

UTILIZAREA BAZELOR DE DATE

CURS: FLORIN RADULESCU
Email: florin.radulescu@cs.pub.ro

SITE: CS.CURS.PUB.RO

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

1

NOTARE

- ◆ 60% IN CURSUL SEMESTRULUI:
 - PREZENTA CURS 10%
 - PREZENTA, ACTIVITATE SI TEST LABORATOR 35%
 - LUCRARE LA MIJLOCUL SEMESTRULUI FARA DEGREVAR (SAPTMAMA 8-9): 15%, ACEASTA LUCRARE NU SE poate REFACE IN SESIUNEA DE EXAMENE
- ◆ 40% VERIFICARE FINALA (EXAMEN)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

2

Incepem?

Administrarea bazelor de date DEFINITII

- Definiti care se pot gasi in Internet:
- ◆ A technical function that is responsible for
 - physical database design
 - security enforcement,
 - database performance,
 - backup and recovery,

(http://wps.prenhall.com/wps/media/objects/1374/1407505/glossary_Terms.htm)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

8

REGULI DE TRECERE

- ◆ 50% din punctajul din timpul semestrului (30 pct. din 60)
și
- ◆ 50% din punctajul de la examen (20 pct. din 40)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

3

Nelamuriri?

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

4

DEFINITII (2)

- ◆ An area of IT that
 - develops,
 - implements,
 - updates,
 - tests, and
 - repairs

a company's server database.

(<http://www.niu.edu/home/fest/1class%203/KnowTerms%20%20less%203.htm>)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

9

DEFINITII (3)

- ◆ Database Administration involves the overall **design** and **management** of the database. Administration tasks include:
 - archiving,
 - consistency checks,
 - developing/maintaining indexing and retrieval functionality,
 - migration,
 - monitoring,
 - performance issues,
 - replication issues, and
 - database sizing/space management,

(http://topus.com/wiki/index.php/Database_Administration)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

10

DRAFT CONTINUT

- ◆ Administrarea bazelor de date relationale (Oracle)
- ◆ Notiuni de Data Mining

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

5

BIBLIOGRAFIE

- ◆ Va fi indicata la fiecare capitol
- ◆ Pentru fiecare capitol, la bibliografie exista documente care aprofundeaza ceea ce s-a predat la curs si care fac parte integranta din materia pentru lucrarea de la mijlocul semestrului si din cea pentru examen.

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

6

JOB PROFILE

- ◆ A database administrator (DBA) is an IT professional responsible for the integrity, performance and security of databases in an organization.
- ◆ The role includes the development and design of database strategies, system monitoring and improving database performance and capacity, and planning for future expansion requirements.
- ◆ They may also plan, co-ordinate and implement security measures to safeguard the database

(<http://en.wikipedia.org>)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

11

JOB PROFILE - cont

- Duties:
- ◆ Managing and upgrading the database server and application tools
 - ◆ Allocating system storage and planning future storage requirements for the database systems
 - ◆ Modifying the database structure, as necessary, from information given by the DBA
 - ◆ Enrolling users and maintaining system security
 - ◆ Ensuring compliance with database vendor license agreement
 - ◆ Controlling and monitoring user access to the database
 - ◆ Monitoring and improving the performance of the database
 - ◆ Planning for backup and recovery of database information
 - ◆ Maintaining archived data
 - ◆ Backing up and restoring databases
 - ◆ Contacting database vendor for technical support
 - ◆ Generating various reports by querying from database as per need

(<http://en.wikipedia.org>)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

12

CU CE INCEPEM?

Administrare ORACLE

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

13

Capitolul 1

Arhitectura Oracle

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

14

SESIUNE

- ◆ O sesiune de lucru reprezinta o conexiune a unui user cu serverul Oracle.
- ◆ Un user poate avea mai multe sesiuni deschise simultan:
 - Pe aceiasi baza de date
 - Cu acelasi cont de user
 - Pe aceiasi masina sau pe masini diferite

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

19

PROCESUL USER

- ◆ Este creat atunci cand un utilizator incepe o sesiune de lucru folosind fie un client care interacționează direct cu Oracle (SQL*Plus) fie prin interfața Oracle (Server Manager, Developer, etc)
- ◆ Ruleaza pe masina utilizatorului
- ◆ Asigura interfață - grafica unei interfețe pentru acesta (GUI, API, User Program Interface)
- ◆ Procesul se încheie cand utilizatoruliese din programul folosit
- ◆ Trimite comenzi/cereri utilizatorului catre procesul server și afisează rezultatele (date, stare, evenualele mesajele de eroare)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

20

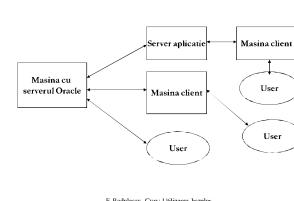
Serverul ORACLE

- ◆ Este un sistem de gestiune a bazelor de date relationale
- ◆ Utilizator poate lucra:
 - Cu un client pe aceeasi masina cu serverul (de exemplu un client SQL*Plus runand pe aceeasi masina cu serverul Oracle)
 - Clientul ruleaza pe o alta masina decat serverul (two-tiered)
 - Aplicatia utilizatorului acceseaza o alta aplicatie iar la randul ei aceasta e in comunicatie cu serverul (three-tiered)

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

15

Serverul ORACLE



F.Radulescu_Curs Utilizarea bazelor de date si IV CS

16

PROCESUL SERVER

- ◆ Ruleaza pe masina unde este instalat serverul Oracle
- ◆ Deserveste un singur proces user sau mai multe procese user, in functie de configuratie: Dedicated Server sau Multithreaded server
- ◆ Foloseste o zona de memorie (PGA – Program Global Area) care va fi descrisa in continuare.
- ◆ Este un intermediar intre procesul user si serverul Oracle, folosind OPI – Oracle Program Interface – pentru a interactiona cu acesta
- ◆ Procesul se incheie cand utilizatorul termina sesiunea.

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

21

INSTANTA ORACLE

- ◆ Un server Oracle consta dintr-o instanta Oracle si o baza de date Oracle.
- ◆ Instanta Oracle este compusa dintr-o structura de date in memoria numita SGA – System Global Area – si procese rulate in background care sunt utilizate de Oracle pentru gestiunea bazei de date.
- ◆ O instanta Oracle deschide o singura baza de date.
- ◆ Este identificata prin ORACLE_SID (variabila la nivel de SO), SID = System ID

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

22

USER vs UTILIZATOR

- ◆ Vom numi in cele ce urmeaza user un cont de utilizator Oracle (exemplu: user 'student' cu parola 'stud1234')
- ◆ Utilizator este o persoana care prin intermediul unui cont de user interactioneaza cu Oracle
- ◆ Utilizator poate fi numit intr-un sens mai larg si un proces, o aplicatie care foloseste un cont user pentru a interactiona cu Oracle

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

17

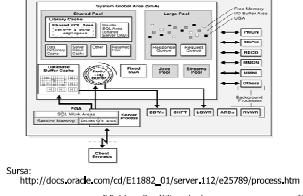
PROCESE

- ◆ Cand userul acceseaza o aplicatie care luceaza cu Oracle sunt create 2 procese:
 - Un "Proces user" (client) pe masina pe care luceaza utilizatorul. Acesta interactioneaza cu utilizatorul si asigura comunicatia cu cel de-al doilea proces
 - Un "Proces server", pe masina unde este instalat serverul Oracle. Acesta comunica cu serverul Oracle in numele procesului user.

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

18

INSTANTA ORACLE



Sursa: http://docs.oracle.com/cd/E11882_01/server.112/e25789/process.htm

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

23

SGA – System Global Area

- ◆ Este alocata in memoria virtuala a sistemului pe care ruleaza serverul Oracle.
- ◆ Contine date si informatii de control
- ◆ Este de tip partajat – shared – mai multi utilizatori pot folosi datele de alii pentru a se evita accesul repetat la disc
- ◆ Contine mai multe componente dintre care cele mai importante sunt:
 - Buffer Cache
 - Redo Log Buffer
 - Shared Pool

F.Radulescu_Curs Utilizarea bazelor de date si IV CS

24

SGA – System Global Area -cont

◆ Mai exista de asemenea:

- Java Pool – utilizata pentru cod si date specific Java de catre Java Virtual Machine (JVM)
- Large Pool – optional, pentru date de mari dimensiuni.
- Streams Pool – folosita de produsul Oracle Streams

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

25

Procese BACKGROUND

Sunt de trei tipuri:

- ◆ A Obligatorii – apar in toate configuratiile uzuale:
- ◆ Database Writer (DBWn) – Este responsabil cu scrierea blocurilor de date modificate/insertate din bufferurile de memorie in fisierle de pe disc. In Oracle 10g pot fi maximum 20 de procese de acest fel
- ◆ Log Writer (LGWR) – Scrie datele din Redo Log Buffer pe disc. Scrierea se face secentual intr-un fisier de Redo Log.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

26

Procese BACKGROUND - cont

B. Optionale

- ◆ Archiver (ARCn) – Copiază fisierile Redo Log în arhiva de pe disc atunci cand acestea sunt pline sau cand acestea se schimbă. Actiunea are loc în anumite condiții. Pot fi mai multe procese de acest fel.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

31

Procese BACKGROUND - cont

B. Optionale – continuare

- ◆ Job Queue Processes: sunt folosite mai ales la lucru pe Ioturi pentru executarea unui job la un moment dat sau repetat. Există:
 - Job queue coordinator process (CJQ0)
 - Job queue slave process (Jnnn) – pot fi mai multe
- ◆ Flashback Data Archiver Process (FDBA) – pastrează copii ale linioilor modificate în anumite tabele
- ◆ Space management Coordinator Process (SMCO) – pentru gestionarea spațiului

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

32

Procese BACKGROUND - cont

A. Obligatorii – continuare

- ◆ Checkpoint (CKPT) – Acest proces scrie periodic pe disc toate blocurile de date (din buffer) care au fost modificate. O astfel de acțiune este denumita **checkpoint**.
- ◆ Procesul anunță acțiunea lui DBWR, actualizează fisierile de date și control ale bazei de date și înregistrează momentul în care s-a facut checkpoint.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

27

Procese BACKGROUND - cont

A. Obligatorii – continuare

- ◆ System Monitor (SMON) – Face verificarea consistenței datelor și inițiază recuperarea după incident atunci când o instanță Oracle are un accident reportez.
- ◆ Process Monitor (PMON) – Dealcă resursele unui proces care are un accident. Folosit pentru procesele user care au incidente.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

28

Procese BACKGROUND - cont

C. Procese 'slave'

- ◆ Sunt lansate de celelalte procese pentru a efectua diverse acțiuni.
- Nota: O listă completă a proceselor de background poate fi găsită de exemplu la adresa:

http://docs.oracle.com/cd/E11882_01/server.112/e40402/bgprocesses.htm

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

33

Baza de date

- ◆ A două componente a serverului (pe langa instanță) este Baza de date
- ◆ Este identificată prin DB_NAME (variabila \$ORACLE_SID)
- ◆ Oracle recomandă ca instanța să își aibă același nume (pot fi diferite) pentru utilizatorul administrativ
- ◆ Este compusă din fisiere de mai multe tipuri

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

34

Procese BACKGROUND - cont

A.Obligatorii – continuare

- ◆ Manageability Monitor Processes : Management monitor process (MMON) - efectuează operații legate de Automatic Workload Repository (AWR). Aceasta conține date istorice despre performanțele sistemului: statistici pentru sistem, sesiuni, cereri SQL individuale și servicii. E folosit pentru tuning.
- Management monitor lite process (MMNL) - scrie statisticile din Active Session History (ASH - buffer în zona SGA) pe disc atunci când acest buffer se umplă.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

29

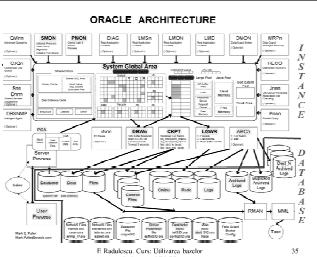
Procese BACKGROUND - cont

A.Obligatorii – continuare

- ◆ Recoverer Process (RECO) – acesta este folosit în cazul bazei de date distribuite pentru rezolvarea incidentelor apărute la tranzacțiile distribuite.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

30



F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

35

Tipuri de fisiere

- ◆ Data files – fisiere de date. Fisierele de date contin:
 - Dictionarul de date al BD (mai stii ce este acesta?)
 - Obiectele utilizatorului (mai stii care sunt acestea?)
 - Contin și vechile valori ale datelor modificate de tranzacțiile curente ('before image')
- ◆ O baza de date are cel putin un astfel de fisier.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

36

Tipuri de fisiere - cont

- ◆ Redo Log Files – Fisiere Redo Log. Contin modificările facute în baza de date. Sunt necesare la reconstrucția ei în caz de incident.
- ◆ Fiecare baza de date conține cel puțin 2 astfel de fisiere scrise în paralel și care pot fi tinute pe dispozitive de stocare diferite, prevenind pierderi de date în cazul în care un disc este distrus.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

37

Tipuri de fisiere - cont

- ◆ Control files – fisiere cu date de control. Contin informații necesare pastrării integrității bazei de date
- ◆ Fiecare baza de date are cel puțin un astfel de fisier.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

38

Library cache

- ◆ Stocă informații despre cele mai recente cereri SQL utilizate:
 - Textul cererii
 - Arborul cererii rezultat în urma etapei Parse (analiza semantică). Acesta este versiunea completă a textului cererii
 - Planul de execuție al cererii rezultat în urma optimizării.
- ◆ Dacă o cerere este re-exjecțuită până să apară alte cereri care să afecteze planul de execuție etapa Parse nu mai este necesară la re-execuție (se folosesc rezultatul existent).

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

43

Data Dictionary Cache

- ◆ Contine cele mai recent folosite informații din dictionarul de date:
 - Descrierile tabelelor
 - Date cont user
 - Drepturi (privilegii)
 - Etc.
- ◆ În etapa Parse acest cache și folosit de procesul server pentru informații necesare compilării cererii (numele folosite în cerere analizată). Dacă nu există sunt încarcate din fisierele de pe disc.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

44

Alte fisiere

- ◆ Pe langa fisierile bazei de date, Oracle mai utilizează și alte fisiere, ca de exemplu:
 - Fisier de parametri: conține parametri care caracterizează instanța Oracle
 - Fisier de parole: utilizat pentru autentificarea utilizatorilor
 - Fisiere Redo Log arhivate: copii offline ale fisierelor Redo Log.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

39

Etape de procesare cerere

- ◆ Există mai multe etape parcuse de o cerere de la emiterea ei de către utilizator până la execuția completă. Exemplu pentru o cerere SELECT:
 - Pasul 1: Parse: cererea primă de la utilizator este analizată sintactic și semantic (compilată). Serverul întoarce o informație de stare (OK sau eroare). Shared Pool este folosit ca zona de memorie pentru această operare

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

40

Database Buffer Cache

- ◆ Partea a SGA. Tine cele mai recent utilizate blocuri de date.
- ◆ Cand o cerere este executată procesul server verifică dacă există blocurile necesare. Dacă nu există le citește și le plasează aici.
- ◆ Dimensiunea sa este data de parametru DB_BLOCK_BUFFERS
- ◆ Dimensiunea unui bloc e data de parametrul DB_BLOCK_SIZE.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

45

Database Buffer Cache – cont.

- ◆ Oracle utilizează algoritm LRU (ce este acesta?) pentru eliberare poziții în buffer.
- ◆ Excepție: operațiile de tip 'full table scan' (când se parcurge întregul tabelă – deci este citită în memoria). În acest caz ultimele blocuri citite pot fi primele dealocate.
- ◆ Algoritmii utilizati sunt complexi, cele de mai sus sunt o prezentare schematică a strategiilor de gestiune a bufferului. (vezi de exemplu: <http://www.oraclenbg.ch/ora/concepts/cache.htm>)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

46

Etape de procesare cerere - cont

- ◆ Pasul 2: Execute. Cererea este executată și datele din tabela rezultat sunt pregătite pentru a fi returnate utilizatorului.
- ◆ Pasul 3: Fetch. Linile sunt returnate de către utilizator (procesul user) pentru a fi procesate (afisare sau procesare). În funcție de dimensiunea datelor returnate se poate executa una sau mai multe operații de tip FETCH

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

41

SHARED POOL

- ◆ Este parte a SGA. Este utilizată și în pasul 1 (parse) de către cererile de la utilizator.
- ◆ Dimensiunea sa e data de parametrul SHARED_POOL_SIZE (din fisierul de parametrii).
- ◆ Pentru Pasul 1 sunt utilizate următoarele componente ale Shared Pool:
 - Library cache
 - Data dictionary cache

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

42

Program Global Area (PGA)

- ◆ Zona de memorie folosită în mod exclusiv de un proces (de tip server sau background). Nu este comună ca în cazul SGA. Ea conține:
 - Sort area – zona sortării, folosită la operațiile de ordonare (sortare).
 - Informații sesiunii, de exemplu drepturile utilizatorului sesiunii,
 - Starea cursorilor folosiți în sesiunea respectivă (dacă există)
 - Stiva, conținând diverse variabile de sesiune,

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

47

Exemplu: procesare UPDATE

- ◆ Se luam exemplul unei cereri de actualizare:


```
UPDATE EMP
SET SAL = SAL * 1.1
WHERE EMPNO=1123
```
- ◆ Se executa într-un Pasul PARSE
- ◆ La etapa EXECUTE (Pasul 2) se efectuează operațiile:

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

48

Exemplu: procesare UPDATE – cont

1. Procesul server citeste blocurile de date si de rollback – valori anterioare modificarilor necomise - din fisierile de date (daca nu sunt deja in Buffer Cache)
2. Copii ale blocurilor sunt puse in Buffer Cache (data nu exista).
3. Procesul server blocheaza datele respective,
4. Procesul server inregistreaza in Redo Log Buffer schimbarile de facut in rollback si in date

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

49

Exemplu: procesare UPDATE – cont

5. Procesul server plaseaza versiunile originale ale blocurilor (nemodificate) in Rollback si modifica blocurile de date, Ambele operatii se fac in Buffer Cache. Ambele blocuri (rollback si date) sunt marcate ca 'dirty blocks' (blocuri modificate), adica blocuri care nu sunt identice cu cele de pe disc.

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

50

Database Writer

- ◆ Este un proces de tip background
- ◆ Scrie blocurile modificate - 'dirty blocks' - din Buffer Cache in fisierile de date, astfel incat sa existe suficiente blocuri libere care sa fie folosite de sistem
- ◆ A fost necesar pentru a degreva procesul server de aceasta operatie – imbunatatirea performantelor

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

55

Database Writer - cont

- ◆ Scrierea se face cand:
 - Numarul de 'dirty blocks' in buffer depaseste o anumita valoare
 - Un proces care cauta locuri libere in buffer nu le gaseste
 - Timeout (la fiecare N secunde)
 - Evenimente care forteaza un checkpoint: exemplu: inchiderea bazei de date.

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

56

Segment de Rollback

- ◆ Inainte de a se face schimbari, serverul salveaza vechile valori de bloc intr-un segment de rollback.
- ◆ Aceasta salvere permite anularea tranzactiei (operatiune ROLLBACK, opusa lui COMMIT), asigura ca alte tranzactii pot citi valorile anterioare inceputului de tranzactie (read consistency – mai stii ce era asta?) si ne permit de asemenea recuperarea dupa incident.

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

51

Segment de Rollback - cont

- ◆ Segmentele de rollback sunt stocate in fisierile de date ale bazei de date si sunt aduse in buffer cache la cerere (cand este nevoie de ele).

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

52

Log Writer

- ◆ Este un proces de tip background
- ◆ Scrie intrari din Redo Log Buffer in fisierile de pe disc.
- ◆ Operatia se efectueaza cand:
 - Redo Log Buffer e aproape plin
 - Timeout (ca mai sus)
 - Inainte ca DBWR sa scrie blocurile modificate pe disc
 - Cand o tranzactie e comisă

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

57

COMMIT

- ◆ Oracle utilizeaza un mecanism rapid de commit care garanteaza recuperarea schimbarilor comise in caz de incident.
- ◆ Oracle asigneaza un "commit System Change Number" (SCN - este de tip timestamp) fiecarei tranzactii care se comite. Aceste numere sunt crescatoare si unice la nivelul bazei de date

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

58

Redo Log Buffer

- ◆ Procesul server inregistreaza schimbarile facute de o instanta in Redo Log Buffer.
- ◆ Are o dimensiune de parametrul LOG_BUFFER (in bytes),
- ◆ Contine inregistrari Redo: blocul care a fost schimbat, pozitia schimbarii, noua valoare.



F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

53

Redo Log Buffer - cont

- ◆ Contine toate schimbarile, atat in date cat si in blocuri de index, rollback, etc.
- ◆ Scrierea acestor inregistrari este securventiala.
- ◆ Contine inregistrari privind schimbarile facute de toate tranzactiile.
- ◆ Este folosit prin parcurgere circulara. Cand un bloc e refolosit, el este scris anterior in fisierile de pe disc.

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

54

COMMIT - Pasii

- ◆ La aparitia unui COMMIT:
 1. Procesul server plaseaza o inregistrare de commit, impreuna cu SCN-ul acestuia in Redo Log Buffer
 2. LGWR scrie o portiune contigua de inregistrari din buffer pana la cea care contine COMMIT-ul in fisierile Redo Log de pe disc. In felul acesta este garantata recuperarea dupa incident

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

59

COMMIT - Pasii

3. Userul este informat ca s-a efectuat COMMIT-ul.
 4. Procesul server inregistreaza ca tranzactia e completa si ca resursele blocate de ea pot fi eliberate.
- Deci:
- ◆ Scrierea efectiva a datelor pe disc este efectuata independent de DBWR.
 - ◆ Dimensiunea tranzactiei nu conteaza

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

60

Sumar

1. Server Oracle = Instanta + Baza de date,
2. Baza de date = Ansamblu de fisiere de diverse tipuri: de date, de redo log, de control, de parametri, samb.
3. Instanta = Zona de memorie SGA + Procese de background
4. SGA contine: Buffer cache, Redo log buffer, Shared Pool si altele; e comun tuturor proceselor server.
5. Procese de background = 3 categorii fiecare continand mai multe tipuri, putand fi uneori mai multe procese de acelasi tip,

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

61

Sumar - continuare

6. Printre procesele de background avem:
 1. DBWR – Database writer: scrie date din Buffer cache pe disc; sunt blocuri de date si de rollback,
 2. LGWR – Log writer: scrie inregistrari din Redo log buffer pe disc; ele contin modificarile efectuate in date,
 7. La momentul unui COMMIT nu se scriu efectiv datele pe disc (DBWR) ci doar inregistrările de redo log (LGWR). Datele vor fi scrise ulterior de DBWR.

