” (niveluri):
 - ✓ clientul – prezinta datele către client,
 - ✓ serverul – furnizeaza servicii către client;
- Arhitectura client-server pe „trei etaje” presupune trei niveluri, fiecare rulat, potențial, pe o platformă diferită:
 1. **nivelul client:**
 - ✓ interfața cu utilizatorul,
 - ✓ este rulat pe calculatorul utilizatorului final;
 2. **nivelul server de aplicație:**
 - ✓ manevrează logica aplicațiilor și prelucrării datelor,
 - ✓ poate servi mai mulți clienți (conectarea la celelalte două straturi se face prin rețele locale LAN / WAN);
 3. **nivelul server de baze de date**
 - ✓ validarea datelor,
 - ✓ accesarea bazei de date (stochează date necesare stratului din mijloc).

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Arhitectura client-server: (cont.)

Arhitectura se potrivește natural mediului *Web*:

- un *browser Web* ≡ client și
- un *server Web* ≡ server de aplicație.

 *Middleware* =

= un strat, evident *software*, între aplicația postului client și serverul de baze de date

- constituit din:
 - ✓ o interfață de programare a aplicațiilor (*API - Application Programming Interface*) și
 - ✓ un protocol de rețea;

API descrie tipul de interacțiune dintre o aplicație client și un server la distanță, *via* un protocol de comunicație și de formatare a datelor

- scopul existenței interfeței de programare a aplicațiilor este de a oferi o interfață unică mai multor servere de baze de date.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

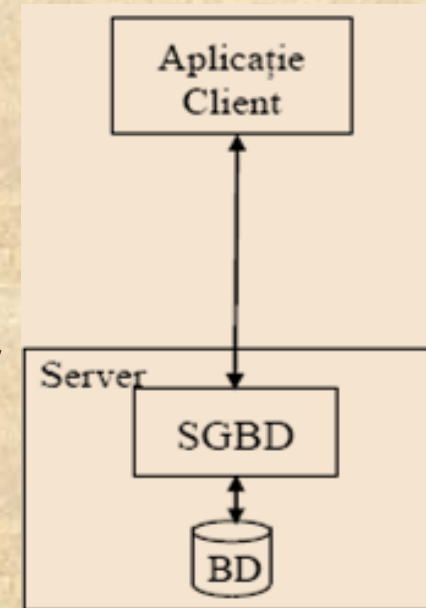
5. Clasificarea dupa modul de implementare a sistemului de baze de date:

- se combina ultimele 3 criterii : numarul de utilizatori, modul de stocare a BD și SGBD și modul de functionare a sistemului de baze de date:
 - I. sisteme client-server centralizate de tip *monouser*
 - II. sisteme client-server centralizate de tip *multiuser*
 - III. sisteme client-server distribuite de tip *multiuser*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

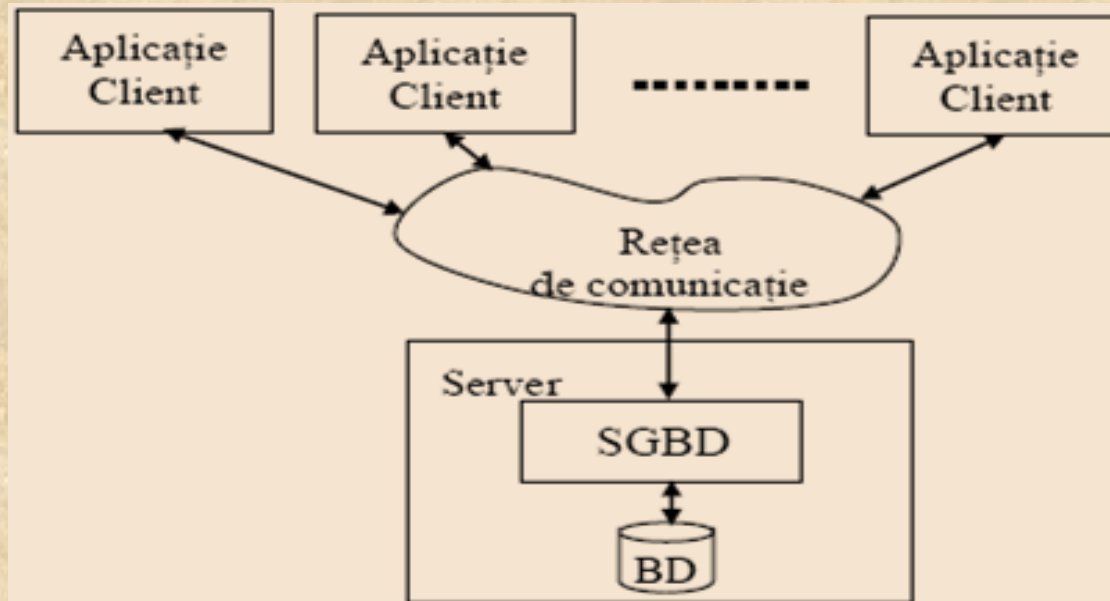
I. Sisteme client-server centralizate de tip *monouser*

- BD și SGBD sunt stocate pe acelasi *server* care raspunde cererilor unui singur *client* care acceseaza BD .