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

62

Bibliografie generala

1. Colin McGregor - Oracle Database 2 Day DBA, 11g
Link: http://docs.oracle.com/cd/E11882_01/server.112/e10897.pdf
2. Ulrike Schwinn, Vijayanandan Venkatachalam – Database administration

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

63

Lecturi obligatorii

1. Colin McGregor - Oracle Database 2 Day DBA, 11g
Link: http://docs.oracle.com/cd/E11882_01/server.112/e10897.pdf
 - ◆ Capitolul 5 (Managing the Oracle Instance) – integral
 - ◆ Capitolul 6 (Managing Database Storage Structure) pag. 6-1 pana la 6-5 (fara Tablespaces)

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

64

Sfârșitul primului capitol

F.Rădulescu_Curs Utilizarea bazei de date_and IV CS.

65

Capitolul 2

Instanta si baza de date

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

1

DBA

- ◆ Există doi utilizatori privilegiati care sunt creati încă de la instalarea Oracle (se cere doar parola pentru ei la instalare):
 - SYS – proprietarul (owner) bazei de date precum și al tuturor tabelelor și vederilor din dicționarul bazei de date. Are rol de DBA
 - SYS are privilegiul SYSDBA - vom vedea ce e asta
 - Atenție: Nu creați/modificați niciodată obiecte în schema SYS (care ce e o schema?)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

2

Etapele pornirii unei BD

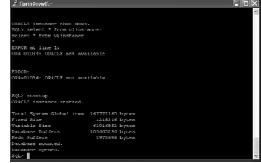
1. Pornirea (start) instantă
2. Montarea bazei de date (Mount)
3. Deschiderea bazei de date (Open)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

7

Etapele pornirii

◆ Exemplu:



8

DBA - cont

2. SYSTEM – are de asemenea rol de DBA. Este proprietarul (owner) celorlalte tabele și vederi de sistem Oracle, altele decât cele din dicționarul de date (ex: cele folosite de uneltele Oracle)
- Este bine să nu creați obiecte în schema SYSTEM

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

3

DBA vs SYSDBA

- ◆ DBA este un rol care conține majoritatea privilegiilor (drepтурilor) de sistem – de tipul "root" din Unix
- ◆ SYSDBA este un privilegiu de sistem
- ◆ DBA nu conține totuști două privilegi importante: SYSDBA și SYSOPER
- ◆ Acestea sunt privilegiu important care permit administratorului sa execute o serie de operații de administrare.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

4

Etapele pornirii unei BD

- ◆ La pornirea instantei Oracle folosește un fisier de parametri (init<SID>.ora) care este un fisier text.
- ◆ Dupa evenuale modificari, instantă trebuie opriță și reportată pentru a fi cînd noile valori.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

9

Exemplu de continut

- ◆ db_name=ORE
- ◆ db_block_size = 8192
- ◆ db_log_buffers = 100
- ◆ shared_pool_size = 3500000
- ◆ log_checkpoint_interval = 10000
- ◆ log_buffer = 32768
- ◆ log_file_size = 10
- ◆ max_dump_file_size = 10240
- ◆ background_dump_dest = ('/home/disk1/UDUMP')
- ◆ user_dump_dest = ('/home/disk1/UDUMP')
- ◆ rollback_segments = (01, 102)
- ◆ control_files = (ora_control1, ora_control2)
- ◆ compatible = 8.0.0

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

10

SYSDBA

- Poate efectua operații:
- ◆ STARTUP și SHUTDOWN
 - ◆ ALTER DATABASE: open, mount, back up, sau schimbarea setului de caractere
 - ◆ CREATE DATABASE
 - ◆ DROP DATABASE
 - ◆ CREATE SPFILE
 - ◆ ALTER DATABASE ARCHIVELOG
 - ◆ ALTER DATABASE RECOVER
 - ◆ Include privilegiul RESTRICTED SESSION

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

5

SYSOPER

- Poate efectua operații:
- ◆ STARTUP și SHUTDOWN
 - ◆ ALTER DATABASE OPEN/MOUNT/BACKUP
 - ◆ CREATE SPFILE
 - ◆ ALTER DATABASE ARCHIVELOG
 - ◆ ALTER DATABASE RECOVER (doar restaurare completă, Restaurarea incompletă - de tip UNTIL TIME | CHANGE| CANCEL| CONTROLFILE necesită privilegiul SYSDBA)
 - ◆ Include privilegiul RESTRICTED SESSION

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

6

Pornirea instantei

- ◆ Dupa momentul pornirii instantiei (fara montarea și deschiderea bazei de date) se pot executa operații:
 - Crearea bazei de date
 - Recrearea fisierelor de control
- ◆ Pornirea instantiei presupune:
 - Citirea fisierului de parametri init<SID>.ora
 - Alocarea SGA
 - Pornirea proceselor de background
 - Deschiderea fisierelor de tip TRACE și ALERT

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

11

Montarea BD

- ◆ În momentul în care instantă este pornita și baza de date montata (dar nu deschisa) se pot executa operații de menținere ca:
 - Redenumirea fisierelor bazei de date (Data files)
 - Activare/dezactivare arhivare fisiere Redo Log
 - Restaurarea bazei de date

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

12

Montarea BD - cont

- ◆ Montarea bazei de date presupune:
 - Asocierea unei baze de date cu o instantă deja pornită
 - Localizarea și deschiderea fisierelor de control specificate în fisierul de parametri
 - Citirea fisierelor de control pentru cunoașterea numărului fisierelor de date și de Redo log (fara a verifica existența lor fizica)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

13

Deschiderea BD

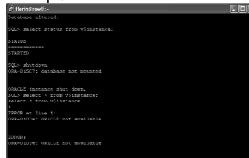
- ◆ Dupa deschiderea BD se poate opera normal cu baza de date. Userii se pot acum conecta și trimite cereri.
- ◆ Deschiderea presupune:
 - Deschiderea fisierelor de date
 - Deschiderea fisierelor Redo log.
 - Verificarea consistenței bazei de date. Daca este necesar, procesul SMON face o recuperare după incidență.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

14

Etapele opririi BD - cont

◆ Exemplu:



19

Deschiderea BD - cont

- ◆ Situațile în care se face recuperarea după incident sunt acelea în care instantă nă a reușit să efectueze toate operațiile (de exemplu în caz de crash de sistem).
- ◆ Recuperarea presupune actualizarea fisierelor de date pe baza modificărilor din fisierul Redo log (care sunt actualizate la fiecare COMMIT, deci efectele tuturor tranzacțiilor încheiate cu succesi sunt înregistrate aici).

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

15

Etapele opririi BD

- ◆ Sunt cele de la pornire, în ordine inversă:
 - Închidere BD
 - Demontare BD
 - Oprire instantă
- ◆ La închiderea BD Oracle scrie pe disc blocurile modificate din Buffer cache și înregistrează din Redo log buffer după care închide fisierile de date și Redo log
- ◆ Fisierile de control sunt închise la demontarea bazei de date.
- ◆ Deallocarea SGA și oprirea proceselor de background se fac la oprirea instantei,

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

16

Oprire normală

- ◆ Nu sunt permise noi conexiuni
- ◆ Oracle așteaptă ca toți userii deja conectați să termine sesiunea de lucru (sa se deconecteze)
- ◆ Închidere și demontare baza de date și oprire instantă
- ◆ Repornire normală (nu este nevoie de recuperare)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

21

Oprire tranzacțională

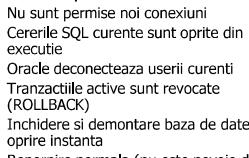
- ◆ Nu sunt permise noi conexiuni și nici tranzacții de la userii deja conectați
- ◆ La terminarea tranzacției curente pentru orice user acesta e deconectat
- ◆ Se executa apoi pasii de la oprirea imediată
- ◆ Repornire normală (nu este nevoie de recuperare)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

22

Etapele opririi BD - cont

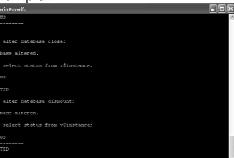
◆ Exemplu:



17

Etapele opririi BD - cont

◆ Exemplu:



18

Oprire imediata

- ◆ Nu sunt permise noi conexiuni
- ◆ Cerelele SQL curente sunt opuse din executie
- ◆ Oracle deconectează userii curenti
- ◆ Tranzacțiile active sunt revocate (ROLLBACK)
- ◆ Închidere și demontare baza de date și oprire instantă
- ◆ Repornire normală (nu este nevoie de recuperare)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

23

Oprire tip ABORT

- ◆ Nu sunt permise noi conexiuni
- ◆ Cerelele SQL curente sunt opuse din executie
- ◆ Oracle deconectează userii curenti
- ◆ Tranzacțiile active sunt revocate (ROLLBACK)
- ◆ Instantă este opriță fără închiderea fisierelor
- ◆ La repornire este necesara recuperarea după incidentă la instantei (procesul SMON)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

24

CREAREA BD

- In Oracle se poate crea o baza de date:
 - Folosind instrumentul DBCA - Database Configuration Assistant, un instrument de creare a bazelor de date
 - Manual, prin comanda SQL
- La instalarea Oracle de obicei se creaza o prima baza de date
- Se poate crea de asemenea o baza de date după instalare în cazuri ca:
 - Să folosiți Oracle Universal Installer (OUI) doar pentru instalare, fară a crea o altă bază de date
 - Criarea unei noi baze de date (și a unei noi instance)
 - Criarea unei baze de date care să fie o copie a uneia existente (donare)

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

49

Preliminarii

- Inainte de crearea unei baze de date trebuie să ne asigurăm că:
 - Oracle este instalat, deci există inclusiv variabilele de mediu și variabilele și sunt stabilite directoriale care vor gazdui datele și opționale
 - Există suficientă memorie internă pe masina în cauză pentru a putea lansa o instanță
 - Există suficient spațiu pe disc pentru crearea fisierelor necesare bazei de date
 - Utilizatorul care efectuează operații are privilegii necesare (este administrator de sistem de exemplu sau folosește un fișier de parole pentru autentificare (http://docs.oracle.com/cd/E11882_01/server.112/e10897.pdf)

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

50

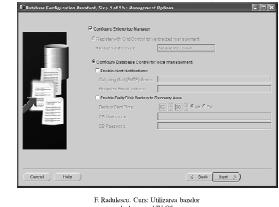
DBCA: 3. - cont

- În campul Global Database Name se tastează numele bazei de date care se creează
- În campul SID se tastează identificatorul instantei pentru baza de date
- Așa cum am spus anterior este recomandat ca SID-ul să fie același cu numele bazei de date din motive de siguranță administrativă

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

55

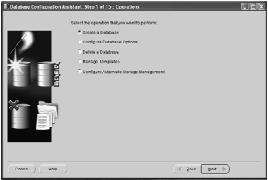
DBCA: 4. Optiuni de gestiune



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

56

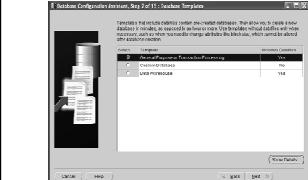
DBCA: 1. Primul pas



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

51

DBCA: 2. Tipul BD (DB template)



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

52

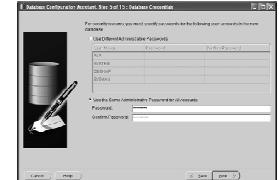
DBCA: 4 - cont

- Se poate configura administrarea bazei de date cu ajutorul unuiași Oracle Enterprise Manager. Acesta conține posibilități de gestionare (web based) pentru fiecare bază de date precum și o gestiune centralizată a întregului mediul Oracle.
- Se poate apoi selecta:
 - fi gestiunea centralizată (dacă Oracle Management Agent este instalat și este conectat la Centralized Register with Grid Control for centralized management)
 - fi gestiunea locală – folosită în continuare – selectând Configure Database Control for local management.
- În al doilea caz se poate opta pentru notificări prin e-mail asupra diverselor probleme aparute și pentru o salvare zilnică a bazei de date

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

57

DBCA: 5. Parole administrator



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

58

DBCA: 2 - cont

- Există câteva sabloane predefinite de Oracle
- General Purpose or Transaction Processing – pentru baze de date folosite tranzacțional (model ales în continuare)
 - Data warehouse (pentru depozite de date)
 - Se poate folosi opțiunea Custom Database care implica însă o bună cunoaștere a sistemului pentru configurație în acest caz. Timpul de creare pentru baza de date crește corespunzător

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

53

DBCA: 3. Numele bazei



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

54

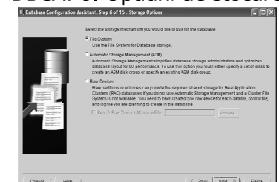
DBCA: 5 - cont

- Se pot specifica fie parole diferite pentru conturile de administrare fie aceeași parola pentru toate
- Conturile sunt cele din figura anterioară:
 - SYS
 - SYSTEM
 - DBSNMP – folosit de Oracle Management Agent, componenta a OEM (Oracle Enterprise Manager)
 - SYSMAN – folosit de asemenea de către OEM

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

59

DBCA: 6. Optiuni de stocare



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

60

DBCA: 6 – cont

- File system** – fisierele care compun baza de date vor fi stocate în sistemul de fisiere al SO folosit de mașina gazdă – este opțiunea folosită implicit
- Automatic Storage Management** – folosit în sisteme cu un mare număr de discuri. Descrierea acestei opțiuni se găsește în anexa A din Oracle Database 2 Day DBA (v. bibliografia)

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

61

DBCA: 6 – cont

- Raw devices** – permite stocarea în zone din afara sistemului de operare. Pentru aceasta trebuie specificată o zona de stocare pe disc neformata (în afara SO).
- Opțiunea Raw devices se folosește mai ales în RAC – Oracle Real Application Cluster.
- Zona respectivă trebuie anterior creată și liberă de orice altă folosire, inclusiv de folosirea ei de către o altă baza de date Oracle

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

62

DBCA: 9. BD de exemple



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

67

DBCA: 10. Parametri de initializare



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

68

DBCA: 7. Localizarea fisierelor



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

63

DBCA: 7. - cont

- Se pot alege opțiunile:**
 - Use Database File from Template** : crearea se face în directoarele din sablon (vezi pasul 2)
 - Use Common Location for All Database Files** : se specifică directorul unde vor fi create fisierile (ca în figura)
 - Use Oracle Managed Files** : Se specifică o zonă (numita database area) unde Oracle își va face singur gestiunea fisierelor. Nu mai trebuie specificate numele fisierelor, locația lor, dimensiunile acestora,

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

64

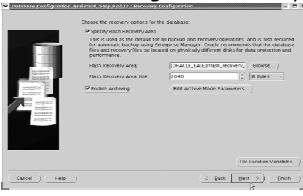
DBCA: 10 - cont

- Se pot seta parametri privind:
 - Tabl Memory (Memoria, cu opțiunile Typical sau Custom).
 - În cazul Typical putem vedea ce s-a alocat cu "Show..."
 - În cazul Custom putem seta pe Automatic (se vad valorile alocate) sau Manual (putem seta noi aceste valori)

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

69

DBCA: 8. Configurare recovery



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

65

DBCA: 8. - cont

- Aceste elemente de configurație se folosesc în caz de incident de sistem pentru recuperarea datelor (data recovery)
 - Se recomandă să fie pe alt disc decât cel pe care se află datele
 - Se specifică Flash Recovery Area (zona de backup si recovery) și dimensiunea ei
 - Se mai poate specifica și arhivarea fisierelor de tip Redo log

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

66

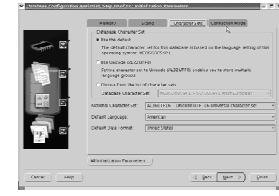
DBCA: 10 - cont

- Tabl Sizing**. Se setează dimensiunea blocului și numarul maxim de procese user care se pot conecta simultan.
 - Pentru dimensiunea blocului, în cauză în care se folosesc sabloane predefinite dimensiunea implicită și de 8KB
 - Pentru procese, numarul implicit este de 150. Trebuie să fie minim 6 pentru a include procesele de background.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

71

DBCA: 10 - cont



F.Rădulescu, Curs Utilizator baza de date, anul IV CS.

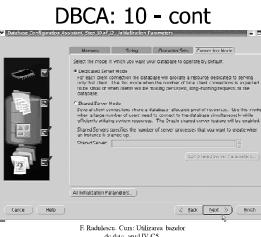
72

DBCA: 10 - cont

- ◆ Tabul Character Set specifica si setul de caractere utilizat pentru acea baza de date (VARCHAR2, CLOB). Se pot selecta:
 - Default – la setul limbii implicite a SO pentru toti utilizatorii BD respective
 - Unicode (AL32UTF8) pentru a putea folosi mai multe seturi de caractere (pentru user si aplicatii I/O)
 - Alegere din lista: ca prima optiune, dar se specifica setul prin alegere din lista
- ◆ Se mai pot specifica National Character Set (NVARCHAR2, NCLOB), Default Language si Default Date Format

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

73



F.Radulescu_Curs Utilizarea bazei de date_and IV CS

Crearea manuala a BD

- ◆ Pasii de urmat sunt urmatori: (vezi documentul)
 http://download.oracle.com/docs/cd/B25359_01/server_111/b10739/create.htm
- 1. Alegerea numeului instantei (SID)
- 2. Stabilierea metodelor de autentificare a administratorului (OS sau fisier de parola)
- 3. Crearea fisierelor de parametri (Initialization Parameter File)
- 4. Crearea DB instance
- 5. Crearea fisierului de parametri server (Server Parameter File)
- 6. Pornirea instantei
- 7. Executarea comenzii CREATE DATABASE
- 8. Crearea de TableSpace aditionala
- 9. Rularea scripturilor de creare a vederelor din Dictionarul de date
- 10. Rularea scripturilor de instalare a optimizatorilor (Optional)
- 11. Salvarea bazei de date astfel create (back up)

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

79

Exemplu pasul 7

Exemplu pasul 7 - cont

Efectul este:

- ◆ Se creeaza o baza de date cu numele **bazamea** (SID-ul creat la pasul 1 este acelasi)
- ◆ Sunt create fisierile de control specificate (ca nume) in fisierul de initializare (pasul 3) la **CONTROL_FILES**
- ◆ Sunt setate paroltele pentru utilizatori privilegiali SYS si SYSTEM (sys\$43 respectiv system\$55). In cazul in care acestia nu au parola, trebuie sa le setamem implicit **change_on_install=1** sau **change**
- ◆ Noua baza de date va avea in cazul din exemplu 3 fisiere de tip Redo log, specificata lor fiind in clauza **LOGFILE**
- ◆ **MAXDATAFILES** specifica numarul maxim de fisiere de date care pot fi deschise in baza de date

1. Din documentul: Colin McGregor - Oracle Database 2 Day DBA, 10g

Link: http://download.oracle.com/docs/cd/B25359_01/server_111/b10739/create.htm#I100870

- ◆ Capitalul 2 (Installing Oracle and Building the Database)
- 2. Crearea manuala a unei baze de date, descrisa in pagina:

http://download.oracle.com/docs/cd/B25359_01/server_111/b10739/create.htm#I100870

Lecturi obligatorii

Exemplu pasul 7 - cont

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

80

DBCA: 10 - cont

- ◆ Tabul Connection Mode:
 - Dedicated Server – fiecare proces server este pentru un proces user. Se folosete cand numarul de clienti nu e foarte mare sau cand clientii sunt conectati mult timp la baza de date (cereri care ruleaza mult timp)
 - Shared Server – mai multe procese client sunt deservite de acelasi proces server. Se folosete cand memoria e limitata sau numarul de clienti este mare

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

75

DBCA: 11. Parametri de stocare



F.Radulescu_Curs Utilizarea bazei de date_and IV CS

Exemplu pasul 7 - cont

Efectul este:

- ◆ Se creeaza o baza de date cu numele **bazamea** (SID-ul creat la pasul 1 este acelasi)
- ◆ Sunt create fisierile de control specificate (ca nume) in fisierul de initializare (pasul 3) la **CONTROL_FILES**
- ◆ Sunt setate paroltele pentru utilizatori privilegiali SYS si SYSTEM (sys\$43 respectiv system\$55). In cazul in care acestia nu au parola, trebuie sa le setamem implicit **change_on_install=1** sau **change**
- ◆ Noua baza de date va avea in cazul din exemplu 3 fisiere de tip Redo log, specificata lor fiind in clauza **LOGFILE**
- ◆ **MAXDATAFILES** specifica numarul maxim de fisiere de date care pot fi deschise in baza de date

1. Din documentul: Colin McGregor - Oracle Database 2 Day DBA, 10g

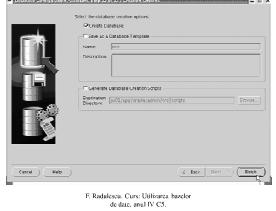
Link: http://download.oracle.com/docs/cd/B25359_01/server_111/b10739/create.htm#I100870

- ◆ Capitalul 2 (Installing Oracle and Building the Database)
- 2. Crearea manuala a unei baze de date, descrisa in pagina:

http://download.oracle.com/docs/cd/B25359_01/server_111/b10739/create.htm#I100870

Lecturi obligatorii

DBCA: 12. Ultima etapa



F.Radulescu_Curs Utilizarea bazei de date_and IV CS

77

DBCA: 12. - cont

- ◆ Sabioanele (a doua optiune) sunt fisiere XML care contin informatii pentru a crea o baza de date.
- ◆ Oracle pune la dispozitie niste sabioane predefinite (cele de la pasul 2)
- ◆ Aceste sabioane se pot crea (cu DBCA):
 - Dint-un alt sablon
 - Dint-o baza de date existenta (doar structura acesteia e folosita, schemele user sunt ignorate)
 - Dint-o baza de date existenta, folosindu-se si datele user existente)

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

Sfârșitul capitolului 2

F.Radulescu_Curs Utilizarea bazei de date_and IV CS

83

Vederi care contin date despre FC

1. V\$DATABASE – contin date despre baza de date (luate din FC)
2. V\$CONTROLFILE, V\$PARAMETER – contin lista cu numele FC

Mai exista si alte vederi care returneaza date privind continutul FC.