1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

II. Sisteme client-server centralizate de tip *multiuser*



- BD și SGBD sunt stocate pe același server care răspunde cererilor mai multor *clienți* care accesează BD ;
- aplicațiile *client* sunt executate pe stații diferite (=> cu puteri de calcul inferioare *serverului*), conectate printr-o rețea de comunicație cu calculatorul pe care rulează *serverul*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

III. Sisteme client-server distribuite de tip *multiuser*

- **O BD distribuita =**

= o colectie de date care, din punct de vedere logic, apartin aceluiasi sistem dar care, din punct de vedere fizic, pot sa fie memorate pe mai multe statii de calcul conectate printr-o retea de comunicatie

- **SGBD distribuit =**

= sistemul software care gestioneaza o astfel de BD

- Caracteristici:

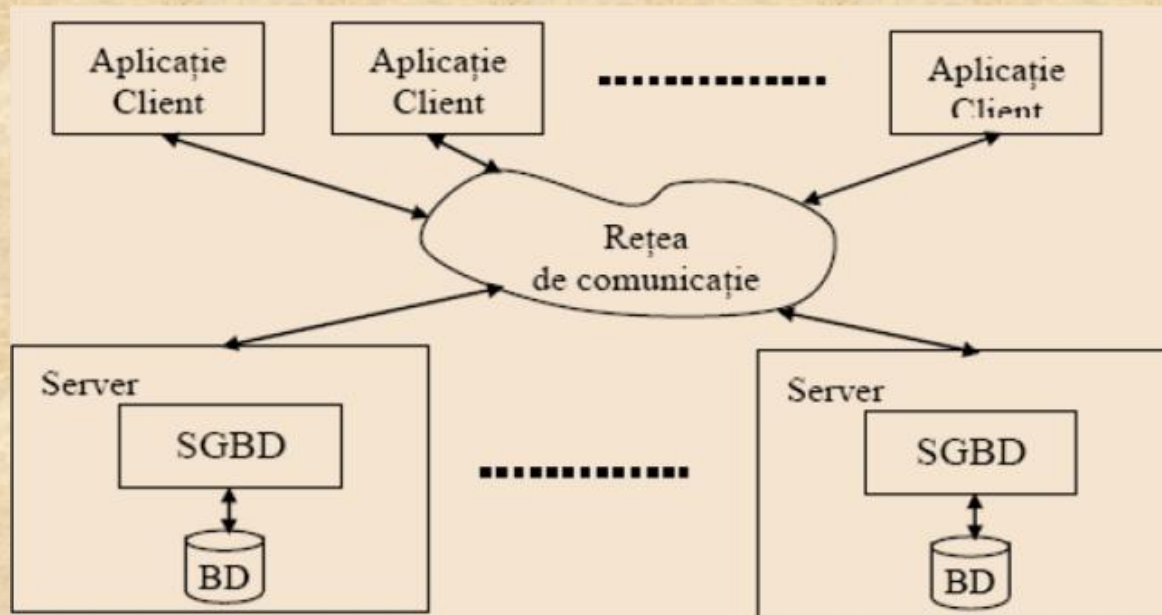
- cresterea capacitatii destocare și prelucrare
- cresterea complexitatii

- Principala cerinta (partial indeplinita):

transparenta = capacitatea unui sistem distribuit de a ascunde detaliile de implementare, astfel încât utilizatorii sa poata accesa datele pe baza unui model de nivel înalt, fara a fi necesara cunoasterea exacta a modului de amplasare, replicare sau comunicare a datelor.

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

III. Sisteme client-server distribuite de tip *multiuser* (cont.)



- BD și SGBD sunt distribuite pe mai multe statii conectate printr-o retea de comunicatie
- aplicatiile *client* sunt executate pe statii diferite , conectate printr-o retea de comunicatie cu calculatoarele (interconectate) pe care ruleaza *serverul*.

Curs 6 : Baze de date Oracle: Generalitati

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

5. Arhitectura sistemului *Oracle*

5.1. Structura bazei de date

5.2. Dicționarul datelor

5.3. Arhitectura internă a sistemului *Oracle*

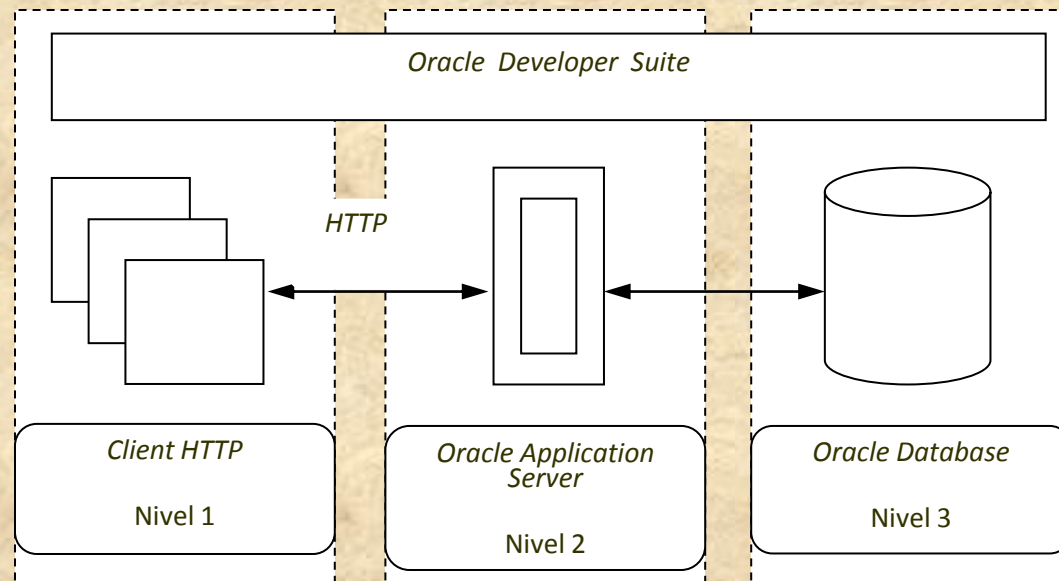
5.4. Prelucrarea bazei de date

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

Oracle (v9i →)

= **SGBD** relational OO

= primul SGBD care a oferit o infrastructură software completă pentru *Internet RAD*: *Oracle Enterprise Edition* =
Oracle Database,
Oracle Application Server,
Oracle Developer Suite;



Arhitectura *three-tier* a sistemului Oracle

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

Oracle Database:

Serverul de baze de date: funcționalități pentru

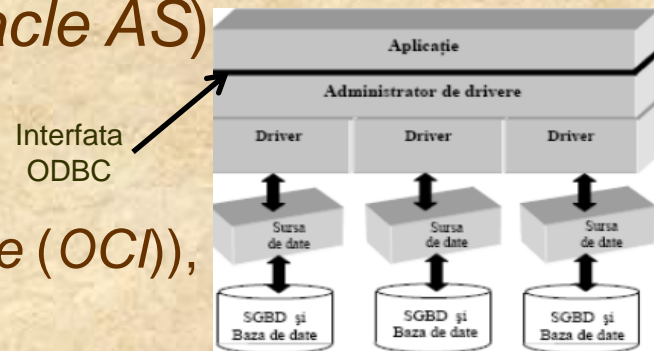
- aplicațiile de baze de date tradiționale,
- aplicațiile *Internet*:
 - ✓ administrarea completa a unei cantitati mari de date în condițiile unui mediu *multiuser*,
 - ✓ asigurarea accesul concurențial la același set de date,
 - ✓ prevenirea accesului neautorizat la date,
 - ✓ recuperarea datelor.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

Oracle Application Server (Oracle AS)

1. acces către orice BD:

- accesul nativ la bazele Oracle (prin extensia *Oracle Call Interface (OCI)*),
- accesul la alte tipuri de baze (prin extensia *Open Database Conectivity (ODBC)*) ;



2. infrastructură completă pentru instalarea, dezvoltarea și administrarea aplicațiilor de tip *e-business* și *Internet* complet independente de dispozitivul de pe care sunt accesate

i.e. ofera suport integral pentru:

- tehnologia *Java (J2EE 1.3)* și
- standardele actuale pentru servicii *Web (XML, XMI, SOAP, UDDI, WSDL, WebDAV)*;

3. asigură confidențialitatea informațiilor transmise prin rețea: suporturi de criptare, autentificare și autorizare.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

Oracle Developer Suite (Oracle DS)

- extinde infrastructura formată din *Oracle AS* și *Oracle Database* (in special in directia dezvoltarii de aplicații *Internet*),
- reprezintă un mediu integrat de dezvoltare (**Integrated Development Environment=IDE**): modelare, dezvoltare, codificare, depanare, optimizare, instalare de aplicatii,

Clasa de utilitare Oracle DS include:

- **componente pentru crearea de aplicații** (*JDeveloper, Designer, Forms Developer, Software Configuration Manager*)
- **componente de tip Business Intelligence** (*Reports Developer, Business Intelligence Beans, Warehouse Builder*),
- **componente pentru dezvoltarea de aplicații complexe** i.e. precompilatoarele *Pro** (*C/C++, PL/1, COBOL, ADA, FORTRAN* și *PASCAL*) care permit încorporarea de instrucțiuni SQL sau blocuri *PL/SQL* în module scrise utilizând alte limbaje de programare (gazdă); precompilatorul citește codul sursă și generează un fișier ce poate fi procesat de către compilatorul limbajului respectiv).

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5. Arhitectura sistemului Oracle

5.1. Structura bazei de date

- BD:
 - ✓ o structură logică și
 - ✓ o structură fizică;
- **Structura fizică** a bazei de date Oracle:
 - A. fișiere de date (*Datafiles*),
 - B. fișiere de reluare (*Redo Log Files*),
 - C. fișiere de control (*Control Files*);
- **Structura logică** a bazei de date Oracle:
 - a. blocurile de date (*data block*),
 - b. extensiile (*extent*),
 - c. segmentele (*segment*),
 - d. spațiile tabel (*tablespace*),
 - e. obiectele schemei (*schema object*).

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

A. Fișierele de date

= fișiere fizice ale SO

- stochează datele tuturor structurilor logice ale bazei;
- alocarea unui fișier de date bazei *Oracle: SO*
 - șterge informațiile nefolosite
 - acordă autorizații pentru fisier;
- primul fișier de date creat: fisierul care stochează dicționarul datelor.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

B. Fișierele de reluare

= înregistrează toate modificările care

- au loc asupra datelor bazei (indiferent dacă au fost permanentizate sau nu) și
- nu au fost scrise încă în fișierele de date;
- sunt specificate în momentul creării sau modificării bazei
- sunt utilizate în manieră circulară (cele care au fost folosite în întregime, pot fi arhivate până când sistemul le va reutiliza)
- asigură protecția BD în cazul defecțiunilor
- o BD *Oracle* conține două sau mai multe fișiere de reluare.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

C. Fișierele de control

= fișiere binare de dimensiune redusă, necesare pentru pornirea și funcționarea bazei de date;

- orice BD *Oracle* deține cel puțin un fișier de control;
- fiecare fișier de control
 - este asociat unei singure BD
 - conține informații despre structura fizică a acesteia
 - este creat odată cu respectiva BD (*Oracle* permite existența fișierelor de control multiplexate)

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

C. Fișierele de control (cont.)

- La pornirea unei instanțe *Oracle*:
 - sistemul folosește fișierul de control pentru:
 - a identifica baza și
 - a determina dacă aceasta este în stare validă pentru utilizare;
 - sunt identificate fișierele de reluare necesare execuției operațiilor bazei de date;
- Fișierele de control reflectă automat schimbările (creare, redenumire sau ștergere) care au loc la nivelul fișierelor de date sau de reluare;
- Informațiile din fișierele de control pot fi modificate doar de serverul *Oracle*;

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

- reprezinta cele trei niveluri de granularitate
- permit gestionarea dinamică a spațiului de pe disc pe măsura utilizării bazei de date
- nivelul cel mai fin de granularitate : blocul.

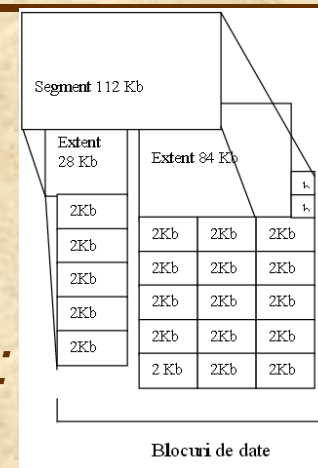
Structura logică a BD Oracle:

- a) blocurile de date (*data block*);
- b) extensiile (*extent*);
- c) segmentele (*segment*);
- d) spațiile tabel (*tablespace*);
- e) obiectele schemei (*schema object*).