F:\Rakines\Car\Utilizare baze de date\and_PCS

23

V\$DATABASE

```

SQL> select controlfile_type, controlfile_created,
2>       sequence#, controlfile_change#, controlfile_time
3>  from v$database;
CONTROL CONTROLFILE SEQUENCER CONTROLFILE_CHANGE# CONTROLFILE_TIME
CURRENT 23-03-2015          4484      3119322 10-10-2015
SQL>

```

F:\Rakines\Car\Utilizare baze de date\and_PCS

Cate grupuri sunt

- ¶ Pentru operarea normala a bazei de date Oracle are nevoie de minimul 2 grupuri de fisiere Redo Log.
- ¶ Pot fi maxim 255 de grupuri diferite.

Observatie: Termenul in engleza este 'online redo log file' pentru ca mai pot exista si fisiere de acest tip arhivate (vom vedea in continuare).

F:\Rakines\Car\Utilizare baze de date\and_PCS

31

PARAMETRI

- ¶ In CREATE DATABASE se folosesc urmatorii parametri:
 - MAXLOGFILES – numarul maxim de grupuri (asa cum spus e <=255)
 - MAXLOGMEMBERS – numarul maxim de membri per grup
 - In fisierul de parametri există LOG_FILES care specifica numarul de fisiere care sunt deschise la run-time (si care poate fi maxim egal cu produsul celor 2 valori maxime de mai sus).

F:\Rakines\Car\Utilizare baze de date\and_PCS

32

V\$CONTROLFILE

```

SQL> select name from v$controlfile;
NAME
C:\ORACLE11G\BIN\ORBRD1.BDF,CONTROLR1.CTL
C:\ORACLE11G\BIN\FLASH_RECOVERY_AREA\BIN\CONTROLR2.CTL
SQL>

```

F:\Rakines\Car\Utilizare baze de date\and_PCS

27

REDO LOG FILES

- ¶ Li se mai spune si 'fisiere jurnal' in cazul altor sisteme de gestiune.
- ¶ In ele se inregistreaza toate modificarile facute in Buffer Cache
- ¶ Se folosesc pentru recuperarea transacțiilor comise ale caror date nu au fost inca scris pe disc pana in momentul incidentului (deci se folosesc DOAR pentru recuperarea datelor – recovery).

F:\Rakines\Car\Utilizare baze de date\and_PCS

28

Utilizare

- ¶ Oracle inregistreaza sequential toate schimbarile facute in baza de date in Redo Log Buffer sub forma unor inregistrari de Redo,
- ¶ Bufferul este folosit circular (cand se ajunge la capat se relia cu inceputul),
- ¶ Din Buffer aceste inregistrari sunt scrise in fisierul (=grup) curent de Redo de catre procesul LGWR in urmatoarele cazuri:

F:\Rakines\Car\Utilizare baze de date\and_PCS

33

Utilizare - cont

1. In momentul in care apare un COMMIT (s-a vorbit de asta la un capitol precedent)
2. In momentul in care Redo log buffer este plin intr-o anumita proportie
3. Cand apare un time-out al LGWR (la fiecare 3 secunde)
4. Inainte ca DBWn sa scrie blocurile de date modificate in fisierile de date

F:\Rakines\Car\Utilizare baze de date\and_PCS

34

Grupuri si membri

- ¶ Fisierile de tip Redo Log folosite la un moment dat de sistem sunt impartite in grupuri, un grup putand contine mai multi membri.
- ¶ Este de preferat ca membrii unui grup sa fie plasati pe dispozitive diferite pentru a evita pierderi in caz de incident de dispozitiv.
- ¶ Procesul care scrie aceste fisiere este LGWR (Log Writer, unul dintre procesele de background ale unei instance)

F:\Rakines\Car\Utilizare baze de date\and_PCS

29

Grupuri si membri – cont

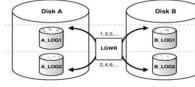
- ¶ Toti membrii (=fisiere) unui grup au dimensiune identica si sunt scrisi in paralel de LGWR
- ¶ Toti membrii unui grup au acelasi numar de secenta (log sequence number). Acesta este un numar dat de Oracle atunci cand incepe sa scrie intr-un grup.
- ¶ Numarul de secenta curent (este unul singur!) este memorat si in fisierile de control si in antetul fisierelor de date.

F:\Rakines\Car\Utilizare baze de date\and_PCS

35

Schimbare grup Redo

- ¶ Cand un fisier (=grup) se umplte se trece la urmatorul fisier (sunt cel putin 2)



F:\Rakines\Car\Utilizare baze de date\and_PCS

36

Schimbare grup Redo - cont

- ¶ Aceasta schimbare se numeste in documentatia Oracle 'Log switch'
- ¶ DBA-ul poate forta o astfel de schimbare si in cazul in care fisierul actual nu e placit
- ¶ La fiecare log switch Oracle asociaza un nou numar de secenta nouului fisier (=grup)
- ¶ Cand apare un log switch este de asemenea initiat si un checkpoint:

F:\Rakines\Car\Utilizare baze de date\and_PCS

37

Checkpoint

- ¶ La aparitia unui checkpoint se executa urmatoarele operatii:
 - > Toate blocurile de date modificate (dirty buffers) din memoria care sunt monitorizate de fisierul de Redo Log respectiv sunt scrise pe disc de catre DBWR
 - > Procesul de checkpoint (CKPT) actualizeaza antetele tuturor fisierelor de control si date pentru a reflecta schimbarea produsa,

F:\Rakines\Car\Utilizare baze de date\and_PCS

37

Checkpoint - cont

- ¶ Un checkpoint apare:
 - > La fiecare log switch;
 - > La oprirea instantei in modulele normal, transnational si imediat
 - > In mod fortat prin setarea parametrului:
 - LOG_CHECKPOINT_INTERVAL si LOG_CHECKPOINT_TIMEOUT
 - > Catre un administrator bazei de date
- ¶ Informatiile despre fiecare checkpoint sunt stocate in fisierul de alerte (dar daca parametrul LOG_CHECKPOINTS_TO_ALERT este setat pe TRUE)

F:\Rakines\Car\Utilizare baze de date\and_PCS

Vederi: v\$instance

```

SQL> show parameter log_archive_start;
NAME                           TYPE        VALUE
log_archive_start              boolean    FALSE
SQL>
SQL> select status from v$instance;
STATUS
STOPPED
SQL>
SQL> show parameter loc_archive_start;
NAME                           TYPE        VALUE
loc_archive_start              boolean    FALSE
SQL>
SQL> select * from v$archive_start;
NAME           SEQUENCE# BYTES  MEMBER# STATUS
LOG_ARCHIVE_1      183 524290000 1 INACTIVE
LOG_ARCHIVE_2      184 524290000 1 CURRENT
LOG_ARCHIVE_3      185 524290000 1 INACTIVE
SQL>

```

F:\Rakines\Car\Utilizare baze de date\and_PCS

43

Parametru

- ¶ Se poate si prin examinarea parametrului log_archive_start (comanda SQL*Plus):

NAME	TYPE	VALUE
log_archive_start	boolean	FALSE

F:\Rakines\Car\Utilizare baze de date\and_PCS

44

Clasificare

- ¶ Fisierile de Redo Log se pot clasifica in
 1. CURRENT - Current = cel in care se scrie la un moment dat
 2. ACTIVE - Activ = scrie in el anterior si modificarile cuprinse in el nu sunt inca scrise in fisierul de date si deci e necesar pentru recuperare instanta,
 3. INACTIVE - Inactiv = scrie in el anterior si modificarile s-au inregistrat si in fisierul de date, deci nu mai e necesar pentru recuperare instanta
 4. UNUSED - Nou - Nu s-a scris niciodata in el pana acum (probabil un fisier nou adaugat)

F:\Rakines\Car\Utilizare baze de date\and_PCS

39

Arhivare

- ¶ Se poate distingea ca fisierile de tip Redo Log File sa fie arhivate de sistemul de gestiune
- ¶ Arhivarea acestor fisiere permite refacerea bazei de date de la 0 pornind de la o salvare la un moment dat si de la fisierul de tip Redo Log care inregistreaza modificarile facute in BD dupa salvarea respectiva.
- ¶ In cazul in care fisierul nu sunt arhivate este posibila in continuare recuperarea instantei dupa incident (pentru asta sunt necesare doar fisierul si cele active de Redo Log)

F:\Rakines\Car\Utilizare baze de date\and_PCS

Vederi: v\$thread

- ¶ Pentru a vizualiza care este fisierul (grupul) curent se poate interoga v\$thread :

```

SQL> select group#, sequence#, bytes, members, status
2>   from v$thread;
GROUP# SEQUENCE# BYTES MEMBERS STATUS
1          1     524290000      1 CURRENT
SQL>

```

F:\Rakines\Car\Utilizare baze de date\and_PCS

45

Vederi: v\$log

- ¶ Informatiile returnate sunt din fisierile de control:

GROUP#	SEQUENCE#	BYTES	MEMBERS	STATUS
1	183	524290000	1	INACTIVE
2	184	524290000	1	CURRENT
3	185	524290000	1	INACTIVE

F:\Rakines\Car\Utilizare baze de date\and_PCS

46

Arhivare - cont

- ¶ O ceasina baza de date poate fi la un moment dat intr-unul din cele 2 moduri:
 1. NOARCHIVELOG – in momentul in care ultimul fisier de Redo Log se umple se revine la primul care este rescris (scris-peste). Bineintele modificarile din asta au fost scrise si in fisierul de date
 2. ARCHIVELOG – dupa un log switch procesele ARCH (de background) arhiveaza fisierile de log inactive,

F:\Rakines\Car\Utilizare baze de date\and_PCS

40

Vederi: v\$database

- ¶ Pentru a vizualiza modul in care este BD: vedere v\$database

F:\Rakines\Car\Utilizare baze de date\and_PCS

Vederi: v\$logfile

- ¶ Coloana STATUS are valoarea NULL daca fisierul este utilizat si INVALID, STALE sau DELETED altfel.

LOG#	MEMBER#	STATUS	TYPE	NAME
1	1	DELETED	LOG	C:\ORACLE11G\BIN\REDBDR1.BDF
2	1	VALID	LOG	C:\ORACLE11G\BIN\REDBDR2.BDF
3	1	VALID	LOG	C:\ORACLE11G\BIN\REDBDR3.BDF

F:\Rakines\Car\Utilizare baze de date\and_PCS

47

Vederi: v\$logfile

- ¶ Un fisier redo log devine INVALID daca sistemul nu il poate accesa,
- ¶ Un fisier redo log devine STALE daca sistemul suspecteaza ca nu este complet sau corect,
- ¶ Un fisier redo log devine VALID din nou cand grupul sau devine activ data viitoare,

F:\Rakines\Car\Utilizare baze de date\and_PCS

48

Fortarea unui log switch

☐ Se face cu cerere:

`ALTER SYSTEM SWITCH LOGFILE;`

Aceasi operatie se poate efectua si din consola de administrare

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

51

Fortarea unui checkpoint

☐ Se face cu cerere:

`ALTER SYSTEM CHECKPOINT;`

Precum si din consola de administrare

☐ De asemenea, in cazul in care baza de date foloseste fisiere de tip Redo Log mari, se poate comanda si efectuarea periodică a checkpoint-urilor prin parametri:

`LOG_CHECKPOINT_INTERVAL si`

`LOG_CHECKPOINT_TIMEOUT`

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

52

Adaugare membrii - cont

☐ Daca fisierul adaugat exista deja pe discul respectiv trebuie sa alba dimensiunea necesara si se va specifica in plus dauer REUSE:

`ALTER DATABASE ADD LOGFILE MEMBER
'/disk01/oracle/dbs/log2b.rdo' REUSE
TO GROUP 1;`

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

53

LOG_CHECKPOINT_INTERVAL

☐ Un checkpoint este initiat dupa ce LGWR a scris un numar de blocuri egal cu acest parametru (blocuri OS)

☐ Cum orice log switch produce de asemenea checkpoint, daca parametrul este mai mare decat fisierul de Log, checkpointul se va face doar la log switch

☐ Daca parametrul este 0 este ignorat (Oracle 10g), Aceasta este valoarea de default.

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

51

LOG_CHECKPOINT_TIMEOUT

☐ Este o valoare specificata in secunde

☐ Se masoara de la inceputul predecesului checkpoint

☐ Valoarea 0 dezactiveaza declansarea de checkpoint-uri pe baza intervalului de timp

☐ Valoarea de default este de 1800 (in Oracle 10g) deci 30 minute,

☐ Garanteaza ca nici un bloc modificat (dirty bloc) nu ramane in memorie mai mult de atatatea secunde cat arata parametrul.

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

52

Redenumire / relocate - cont

1. Oprirea bazei de date (SHUTDOWN)

2. Copierea fisierelor Redo Log in noua locatie. Se face cu comanda OS.

Exemplu:

`mv /diska1/logs/log1a.rdo /diskc1/logs/log1c.rdo`

`mv /diska1/logs/log2a.rdo /diskc1/logs/log2c.rdo`

3. Repozitionare in modul MOUNT:

`CONNECT / AS SYSDBA`

`STARTUP MOUNT`

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

53

Redenumire / relocate - cont

4. Redenumirea (in sistem) a fisierelor:

`ALTER DATABASE RENAME FILE
'/diska1/logs/log1a.rdo', '/diska1/logs/log2a.rdo' TO
'/diskc1/logs/log1c.rdo', '/diskc1/logs/log2c.rdo';`

5. Deschiderea bazei:

`ALTER DATABASE OPEN`

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

54

Adaugare grup Redo Log

☐ Se face cu ALTER DATABASE:

`ALTER DATABASE ADD LOGFILE GROUP 10
('/disk01/oracle/dbs/log10a.rdo',
 '/disk04/oracle/dbs/log10b.rdo') SIZE 500K;`

☐ Specificarea numarului de grup este optional (doar cand dorim sa le cream in alta ordine decat cea normala).

☐ Nu este bine sa cream grupuri cu 10, 20, 30, ... (pe sanse) pentru ca vom consuma inutil spatiu in fisierile de control

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

55

Adaugare membri

☐ Se face cu ALTER DATABASE. Exemplu:

`ALTER DATABASE ADD LOGFILE MEMBER
'/disk01/oracle/dbs/log3b.rdo'
TO GROUP 1,
 '/disk07/oracle/dbs/log2b.rdo'
TO GROUP 3;`

☐ Se poate specifica grupul si print membrii sai:

`ALTER DATABASE ADD LOGFILE MEMBER
'/disk06/oracle/dbs/log10c.rdo'
TO ('/disk01/oracle/dbs/log10a.rdo',
 '/disk04/oracle/dbs/log10b.rdo');`

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

56

Stergere grup

☐ Trebuie sa ramana cel putin 2 grupuri (nu se poate sterge mai mult de atat)

☐ Un grup se poate sterge doar daca e INACTIVE.

☐ Daca se doreste stergera grupului curent trebuie fortat un log switch.

☐ Grupul trebuie sa fie arhivat (daca arhivarea e posibila).

☐ Pentru a vedea starea grupurilor putem utiliza comanda:

Stergere grup - cont

`SELECT GROUP#, ARCHIVED, STATUS
FROM V$LOG;`

GROUP#	ARC STATUS
1	YES ACTIVE
2	NO CURRENT
3	YES INACTIVE
4	YES INACTIVE

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

57

Stergere grup - cont

☐ Stergera efectiva se face cu ALTER DATABASE:

`ALTER DATABASE DROP LOGFILE GROUP 3;`

☐ Grupul se poate da nu numai ca numar ci si prin lista membrilor sai.

☐ Operația de stergere nu implica stergera fisierelor de pe disc ci doar actualizarea informatiilor interne ale sistemului prin eliminarea grupului respectiv.

☐ Putem sa utilizam apoi comenzi GO pentru stergerea efectiva a fisierelor respective.

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

60

Stergere membri

☐ Se poate sterge un membru doar daca nu este in grupul curent sau int-va un grup activ.

☐ Este bine ca grupul respectiv sa fie in acel moment arhivat

☐ Nu putem sterge ultimul membru valid al unui grup daca in felul acesta nu ramana cel putin 2 grupuri continand membri valizi.

☐ In cazul in care grupurile au cat de multi membri de exemplu, se poate sterge un membru din unul dintre ele dar este bine ca ulterior grupul sa fie completat (pentru siguranta in functionare)

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

61

Stergere membri - cont

☐ Comanda este:

`ALTER DATABASE`

`DROP LOGFILE MEMBER 'oracle/dbs/log3c.rdo';`

☐ Si aici operatia nu implica stergera fisierului de pe disc ci doar actualizarea informatiilor interne ale sistemului

☐ Dupa terminarea operatiei Oracle, putem sa utilizam comenzi GO pentru stergerea efectiva a fisierului respectiv.

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

61

CLEAR LOGFILE

☐ In cazul in care un grup devine corupt el poate fi reinitalizat cu comanda:

`ALTER DATABASE CLEAR LOGFILE GROUP 3;`

☐ Grupul se poate da ca numar sau ca lista de membri (intre paranteze).

☐ In cazul in care fisierul nu a fost arhivat trebuie mentionat in comanda:

`ALTER DATABASE CLEAR UNARCHIVED LOGFILE GROUP 3;`

☐ In acest caz insa recuperarea bazei din salvare+Loguri nu mai e posibila pentru salvante care ar folosi si logul astfel initializat.

F.Rădulescu, Curs Utilizator baze de date, anul IV, CC

62

Lecturi obligatorii

1. Oracle Database Administrator's Guide - Cap 5: Managing Control Files
http://download.oracle.com/docs/cd/B14117_01/server.101/b10739/control.htm

2. Oracle Database Administrator's Guide - Cap 5: Managing the Redo Log
http://download.oracle.com/docs/cd/B14117_01/server.101/b10739/redredo.htm

Sfârșitul capitoului 3

63

Capitolul 4

Fisiere de date si Tablespace

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

1

Continut capitol

Ca structura fizica, baza de date contine fisiere de control, de date si de Redo log.

Ca structura logica o baza de date se compune din:

Tablespace \supset Segment \supset Extensie (extent) \supset Bloc (stocare in fisierile de date)

Din punct de vedere fizic avem:

Tablespace \supset Fisiere de date

In acest capitol vom discuta despre:

- Tablespace (element in structura logica a fisierelor de date)
- Fisiere de date

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

2

SEGMENTE - cont

◆ Segmentele temporare sunt in general cele folosite pentru sortari,

◆ Urmatorele cereri SQL au nevoie de segment temporar in cazul in care sortarile nu pot fi efectuate in memorie:

```

• create index
• select * into cursor
• select distinct
• select group by
• select ... union
• select ... intersect
• select ... minus
• analyze table
• joinuri care nu folosesc indexi
• anumite subcriterii corelate

```

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

3

Tablespace

- ◆ O baza de date contine (unul sau) mai multe subdiviziuni numite 'tablespace'.
- ◆ Un tablespace apartine unei singure baze de date.
- ◆ Un tablespace poate fi stocat in unul sau mai multe fisier de date.
- ◆ Un fisier de date aparține unui singur tablespace.
- ◆ Cu unele exceptii (SYSTEM de ex.) un tablespace poate fi trecut intre stările online \leftrightarrow offline si read-write \leftrightarrow read-only

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

3

Fisiere de date

- ◆ Aceste fisiere se creaza:
 - la crearea bazei de date (pentru tablespace-urile care sunt create atunci)
 - la crearea unui nou tablespace
 - la adaugarea unui nou fisier de date la un tablespace
- ◆ Dimensiunea fisierelor este specificata la creare.
- ◆ Unui fisier de date existent i se poate modifica dimensiunea ulterior.
- ◆ Unui fisier de date i se poate seta optiunea 'AUTOEXTEND' pentru a creste automat ca dimensiune cand este necesar.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

4

EXTENSII

- ◆ Un segment este format din una sau mai multe extensi (eng.: extent).
- ◆ O extensie e formata dintr-o succesiune **contigua** de blocuri de date (database blocks).
- ◆ O extensie se gaseste in intregime intr-un singur fisier de date de la care formeaza tablespace-ul.
- ◆ Faptul ca este contigua este relevant pentru cresterea vitezei de exploatare a datelor (citire - scriere)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

5

EXTENSII

- ◆ Nota privind contiguitatea blocurilor unei extensi: acestea sunt blocuri logice, apartinand unui fisier de date (fisierul in care se gaseste extensia).
- ◆ In documentatia Oracle se precizeaza:
 - File system extents are not the same as Oracle Database extents,
 - File system extents are physical contiguous blocks of data written to a device as managed by the file system,
 - Oracle Database extents are logical structures managed by the database, such as tablespace extents.

(http://docs.oracle.com/cd/B28359_01/server.111/b28310/dl_ext009.htm)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

10

SEGMENTE

- ◆ Un tablespace contine segmente.
- ◆ Un segment contine un obiect (tabela, index, etc)
- ◆ Sunt de 4 tipuri generice (si 11 tipuri efective):
 - Segment de tip date (tabele si cluster)
 - Segment de tip index
 - Segment temporar
 - Segment de rollback
- ◆ Un segment se poate intinde pe mai multe fisier de date care aparțin aceluiasi tablespace.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

5

SEGMENTE

- ◆ Cele 11 tipuri de segmente sunt:
 1. table
 2. table partition
 3. index
 4. index partition
 5. cluster
 6. rollback
 7. deferred rollback
 8. temporary
 9. cache
 - 10.lobsegment
 - 11.lobindex

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

6

BLOC

- ◆ O extensie e formata din blocuri.
- ◆ Este vorba despre blocuri ale bazei de date (de dimensiune DB_BLOCK_SIZE)
- ◆ Un astfel de bloc poate fi format din unul sau mai multe blocuri fizice (de disc)
- ◆ Un bloc este cea mai mica unitate de intrare – ieșire pentru SGBD.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

7

Revenim la TABLESPACE

- ◆ Avantajele folosirii mai multor tablespace-uri:
 - Se pot separa datele user de datele de sistem (prin stocarea in tablespace-uri diferite), in felul acesta se micoresca si traficul de date pe tablespace-ul de sistem.
 - Se pot separa datele unui aplicatie de ale altora (prin stocarea in tablespace-uri diferite). In cazul in care un tablespace trece in starea offline – din diverse motive – doar o aplicatie va avea de suferit.
 - Se pot stoca pe discuri diferite, micorsand astfel traficul de date pentru fiecare disc in parte,

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

11

Avantaje - cont

- Se poate optimiza utilizarea tablespace-urilor prin crearea de tablespace-uri dedicate:
 - Altitudine pentru aplicatii update-intensive
 - Altitudine pentru exploatare read-only
 - Altitudine pentru date temporare (segmente temporare)
- Se pot efectua operatii de salvare la nivel de tablespace deci se pot astfel salva doar datele aferente unor aplicatii importante care ruleaza in sistem.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

13

Tablespace-ul SYSTEM

- ◆ La crearea bazei de date se creaza automat tablespace-ul SYSTEM care contine printre altii dictionarul de date al sistemului si segmentul de rollback de sistem.
- ◆ Aceasta este primul tablespace creat si are caracteristici speciale:
 - Nu poate fi redenumit
 - Nu poate fi sters
 - Nu poate fi trecut in starea offline
 - Necesa privilegii speciale pentru operare

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

14

Exemplu

- ◆ Pentru a afla informatii despre tablespace-urile existente, se poate interoga vederea dba_tablespaces din dictionarul de date al sistemului.
- ◆ Un exemplu de cerere este urmatorul:

```
SQL> select tablespace_name,
  extent_management, allocation_type
  from dba tablespaces;
```

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

19

Tablespace-ul SYSAUX

- ◆ La crearea bazei de date se creaza de asemenea si tablespace-ul SYSAUX care contine informatii despre schimbarile de date folosite de uneltele Oracle – astfel ele nu vor avea nevoie de un alt tablespace suplimentar.
- ◆ Aceasta are de asemenea caracteristici speciale:
 - Nu poate fi redenumit
 - Nu poate fi sters
 - Nu poate fi trecut in starea offline
 - Necesa privilegii speciale pentru operare

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

15

Clasificare

- ◆ Un tablespace poate fi – din punct de vedere al datelor continue – de unul dintre urmatoarele tipuri:
 - Permanent – sunt tablespace-urile uzuale, inclusiv cele de sistem
 - Temporar – contin segmente temporare – am restringere de tip
 - De tip Undo – introduce incepand cu versiunea 9i – contin segmente de undo (rollback), necesare in cazul revocarii operatorilor de actualizare. Anterior versiunii 9i existau doar segmente de rollback – nu si tablespace.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

16

Sintaxa CREATE TABLESPACE

```

CREATE [TEMPORARY / UNDO] TABLESPACE <tblspc_name>
DATAFILE / TEMPFILE
<datafile>[,...] name Path where file is created SIZE <n>M[,...]
<datafile>[,...] name Path where file is created SIZE <n>M[,...]
<datafile>[,...] name Path where file is created SIZE <n>M[,...]
<datafile>[,...] name Path where file is created SIZE <n>M[,...]
BLOCKSIZE <DB_BLOCK_SIZE parameter>/2K/4K/8K/16K/32K>
AUTOMATICEXTENSION <ON (NEXT <n>K/M) or MAXSIZE <n>K/M> / UNLIMITED>
LOGGING / NOLOGGING (implicit: logging)
PORTAL / NOPORTAL (implicit: on/off)
ONLINE / OFFLINE (implicit: on/off)
SEGMENT SPACE MANAGEMENT { AUTO | MANUAL }
FLASHBACK { ON | OFF }
EXTENT MANAGEMENT { DICTIONARY / 
  { LOCAL / AUTOMATIC / UNIFORM } <n>K/M } } ]
PERMANENT / NOT PERMANENT (implicit: permanent)
MINIMUM EXTENT <n>K
<Clauza DEFAULT STORAGE>
NOCACHE;

```

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

21

CREATE TABLESPACE - cont

- ◆ EXTENT MANAGEMENT (DICTIONARY | LOCAL / AUTOALLOCATE | UNIFORM [SIZE <n>K M I])
- ◆ Aceasta optiune arata ca acest tablespace va fi de tip DMT sau LMT.
- ◆ In cazul LMT se poate specifica suplimentar:
 - ◆ optiunea AUTOALLOCATE (cand se interrogeaza vederile din dictionarul de date se afiseaza in acest caz 'SYSTEM':
 - tablespace-ul va contin extensiuni de dimensiuni diferite, gestiunea fiind facuta automat de catre sistem.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

23

CREATE TABLESPACE - cont

- ◆ optiunea AUTOALLOCATE – cont,
 - Aceasta optiune este buna atunci cand in acel tablespace vor fi stocate obiecte (segmente) de dimensiuni variabile, fiecare putand avea mai multe extensi.
 - Este un mod simplificat de gestiune (pt. ca e facuta de sistem) dar poate duce uneori la imblocarea unor spati pe disc.
 - Dimensiunea minima a unei extensi este de 64K. Daca blocul de date al BD este 16K sau mai mare atunci dimensiunea minima a unei extensi este de 1M.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

24

Alta clasificare: DMT si LMT

- ◆ Fiecare Tablespace este format dintr-o multime de extensi (continute in segmentele componente).
- ◆ Gestiunea acestora (care sunt libere si care sunt ocupate) se poate face in doua feluri: fie informatiile respective se stocheaza in dictionarul de date fie se memoreaza in tablespace.
- ◆ Tablespace-urile pentru care gestiunea se face prin intermediul dictionarului de date (o solutie costisitoare ca timp) se numesc DMT – dictionary managed tablespaces

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

17

DMT si LMT - cont

- ◆ Tablespace-urile pentru care gestiunea se face local, prin stocarea datei privind starea extensiilor in interiorul tablespace-ului se numesc LMT – locally managed tablespaces
- ◆ Informatiile se tin in headerul acestuia, de fapt in headerul fiecarui fisier de date component.
- ◆ In acest caz headerul contine un bitmap unde fiecare bit este un bloc sau un grup de blocuri. Bitul arata daca zona respectiva este ocupata sau nu.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

18

CREATE TABLESPACE - cont

- ◆ EXTENT MANAGEMENT (DICTIONARY | LOCAL / AUTOALLOCATE | UNIFORM [SIZE <n>K M I])
- ◆ Aceasta optiune arata ca acest tablespace va fi de tip DMT sau LMT.
- ◆ In cazul LMT se poate specifica suplimentar:
 - ◆ optiunea AUTOALLOCATE (cand se interrogeaza vederile din dictionarul de date se afiseaza in acest caz 'SYSTEM':
 - tablespace-ul va contin extensiuni de dimensiuni diferite, gestiunea fiind facuta automat de catre sistem.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

23

CREATE TABLESPACE - cont

- ◆ optiunea AUTOALLOCATE – cont,
 - Aceasta optiune este buna atunci cand in acel tablespace vor fi stocate obiecte (segmente) de dimensiuni variabile, fiecare putand avea mai multe extensi.
 - Este un mod simplificat de gestiune (pt. ca e facuta de sistem) dar poate duce uneori la imblocarea unor spati pe disc.
 - Dimensiunea minima a unei extensi este de 64K. Daca blocul de date al BD este 16K sau mai mare atunci dimensiunea minima a unei extensi este de 1M.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS

24

CREATE TABLESPACE - cont

- ◆ optiunea UNIFORM
 - Specifica faptul ca acel tablespace este gestionat folosindu-se extensiile dimensiuni fixe.
 - Valoarea implicită a dimensiunii este 1M
 - Fiecare extensie trebuie sa aiba minim 5 blocuri (blocuri BD), Deci:
 - Daca bloc este de 8192 octeti (8K) atunci dimensiunea minima pentru UNIFORM este de 40K.
 - Pentru 16384 octeti (16K) minimul pentru UNIFORM este 80K.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

25

CREATE TABLESPACE - cont

- ◆ optiunea UNIFORM - cont
 - UNIFORM nu este o optiune valida pentru tablespace-ul SYSTEM
 - Aceasta optiune permite o alocare mai precisa a spatiului astfel incat sa se minimizeze pierderile de spatiu pe disc.
 - Se foloseste atunci cand avem o estimare asupra spatiului ocupat de fiecare obiect din acel tablespace.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

26

CREATE TABLESPACE - cont

Claузă Storage arată cum va stoca Oracle fiecare obiect în acel tablespace. Optiunile sunt:

- ◆ INITIAL int|K|M
- ◆ NEXT int|K|M
- ◆ MINEXTENTS int
- ◆ MAXEXTENTS int|UNLIMITED
- ◆ PCTINCREASE int
- ◆ FREELISTS int
- ◆ FREELIST GROUPS int
- ◆ OPTIMAL int|NULL
- ◆ BUFFER_POOL {KEEP | RECYCLE | DEFAULT}

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

31

CREATE TABLESPACE - cont

Detaliere:

- ◆ **INITIAL int|K|M** – defineste dimensiunea primei extensii (minim 2 blocuri). Valoarea implicită este 5 blocuri ale RD.
- ◆ **NEXT int|K|M** - da dimensiunea celei de-a doua extensiile. Valoarea minima este de 1 bloc, valoarea implicită este de asemenea 5 blocuri.
- ◆ **MINEXTENTS int** - este numarul de extensiile care sunt alocate cand segmentul este creat. Valoarea minima – și implicită – este 1.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

32

Exemple

Exemplu:

- ◆ **Cazul AUTOALLOCATE:**
CREATE TABLESPACE user
DATAFILE '/u02/oracle/data/user01.dbf'
SIZE 50M
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
- ◆ **Cazul UNIFORM:**
CREATE TABLESPACE user
DATAFILE '/u02/oracle/data/user01.dbf'
SIZE 50M
EXTENT MANAGEMENT LOCAL UNIFORM SIZE
128K;

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

27

CREATE TABLESPACE - cont

- ◆ **MINIMUM EXTENT int|K|M** – arată dimensiunea minimă a unei extensii (8K sau după număr urmăzii K sau M). O extensie este de acea dimensiune sau **multiplu** de aceea dimensiune. Se foloseste pentru a împiedica o prea mare fragmentare a spațiului.
- ◆ **BLOCKSIZE int|K** – se poate specifica o dimensiune non-standard a blocului pentru acel tablespace. E legată de extensia MAXEXTENTS.
- ◆ **LOGGING | NOLOGGING** – operatii (cum ar fi crearea unui index sau încarcarea de date cu loaderul) nu sunt înregistrate în fisierul Redo Log în caz de NOLOGGING. Se aplică obiectelor din acel tablespace. Nu este recomandat!

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

28

CREATE TABLESPACE - cont

Claузă Storage - cont

- ◆ **MAXEXTENTS int** – determină numărul maxim de extensiile pe care le poate avea un segment. Valoarea minima este 1 iar valoarea maxima depinde de dimensiunea blocului.
- ◆ **MAXEXTENTS UNLIMITED** – este echivalentă cu 2G extensiile
- ◆ **PCTINCREASE int** – este procentul cu care crește dimensiunea extensiilor. Valoarea minima este 0, cea implicită 50.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

33

CREATE TABLESPACE - cont

◆ Există o formulă care ne da dimensiunea extensiiei cu numărul n:

$$\text{Size}(n) = \text{NEXT} * (1 + \text{PCTINCREASE}/100)^{(n-1)}$$

Deci dacă NEXT = 200K iar PCTINCREASE este 50 atunci:

- Size(2) = 200K,
- Size(3) = 300K,
- Size(4) = 450K, etc

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

34

CREATE TABLESPACE - cont

- ◆ **FORCE LOGGING** – Se forțează înregistrările în Redo Log a modificarilor pe obiectele din acel tablespace chiar dacă ele au fost create cu NOLOGGING.
- ◆ **ONLINE | OFFLINE** – în cazul OFFLINE acel tablespace nu este disponibil imediat după creare (trebuie ca ulterior să fie adus în starea online)
- ◆ **PERMANENT | TEMPORARY** – tablespace permanent sau temporar.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

29

CREATE TABLESPACE - cont

- ◆ **FLASHBACK ON | OFF**
Se folosesc în conjuncție cu operații de tip:
• ALTER DATABASE FLASHBACK ON si
• FLASHBACK DATABASE TO pentru a redauce baza de date la o stare anterioară.

Un exemplu ilustrativ se găsește la adresa:

<http://www.orafaq.com/node/1847>

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

30

CREATE TABLESPACE - cont

- ◆ Optiunile FREELISTS int și FREELIST GROUPS int sunt legate de clauza:
SEGMENT SPACE MANAGEMENT {MANUAL | AUTO}
- ◆ Aceasta clauză spune cum este gestionat spațiul liber dintr-un segment:
 - MANUAL
 - AUTO

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

35

CREATE TABLESPACE - cont

- ◆ **MANUAL**: Sunt utilizate liste ale spațiului liber pentru fiecare segment. Acestea sunt liste de blocuri care corespund spațiului disponibil pentru noi operații de inserție.
- ◆ **MANUAL** este valoarea implicită pentru aceasta clauză. În acest caz FREELISTS este un parametru care specifică numărul de liste de blocuri care pot primi înregistrări. În aplicații de tip paralel sau distribuite se folosesc grupuri de liste (cate unul pentru fiecare nod).
- ◆ **AUTO**: În acest caz sunt utilizate bitmapuri pentru spațiul liber din segmente. Acestea permit o gestionare automată a spațiului disponibil. Optiunea AUTO poate fi lăsată în casă în care se face multe actualizări.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

36

Creare TABLESPACE - cont

- ◆ În cazul LMT nu este nevoie de a specifica în CREATE sau ALTER opțiuni de stocare (gestionați-se automatic).
- ◆ Deși nu vor apărea clauzele:
 - next
 - pctincrease
 - minextents
 - maxextents
 - default storage
- ◆ În cazul DMT însă aceste clauze pot să apară atât la crearea unui tablespace cat și în comanda de modificare ALTER TABLESPACE.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

37

CREATE TABLESPACE - cont

- Dacă nu există clauza EXTENT MANAGEMENT atunci pentru determinarea tipului (DMT sau LMT) se folosesc informații de compatibilitate (parametrii în fisierul init.ora) precum și clauzele MINIMUM EXTENT și DEFAULT clauza _storage astfel:
1. If compatibil < 9.0.0 se creează un tablespace DMT.
 2. If compatibil >= 9.0.0 și **DEFAULT clauza _storage** nu a fost specificată se creează un LMT cu AUTOALLOCATE.

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

38

ALTER TABLESPACE

- ◆ Sintaxa (parțială) este:


```
ALTER TABLESPACE tablespace
  ADD DATAFILE 'filename'
  [AUTOEXTEND [ OFF | ON ]NEXT integer [K|M]
  [MAXSIZE (UNLIMITED | integer[K|M] ) ]
  [FILESPERGROUP integer]
  | REUSE [filename [, filename'] ...]
  | DEFALUT STORAGE storage_clause
  | ONLINE
  | OFFLINE [NORMAL | TEMPORARY | IMMEDIATE]
  | READ ONLY
  | READ WRITE
  | {BEGIN | END} BACKUP
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

43

Adaugare fisier

- ◆ Se adaugă un nou fisier de date la un tablespace folosind cereri de tip ALTER TABLESPACE:

```
ALTER TABLESPACE user
ADD DATAFILE '/u02/oracle/data/user01.dbf' SIZE 50M
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

44

CREATE TABLESPACE - cont

- 3. If compatibil >= 9.0.0 și clauza **DEFAULT** storage a fost specificată atunci:
 - MINIMUM EXTENT a fost specificată atunci:
 - a. Dacă MINIMUM EXTENT, INITIAL și NEXT sunt egale între ele și PCTINCREASE = 0 atunci se creează un LMT cu UNIFORM.
 - b. MINIMUM EXTENT, INITIAL și NEXT sunt egale și PCTINCREASE nu sunt atunci se creează un LMT cu AUTOALLOCATE.
 - MINIMUM EXTENT, INITIAL și NEXT sunt specificate atunci:
 - Dacă INITIAL și NEXT sunt egale iar PCTINCREASE = 0 atunci LMT cu UNIFORM și altele LMT cu AUTOALLOCATE,

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

39

Exemplu

- ◆ Tablespace permanent:

```
create tablespace TS1
logging
datafile '/home/oracle/data/ts1.dbf'
size 16M
autoextend on next 32M maxsize 2048M
extent management local
```

```
create tablespace TSDATE
datafile '/home/oracle/data/date/tb1.dbf'
size 10M
autoextend on maxsize 200M
extent management local uniform size 64K;
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

40

Adaugare fisier - cont

- ◆ Se pot adăuga mai multe fisiere cu aceeași comandă:

```
ALTER TABLESPACE user
ADD DATAFILE '/u02/oracle/data/user01.dbf' SIZE 50M,
```

```
'/u02/oracle/data/user02.dbf' SIZE 50M,
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

45

AUTOEXTEND

- ◆ Clauza AUTOEXTEND permite / inhibă extinderea automată a fisierelor de date:
 - AUTOEXTEND OFF inhibă creșterea automată a acestora în dimensiune
 - În cazul AUTOEXTEND ON fisierile de date se extind automat la nevoie

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

46

Exemplu

- ◆ Tablespace permanent cu mai multe fisiere de date:

```
create tablespace TS1
datafile '/home/oracle/data/ts1.dbf' size 16M autoextend on next 32M maxsize 2048M,
'/home/oracle/data/ts2.dbf' size 32M autoextend on next 32M maxsize 2048M
extent management local;
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

41

Exemplu

- ◆ Tablespace temporar:

```
create temporary tablespace ttemp
tempfile '/home/oracle/temp/temp01.dbf'
size 16M
autoextend on next 16M maxsize 2048M
extent management local;
```

- ◆ Tablespace pe tip UNDO:

```
create undo tablespace tsundo
datafile '/home/oracle/data/undo.dbf'
size 100M;
```

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

42

AUTOEXTEND - cont

- ◆ NEXT specifică dimensiunea minimă a incrementului (în MB sau în MB) în care sunt necesare noi extensii (extensii) și spațiu disponibil din fisier nu este suficient pentru acestea.
- ◆ Valoarea implicită pentru NEXT este de 1 bloc al BD
- ◆ MAXSIZE specifică dimensiunea maximă a spațiului care se poate aloca pentru acel fisier de date (pană la ce dimensiune poate crește).
- ◆ UNLIMITED specifică faptul că nu este setată o dimensiune maximă permisă (se poate extinde oricât, în limita spațiului existent).

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

47

Exemplu

```
ALTER TABLESPACE user
ADD DATAFILE '/u02/oracle/data/user01.dbf' SIZE 200M
AUTOEXTEND ON
NEXT 10M
MAXSIZE 500M
```

Clauza AUTOEXTEND poate fi prezenta în ceriere:

- ◆ CREATE DATABASE
- ◆ ALTER DATABASE
- ◆ CREATE TABLESPACE
- ◆ ALTER TABLESPACE

F.Rudolec, Curs Utilizarea bazei de date, anul IV CS

48

Specificare AUTOEXTEND pentru fisier existent:

- ◆ Se face folosind ALTER DATABASE:

```
ALTER DATABASE ora
DATAFILE '/u02/oracle/data/user01.dbf'
AUTOEXTEND ON
NEXT 10M
MAXSIZE 500M
```

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

49

RESIZE

- ◆ Pentru schimbarea manuala a dimensiunii unui fisier care nu este read-only se poate folosi ALTER DATABASE. De dator la DATAFILE-urile pot fi prezente mai multe nume de fisiere (sunt toate afectate):

```
ALTER DATABASE ora
DATAFILE '/u02/oracle/data/user01.dbf'
RESIZE 500M
```

- ◆ Pentru cazul micsorarii dimensiunii, aceasta se poate face doar cu spatiul liber de la sfarsitul fisierului (daca exista!)

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

50

Read Only – Read Write

- ◆ READ ONLY specifica faptul ca nu sunt permise operatii de scriere in acel tablespace.
- ◆ Inainte de a trece un tablespace in acest mod trebuie sa fie inedepnse urmatoarele:
 - Acel tablespace trebuie sa fie online,
 - Nu trebuie sa existe tranzactii active in baza de date respectiva,
 - Acel tablespace nu trebuie sa contina segmente active de rollback,

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

55

Read Only – Read Write

- ◆ Conditiile de trecere R/O – cont:
- Acel tablespace nu trebuie sa fie implicat in acel moment intr-o operatie de salvare (online backup),
- Parametrul de initializare COMPATIBLE trebuie setat la versiunea 7.1.0 sau la una ulterioara acesteia,
- ◆ READ WRITE specifica faptul ca acel tablespace revine din starea de READ ONLY in starea READ WRITE in care poate fi scris.
- ◆ In acest caz toate fisierile de date ale acelui tablespace trebuie sa fie online,

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

56

ONLINE / OFFLINE

- ◆ Sintaxa clauzelor:
- ONLINE | OFFLINE [NORMAL | TEMPORARY | IMMEDIATE]**
- ◆ Trecerea in modul ONLINE aduce un tablespace care nu era asa in mod online.
- ◆ OFFLINE este optiunea inversa, caz in care se inhiba accesul la acel tablespace si la segmentele care se afla in el.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

51

ONLINE / OFFLINE - cont

- ◆ Trecerea OFFLINE se poate face in trei feluri:
 - NORMAL – se executa checkpoint pentru toate fisierele de date din acel tablespace (aceste fisiere trebuie sa fie toate online – si ele pot fi trecute offline!)
 - In cazul NORMAL, la revenirea online nu este necesar sa se execute operatii de recovery,
 - NORMAL este valoarea implicita (in caz in care la trecerea OFFLINE nu se specifica nici una din cele trei optiuni).
 - Daca baza de date este in modul NOARCHIVELOG, NORMAL este optiunea cea mai buna pentru a se evita recuperarea (pentru ca in acest caz nu se poate face recuperarea).

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

52

Mutarea fisierelor de date

- ◆ Fisierile de date ale unui tablespace pot fi mutate astfel:
 1. Se trece acel tablespace offline.
 2. Cu comenzi SO copiem fisierile in noua locatie.
 3. Se executa ALTER TABLESPACE RENAME.
 4. Se reduse acel tablespace online.
 5. Se pot apoi sterge vechile fisiere de date cu comenzi SO.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

57

Mutarea fisierelor de date - cont

- ◆ Iata un exemplu de comanda:

```
ALTER TABLESPACE user
RENAME DATAFILE
  '/u02/oracle/data/user01.dbf' TO
  '/u15/oracle/data/user01.dbf'
```

- ◆ Oracle nu face efectiv vreo redenumire de fisiere ci doar inlocuieste in fisierile de control vechil nume de fisier cu cel nou.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

58

ONLINE / OFFLINE - cont

2. In cazul TEMPORARY se face checkpoint pentru toate fisierile de date care sunt online dar Oracle nu se asigura ca toate fisierile pot fi stocate.
- ◆ Orice fisier care e in acel moment offline poate avea nevoie de recovery cand revenim online.
3. IMMEDIATE nu face checkpoint si nici nu verifica daca fisierile sunt disponibile sau nu. La revenirea online este nevoie de recovery.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

53

Exemple