- tabele, vizualizări, vizualizări materializate, secvențe, unități de program, sinonime, indecși, grupări, dimensiuni, legături de baze de date.
- sunt structuri logice care referă în mod direct datele bazei

permit

- definirea logică a organizării fizice a BD
- efectuarea legăturii dintre nivelul fizic și nivelul logic al acesteia.



- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

a) Blocuri de date

= unitati logice prin care sistemul administrează spațiul de stocare al fișierelor de date;

Blocul = cea mai mică unitate I/O folosită de baza de date,
= corespunzătoare unui bloc fizic de octeți de pe disc,
= dimensiunea sa: este definită în momentul creării BD,
poate fi modificată ulterior,
este un multiplu al dimensiunii
blocurilor fizice de la nivelul SO;

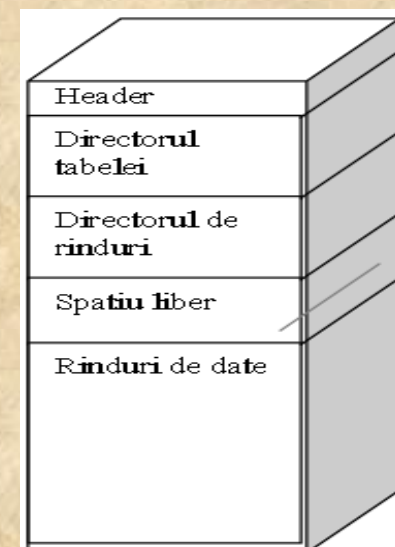
Structura blocului de date Oracle:

- un antet (*header*),
- un spațiu liber (*free space*),
- un spațiu pentru date (*data space*).

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

a) Blocuri de date (cont.)


- **Antetul** conține
 - informații generale referitoare la bloc
 - un catalog al tabelelor (*table directory*):
 - un catalog al liniilor (*row directory*):




- **Spațiul liber al blocului de date** este alocat pentru inserarea de noi linii sau actualizarea liniilor care necesită spațiu suplimentar.
 - Alegerea blocului în care va fi inserată o linie nouă depinde de spațiul liber al acestuia și de valorile parametrilor *PCTFREE* și *PCTUSED*.
 - Într-un bloc, se pot introduce date atâta timp cât dimensiunea spațiului liber este mai mare decât limita fixată de parametrul *PCTFREE*. Sistemul Oracle va considera acest bloc indisponibil pentru inserarea de noi linii, până când procentajul spațiului utilizat coboară sub valoarea dată de *PCTUSED*.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

b) c) Extensia și segmentul

 **Extensia** = unitate logica de alocare a spațiului BD,
= compusă dintr-o mulțime contiguă de blocuri de date (din același fișier de date);

 **Segmentul** = unitate logica formata din una sau mai multe extensii;

Inițial, segmentul are o singură extensie (*initial extent*).

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

b) c) Extensia și segmentul (cont.)

O extensie

- este alocată atunci când este creat sau extins un segment,
- este dezalocată (in general) când segmentul este suprimat sau trunchiat;
- Eliberarea unei extensii implică ștergerea datelor existente în blocurile de date alocate acesteia (ele vor fi reutilizate pentru extensiile nou create);

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

b) c) Extensia și segmentul (cont.)

Segmentul

= corespunde unui singur obiect fizic stocat

= folosește blocuri de date care se găsesc în același spațiu tabel;

Tipuri de segmente din BD *Oracle* :

- segmente de date (*data segment*),
- segmente index (*index segment*),
- segmente temporare (*temporary segment*),
- segmente de revenire (*undo segment*) etc.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

b) c) Extensia și segmentul (cont.)

Segmentele de date

- sunt definite atunci când este folosită comanda de creare a unui tabel sau a unei grupări
- un singur segment de date este folosit pentru stocarea tuturor datelor dintr-un tabel nepartiționat care nu face parte din nicio grupare, dintr-o partiție a unui tabel partiționat sau dintr-o grupare de tabele

Segmentele index

- sunt folosite pentru a stoca datele unui index
- fiecare index nepartiționat este conținut într-un singur segment. În cazul indecșilor partiționați, fiecărei partiții i se asociază câte un segment index

Segmentele temporare

- sunt utilizate de sistem pentru analiza și execuția comenzilor SQL care necesită un spațiu temporar de stocare
- sistemul alocă în mod automat segmente temporare atunci când este necesar și le suprimă după execuția comenzii SQL
- segmentele temporare sunt alocate în majoritatea cazurilor de sortare (atunci când operația respectivă nu se poate face în memorie sau dacă folosirea indecșilor nu presupune o soluție mai eficientă).

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

b) c) Extensia și segmentul (cont.)

Segmentele de revenire

- ✓ O BD conține unul sau mai multe segmente de revenire, folosite pentru:
 - anularea acțiunii tranzacțiilor
 - asigurarea consistenței la citire,
 - efectuarea operațiilor de recuperare a bazei de date;
- ✓ Segmentele de revenire **nu pot fi** accesate de către utilizatorii sau administratorii bazei de date
- ✓ Ele pot fi scrise și citite **doar** de către sistem.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |

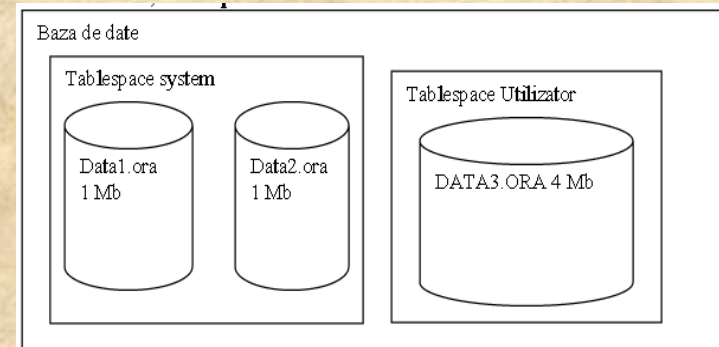
d) Spatiul tabel

= unitate logica de stocare formata din 1,2,... segmente

- grupeaza logic o mulțime de obiecte:
 - fiecare obiect al BD are specificat un spațiu tabel în care trebuie să fie creat ->
 - datele care alcătuiesc obiectul sunt apoi stocate în fișierele de date alocate spațiului tabel respectiv ->
 - un fișier de date poate fi alocat unui singur spațiu tabel;
- fiecarui utilizator i se poate aloca explicit un spațiu tabel, în care vor fi stocate toate obiectele create de el
- alocarea se efectueaza automat
- folosirea mai multor spații tabel -> flexibilitate în utilizarea BD
- BD = {spații tabel}

Tipuri de spatiu tabel in BD Oracle:

- spațiul tabel *SYSTEM*,
- spații tabel non-*SYSTEM*.



- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

d) Spatiul tabel (cont.)

Spațiul tabel **SYSTEM**:

- primul spațiu tabel creat
- i se alocă automat (în timpul creării BD) primul fișier de date,
- conține:
 - dicționarul datelor, inclusiv unitățile de program stocate,
 - segmentul de revenire **SYSTEM**;

Spațiile tabel non-**SYSTEM**:

- permit administrarea flexibilă a BD,
- separă segmentele de revenire, segmentele temporare, segmentele de date și segmentele index,
- separă datele dinamice de cele statice,
- controlează spațiul alocat pentru obiectele utilizatorilor.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

d) Spatiul tabel (cont.)

Relația dintre baze de date, spații tabel și fișiere de date presupune că:

- fiecare bază de date este împărțită din punct de vedere logic în unul sau mai multe spații tabel;
- unul sau mai multe fișiere de date sunt create explicit pentru fiecare spațiu tabel, cu scopul de a stoca fizic datele din structurile sale logice;
- suma mărimilor tuturor fișierelor de date asociate unui spațiu tabel reprezintă capacitatea totală de stocare a spațiului tabel;
- suma capacităților de stocare a spațiilor tabel ale unei baze de date reprezintă capacitatea totală de stocare a bazei.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

e) Schema

- = mulțimea obiectelor bazei de date, aflate în posesia unui utilizator (fiecare utilizator deține o singură schemă).
 - numele schemei este același cu numele utilizatorului
 - nu există o corespondență biunivocă între spațiile tabel și schemele de obiecte
 - obiectele aceleiași scheme pot fi în spații tabel diferite
 - un spațiu tabel poate conține obiecte din mai multe scheme
- ⇒ pentru a accesa un obiect din propria schemă, utilizatorul poate folosi doar numele acestuia
- ⇒ pentru referirea unui obiect din schema altui utilizator, trebuie specificat atât numele obiectului, cât și schema din care face parte, prin folosirea notației
- schema.obiect*

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5. Arhitectura sistemului *Oracle*

5.2. Dictionarul datelor (catalogul de sistem)

= conține „date despre date” (metabaza de date) i.e. informații despre baza de date:

- definițiile tuturor obiectelor din schemele bazei
- cantitatea de spațiu alocat pentru obiectele schemelor
- cantitatea de spațiu utilizat de acestea la momentul curent
- valorile implicite ale coloanelor
- constrângerile de integritate
- numele utilizatorilor *Oracle*
- privilegiile și *role*-urile acordate fiecărui utilizator
- informații de auditare etc.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.2. Dictionarul datelor (cont.)

- este generat automat la crearea BD;
 - este reactualizat de către serverul *Oracle* după fiecare comandă LDD sau LCD;
 - conținutul său reflectă imaginea bazei de date (structura fizică și logică) la un moment dat;
 - din punct de vedere structural este compus:
 - tabele de bază ale dictionarului și
 - vizualizări publice asupra acestora;
- => “vizibil” și pt sistem și pt utilizatori.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.2. Dictionarul datelor (cont.)

✓ *Tabelele de bază*

- stochează informațiile asociate BD,
- sunt primele obiecte create;

✓ *Vizualizările*

- decodifică informațiile stocate în tabelele de bază și
- le sintetizează pentru a fi disponibile utilizatorilor;

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.2. Dictionarul datelor (concl.)

- este deținut de către utilizatorul *SYS* și se află în spațiul tabel *SYSTEM*;
- sistemul poate accesa dicționarul datelor pentru a obține informații despre:
 - utilizatori,
 - obiecte,
 - structurile de stocare;
- orice utilizator poate consulta dicționarul datelor pentru a afla informații despre baza de date (documentare sau administrare)
- utilizatorii fără privilegii de administrare pot accesa doar vizualizările prefixate de *USER_* sau *ALL_*
- pentru a obține lista vizualizărilor disponibile se poate interoga vizualizarea *DICTIONARY* care are sinonimul *DICT*
 - se utilizeaza instrucțiunea *SELECT* din *SQL*.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5. Arhitectura sistemului *Oracle*

5.3. Arhitectura interna a sistemului *Oracle*

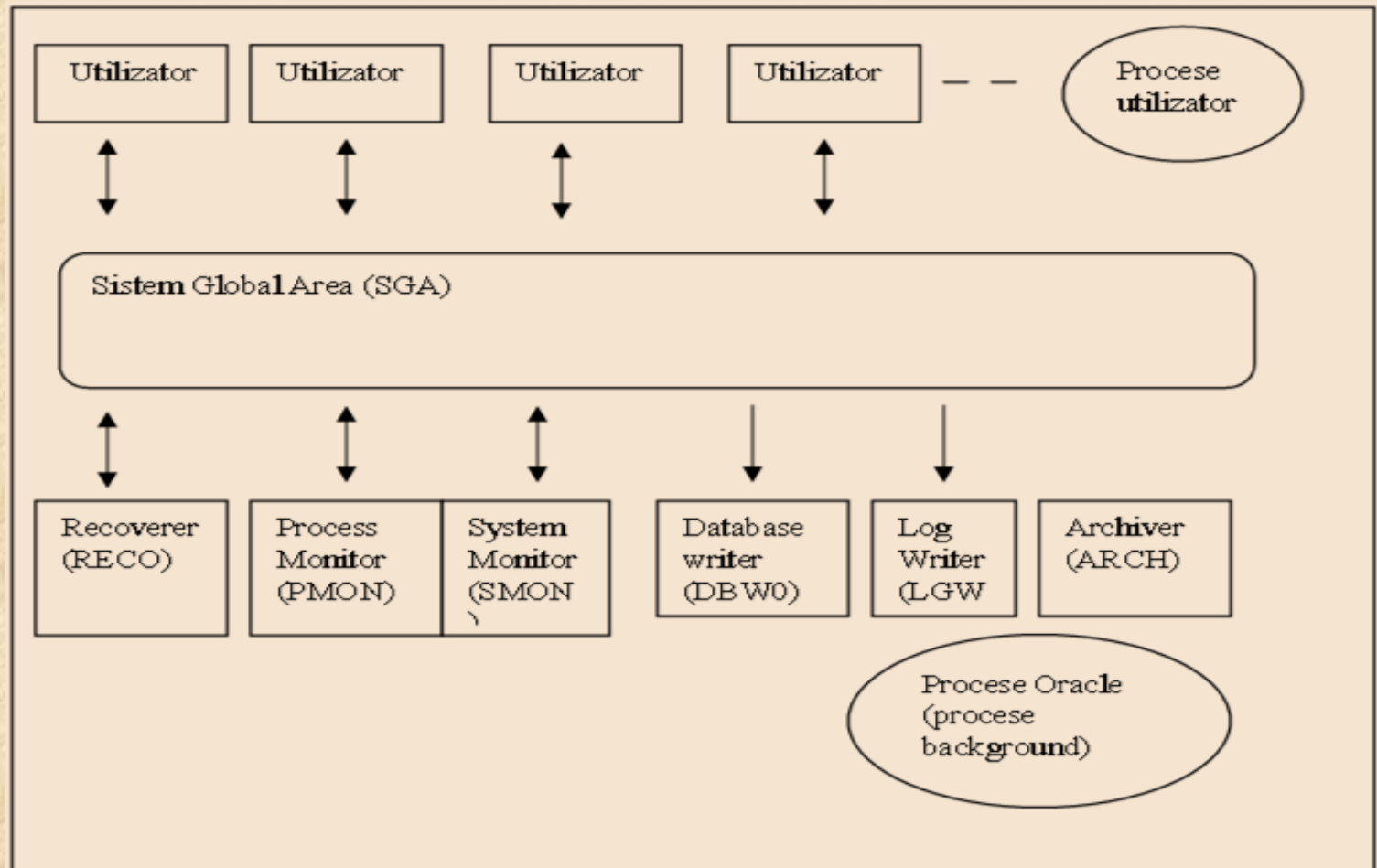
Observatie

Serverul *Oracle* = SGBD relațional OO

- constă
 - o bază de date *Oracle*
 - o instanță, care constă din:
 - o structură de memorie numită *SGA* (*System Global Area*)
 - procese *background* ,
- pentru a putea accesa datele din BD, trebuie pornită o instanță,
- o instanță poate fi asociată unei singure BD,
- când este pornită BD:
 - se localizează structura *SGA* și
 - sunt lansate procesele *background* (*DBWn*, *LGWR*, *CKPT*, *SMON*, *PMON*, *Dnnn*, *ARCn*, *RECO*, *LMS*, *QMnN* etc.).
 - dacă unul dintre aceste procese se termină anormal, atunci instanța se oprește.

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. *Oracle*
6. Grid computing și SOA in Oracle.

5.3. Arhitectura interna a sistemului *Oracle* (cont.)



- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

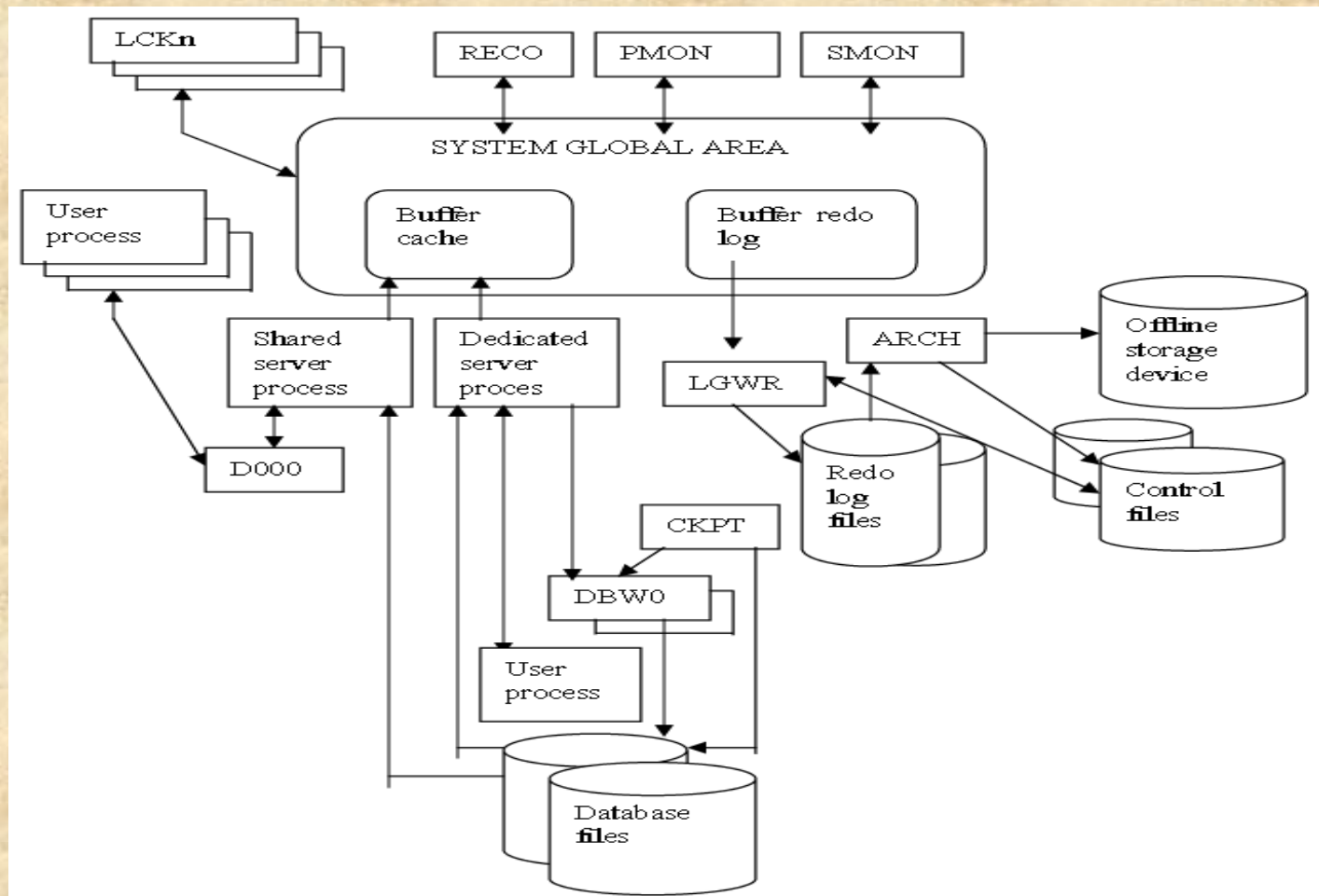
5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Sistemul *Oracle* este structurat pe trei nivele:
 - nivelul Procese: corespunde diverselor procese de sistem care asigură gestiunea datelor
 - nivelul Memorie: constă dintr-o mulțime de zone tampon alocate pentru a stoca date și anumite informații de control
 - nivelul Fișiere: corespunde structurii (fizice) a BD și modului în care sunt stocate datele.