```
ALTER TABLESPACE user ONLINE
ALTER TABLESPACE user OFFLINE
ALTER TABLESPACE user OFFLINE TEMPORARY
ALTER TABLESPACE user OFFLINE IMMEDIATE
```

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

54

Mutarea fisierelor – v2

- ◆ Exista si posibilitatea de a muta fisierile de date cu comanda ALTER DATABASE. Pentru aceasta:
 1. Se opreste baza de date.
 2. Se muta fisierile cu comenzi SO.
 3. Se monteaza baza.
 4. Se executa ALTER DATABASE RENAME FILE.
 5. Se deschide baza.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

59

Mutarea fisierelor de date - cont

- ◆ Iata un exemplu de comanda:

```
ALTER DATABASE ora
RENAME DATAFILE
  '/u02/oracle/data/user01.dbf' TO
  '/u15/oracle/data/user01.dbf'
```

- ◆ La fel ca inainte, Oracle nu face efectiv vreo redenumire de fisiere ci doar inlocuieste in fisierile de control vechil nume de fisier cu cel nou.

- ◆ In ambele cazuri se pot redenumi cu o singura comanda mai multe fisiere (RENAME FILE lista-old TO lista-new).

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

60

Stergere TABLESPACE

- ◆ Se face cu DROP TABLESPACE.
- ◆ Sintaxa:

```
DROP TABLESPACE nume
[INCLUDING CONTENTS
[CASCADE CONSTRAINTS ] ]
```

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

61

Stergere TABLESPACE

- ◆ INCLUDING CONTENTS specifica faptul ca se sterg inclusiv acel tablespace-uri care contin date (altfel acesta nu pot fi stocate).
- ◆ CASCADE CONSTRAINTS – sterge si constrangerile referentiale aferente obiectelor din acel tablespace.
- ◆ In cazul in care CASCADE CONSTRAINTS este omisă si există astfel de constrangeri Oracle va returna o eroare si nu va efectua stergerea.

F.Rădulescu, Curs Utilizator baza de date, anul IV CS

62

Vederi - DBA_TABLESPACES

TABLESPACE_NAME	LOGGING	DATAFILE_NAME	INITIAL_SIZE	MAX_SIZE	MIN_EXTENT_SIZE	SEGMENT_SPACE_LIMIT
SYSTEM	YES	/u02/oracle/data/system01.dbf	100M	UNLIMITED	1M	100M
SYSAUX	NO	/u02/oracle/data/sysaux01.dbf	10M	100M	1M	100M
TEMP	NO	/u02/oracle/temp01.dbf	10M	100M	1M	100M
USERS	NO	/u02/oracle/users01.dbf	10M	100M	1M	100M
ORDERS	NO	/u02/oracle/orders01.dbf	10M	100M	1M	100M
EMPLOYEES	NO	/u02/oracle/employees01.dbf	10M	100M	1M	100M
DEPARTMENTS	NO	/u02/oracle/departments01.dbf	10M	100M	1M	100M
JOBS	NO	/u02/oracle/jobs01.dbf	10M	100M	1M	100M
DEPTLOCATIONS	NO	/u02/oracle/deptlocations01.dbf	10M	100M	1M	100M
JOBLocations	NO	/u02/oracle/joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS	NO	/u02/oracle/joblocations_deptlocations01.dbf	10M	100M	1M	100M
JOBLocations_JOBS	NO	/u02/oracle/joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBS	NO	/u02/oracle/joblocations_deptlocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_jobs01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations_joblocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBS	NO	/u02/oracle/joblocations_deptlocations_joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations_JOBLocations	NO	/u02/oracle/joblocations_deptlocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle/joblocations01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations	NO	/u02/oracle01.dbf	10M	100M	1M	100M
JOBLocations_DEPTLOCATIONS_JOBLocations_JOBLocations_JOBLocations						

Exemple: DBA_TEMP_FILES

Exemple: DBA_ROLLBACK_SEGS

Lecturi obligatorii

1. Locally vs. Dictionary Managed Tablespaces
<http://www.oracle.com/node/3>
 2. Oracle Database Administrator's Guide – Cap 8:
Managing Tablespaces
http://download.oracle.com/docs/cd/B14117_01/server.101/b10739/tspaces.htm
 3. Oracle Concepts – Tablespaces
<http://www.gdmgbmzch.your/concepts tablespaces.html>

E. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Sfârșitul
capitolului 4

F. Radulescu. Curs: Utilizarea bazelor de date, anul IV CS.

7

Capitolul 5

Gestiunea tabelelor

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

1

Tipuri de organizare

- Există patru tipuri de organizare pentru tabelă: unel baze de date:
 1. Tabele uzuale – heap-organized tables – este tipul de bază, ușor. O astfel de tabelă reprezintă o mulțime neorganizată (heap) de linii. Aceast tip de tabelă este subiectul principal al capitolului de fata.
 2. Tabele particionate – partitioned tables – în care linile sunt împărțite în mai multe grupuri, numite partitii, fiecare astfel de partitie (sau subpartitie) putând fi gestionată separat,

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

2

Clauze CREATE – PCTFREE(1)

- PCTFREE specifică procentul de spațiu din fiecare bloc rezervat creșterii în lungime a înregistrărilor (liniilor) determinată de operații de tip update.
- Valoarea trebuie să fie între 1 și 99.
- În cazul specificării valorii 0 intregul bloc poate fi umplut prin inserare de noi linii.
- Valoarea implicită a acestui parametru este 10 (deci 10% spațiu disponibil pentru update, 90% spațiu disponibil pentru insert).

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

Clauze CREATE – PCTFREE(2)

- În cazul în care înregistrările dintr-un bloc cresc în lungime și se depășește spațiul alocat lor (inclusiv cel initial reținut pentru creștere prin PCTFREE) Oracle ia o linie din acel bloc și o mută în alt bloc, lăsând în locul ei doar un pointer.
- Acest proces este numit "migrarea liniilor".
- În acest caz performanțele scad, deoarece pentru cîndrarea acelei linii sunt cîndră două blocuri

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

8

Tipuri de organizare - cont

- 3. Tabele de tip "clustered tables". O astfel de tabelă este parte a unui cluster. Un cluster conține mai multe tabele care au în comun blocuri de date deoarece ele au în comun anumite coloane și, de asemenea, sunt folosite frecvent împreună.
- 4. Tabele "index organized". Spre deosebire de primele (heap organized), înregistrările (liniile) unei astfel de tabele sunt organizate sub forma unui arbore B, sortate după cheia primă,

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

3

Tipuri de organizare - cont

- În cazul tabelelor particionate, împărțirea linilor în partitii se face după valoarea uneia sau mai multor coloane.
- O linie la poziția sa apără una singură partită.
- Fiecare partită are un număr și este stocată într-un segment. Aceste segmente pot fi în tablespace-uri diferite.
- Performanța se practice în cazul tabelelor de mari dimensiuni care sunt accesate concomitent.
- Se pot participa și tabelele de tip "index-organized" cu condiția ca atributul (coloanele) după care se face particionarea să fie o submultime a cheii primăre.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

4

Clauze CREATE – PCTFREE(3)

- În cazul în care o înregistrare este prea lungă pentru a începea într-un bloc aceasta înregistrare este spartă în mai multe bucăți care sunt stocate în mai multe blocuri, împreună cu pointerii necesari recuperării întregii linii. Această situație se numește "row chaining".
- Dacă se pot particiona linii în mai multe blocuri.
- Valoarea implicită a acestui parametru este 10.
- Parametrul PCTFREE poate fi prezent în comenzi create/alter și în punctul altor obiecte (ex. indecs).

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

9

Clauze CREATE - PCTUSED

- Combinarea PCTFREE - PCTUSED duce la direcționarea noulor înregistrării fie în blocuri existente fie în blocuri noi (goale la moment).
- PCTUSED specifică procentul minim de spațiu utilizat din fiecare bloc. Dacă spațiu utilizat scade sub acea valoare blocul devine candidat pentru inserarea de noi înregistrări (linii).
- PCTUSED are valori între 1 și 99.
- Valoarea de default este 40.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

10

CREATE TABLE

- Pe lângă elementele cunoscute din cursurile anterioare, cererea SQL CREATE TABLE poate avea și altă clauze, suplimentare, specificând parametrii de stocare pentru datele tabelei respective.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

5

Syntaxa CREATE TABLE (9i)

```
CREATE TABLE [schema.]table
  ( column [,constraint] ) - sintaxa clasica
  [ [PCTFREE integer] [PCTUSED integer]
    [INITTRANS integer] [MAXTRANS integer]
    [PARALLEL integer]
    [STORAGE storage_clause]
    [PARALLEL | NOCACHE]
    [NOPARALLEL]
    [CACHE | NOCACHE]
    [LOGGING | NOLOGGING]
    [CLUSTER cluster (column [, column]...)]
    [ENABLE enable_clause | DISABLE disable_clause] ...
  [AS subquery]
```

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

6

Clauze CREATE - PCTUSED

- Acest parametru poate fi prezent și în comenzi de creare pentru alte obiecte (ex. Indeks)
- Suma dintre PCTFREE și PCTUSED trebuie să fie mai mică sau egală cu 100.
- Cu cat diferența între 100 și aceasta suma este mai mică, cu atât este mai eficientă folosirea spațiului pe disc, însă pot să scada performanțele.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

11

Clauze CREATE - MAXTRANS

- MAXTRANS specifică numărul maxim de tranzacții concurente care pot actualiza un bloc la tabelă – deci numărul maxim de „transacții entries“ care pot fi alocate unui bloc.
- Acestea se aloca dinamic de Oracle după depășirea INITTRANS.
- Valoarea poate fi între 1 și 255.
- Valoarea de default este în funcție de dimensiunea blocului (255 în Oracle 8i)
- Este recomandat ca valoarea de default pentru MAXTRANS sa nu fie schimbată.
- MAXTRANS este parametrul și în alte operații de creare obiecte ale bazei de date.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

13

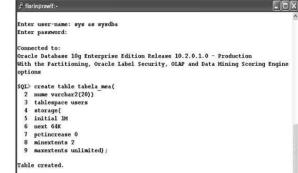
TABLESPACE, STORAGE

- TABLESPACE specifică unde se va crea tabela respectivă.
- Dacă optiunea lipsesc, tabela se creează în tablespace-ul implicit (default) al utilizatorului care definește schema în care se face creația.
- STORAGE specifică modul în care extensiile vor fi alocate tabeliei.
- Sintaxa clauzei STORAGE a fost prezentată în capitolul anterior (cel despre tablespaci-uri).
- Aceasta clauza are implicații în performanțele obținute în cazul tabelelor de mari dimensiuni.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

14

Rulare



```
SQL> create table tabel1_mea(
  1   segment_size 128K,
  2   tablespace users,
  3   extent_management,
  4   initial 3M,
  5   next 4M,
  6   extent_size 4K,
  7   minextents 2,
  8   maxextents unlimited);
Table created.
```

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

15

Clauze CREATE - INITTRANS

- INITTRANS specifică numărul initial de „transaction entries“ alocate în fiecare bloc. Fiecare transacție care actualizează un bloc are nevoie de o astfel de intrare la nivelul blocului.
- Valoarea poate fi de la 1 la 255.
- Valoarea implicită este 1 în cazul tabelelor (2 la indecs).
- În general Oracle recomandă să se păstreze valoarea implicită.
- Acest parametru asigură un număr minim de tranzacții per bloc, fară overtheadul alocării dinamice a unei intrări („transaction entry“)

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

12

Reamintire:

Dacă Storage are opțiuni ca:

- INITIAL int K | M
- NEXT int K | M
- MINEXTENTS int
- MAXEXTENTS int
- MAXEXTENTS UNLIMITED
- PCTINCREASE int
- FREELISTS int
- FREELIST GROUPS int

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

15

Reamintire:

- Unde:
- INITIAL int K | M – defineste dimensiunea primei extensii (minimum 2 blocuri). Valoarea implicită este 5 blocuri ale BD.
 - NEXT int K | M – da dimensiunea celei de-a doua extensii. Valoarea minima este de 1 bloc, valoarea implicită este de asemenea 5 blocuri.
 - MINEXTENTS int – este numărul de extensii care sunt alocate cand segmentul este creat. Valoarea minima – și implicită – este 1.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

16

PARALLEL - cont

- Cererea următoare specifică parcurgerea întregii tabele cu un grad de paralelism egal cu 5. Observăm că daca se definește un alias de tabelă hintul trebuie să folosească acest alias:

```
SELECT /*+ FULL(s)
           PARALLEL(s, 5) */
           ename
      FROM emp s;
```

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

21

PARALLEL - cont

- Cererea următoare specifică parcurgerea tabelei fară paralelizare. De asemenea trebuie să se folosească aliasul definit:

```
SELECT /*+ NOPARALLEL(s) */
           ename
      FROM emp s;
```

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

22

Reamintire:

- MAXEXTENTS int – determină numărul maxim de extensii pe care le poate avea un segment. Valoarea minima este 1 iar valoarea maximă depinde de dimensiunea blocului.
- MAXEXTENTS UNLIMITED – este echivalentă cu 2G extensii
- PCTINCREASE int – este procentul cu care crește dimensiunea extensiilor. Valoarea minima este 0, cea implicită 50.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

17

Exemplu:

```
create table tabela_mea (
  1   nume  varchar2(30),
  2   descriere varchar2(4000)
  3   tablespace users
  4   storage (
  5     initial 1M
  6     next 512K
  7     pctincrease 0
  8     minextents 2
  9     maxextents unlimited )
 10  /
 11  /
```

18

Clauze CREATE - CACHE

- CACHE se folosește mai ales pentru tabele de mici dimensiuni și specifică faptul ca aceea tabelă va fi pastrată în buferele de memorie (deci nu va fi dealocată) prin plasarea blocurilor sale în zona celor mai recent utilizate chiar și atunci când se execută o parcurgere completă a tabelei.
- NOCACHE (valoare implicită) specifică faptul ca blocurile tabelii din buffer cache vor fi supuse algoritmului LRU standard atunci când se execută o parcurgere completă a unei tabele (full table scan), și deci vor fi plasate în zona celor mai puțin recent utilizate.

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

23

Clauze CREATE - LOGGING

- LOGGING arată ca atât operația de creare a unei tabele cat și operațiile care vor fi facute apoi asupra acesteia vor fi înregistrate în fisierile Redo Log.
- NOLOGGING specifică faptul ca operația de creare a unei tabele precum și unele operații de încarcare cu date (nu însă și operațiile obisnuite de insert) nu vor fi înregistrate în fisierile Redo Log.
- În lipsa acestor opțiuni se folosesc parametrii de la creația tablespace-ului în care este gazduita tabela (să acolo avem acestea două opțiuni)

F: Radulessu_Curs_Ultimele_baze_de_date_mii_IV_CS.

24

Clauze CREATE - CLUSTER

- CLUSTER arata ca tabela este parte a unui cluster.
- Coloanele din clauza sunt coloane ale tabeliei care corespund cu coloanele clusterului.
- In general coloanele sunt parte a cheii primare (sau intreaga cheie primara).
- Trebuie specificata o coloana a tabeliei pentru fiecare coloana a clusterului.
- Corespondenta este pozitionala (nu prin nume)
- Deoarece tablelele de tip cluster folosesc o alta alocare a spatiului NU se pot folosi in paralel clauzele PCTFREE, PCTUSED, INITTRANS, MAXTRANS, TABLESPACE in conjunctie cu clauza CLUSTER

F: Radulessu_Curs_Ultimele_bucuri_de_date_and_IV_CS

25

Exemplu

- 1. Creare cluster:

```
CREATE CLUSTER pers
(dept NUMBER(2))
SIZE 512
STORAGE (initial 100K next 50K);
```
- 2. Creare index pentru cheia cluster:

```
CREATE INDEX idx_pers ON CLUSTER
pers;
```

26

Clauze CREATE - AS

- AS specifica faptul ca noua tabela va fi populata cu linile rezultante din cererea select prezentata in clauza AS.
- Crearea unei tablele ca rezultat al unei cereri SELECT a fost studiata in semestrelle trecute.

F: Radulessu_Curs_Ultimele_bucuri_de_date_and_IV_CS

31

CREATE .. TEMPORARY

- Se pot crea table temporare cu cereri CREATE GLOBAL TEMPORARY TABLE
- Definitia tablelor este vizibila tuturor sesiunilor active
- Datele din aceste table sunt vizibile doar sesiunii care le inseraza (doar una la un moment dat)
- Linile din tabela se sterg la sfarsitul fiecarei tranzactii sau la sfarsitul sesiunii, dupa cum se specifica in clauza ON COMMIT:

 - ON COMMIT DELETE ROWS – linile se sterg la sfarsitul fiecarei tranzactii
 - ON COMMIT PRESERVE ROWS – se sterg la sfarsit de sesiune.

F: Radulessu_Curs_Ultimele_bucuri_de_date_and_IV_CS

32

Exemplu

- 3. Adaugare tablele la cluster
- ```
CREATE TABLE dept_10 CLUSTER pers
(deptno)
AS SELECT * FROM scott.emp
WHERE deptno = 10;

CREATE TABLE dept_20 CLUSTER pers
(deptno)
AS SELECT * FROM scott.emp
WHERE deptno = 20;
```

## Rulare

```
f:\fisieripr\>
CREATE TABLE dept_10 CLUSTER pers
(deptno)
AS SELECT * FROM scott.emp
WHERE deptno = 10;

CREATE TABLE dept_20 CLUSTER pers
(deptno)
AS SELECT * FROM scott.emp
WHERE deptno = 20;
```

28

## Exemplu

```
CREATE GLOBAL TEMPORARY TABLE test_temp
(name VARCHAR2(20)) ON COMMIT DELETE
ROWS;
```

- Se va crea in acest caz o tabela temporara in care linile se sterg la sfarsitul fiecarei tranzactii.

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

33

## Rulare

```
f:\fisieripr\>
CREATE GLOBAL TEMPORARY TABLE test_temp
(name VARCHAR2(20)) ON COMMIT DELETE
ROWS;
Table created.

SQL> INSERT INTO test_temp VALUES ('Ion');

1 row created.

SQL> SELECT * FROM test_temp;

NAME

ION

SQL> COMMIT;

Commit complete.

SQL> SELECT * FROM test_temp;

no rows selected

SQL>
```

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

34

## Rulare

```
f:\fisieripr\>
SELECT *
FROM dept;
2 rows selected.

SQL> SELECT *
 2 FROM dept
 3 WHERE deptno = 10;
NO rows selected.

SQL> SELECT *
 2 FROM dept
 3 WHERE deptno = 20;
NO rows selected.

SQL> SELECT *
 2 FROM dept
 3 WHERE deptno = 30;
NO rows selected.

SQL>
```

## ENABLE / DISABLE

- ENABLE si DISABLE activeaza / inhiba constrangerile de integritate.
- Aceste constrangeri sunt dintr-o lantul de create in aceiasi comanda.
- In mod implicit, la creare, Oracle activeaza o constrangerare de integritate

30

## Copierea unei table

- Se poate copia o tabela schimbandu-i la momentul copierii anumiti parametri folosind CREATE TABLE .. AS SELECT
- In acest caz se pot specifica noi nume ale coloanelor, noi parametrii de stocare fizica etc.
- Atentie: nu sunt copiate in acest caz si constrangerile de integritate (cu exceptia coloanelor definite cu NOT NULL, care vor fi la fel si in noua tabela).

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

35

## ALTER TABLE - ALLOCATE

- ALLOCATE EXTENT aloca explicit o noua extensie pentru acea tabela.
- SIZE specifica dimensiunea extensiei in octeti (bytes). Se poate folosi K ori M pentru a specifica KB sau MB. In cazul absentei acestei clauze Oracle determina dimensiunea pe baza valorilor de STORAGE ale tabeliei.
- DATAFILE specifica numele fisierului (afinat tablespace-ului in care se gaseste tabela) in care se va aloca noua extensie. In lipsa alegerea este facuta de Oracle.

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

37

## ALTER TABLE - ALLOCATE

- INSTANCE face acea extensie disponibila pentru instanta specificata. Numarul instantei e dat de parametrul de initializare INSTANCE\_NUMBER. In lipsa extensiei va fi disponibila pentru toate instantele. Acest parametru este util in conjunctie cu Parallel Server.
- Alocarea explicita a unei extensii afecteaza dimensiunea urmatoarei extensii care va fi alocata pe baza parametrilor NEXT si PCTINCREASE (prezentati in capitoul anterior)

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

38

## Parametrii de stocare

- In cazul lui ALTER TABLE, semnificativa noulor valori ale acestor parametri este urmatoarea:
- NEXT – In momentul alocarii unei noi extensii Oracle va folosi noua valoare a lui NEXT dupa care cresterea se face pornind de la aceasta valoare si cea a lui PCTINCREASE (vezi formula din capitoul precedent pentru dimensiunea extensiei n)

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

43

## Parametrii de stocare - cont

- PCTINCREASE – de asemenea schimbarea acestuia va afecta dimensiunea noilor extensii care se aloca,
- MINEXTENTS – poate fi schimbat cu orice valoare care e mai mica sau egala cu numarul de extensii pe care le are tabela la acel moment
- MAXEXTENTS – poate fi schimbat cu orice valoare mai mare sau egala cu numarul de extensii ale tablei la acel moment

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

44

## ALTER TABLE - cont

- PARALLEL
- NOPARALLEL
- CACHE
- NOCACHE
- ENABLE
- DISABLE
- Sunt folosite pentru a schimba setarile curente.
- Aceste duse au fost prezentate la CREATE TABLE

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

39

## ALTER TABLE - LOCK

- ENABLE TABLE LOCK - Activeaza posibilitatea obtinerii de blocari asupra tablei de catre comenzi DDL.
- Aceste comenzi nu se pot executa daca blocarea nu este permisa.
- DISABLE TABLE LOCKS duce implicit la interzicerea operatiilorDDL asupra tablei.

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

40

## Parametri utilizare bloc

- PCTFREE – schimbarea sa afecteaza urmatoarele operatii de inserare. Blocurile 'umplute' dupa vechea valoare nu sunt afectate decat in momentul in care ele ajung in lista de blocuri cu spatiu liber – FREELIST, deci deasupra care se pot face operatii de inserare.
- Un bloc ajunge in lista de blocuri cu spatiu liber doar daca se efectueaza stergeri din el pana sub PCTUSED

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

45

## Parametri utilizare bloc - cont

- PCTUSED – orice modificar a acestui parametru va afecta toate blocurile din tabela. Daca o linie e actualizata sau stearsa blocul care o contine va fi testat daca poate fi pus in lista de blocuri cu spatiu liber.
- INITTRANS – schimbarea acestui parametru va afecta doar blocurile noi
- MAXTRANS – schimbarea acestui parametru va afecta toate blocurile tablei (pentru ca diferența de la INITTRANS pana la MAXTRANS sunt 'transaction entries' care se aloca dinamic).

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

46

## ALTER TABLE - TRIGGERS

- ENABLE ALL TRIGGERS
- DISABLE ALL TRIGGERS
- Permit activarea / dezactivarea tuturor dedansatorilor asociati unei table
- Pentru activarea / dezactivarea unui singur dedansator se poate folosi comanda ALTER TRIGGER.
- Clauza DROP – specifica stergerea unei constrangeri de integritate.

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

41

## Comanda ALTER TABLE - cont

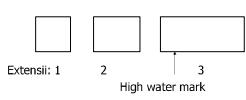
- PCTFREE,
- PCTUSED,
- INITTRANS,
- MAXTRANS,
- STORAGE
- Schimba valorile si optiunile care exista la acel moment. Explicatia semnificativei acestor parametri s-a facut la descrierea CREATE TABLE

F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

42

## High water mark

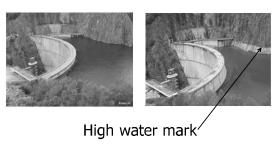
- Pentru orice segment (inclusiv deci pentru segmentele continand table) exista un marcat al ultimului bloc care a fost vredodata utilizat.
- Acest marcat se numeste 'high water mark' (HWM)



F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

47

## De ce se numeste asa?



F: Radulessu\_Curs\_Ultimele\_bucuri\_de\_date\_and\_IV\_CS

48





## Modificare date user

- Datele privind autentificarea utilizatorului:
 

```
ALTER USER user_name
 IDENTIFIED BY password
 EXTERNALLY
 | GLOBALLY AS 'external_name'
 [PASSWORD EXPIRE]
 [ACCOUNT [LOCK|UNLOCK]]
```
- În momentul blocării unui cont (LOCK), dacă utilizatorul este logat la acel moment nu va fi afectat. Modificările date de comandă de mai sus sunt valabile începând cu următoarea sesiune de lucru.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

21

## Modificare date user

```
+ SQL Plus:1g
SQL> ALTER SYSTEM
 SET RESOURCE_LIMIT = FALSE
 SCOPE=MEMORY;
ALTER SYSTEM
 SET RESOURCE_LIMIT = FALSE
 SCOPE=MEMORY
 FAILED: The account is locked
SQL> CONNECT STUDIO
 IDENTIFIED BY studio
 SCOPE=MEMORY;
CONNECT STUDIO
 IDENTIFIED BY studio
 FAILED: The account is locked
SQL> CONNECT STUDIO
 IDENTIFIED BY studio
 SCOPE=BOTH;
```

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

26

## Stergere user – cont.