1. Notiuni generale
2. Arhitectura SGBD
3. Evolutia bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistem. Oracle
6. Grid computing și SOA in Oracle.

5.3. Arhitectura internă a sistemului Oracle (cont.)

Structura memoriei si procesele Oracle



- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Serverul *Oracle* poate folosi fișiere care nu fac parte din baza de date; ele permit
 - configurarea instanței,
 - autentificarea utilizatorilor
 - recuperarea bazei de date;
- Exemple:
 - fișierele de parole (*password file*) = fișiere binare folosite pentru autentificarea utilizatorilor bazei de date
 - fișierul parametrilor de inițializare (*parameter file*) = principalul mijloc de configurare a sistemului; este utilizat pentru definirea caracteristicilor unei instanțe *Oracle*
 - fișierele *archived redo log* = copii *offline* ale fișierelor de reluare, folosite pentru recuperarea bazei de date în cazul defecțiunilor *hardware*
 - fișierele istorice (*trace file* și *alert file*) = conțin toate mesajele, erorile și evenimentele importante.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
- Arhitectura memoriei.

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
 - Arhitectura memoriei (SGA; PGA).
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor

Observatie

Pentru a accesa o instanță a unei BD *Oracle*, se execută:

- o aplicație sau un utilitar *Oracle* (prin intermediul cărora se lansează comenzi SQL asupra bazei – de ex. *Recovery Manager*, *Oracle Enterprise Manager*, *Oracle Forms*)
- un cod *Oracle* server (cu ajutorul căruia sunt interpretate și procesate comenzile SQL);

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
 - Arhitectura memoriei (SGA; PGA).
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor
 - Un proces = un mecanism al sistemului de exploatare care permite executarea unor operații de calcul sau operații I/O;
 - Fiecărui proces i se alocă o zonă privată de memorie
 - Serverul *Oracle* : două tipuri generale de procese:
 - procese *user*: execută aplicațiile,
 - procese *Oracle* (procese *server* și *background*): asigură gestiunea informațiilor dintr-o bază de date.

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
 - Arhitectura memoriei (SGA; PGA).
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor (cont.)
 - Un proces *user*
 - creat de sistemul *Oracle* pentru:
 - a executa codul unei aplicații program sau
 - ca urmare a lansării unui utilitar *Oracle*
 - se execută pe mașina client
 - începe și se termină odată cu aplicația utilizatorului resp.;
 - nu interacționează în mod direct cu serverul *Oracle* ci generează mesaje printr-un program interfață (*UPI* = *User Program Interface*).

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
 - Arhitectura memoriei (SGA; PGA).
-

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor (cont.)
 - Un proces *Oracle*
 - = execută instrucțiunile interne ale serverului *Oracle*
 - este invocat de alte procese pentru a îndeplini anumite operații în favoarea acestora;
 - două tipuri de procese *Oracle*:
 - procese server (*server process*),
 - procese de fundal (*background process*).

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
- Arhitectura memoriei (SGA; PGA).

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor (cont.)
 - Procese *Oracle*
 - Un proces *server*
 - interacționează cu procesele *user* ,
 - comunică în mod direct cu serverul *Oracle* pentru a transmite cererile acestora printr-un program de interfata *Oracle* (*OPI* = *Oracle Program Interface*),
 - este lansat când utilizatorul inițiază o sesiune.

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
- Arhitectura memoriei (SGA; PGA).

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura proceselor (cont.)
 - Procesele *Oracle*
 - Un proces de fundal (*background process*)
 - reunește funcțiile executate pentru fiecare proces *user*
 - execută operațiile I/O asincrone,
 - monitorizează alte procese *Oracle*;
 - folosit pentru a îmbunătăți performanțele unui sistem multiprocesor, în prezența mai multor utilizatori,
 - serverul *Oracle* creează câte un set de procese *background* pentru fiecare instanță;

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
- Arhitectura memoriei (SGA; PGA).

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura memoriei
 - structural, memoria este compusă din:
 - o zona de memorie partajata = zona globală sistem (SGA = *System Global Area*),
 - o zona de memorie nepartajata = zona globală program (PGA = *Program Global Area*);
 - toate structurile de memorie se găsesc în memoria centrală,
 - sunt create și utilizate pentru a depozita:
 - codul programelor executate,
 - datele necesare în timpul execuției acestora,
 - datele folosite în comun de mai multe procese *Oracle*,
 - informațiile referitoare la sesiunile curente etc.

5.3. Arhitectura interna a sistemului Oracle

- Arhitectura proceselor
 - procese *user*
 - procese Oracle
 - procese *server*
 - procese *background*
- Arhitectura memoriei (SGA; PGA).

5.3. Arhitectura interna a sistemului Oracle (cont.)

- Arhitectura memoriei (cont.)
 - **SGA** = *System Global Area* = *zona globala sistem* =
= este un grup de structuri partajate de memorie care conțin date și informații de control relative la BD și la o instanță;
 - fiecare instanță are propria sa SGA care:
 - este alocata atunci când este pornita instanța
 - este eliberata în momentul opririi instantei;
 - datele conținute în SGA sunt folosite în comun de către utilizatorii conectați la instanță,
 - informațiile conținute în SGA sunt repartizate în diferite zone (*database buffer cache*, *redo log buffer*, *shared pool* etc.), care sunt alocate la pornirea instanței;
 - **SGA fixă**
= o zona specială a SGA folosită pentru stocarea informațiilor despre starea bazei de date și a instanței
 - informațiile sunt accesate de către procesele *background*,
 - nu poate conține date ale utilizatorilor.

5.3. Arhitectura interna a sistemului *Oracle*

- Arhitectura proceselor
 - procese *user*
 - procese *Oracle*
 - procese *server*
 - procese *background*
- Arhitectura memoriei (SGA; PGA).

5.3. Arhitectura interna a sistemului *Oracle* (cont.)

- Arhitectura memoriei (cont.)
 - **PGA** = *Program Global Area* = zona globala program =
= zonă de memorie care conține date și informații de control relative la un singur proces *Oracle*
 - poate fi folosită de un singur proces,
 - este alocată la crearea procesului,
 - este dealocată la terminarea acestuia,
 - este formată - in general – din:
 - o zonă privată *SQL* (conține date - de ex., informații de legătură și structuri de memorie necesare rulării comenzilor)
 - o zonă de memorie alocată sesiunii,
 - zone de lucru *SQL*.

- | | | |
|---------------------|---------------------------------|--------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. <i>Oracle</i> |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5. Arhitectura sistemului *Oracle*

5.4. Prelucrarea bazei de date

- Operatii esentiale:
 - crearea BD,
 - inchiderea BD,
 - [izolarea BD];
- O BD *Oracle* = disponibilă utilizatorilor din momentul deschiderii și până la închiderea ei
- Un utilizator *Oracle* = o persoană
 - care posedă un cont înregistrat în serverul *Oracle* și
 - pentru care au fost acordate privilegii corespunzătoare de acces la date.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.4. Prelucrarea bazei de date (cont.)

- **Crearea bazei de date =**
 - definirea și
 - implementarea celor 2 tipuri de structuri (logică și fizică) ce compun baza;
 - ✓ Modalitățile de creare a unei BD:
 - în mod automat, la instalarea lui *ODBServer*, utilizând:
 - *Oracle Universal Installer*
 - comanda *CREATE DATABASE* din SQL
 - instrumentul *Oracle Database Configuration Assistant*;
 - în timpul migrării de la o versiune anterioară a bazei deja existente, folosind *Oracle Data Migration Assistant* etc.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.4. Prelucrarea bazei de date (cont.)

- **Inchiderea bazei de date**
 - se folosește comanda *SHUTDOWN* (este necesară conectarea ca *SYSDBA* sau *SYSOPER*).
 - se poate realiza în mai multe moduri:
 - normal,
 - imediat,
 - tranzacțional sau
 - renunțare (*abort*);
 - procesul de închidere a unei BD presupune trei etape:
 - închiderea BD:
 - datele din *SGA* sunt înregistrate în fișierele de date și în cele de reluare,
 - aceste fișiere sunt închise,
 - fișierele de control rămân deschise;
 - demontarea BD:
 - BD este disociată de instanță,
 - sunt închise fișierele de control;
 - oprirea instanței:
 - se eliberează memoria utilizată de *SGA*,
 - se opresc toate procesele *Oracle*,
 - se închid fișierele istorice.

- | | | |
|---------------------|---------------------------------|-------------------------------------|
| 1. Notiuni generale | 3. Evolutia bazelor de date | 5. Arhitectura sistem. Oracle |
| 2. Arhitectura SGBD | 4. Clasificarea bazelor de date | 6. Grid computing și SOA in Oracle. |
-

5.4. Prelucrarea bazei de date (cont.)

- **Izolarea bazei de date**

= operatie optionala executata de DBA (de ex.: cand trebuie sa izoleze anumite actiuni pe care trebuie sa le efectueze de actiunile concurente realizate de utilizatorii obișnuiți (tranzacții, interogări sau comenzi *PL/SQL*): modificarea structurii unui tabel poate fi impune interzicerea tranzacțiilor concurente care accesează datele acelui tabel)

- presupune închiderea acesteia și redeschiderea sa în mod restrictiv (*STARTUP RESTRICT*)
- in cazul sistemelor care trebuie să funcționeze continuu, oprirea temporară a BD poate cauza probleme importante.
 - s-a introdus posibilitatea de izolare a BD prin trecerea sa în **stare de repaus**, fără a deconecta utilizatorii.

Curs 6 : Baze de date Oracle: Generalitati

1. Notiuni generale
2. Arhitectura SGBD
3. Clasificarea bazelor de date
4. Clasificarea bazelor de date
5. Arhitectura sistemului *Oracle*
6. *Grid computing* și *SOA* in *Oracle*.