```
+ SQL Plus:1g
SQL> DROP USER studio
 SCOPE=BOTH;
DROP USER studio
 SCOPE=BOTH
 FAILED: The account is locked
SQL> DROP USER studio
 SCOPE=BOTH;
DROP USER studio
 SCOPE=BOTH
 FAILED: The account is locked
SQL> CONNECT STUDIO
 IDENTIFIED BY studio
 SCOPE=BOTH;
CONNECT STUDIO
 IDENTIFIED BY studio
 FAILED: The account is locked
SQL> CONNECT STUDIO
 IDENTIFIED BY studio
 SCOPE=BOTH;
```

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

31

## Modificare date user - cont

- Datele privind tablespace și cota:
 

```
ALTER USER username
 [DEFAULT TABLESPACE tablespace]
 [TEMPORARY TABLESPACE tablespace]
 [QUOTA [K|M] ON tablespace]
 [QUOTA UNLIMITED ON tablespace]
```
- La trecerea pe 0 a cotei nu se mai pot crea obiecte și cele existente nu mai pot crește. Exemplu:
 

```
ALTER USER mihai341c5
 QUOTA 0 ON users;
```

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

27

## Modificare date user - cont

- Observație: trecerea pe 0 a cotei nu are efect dacă utilizatorul este asignat rolui (colectia de privilegii) RESOURCE deoarece aceasta implica o cota nelimitată.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

28

## Exemplu

```
SELECT TABLESPACE_NAME, BLOCKS, MAX_BLOCKS, BYTES,
 MAX_BYTES
 FROM DBA_TS_QUOTAS
 WHERE USERNAME = 'SCOTT';
 □ Se obține un rezultat care conține date despre cota utilizatorului:
```

| TABLESPACE_NAME | BLOCKS | MAX_BLOCKS | BYTES | MAX_BYTES |
|-----------------|--------|------------|-------|-----------|
| DATE            | 10     | -1         | 20480 | -1        |

□ Valoarea -1 reprezintă cota nelimitată. Restul valorilor reprezintă spațiul ocupat la acel moment.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

33

## Alt exemplu

```
SELECT USERNAME, ACCOUNT_STATUS,
 TEMPORARY_TABLESPACE
 FROM DBA_USERS
 □ Se obține o listă cu starea fiecărui cont (și altele):
 USERNAME ACCOUNT_STATUS TEMPORARY_TABLESPACE
```

| SYS    | OPEN | TEMP |
|--------|------|------|
| SYSTEM | OPEN | TEMP |
| DBSNMP | OPEN | TEMP |
| SCOTT  | OPEN | TEMP |

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

34

## Modificare date user - cont

- În exemplul de mai jos STUD1 sunt asignați roluri RESOURCE:
 

```
+ SQL Plus:1g
SQL> connect stud1/stud1
SQL> create table tta number;
Table created.
SQL> create table ttb number;
Table created.
SQL> connect stud1/stud1
SQL> alter user stud1 quota 0 on HEHE;
 Utilizator modificat.
SQL> connect stud1/stud1
SQL> create table ttc number;
Table created.
SQL> create table ttb number;
Table created.
SQL> create table ttcc number;
Table created.
```

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

29

## Stergere user

- Stergerea unui utilizator se face cu comanda DROP USER:
 

```
DROP USER name [CASCADE]
```
- Optiunea CASCADE stergă întâi toate obiectele din schema utilizatorului respectiv (altfel se obține un mesaj de eroare).
- Fara CASCADE se pot sterge utilizatorii care nu detin nici un obiect în schema proprie.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

30

## PROFIL

- Profilurile sunt o modalitate prin care se pot limita resursele care pot fi utilizate de un utilizator.
- Un profil se creează cu CREATE PROFILE și se asignează utilizatorului la creare sau ulterior prin comanda ALTER USER.
- Există un profil DEFAULT care se asociază implicit utilizatorului pentru care la creare nu s-a specificat un profil.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

35

## Resurse ale sistemului

- Timpul maxim de conectare măsurat în minute (CONNECT\_TIME). Sesiunile utilizatorului sunt închise de Oracle după expirarea acestui timp.
- Timp maxim de așteptare (IDLE\_TIME) – măsurat în minute – sesiunile vor fi închise de Oracle după expirarea perioadei specifică daca în sesiunea respectivă nu s-a facut nimic (e "idle"). Atenție: cererile a căror execuție este lungă nu intră în această categorie!

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

37

## Resurse ale sistemului

- Numărul maxim de blocuri citite de pe disc sau din memorie. Acest parametru este gândit pentru a limita cererile care fac citiri intensive (LOGICAL\_READS\_PER\_SESSION).
- Numărul maxim de blocuri citite per operatie (call) (LOGICAL\_READS\_PER\_CALL).
- Dimensiunea maximă de memorie ocupată în shared pool – parte a SGA – de o sesiune de lucru – în bytes (PRIVATE\_SGA).

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

38

## LIMITARI

- Dacă este atinsă o limită la nivel de operare (call) atunci:
  - Procesarea cererii curente este opriță
  - Cererea curentă este revocată (rollback)
  - Efectul cererilor anterioare persistă
  - Utilizatorul ramane conectat.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

43

## Creare și utilizare profil

- În continuare vom discuta despre cum se crează și se modifică un profil și despre cum este asignat un profil la un utilizator.
- Sintaxa cererii SQL de creare a unui nou profil este următoarea:

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

44

## Resurse legate de parola

- Numărul maxim de încercări eronate de login (FAILED\_LOGIN\_ATTEMPTS)
- Timpul maxim (în zile) cat parola este validă (PASSWORD\_LIFE\_TIME)
- Numărul minim de parole diferite utilizate până cand o parola poate fi reutilizată (PASSWORD\_REUSE\_MAX)
- Numărul minim de zile după care o parola poate fi utilizată (PASSWORD\_REUSE\_TIME)

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

39

## Resurse legate de parola

- Mai există:
  - PASSWORD\_LOCK\_TIME**: Cate zile se blochează contul după încercări repetate de login eronate.
  - PASSWORD\_GRACE\_TIME**: Cate zile sunt disponibile pentru a schimba parola după expirarea acesteia.
  - PASSWORD\_VERIFY\_FUNCTION**: bloc (program) PL/SQL utilizat pentru verificarea parolei.
  - SEC\_CASE\_SENSITIVE\_LOGON**: literale mari și mici sunt considerate identice sau nu într-o parolă.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

40

## Creare profil

```
CREATE PROFILE profile
 LIMIT
 [SESSIONS_PER_USER [integer | UNLIMITED | DEFAULT]]
 [CPU_PER_SESSION [integer | UNLIMITED | DEFAULT]]
 [CPU_PER_CALL [integer | UNLIMITED | DEFAULT]]
 [CONNECT_TIME [integer | UNLIMITED | DEFAULT]]
 [IDLE_TIME [integer | UNLIMITED | DEFAULT]]
 [LOGICAL_READS_PER_SESSION [integer | UNLIMITED | DEFAULT]]
 [LOGICAL_READS_PER_CALL [integer | UNLIMITED | DEFAULT]]
 [COMPOSITE_LIMIT [integer | UNLIMITED | DEFAULT]]
 [PRIVATE_SGA [integer [K|M] | UNLIMITED | DEFAULT]]
 [RESOURCE_NAME]

```

□ Nu sunt listate mai sus toate opțiunile.

□ Pentru a executa această operare trebuie să se execute CREATE PROFILE.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

45

## Creare profil – cont.

- UNLIMITED: ană apăsați pe acel profil resursa respectivă poate fi folosită în cota nelimitată.
- DEFAULT: ană apăsați pe resursa respectivă poate fi folosită în limită, valoarea fiind același cu a profilului DEFAULT.
- COMPOSITE\_LIMIT limitează costul total al resurselor pentru o sesiune de utilizator. Oracle calculează acest cost ca o sumă ponderată între:
  - CPU\_PER\_SESSION
  - CPU\_PER\_CALL
  - CONNECT\_TIME
  - LOGICAL\_READS\_PER\_SESSION
  - PRIVATE\_SGA

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

46

## alte informații

- Lista de mai sus nu este exhaustivă. Un tabel cu limitările care se pot folosi în Oracle 11.1 se găsește în pagina:
 <http://www.psoug.org/reference/profiles.html>
- Am dat numele parametrilor pentru ca fiecare în parte se poate modifica ulterior prin comenzi ALTER PROFILE.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

41

## LIMITARI

- Dacă este atinsă o limită la nivel de sesiune atunci:
  - Fișe se afisează un mesaj de eroare (de exemplu cand se încerce deschiderea unei noi sesiuni și se depășesc sessions\_per\_user)
  - Fișe Oracle deconectează utilizatorul (sesiunea), de exemplu cand s-a atins durata ei maximă.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

42

## COMPOSITE\_LIMIT

- Exemplu de setare a costului resurselor:
 

```
ALTER RESOURCE COST
 CPU_PER_SESSION 100
 CONNECT_TIME 100
 LOGICAL_READS_PER_SESSION 0
 PRIVATE_SGA 0
```
- Costul este determinat astfel:
 
$$\text{COST} = (100 * \text{CPU\_PER\_SESSION}) + (1 * \text{CONNECT\_TIME})$$
- Dacă nu se depășește costul stabilit de COMPOSITE\_LIMIT sesiunea este închisă de Oracle.
- Pentru parametri neșătăți se iau valori din DEFAULT.

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

47

## Exemplu

```
CREATE PROFILE EXEMPLU LIMIT
 SESSIONS_PER_USER 1
 IDLE_TIME 1
 FAILED_LOGIN_ATTEMPTS 3;
 □ Pentru a vizualiza restricțiile aferente profilului creat cu cererea de mai sus se poate executa cererea:
 SELECT RESOURCE_NAME, LIMIT
 FROM DBA_PROFILES
 WHERE PROFILE = 'EXEMPLU';
```

F.Rakitov\_Curs Utilizatori Oracle  
de laici, edit 1IV.CS

48

## Exemplu - cont

❑ Rezultatul va contine lista urmatoare:

```

CONNECTED AS SYSTEM
SESSIONS_PER_USER 1
CPU_PER_SESSION DEFAULT
CPU_PER_CALL DEFAULT
LOGICAL_READS_PER_SESSION DEFAULT
LOGICAL_READS_PER_CALL DEFAULT
TIDLE_TIME DEFAULT
PRIVATE_SGA DEFAULT
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_GRACE_TIME DEFAULT
PASSWORD_REUSE_TIME DEFAULT
PASSWORD_REUSE_FUNCTION DEFAULT
PASSWORD_LOCK_TIME DEFAULT
PASSWORD_GRACE_TIME DEFAULT

```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

49

## Asignarea unui profil

A asignarea unui profil la un user se poate face:

- ❑ La crearea userului (CREATE USER) exista clauza PROFILE care specifica un profil asociat acelui user (in lipsa se ia profilul DEFAULT).
- ❑ Ulterior se poate schimba profilul cu ALTER USER:

```
ALTER USER scott
PROFILE exemplu;
```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

50

## RESOURCE\_LIMIT - cont

❑ Putem schimba valoarea curenta cu ALTER SYSTEM:

```
ALTER SYSTEM
SET RESOURCE_LIMIT=TRUE;
```

❑ Efectul acestor comenzi dureaza pana la o noua schimbare a valorii cu ALTER SYSTEM sau pana cand se opreste baza de date (la reponire va fi citi din nou valoarea din fisierul de parametri)

## Modificare profil

❑ Modificarea unui profil se poate face cu ALTER PROFILE (similar cu CREATE):

```
ALTER PROFILE profile
LIMIT
[SESSIONS_PER_USER {integer | UNLIMITED | DEFAULT}]
[CPU_PER_SESSION {integer | UNLIMITED | DEFAULT}]
[CPU_PER_CALL {integer | UNLIMITED | DEFAULT}]
[LOGICAL_READS_PER_CALL {integer | UNLIMITED | DEFAULT}]
[CONNECT_TIME {integer | UNLIMITED | DEFAULT}]
[DISK_TIME {integer | UNLIMITED | DEFAULT}]
[LOGICAL_READS_PER_SESSION {integer | UNLIMITED | DEFAULT}]
[PRIVATE_SGA {integer | UNLIMITED | DEFAULT}]
```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

56

## Exemplu de limitare

```

SQL> CREATE PROFILE STUD1
 UNLIMITED SESSIONS_PER_USER;
Profile created.
SQL> ALTER USER SCOTT PROFILE STUD1;
User altered.
SQL> CONNECT SCOTT/pwd123
Connected.
SQL> SELECT * FROM DBA_PROFILES;
 PROFILE RESOURCE_NAME RESOURCE LIMIT
----- -----
 DEFAULT DISK_CONNECT_TIMEOUT UNLIMITED
 DEFAULT LOGICAL_READS_PER_CALL UNLIMITED
 DEFAULT LOGICAL_READS_PER_SESSION UNLIMITED
 DEFAULT PRIVATE_DISK_TIMEOUT UNLIMITED
 DEFAULT PRIVATE_DISK_TIME UNLIMITED
 DEFAULT PRIVATE_NET_TIMEOUT UNLIMITED
 DEFAULT PRIVATE_NET_TIME UNLIMITED
 DEFAULT PRIVATE_PWD_ATTEMPTS UNLIMITED
 DEFAULT PRIVATE_PWD_TIME UNLIMITED
 DEFAULT PRIVATE_PWD_UNLOCK_ATTEMPTS UNLIMITED
 DEFAULT PRIVATE_PWD_UNLOCK_TIME UNLIMITED
 DEFAULT PRIVATE_PWD_UNLOCK_FUNCTION UNLIMITED
 DEFAULT PRIVATE_PWD_UNLOCK_TYPE UNLIMITED
 STUD1 DISK_CONNECT_TIMEOUT UNLIMITED
 STUD1 LOGICAL_READS_PER_CALL UNLIMITED
 STUD1 LOGICAL_READS_PER_SESSION UNLIMITED
 STUD1 PRIVATE_DISK_TIMEOUT UNLIMITED
 STUD1 PRIVATE_DISK_TIME UNLIMITED
 STUD1 PRIVATE_NET_TIMEOUT UNLIMITED
 STUD1 PRIVATE_NET_TIME UNLIMITED
 STUD1 PRIVATE_PWD_ATTEMPTS UNLIMITED
 STUD1 PRIVATE_PWD_TIME UNLIMITED
 STUD1 PRIVATE_PWD_UNLOCK_ATTEMPTS UNLIMITED
 STUD1 PRIVATE_PWD_UNLOCK_TIME UNLIMITED
 STUD1 PRIVATE_PWD_UNLOCK_FUNCTION UNLIMITED
 STUD1 PRIVATE_PWD_UNLOCK_TYPE UNLIMITED
16 rows selected.
SQL>

```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

51

## Profilul DEFAULT

| PROFILE | RESOURCE_NAME               | RESOURCE LIMIT |
|---------|-----------------------------|----------------|
| DEFAULT | DISK_CONNECT_TIMEOUT        | UNLIMITED      |
| DEFAULT | CPU_PER_SESSION             | UNLIMITED      |
| DEFAULT | CPU_PER_CALL                | UNLIMITED      |
| DEFAULT | LOGICAL_READS_PER_SESSION   | UNLIMITED      |
| DEFAULT | LOGICAL_READS_PER_CALL      | UNLIMITED      |
| DEFAULT | PRIVATE_DISK_TIMEOUT        | UNLIMITED      |
| DEFAULT | PRIVATE_DISK_TIME           | UNLIMITED      |
| DEFAULT | PRIVATE_NET_TIMEOUT         | UNLIMITED      |
| DEFAULT | PRIVATE_NET_TIME            | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_ATTEMPTS        | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_TIME            | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_UNLOCK_ATTEMPTS | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_UNLOCK_TIME     | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_UNLOCK_FUNCTION | UNLIMITED      |
| DEFAULT | PRIVATE_PWD_UNLOCK_TYPE     | UNLIMITED      |

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

52

## Modificare profil - cont

❑ Modificările de profil nu afectează sesiunile curente ci doar pe cele deschise după modificare

❑ Pentru execuția comenzi trebuie privilegiul ALTER PROFILE.

## Stergerea unui profil

❑ Se face cu DROP PROFILE:  
DROP PROFILE nume\_profil [CASCADE]  
> Profilul DEFAULT nu se poate sterge  
> Stergerea unui profil nu afectează sesiunile curente  
> Optiunea CASCADE revoca accesul la userii care îl au  
> În useri care au profilul sters vor trece automat pe profilul DEFAULT  
> Pentru executia operatiiei trebuie privilegiul DROP PROFILE

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

56

## Profilul STUD1

```

SQL> CREATE PROFILE STUD1
 UNLIMITED SESSIONS_PER_USER;
Profile created.
SQL> ALTER USER SCOTT PROFILE STUD1;
User altered.
SQL> SELECT * FROM DBA_PROFILES;
 PROFILE RESOURCE_NAME RESOURCE LIMIT
----- -----
 DEFAULT DISK_CONNECT_TIMEOUT UNLIMITED
 DEFAULT LOGICAL_READS_PER_CALL UNLIMITED
 DEFAULT LOGICAL_READS_PER_SESSION UNLIMITED
 DEFAULT PRIVATE_DISK_TIMEOUT UNLIMITED
 DEFAULT PRIVATE_DISK_TIME UNLIMITED
 DEFAULT PRIVATE_NET_TIMEOUT UNLIMITED
 DEFAULT PRIVATE_NET_TIME UNLIMITED
 DEFAULT PRIVATE_PWD_ATTEMPTS UNLIMITED
 DEFAULT PRIVATE_PWD_TIME UNLIMITED
 STUD1 DISK_CONNECT_TIMEOUT UNLIMITED
 STUD1 LOGICAL_READS_PER_CALL UNLIMITED
 STUD1 LOGICAL_READS_PER_SESSION UNLIMITED
 STUD1 PRIVATE_DISK_TIMEOUT UNLIMITED
 STUD1 PRIVATE_DISK_TIME UNLIMITED
16 rows selected.
SQL>

```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

51

## RESOURCE\_LIMIT

❑ Aşa cum s-a specificat, parametrul de initializare RESOURCE\_LIMIT trebuie sa fie TRUE

❑ Pentru a vedea care este valoarea curenta a acestui parametru se poate folosi in SQL\*Plus comanda SHOW PARAMETER:

```
SQL> SHOW PARAMETER RESOURCE_LIMIT
RESOURCE_LIMIT BOOLEAN FALSE
```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

54

## Vizualizari

❑ Pentru a vedea informatii despre stare cont, blocare, data expirarii parolei si alte limitari ale acesteia se poate folosi vedereaza DBA\_USERS sau vedereaza DBA\_PROFILES;

```
SELECT USERNAME, PASSWORD, ACCOUNT_STATUS, EXPIRY_DATE
FROM DBA_USERS;
```

❑ Se obtine un rezultat similar decesc fice user, de tipul:

```
USERNAME PASSWORD ACCOUNT_STATUS EXPIRY_DATE
----- -----
SYS D4C5123456BDC08 OPEN 19-DEC-20
```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

56

## PRIVILEGII

❑ Există două tipuri de privilegii:

1. Privilegii sistemele permit utilizatorilor sa execute operatii pe baza de date: creare, stergere, modificarie pe tabele, vederi, sechide de roluri sau proceduri
2. Privilegii obiect: permit utilizatorilor sa efectueze anumite operatii pe un obiect: tabela, secheta, vedere, procedura, functie sau pacchet

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

62

## ROLURI

❑ Rolurile sunt colectii de privilegii (ca niste "cosuri" de privilegii) care pot fi asignate si revocate impreuna.

❑ Un rol poate fi creat, i se pot asocia privilegii (umprem cosul) dupa care el se poate asigna cu GRANT unui user sau unui alt rol.

❑ Există o serie de roluri predefinite (CONNECT, RESOURCE, DBA, etc.).

## Vizualizari - cont

❑ Pentru limitari asupra parolei, in cererile asupra lui DBA\_PROFILES se poate folosi in clauza WHERE conditia WHERE RESOURCE\_TYPE='PASSWORD'

❑ Exemplu:

```
SELECT PROFILE, RESOURCE_NAME, LIMIT
FROM DBA_PROFILES
WHERE RESOURCE_TYPE='PASSWORD'
```

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

60

## Privilegii sistem

❑ Există un numar mare de privilegii sistem.  
([http://www.psoug.org/reference/system\\_privs.html](http://www.psoug.org/reference/system_privs.html))

❑ Privilegii de sistem pot fi clasificate in:

- Privilegii pentru operatii care afecteaza intreg sistemul, ca de exemplu CREATE SESSION, CREATE TABLESPACE
- Privilegii care afecteaza obiectele din orice schema, de ex. CREATE ANY TABLE
- Privilegii care afecteaza doar obiectele din schema proprie, de ex. CREATE TABLE

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

62

## Privilegii sistem – cont.

❑ Particula ANY arata ca userul are acel privilegiu in orice schema.

❑ Pentru a adauga privilegii la un user se folosete GRANT.

❑ Pentru a inlatura privilegii de la un user se folosete REVOKE.

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

64

## ROLURI - Oracle 9

❑ Rolul RESOURCE contine privilegii:

- ❑ CREATE CLUSTER,
- ❑ CREATE INDEXTYPE,
- ❑ CREATE OPERATOR,
- ❑ CREATE PROCEDURE,
- ❑ CREATE SEQUENCE,
- ❑ CREATE TABLE,
- ❑ CREATE TRIGGER,
- ❑ CREATE TYPE

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

67

## ROLURI - Oracle 10

❑ Nota: In documentatia Oracle 10 se mentioneaza:

"Customers should discontinue using the CONNECT and RESOURCE roles, as they will be deprecated in future Oracle Database releases. The CONNECT role presently retains only the CREATE SESSION privilege."

(se参见 [http://docs.oracle.com/cd/B19306\\_01/doc/appdev/010/index.htm](http://docs.oracle.com/cd/B19306_01/doc/appdev/010/index.htm))

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

66

## ROLURI - Oracle 12c

❑ Legat de aceste roluri, in Oracle 12c:

- ❑ CONNECT: Contine doar CREATE SESSION si SET CONTAINER – ca privilegii de sistem
- ❑ RESOURCE: Contine toate privilegii mentionate anterior dar se specific faptul ca in versiunile ulterioare de Oracle ar putea sa nu mai fie prezenti.
- ❑ DELEGATORI sunt sfatuiti sa-și creeze propriile roluri, adaptate nevoilor lor de securitate.

F:\Rakibou\_Curs Utilizatori Oracle\_baza de date\_and PC.cs

71

## Exemplu de privilegii - CREATE

- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type
- ❑ Create Any View
- ❑ Create Any Index
- ❑ Create Any IndexType
- ❑ Create Any Materialized View
- ❑ Create Any Procedure
- ❑ Create Any Operator
- ❑ Create Any Job
- ❑ Create Any Sequence
- ❑ Create Any SQL Profile
- ❑ Create Any Synonym
- ❑ Create Any Table
- ❑ Create Any Trigger
- ❑ Create Any Type

**Aaignarea privilegiilor: GRANT**

- In cazul specificarii WITH ADMIN OPTION cel care primește privilegiul poate să îl asigne mai departe, inclusiv cu WITH ADMIN OPTION.
- Utilizari care au privilegiul GRANT ANY ROLE pot să asigneze orice rol în sistem.
- Cel care primește un privilegiu cu WITH ADMIN OPTION îl poate de asemenea revoca de la orice user sau rol din sistem (nu numai de la cel căruia îl a asignat).
- În general privilegiile SYSDBA și SYSOPER (căci discutat despre de anterior) trebuie asignate doar utilizatorilor de tip administrator pentru ca ele dă acces la orice operație în baza de date (SYSDBA).

F.Rădulescu, Curs Utilizator Oracle

73

**Vizualizare privilegii**

- Există vederile DBA\_SYS\_PRIVS și SESSION\_PRIVS care pot fi interogate pentru a vedea privilegiile asociate fiecarui user (DBA\_SYS\_PRIVS) sau sesiunii curente (SESSION\_PRIVS).
- Privilegiile afisate provin atât din privilegi assignate individual cat și din privilegi asociate cu rolurile assignate utilizatorilor.

F.Rădulescu, Curs Utilizator Oracle

74

### Exemple

**Asignare:**

```
GRANT SELECT, INSERT, UPDATE, DELETE ON emp TO stud1;
GRANT ALL ON emp TO stud1;
GRANT SELECT ON emp TO public;
GRANT UPDATE(punctaj) ON stud TO scott
```

**Revocare:**

```
REVOKE DELETE ON emp FROM stud1;
REVOKE ALL ON emp FROM stud1;
REVOKE ALL ON emp FROM public;
```

F.Rădulescu, Curs Utilizator Oracle

79

**Listare privilegii și roluri**

```
SELECT LPAD(' ', 2*level) || granted_role
 "USER PRIVS"
 FROM (
 SELECT NULL grantee, username granted_role
 FROM dba_users
 WHERE username LIKE UPPER('%&uname%')
 UNION
 SELECT grantee, granted_role
 FROM dba_role_privs
 UNION
 SELECT grantee, privilege
 FROM dba_sys_privs
 START WITH grantee IS NULL
CONNECT BY grantee = prior granted_role;
```

F.Rădulescu, Curs Utilizator Oracle

80

### Revocarea privilegiilor: REVOKE

**Sintaxa este:**

```
REVOKE { priv_sistem | rol },
 { priv_sistem | rol }...
 FROM { user | PUBLIC },
 { user | rol | PUBLIC }...
```

**Revoca aceléle privilegi care au fost assignate cu GRANT.**

**Revocarea unor privilegi poate face ca anumite proceduri sau vederi care aveau nevoie de acel privilegi să devină invalide.**

**Revocarea unui privilegiul de la un user nu afectează userii carora acesta le-a transmis privilegiul – deci REVOKE nu are efect în cascada.**

F.Rădulescu, Curs Utilizator Oracle

75

### Privilegii obiect

**Sintaxă:**

```
GRANT {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)]
[, {object_priv | ALL [PRIVILEGES]}
[(column [, column] ...)] ...]
```

**ON [schema.]object**

```
TO {user | role | PUBLIC}
[, {user | role | PUBLIC}] ...
[WITH GRANT OPTION]
```

F.Rădulescu, Curs Utilizator Oracle

76

SQL> SELECT \* FROM dba\_objects WHERE object\_name='EMP'

1| 2| 3| 4| 5| 6| 7| 8| 9| 10| 11| 12| 13| 14| 15| 16| 17| 18| 19| 20| 21| 22| 23| 24| 25| 26| 27| 28| 29| 30| 31| 32| 33| 34| 35| 36| 37| 38| 39| 40| 41| 42| 43| 44| 45| 46| 47| 48| 49| 50| 51| 52| 53| 54| 55| 56| 57| 58| 59| 60| 61| 62| 63| 64| 65| 66| 67| 68| 69| 70| 71| 72| 73| 74| 75| 76| 77| 78| 79| 80| 81| 82| 83| 84| 85| 86| 87| 88| 89| 90| 91| 92| 93| 94| 95| 96| 97| 98| 99| 100| 101| 102| 103| 104| 105| 106| 107| 108| 109| 110| 111| 112| 113| 114| 115| 116| 117| 118| 119| 120| 121| 122| 123| 124| 125| 126| 127| 128| 129| 130| 131| 132| 133| 134| 135| 136| 137| 138| 139| 140| 141| 142| 143| 144| 145| 146| 147| 148| 149| 150| 151| 152| 153| 154| 155| 156| 157| 158| 159| 160| 161| 162| 163| 164| 165| 166| 167| 168| 169| 170| 171| 172| 173| 174| 175| 176| 177| 178| 179| 180| 181| 182| 183| 184| 185| 186| 187| 188| 189| 190| 191| 192| 193| 194| 195| 196| 197| 198| 199| 200| 201| 202| 203| 204| 205| 206| 207| 208| 209| 210| 211| 212| 213| 214| 215| 216| 217| 218| 219| 220| 221| 222| 223| 224| 225| 226| 227| 228| 229| 230| 231| 232| 233| 234| 235| 236| 237| 238| 239| 240| 241| 242| 243| 244| 245| 246| 247| 248| 249| 250| 251| 252| 253| 254| 255| 256| 257| 258| 259| 260| 261| 262| 263| 264| 265| 266| 267| 268| 269| 270| 271| 272| 273| 274| 275| 276| 277| 278| 279| 280| 281| 282| 283| 284| 285| 286| 287| 288| 289| 290| 291| 292| 293| 294| 295| 296| 297| 298| 299| 300| 301| 302| 303| 304| 305| 306| 307| 308| 309| 310| 311| 312| 313| 314| 315| 316| 317| 318| 319| 320| 321| 322| 323| 324| 325| 326| 327| 328| 329| 330| 331| 332| 333| 334| 335| 336| 337| 338| 339| 340| 341| 342| 343| 344| 345| 346| 347| 348| 349| 350| 351| 352| 353| 354| 355| 356| 357| 358| 359| 360| 361| 362| 363| 364| 365| 366| 367| 368| 369| 370| 371| 372| 373| 374| 375| 376| 377| 378| 379| 380| 381| 382| 383| 384| 385| 386| 387| 388| 389| 390| 391| 392| 393| 394| 395| 396| 397| 398| 399| 400| 401| 402| 403| 404| 405| 406| 407| 408| 409| 410| 411| 412| 413| 414| 415| 416| 417| 418| 419| 420| 421| 422| 423| 424| 425| 426| 427| 428| 429| 430| 431| 432| 433| 434| 435| 436| 437| 438| 439| 440| 441| 442| 443| 444| 445| 446| 447| 448| 449| 450| 451| 452| 453| 454| 455| 456| 457| 458| 459| 460| 461| 462| 463| 464| 465| 466| 467| 468| 469| 470| 471| 472| 473| 474| 475| 476| 477| 478| 479| 480| 481| 482| 483| 484| 485| 486| 487| 488| 489| 490| 491| 492| 493| 494| 495| 496| 497| 498| 499| 500| 501| 502| 503| 504| 505| 506| 507| 508| 509| 510| 511| 512| 513| 514| 515| 516| 517| 518| 519| 520| 521| 522| 523| 524| 525| 526| 527| 528| 529| 530| 531| 532| 533| 534| 535| 536| 537| 538| 539| 540| 541| 542| 543| 544| 545| 546| 547| 548| 549| 550| 551| 552| 553| 554| 555| 556| 557| 558| 559| 5510| 5511| 5512| 5513| 5514| 5515| 5516| 5517| 5518| 5519| 5520| 5521| 5522| 5523| 5524| 5525| 5526| 5527| 5528| 5529| 5530| 5531| 5532| 5533| 5534| 5535| 5536| 5537| 5538| 5539| 55310| 55311| 55312| 55313| 55314| 55315| 55316| 55317| 55318| 55319| 55320| 55321| 55322| 55323| 55324| 55325| 55326| 55327| 55328| 55329| 55330| 55331| 55332| 55333| 55334| 55335| 55336| 55337| 55338| 55339| 55340| 55341| 55342| 55343| 55344| 55345| 55346| 55347| 55348| 55349| 55350| 55351| 55352| 55353| 55354| 55355| 55356| 55357| 55358| 55359| 55360| 55361| 55362| 55363| 55364| 55365| 55366| 55367| 55368| 55369| 55370| 55371| 55372| 55373| 55374| 55375| 55376| 55377| 55378| 55379| 55380| 55381| 55382| 55383| 55384| 55385| 55386| 55387| 55388| 55389| 55390| 55391| 55392| 55393| 55394| 55395| 55396| 55397| 55398| 55399| 553100| 553101| 553102| 553103| 553104| 553105| 553106| 553107| 553108| 553109| 553110| 553111| 553112| 553113| 553114| 553115| 553116| 553117| 553118| 553119| 553120| 553121| 553122| 553123| 553124| 553125| 553126| 553127| 553128| 553129| 553130| 553131| 553132| 553133| 553134| 553135| 553136| 553137| 553138| 553139| 553140| 553141| 553142| 553143| 553144| 553145| 553146| 553147| 553148| 553149| 553150| 553151| 553152| 553153| 553154| 553155| 553156| 553157| 553158| 553159| 553160| 553161| 553162| 553163| 553164| 553165| 553166| 553167| 553168| 553169| 553170| 553171| 553172| 553173| 553174| 553175| 553176| 553177| 553178| 553179| 553180| 553181| 553182| 553183| 553184| 553185| 553186| 553187| 553188| 553189| 553190| 553191| 553192| 553193| 553194| 553195| 553196| 553197| 553198| 553199| 553200| 553201| 553202| 553203| 553204| 553205| 553206| 553207| 553208| 553209| 553210| 553211| 553212| 553213| 553214| 553215| 553216| 553217| 553218| 553219| 553220| 553221| 553222| 553223| 553224| 553225| 553226| 553227| 553228| 553229| 553230| 553231| 553232| 553233| 553234| 553235| 553236| 553237| 553238| 553239| 553240| 553241| 553242| 553243| 553244| 553245| 553246| 553247| 553248| 553249| 553250| 553251| 553252| 553253| 553254| 553255| 553256| 553257| 553258| 553259| 553260| 553261| 553262| 553263| 553264| 553265| 553266| 553267| 553268| 553269| 553270| 553271| 553272| 553273| 553274| 553275| 553276| 553277| 553278| 553279| 553280| 553281| 553282| 553283| 553284| 553285| 553286| 553287| 553288| 553289| 553290| 553291| 553292| 553293| 553294| 553295| 553296| 553297| 553298| 553299| 553200| 553201| 553202| 553203| 553204| 553205| 553206| 553207| 553208| 553209| 553210| 553211| 553212| 553213| 553214| 553215| 553216| 553217| 553218| 553219| 553220| 553221| 553222| 553223| 553224| 553225| 553226| 553227| 553228| 553229| 553230| 553231| 553232| 553233| 553234| 553235| 553236| 553237| 553238| 553239| 553240| 553241| 553242| 553243| 553244| 553245| 553246| 553247| 553248| 553249| 553250| 553251| 553252| 553253| 553254| 553255| 553256| 553257| 553258| 553259| 553260| 553261| 553262| 553263| 553264| 553265| 553266| 553267| 553268| 553269| 553270| 553271| 553272| 553273| 553274| 553275| 553276| 553277| 553278| 553279| 553280| 553281| 553282| 553283| 553284| 553285| 553286| 553287| 553288| 553289| 553290| 553291| 553292| 553293| 553294| 553295| 553296| 553297| 553298| 553299| 553300| 553301| 553302| 553303| 553304| 553305| 553306| 553307| 553308| 553309| 553310| 553311| 553312| 553313| 553314| 553315| 553316| 553317| 553318| 553319| 553320| 553321| 553322| 553323| 553324| 553325| 553326| 553327| 553328| 553329| 553330| 553331| 553332| 553333| 553334| 553335| 553336| 553337| 553338| 553339| 553340| 553341| 553342| 553343| 553344| 553345| 553346| 553347| 553348| 553349| 553350| 553351| 553352| 553353| 553354| 553355| 553356| 553357| 553358| 553359| 553360| 553361| 553362| 553363| 553364| 553365| 553366| 553367| 553368| 553369| 553370| 553371| 553372| 553373| 553374| 553375| 553376| 553377| 553378| 553379| 553380| 553381| 553382| 553383| 553384| 553385| 553386| 553387| 553388| 553389| 553390| 553391| 553392| 553393| 553394| 553395| 553396| 553397| 553398| 553399| 553400| 553401| 553402| 553403| 553404| 553405| 553406| 553407| 553408| 553409| 553410| 553411| 553412| 553413| 553414| 553415| 553416| 553417| 553418| 553419| 553420| 553421| 553422| 553423| 553424| 553425| 553426| 553427| 553428| 553429| 553430| 553431| 553432| 553433| 553434| 553435| 553436| 553437| 553438| 553439| 553440| 553441| 553442| 553443| 553444| 553445| 553446| 553447| 553448| 553449| 553450| 553451| 553452| 553453| 553454| 553455| 553456| 553457| 553458| 553459| 553460| 553461| 553462| 553463| 553464| 553465| 553466| 553467| 553468| 553469| 553470| 553471| 553472| 553473| 553474| 553475| 553476| 553477| 553478| 553479| 553480| 553481| 553482| 553483| 553484| 553485| 553486| 553487| 553488| 553489| 553490| 553491| 553492| 553493| 553494| 553495| 553496| 553497| 553498| 553499| 553500| 553501| 553502| 553503| 553504| 553505| 553506| 553507| 553508| 553509| 553510| 553511| 553512| 553513| 553514| 553515| 553516| 553517| 553518| 553519| 553520| 553521| 553522| 553523| 553524| 553525| 553526| 553527| 553528| 553529| 553530| 553531| 553532| 553533| 553534| 553535| 553536| 553537| 553538| 553539| 553540| 553541| 553542| 553543| 553544| 553545| 553546| 553547| 553548| 553549| 553550| 553551| 553552| 553553| 553554| 553555| 553556| 553557| 553558| 553559| 553560| 553561| 553562| 553563| 553564| 553565| 553566| 553567| 553568| 553569| 553570| 553571| 553572| 553573| 553574| 553575| 553576| 553577| 553578| 553579| 553580| 553581| 553582| 553583| 553584| 553585| 553586| 553587| 553588| 553589| 553590| 553591| 553592| 553593| 553594| 553595| 553596| 553597| 553598| 553599| 553600| 553601| 553602| 553603| 553604| 553605| 553606| 553607| 553608| 553609| 553610| 553611| 553612| 553613| 553614| 553615| 553616| 553617| 553618| 553619| 553620| 553621| 553622| 553623| 553624| 553625| 553626| 553627| 553628| 553629| 553630| 553631| 553632| 553633| 553634| 553635| 553636| 553637| 553638| 553639| 553640| 553641| 553642| 553643| 553644| 553645| 553646| 553647| 553648| 553649| 553650| 553651| 553652| 553653| 553654| 553655| 553656| 553657| 553658| 553659| 553660| 553661| 553662| 553663| 553664| 553665| 553666| 553667| 553668| 553669| 553670| 553671| 553672| 553673| 553674| 553675| 553676| 553677| 553678| 553679| 553680| 553681| 553682| 553683| 553684| 553685| 553686| 553687| 553688| 553689| 553690| 553691| 553692| 553693| 553694| 553695| 553696| 553697| 553698| 553699| 553700| 553701| 553702| 553703| 553704| 553705| 553706| 553707| 553708| 553709| 553710| 553711| 553712| 553713| 553714| 553715| 553716| 553717| 553718| 553719| 553720| 553721| 553722| 553723| 553724| 553725| 553726| 553727| 553728| 553729| 553730| 553731| 553732| 553733| 553734| 553735| 553736| 553737| 553738| 553739| 5537340| 5537341| 5537342| 5537343| 5537344| 5537345| 5537346| 5537347| 5537348| 5537349| 5537340| 5537341| 5537342| 5537343| 5537344| 5537345| 5537346| 5537347| 5537348| 5537349| 5537340| 5537341| 5537342| 5537343| 5537344| 5537345| 5537346| 5537347| 5537348| 5537349| 5537

## Prelucrarea datelor cu tehnici de Data Mining

### Cuprins

- Ce este data mining
- Etapele procesului de data mining
- Metode si subdomenii in data mining
- Preprocesarea datelor
- Sumar

### Concluzii

- Procesul de data mining converteste datele in cunostinte valoioase care pot fi folosite pentru suportul deciziei.
- Domeniul Data mining consta intr-o colectie de metodologii de analiza, tehnici si algoritmi pentru descoperirea noii sabioane.
- Data mining se foloseste cu precadere pentru seturi mari de date.
- Procesul de data mining este automat (nu necesita interventie umana).
- Data mining si Knowledge Discovery in Databases (KDD) sunt concepte apropiate, dar nu sunt identice. KDD este un set de etape si algoritmi care transforma datele dintr-o baza de date in cunostinte utile. Data mining este o parte a KDD-ului. Aceasta este etapa de analiza si de extragere a sabioanelor a procesului de descoperire a cunostintelor (KDD), etapa care se desfasoara dupa curatare si transformare a datelor si inainte de vizualizarea si evaluarea rezultatelor.

### Exemple de succes

- In cursurile profesorului J. Ullman de la Stanford sunt enumerate cateva exemple de succes ale folosirii tehnicielor de data mining (preluate din [Ullman 03]):
  - Arboar de decizie construit din istoria imprumuturilor bancare pentru a genera algoritmi de acordante si impreunaturi.
  - Sabioanele ale comportamentului calatorilor pentru a optimiza vanzarile cu pret redus a locurilor la avion sau a camerelor de hotel.
  - Scutece si bere: "S-a observat ca cel care cumpara scutece sunt mai inclinati decat ceilalati sa cumpere bere. Datorita acestor cunostinte, compania care produce scutece a deschis un supermarket unul in apropierea celuilalt, astfel incat multi cumparatori vor circula intre cele doua raiabane. Plasarea cipurilor pe trasee a dus la creșterea vanzarilor pentru toate cele 3 produse."

## Definitie ([Liu 11])

- Data mining este cunoscut si sub numele KDD - Knowledge Discovery in Databases (descoperirea de cunostinte in date).
- In mod obisnuit este definit ca procesul de descoperire a unor sabioane/modele (eng. patterns) / cunostinte utile in diverse surse de date/baze de date, colectii de texte, de imagini sau de pagini web, etc.
- Sabioanele trebuie sa fie valide, utile si inteligibile.

## Definitie ([Ullman 09, 10])

- Descoperirea de sabioane utile, uneori neasteptate, in date.
- Descoperirea de "modele" pentru date prin tehnici de tipul:
  - Modelare statistica
  - Invatare automata (Machine learning)
  - Abordari computationale in modelare
  - Sumarizare
  - Extragera proprietatilor (Feature Extraction)

## Exemple de succes

- Skycat si Sloan Sky Survey: gruparea obiectelor ceresti dupa nivelele radiatiei electromagnetice intr-un numar de benzi de frecventa sau permis identificarea galaxiilor, stelilor apropiate si a altor categorii de obiecte celeste.
- Compararea genotipului persoanelor avand sau neavand o anumita problema de sanatate a permis descoperirea unor gene care sunt asociate cu respectiva problema (de exemplu diabet). In acest fel se pot lua masuri de preventie inainte de declararea bolii in cazul persoanelor care prezinta respective caracteristici.

## Ce NU este Data Mining:

- Cautarea unei percase in baza de date a unei organizatii.
- Calcularea de valori minime, maxime, medii, sume sau numararea valorilor afilate in tabelele unei baze de date (operatii pur statistice).
- Folosirea unui motor de cautare pentru a gasi referintele asociate numelui tau.

## Definitie ([Wikipedia])

- Data mining (etapa de analiza a procesului KDD - knowledge discovery in databases) este procesul de descoperirea de noi sabioane in seturi mari de date prin metode aflate la intersecția dintre inteligența artificială, învățarea automată, statistică și baze de date.
- Obiectivul procesului de data mining este acela de a extraage cunostinte dintr-un set de date intr-o forma inteligibila pentru utilizatorii umani.
- Procesul implica baze de date si gestiunea informatiei, preprocesarea datelor, probleme de modelare si inferenta, metriki pentru performanta rezultatelor, probleme de complexitate, post procesare si vizualizare.

## Definitie ([Kimball, Ross 02])

- O categorie de cereri execute adesea pe datele atomice pentru a gasi sabioane/modele neasteptate in date.
- Cele mai cunoscute subdomenii ale data mining sunt clasificarea, gruparea (clustering), estimarea, predictia si gasirea evenimentelor care apar impreuna.
- Există multe tipuri de unelete pentru data mining. Cele mai importante sunt arborii de decizie, retele neurale, metode pentru vizualizarea seturilor de date, algoritmi genetici, logica fuzzy si unelete folosite in statistica obisnuita.
- In cadrul aplicatiilor concrete pentru mediul economic data mining foloseste de obicei datele afilate intr-un depozit de date (data warehouse).

## Date, informatii, cunostinte

- Să zicem că ai un magazin și înregistri toate detalii clientilor tăi in baza de date a acestuia. Să îți numești clientul și produsele pe care le cumpără în fiecare săptămână.
- De exemplu, Alex, Jessie și Paul vizitează magazinul tău in fiecare duminică și cumpără lumanări. Stochări acestea informații în aplicația informatică a magazinului. Acestea sunt date. Or de către ai dorești să afli cine sunt clientii care cumpără lumanări, porțiajăgăzărie baza de date și primești răspunsul. Acesta este un rezultat. Deci, să te întrebă: ce informații sunt văzute in fiecare zi in săptămâna în magazinul tău, pot să întregești din nou baza de date și vei primi răspunsul - accesați-o și aflați informație.
- Să presupunem că există și alti 1000 de clienti care, de asemenea, cumpără lumanări de la tine in fiecare duminică (importanță - cu un anumit procent de variabilitate). De ceva văzută în fiecare săptămână. Rezultat. Deci, pot concluziona că Alex, Jessie și Paul trebuie să fie și ei crestători.

## Date, informatii, cunostinte

- Acum, religia lui Alex, la lui Jessie și la lui Pavel nu o ai ca date in datele stocate. Acest lucru nu poate fi regăsit in baza de date ca informație. Dar ai obținut direct aceasta informație. Aceasta este o "cunostință" pe care ai descoperit-o. și așa cum se poate face printind un proces numit "Data Mining".
- Există sansa să te înscrii in legătura cu Alex, Jessie și Paul. Dar există o rezoluție de să te să înscrii. De aceea este foarte importantă să "evaluați" rezultatul și să îl depărtați. De asemenea, trebuie să îl evaluați și în ceea ce privește siguranța.
- Am dat acest exemplu pentru că am vrut să fac o distincție clară între cunoștințe și informații în contextul domeniului Data Mining. Este important de înțeles... de ce regăsesc informații de orunde s-ar afla in baza de date **nu este același lucru cu Data Mining**. Indiferent cat de complex este procesul de descoperire a cunostintelor, indiferent cat de adânc este localizată informația, asta nu este Data Mining.

## Software pentru DM

- In lucrarea [Mikut, Reichert 11] programele software pentru data mining sunt clasificate in 9 categorii:
  - Suite pentru data mining (Data mining suites - DMS) sunt pachete de instrumente de lucru cu date folosite in aplicatii de business. Inclusiv IBM SPSS Modeler, SAS Enterprise Miner, DataMining, Oracle Data Mining, Knowledge Studio, NAG Data Mining Components, STATISTICA
  - Open source
- Pachete de Business Intelligence (Business intelligence packages - BIP) includ functionalitatea de baza pentru DM - de exemplu metode statistice pentru aplicatii de business. Exemplu:
  - Comerciale: IBM Cognos 8 BI, Oracle DataMining, SAPNetweaver Business Warehouse, Teradata Database, DB2 Data Warehouse from IBM
  - Open source: Pentaho

## Software pentru DM

- **Pachete matematice** (Mathematical packages - MATs) contin o extensie mare si extensibilitatea de algoritmi precum si rutine de visualizare:
  - Comerciale: MATLAB, R-PLUS
  - Open source: R, Kepler
- **Pachete integrate** (Integration packages - INTs) sunt colectii extensibile de algoritmi de tip open-source. Exemplu:
  - Softwareurile KNIME, versiunea GUI pentru WEKA, KEEL, TANAGRA
  - Extensiile pentru unelele de tip MAT descrie mai sus.
- **Extensiile (EXT)** sunt accesorii (add-ons) pentru unelele au Excel, Matlab, R, cu o functionalitate limitata dar utila in diverse cazuri. Exemplu:
  - Rețele neuronale pentru Excel (Forecaster XL, XLMiner)
  - MATLAB (Matlab Neural Networks Toolbox).

## Preprocesarea datelor

- Curatarea datelor (Data cleaning):
  - înlocuirea sau înălțarea valorilor lipsa,
  - netezirea datelor care contine zgromadit,
  - identificarea si eventual înălțarea punctelor izolate (eng. outliers - valori izolate, departate de toate celelalte),
  - înălțarea inconsistentelor.

Nota: in multe documente termenul 'outliers' a fost tradus in limba romana prin "valori abervative" - prin similaritate cu traducerea in limba franceza.

## Preprocesarea datelor

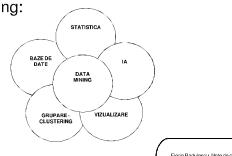
- Integrarea datelor (Data integration):
  - integrarea intr-un corp unic a datelor proveniente din mai multe surse,
  - conversii de tip si de structura,
  - eliminarea duplicatelor,
  - tratarea inconsistentelor datorate integrarii.

## Software pentru DM

- Biblioteci de DM (Data mining libraries - LIBs) contin functii care pot fi folosite in alte aplicatii de business. Exemplu:
  - Numerical Recipes in C, Java, C++, M-File si Data Mining Package, libSVM
- Specialisti (Specialists - SPECS) sunt similare cu DMS dar implementeaza doar o categorie/familie de metode. Exemplu:
  - CART, Bayesia Lab, C3D, WitRule, Rule Discovery System, MagnumOpus, JavaBayes, Naive Bayes, Neural Network, Random Forest, etc.
- Cercurile (Research - RES) sunt prima implementare ale unor noi algoritmi, avand de obicei un grafic restrans si fara foarte multe facilitati in utilizare. Suntem de obicei open source. WEKA si RapidMiner sunt cele mai cunoscute.
- Solutii (Solutions - SOLs) descriu unele care sunt adaptate la un domeniu ingust de aplicatii. Exemplu: pentru text mining: GATE; pentru prelucrare de imagini: ITK, ImageJ; cercetarea farmaceutica: Molgen Data Modeler

## Comunitati implicate

Cele mai importante comunitati implicate in data mining:



## Preprocesarea datelor

- Transformarea datelor (Data transformation):
  - normalizarea (sau standardizarea) datelor,
  - sumarizari,
  - generalizari,
  - constructia de noi atribute, etc.

## Preprocesarea datelor

- Reducerea datelor (Data reduction): nu toate atributele existente sunt necesare pentru un anumit proces de data mining.
- Prin reducere se opresc doar acele atribute care sunt necesare diminuand astfel volumul de date prelucrate (si implicit timpul de rulare al algoritmilor).

## Cuprins

- Ce este data mining
- Etapele procesului de data mining
- Metode si subdomenii in data mining
- Preprocesarea datelor
- Sumar

## Etapele procesului de DM (1)

1. **Colectarea datelor:** Datele sunt culese din baze de date existente sau prin parcurgerea paginilor web (Web crawling).
2. **Preprocesarea datelor:** presupune mai multe activitati desfasurate in scopul pregatirii datelor pentru aplicarea algoritmilor specifici

## Preprocesarea datelor

- **Discretizare:** Anumiti algoritmi lucreaza doar pe date discrete. De aceea pentru atributele avand valori continue trebuie efectuata discretizarea constant in inlocuirea acestor valori cu altele dintr-o multime de valori discrete.
- De exemplu varsta se poate inlocui cu un atribut avand doar trei valori: Tanar, Adult si Batran.

## Etapele procesului de DM (2)

3. **Extragererea sabioanelor si descoperirea de cunostinte.** Aceasta este etapa in care sunt efectiv utilizati algoritmi de DM pentru obtinerea rezultatului. Unii autori reduc domeniul Data Mining la aceasta etapa, intregul proces fiind numit KDD.
4. **Vizualizarea:** deosebita data mining proprieta/informatie ascunsie din date este necesara o vizualizare a rezultatelor pentru o mai buna intelegeri a lor si pentru a le evalua. Vizualizarea este uneori necesara si pentru datele de intrare.

## Etapele procesului de DM (3)

- Evaluarea rezultatului:** nu tot ce ieșe dintr-un proces de data mining este valoare.
- Unele rezultate sunt adevaruri pur statistică care se poate deduce și fără aplicarea algoritmilor, iar altele nu prezintă interes pentru utilizator.
- De aceea este necesară evaluarea rezultatelor de către expert pentru a separa cunoștințele valuroase de celelalte lucruri obținute în urma rularii algoritmului.

25 

## Principiul lui Bonferroni



O cunoștință descoperită în urma procesului de data mining poate fi un adevar pur statistic. Exemplu (din [Ullman 03]):

- În 1950 David Rhine, un parapsiholog de la universitatea Duke a testat studentul pentru a afla dacă au sau nu percepție extrasenzorială (ESP).
- Pentru acesta l-a cerut să ghicească culoarea a 10 carti de joc successive - rosu sau negru. Rezultatul a fost că 1/1000 din participanți au ghicit toate cele 10 carti - în consecință el a declarat că având ESP.

26 

## Algoritmi



### Algoritmi de predicție:

- Clasificare
- Regresie
- Detectia deviatiei

### Algoritmi de descriere:

- Grupare - Clustering
- Gasirea de reguli de asociere
- Descoperirea de sabloane secențiale

31 

## Clasificare



### Intrare:

- Un set având un număr k de clase  $C = \{c_1, c_2, \dots, c_k\}$ .
- Un set de  $n$  articole etichetate  $D = \{(d_1, c_1), (d_2, c_2), \dots, (d_n, c_n)\}$ . Articolele sunt  $d_i$ ,  $c_i$ , fiecare articol  $d_i$  fiind etichetat cu clasa  $c_i \in C$ . D este numit și setul (multimea) de antrenare (training set).
- Pentru calibrarea unor algoritmi este necesar și un set (multime) de validare (validation set). Acesta conține de asemenea articole etichetate, neniechis în setul de antrenare.
- Un model sau metodă de a clasifica noi articole (clasificator). Seul conținând noile articole se numește set de test (test set)

32 

## Principiul lui Bonferroni

- Re-testarea efectuată doar asupra acestora nu a mai avut aceleși rezultate, el considerand că în momentul în care au alături ca ESP au pierdut această capacitate.
- David Rhine nu a realizat ca statistica spune că probabilitatea de a ghici 10 carti consecutive este  $1/10^{10}$  deoarece probabilitatea de a ghici o carte este  $1/2$  (rosu sau negru)!

27 

## Principiul lui Bonferroni



- Astfel de rezultate pot fi regăsite în lecția unui algoritm de data mining și trebuie recunoscute ca adevaruri statistică și nu ca rezultate reale ale procesului de data mining.
- Astfel de rezultate sunt obiectul principiului lui Bonferroni. Aceasta poate fi sintetizat astfel:

Daca sunt prea multe concluzii, unele vor fi adevărate din motive pur statistică.

Data Mining, note: "Tuning data until it converges... and if you torture it enough, it will confess to anything" - Jeff Jones, 1994 (<http://www.cs.cmu.edu/~mason/courses/Deployment.html>)

28 

## Exemplu



- Să considerăm că articolele sunt pacienti ai unui serviciu de urgențe medicale.
- Există 5 clase,  $C_1, C_{10}, C_{20}, \dots, C_{100}$ , unde eticheta  $C_i$  înseamnă că pacientul poate fi lăsat să aștepte maxim  $k$  minute fară că starea sa se înrăutătească.
- Vom reprezenta datele setului de antrenare sub forma tabelară.
- Iesirea unui algoritm de creare a unui clasificator poate fi de exemplu un arbore de decizie sau un set ordonat de reguli.
- Modelul obținut poate fi utilizat apoi pentru a clasifica noi pacienți asignându-le o etichetă din multimea celor 5 de mai sus.

33 

## Multimea de antrenare UPU



| Name (or ID) | Vital risk? | Danger if water? | 0 resource needed | 1 resource needed | >1 resource needed | >1 resource needed and vital resources affected | Waiting time before new label? |
|--------------|-------------|------------------|-------------------|-------------------|--------------------|-------------------------------------------------|--------------------------------|
| John         | No          | No               | Yes               | No                | No                 | No                                              | 00                             |
| Maria        | No          | No               | No                | No                | Yes                | No                                              | 010                            |
| Natalia      | Yes         | Yes              | Yes               | No                | No                 | No                                              | 010                            |
| Oliver       | No          | No               | No                | No                | Yes                | Yes                                             | 030                            |
| Kiril        | No          | No               | No                | Yes               | No                 | Yes                                             | 060                            |
| Denis        | No          | No               | No                | No                | Yes                | No                                              | 010                            |
| Jean         | No          | No               | Yes               | Yes               | No                 | No                                              | 010                            |
| Patricia     | Yes         | Yes              | No                | No                | Yes                | Yes                                             | 030                            |

34 

## Cuprins

- Ce este data mining
- Etapele procesului de data mining
- Metode si subdomenii in data mining
- Preprocesarea datelor
- Sumar

29 

## 2 tipuri de metode



- Metode predictive:** Aceste metode utilizează anumite variabile pentru a prezice valoarea altă variabilă. Un exemplu din acesta categoria este clasificarea; bazat pe date cunoscute și de etichetă (clasificate), algoritmii de clasificare crează modele care pot fi folosite pentru predicție.
- Metode descriptive:** Algoritmii din această categorie găsești sabioane / modele care descriu structura internă a unui set de date. De exemplu algoritmii de grupare (clustering) găsești grupuri de obiecte similară în setul de date (grupuri numite cluster) și de asemenea detectează posibile obiecte izolate, departate de oricare dintre cluster - acea numită "outliers".

30 

## Regresia (2)

Există multe tipuri de regresie, de exemplu:

- Regresie liniara
- Regresie liniara simplă
- Regresie logistică
- Regresie neliniera
- Regresie nonparametrică
- Regresie robustă

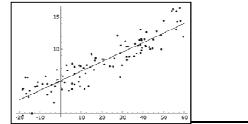
37 

## Exemplu



### Exemplu de regresie liniara

(sursa: [http://en.wikipedia.org/wiki/File:Linear\\_regression.svg](http://en.wikipedia.org/wiki/File:Linear_regression.svg))

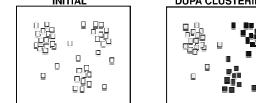


38 

## Exemplu



- Dacă avem un set de 2 puncte într-un spațiu 2D, să se găsească grupurile/clusterurile naturale formate de ele



43 

## Regresia (1)



- Regresia provine din statistică.
- Însemnată: precizarea (predicția) valorii unei anumite variabile continuu pe baza valorilor altor variabile, considerând un model de dependență liniară sau neliniară (Tian, Steinbach, Kumar 06).
- Utilizată în predicție sau în proiecție;
- Analizează bazate pe regresie sunt folosite pentru înțelegerea relațiilor dintre variabile dependente și independente.
- Interpretabilitatea: care e deosebită între predicție și proiecție?

36 

## Detectia deviației

- Detectia deviației (sau a anomalilor) presupune descompunerea de detaliu descriptiv de la componentă normală. Putem să identificăm ceva care este semnificativ de astfel de date normale.
- Detectia deviației poate fi utilizată în multe situații:
  - În fază de rulare a algoritmilor de data mining, datele anormale pot indica că există un atac.
  - Audit: astfel de informații pot arăta existența unor probleme sau practici gresite.
  - Detectia fraudei: cercetă frauduloase conturi de utilizatori.
  - Detectia intruziunilor: într-o rețea de calculatoare poate fi facuta să detecteze datele anormale.
  - Corectarea datei (data cleaning): astfel de informații anormale pot reprezenta datele oronice cără trebuie corectate

39 

## Metode de detectie a deviației



- Tehnici bazate pe distanțe (ex.: k-nearest neighbor).
- Folosirea "One Class Support Vector Machines" (Ca un set de date să dezbată exemplul de antrenare din aceeași clasă și să determine că este de-a doua clasă).
- Metode predictive (arbori de decizie, rețele neurale).
- Detectia punctelor izolate folosind clustering.
- Înregistrari care nu respectă regulile de asociere într-o rețea.
- Analize Hotspot (clusterizare având o valoare ridicată a anumitor parametri; de exemplu poliția folosește astfel de analize pentru analiza zonelor unde criminalitatea are valori ridicate)

40 

## Reguli de asociere



- O regula** este o construcție de tipul  $X \rightarrow Y$  unde  $X$  și  $Y$  sunt multimi de articole (itemsets).
- Suportul** unei regule  $X \rightarrow Y$  este numarul de aparitii ale  $X \cup Y$  în  $T$  (egal cu suportul acestei reunii ca itemset):  $\text{support}(X \rightarrow Y) = \text{support}(X \cup Y)$
- Confidența** unei regule este proporția transacțiilor continându-l pe  $Y$  în multimea transacțiilor continându-l pe  $X$ :  $\text{confidence}(X \rightarrow Y) = \text{support}(X \cup Y) / \text{support}(X)$
- Acceptăm o regula ca validă doar dacă suportul și confidența ei sunt mai mari sau egale cu pragurile date.

40 

## Exemplu



- Sa considerăm următorul set de tranzacții:

| Transacție ID | Items                                                        |
|---------------|--------------------------------------------------------------|
| 1             | Bread, Milk, Butter, Orange Juice, Onion, Beer               |
| 2             | Bread, Milk, Butter, Onion, Garlic, Beer, Orange Juice, Soda |
| 3             | Milk, Butter, Onion, Garlic, Beer                            |
| 4             | Orange Juice, Soda, Sheep, Bread, Milk                       |
| 5             | Bread, Onion, Beer, Orange Juice                             |
- Dacă  $s = 60\%$  atunci (Bread, Milk, Orange Juice) sau (Onion, Garlic, Beer) sunt multimi frecvente de articole.
- De asemenea, dacă  $s = 60\%, c = 70\%$  atunci regula (Onion, Beer) → (Garlic) și validă având suport de 60% și incidență de 75%.

46 

## Algoritmi

- Algoritmi de predicție:**
  - Clasificare
  - Regresie
  - Detectia deviației
- Algoritmi de descriere:**
  - Grupare - Clustering
  - Gasirea de reguli de asociere
  - Descoperirea de sabloane secențiale

41 

## Clustering



- Intrare:**
  - Un set de obiecte  $D = \{d_1, d_2, \dots, d_n\}$  (numite uzual puncte). Obiectele nu sunt etichetate și nu există definitiv reuniunea de clase.
  - Metoda de distanță (măsură de similaritate) care poate fi utilizată pentru a calcula distanța dintre oricare două puncte. O distanță mica înseamnă "aproape" iar una mare "departe".
  - Un algoritm necesită introducerea unei valori pentru numărul de clusteruri de către utilizator.
- Test:**
  - Un set de grupuri de obiecte/puncte, numite clustere unde punctele din același cluster sunt "aproape" împreună, iar cele din clustere diferențiate "departe" unele de altele, considerând funcția de distanță existentă.

42 

## Descoperire sabloane secențiale



- Intrare:**
  - O mulțime de m articole  $I = \{i_1, i_2, \dots, i_n\}$ . O seconță este o listă ordonată de elemente din articolele  $I$ .
  - Mulțimea de seconde este  $S_m$  (numită "sequence database").
  - O funcție booleană care poate testa dacă o seconță  $S_m$  este inclusă (este o subsecvență) în secvența  $S_n$ . În acest caz  $S_n$  este numita o supersetă a lui  $S_m$ .
  - Un prag (percent sau valoare absolută) necesar pentru a gasi secondele frecvente.
- Iesire:**
  - Multimea de seconde frecvente, i.e. multimea de seconde incluse în cel puțin s seconțe din  $S_n$ .
  - Uneori se poate obține și un set de reguli, fiecare regula fiind de forma  $S_n \rightarrow S_1$  unde  $S_1$  și  $S_n$  sunt seconde.

47 

## Exemplu



- Într-o librărie putem găsi seconde de tipul:
  - `Book_on_C, Book_on_C++, Book_on_Perl`
- Din această seconță se poate obține și o regulă de tipul:
  - după cumpărarea unor cărți de C și C++, clientul cumpără cărți de Perl:
    - `Book_on_C, Book_on_C++ → Book_on_Perl`

48 

## Cuprins

- Ce este data mining
- Etapele procesului de data mining
- Metode si subdomenii in data mining
- Preprocesarea datelor
- Sumar

49

Planul Răspunsului la Vizualizarea UBD-7

## Preprocesarea datelor

- Tipuri de date
- Masurarea datelor
- Curatarea datelor
- Integrarea datelor
- Transformarea datelor
- Reducerea datelor
- Discretizarea datelor

50

Planul Răspunsului la Vizualizarea UBD-7

## Nominala

- Valorile aparținând scălei nominale sunt caracterizate prin etichete (nume).
- Valorile sunt neordonate și au importanță (pondere) egală.
- Nu putem calcula valoarea medie sau mediana a unui astfel de set.
- Putem totuși calcula valoarea modală – valoarea cea mai frecventă.
- Datele nominale sunt de tip categoric și uneori este nevoie să fie convertite în valori numerice.

55

Planul Răspunsului la Vizualizarea UBD-7

## Ordinala

- Valorile de acest tip sunt ordonate dar diferența / distanța între două valori nu poate fi calculată.
- Putem specifica doar ordinea / poziția în sirul ordonat pentru aceste valori.
- Exemplu: lista gradelor militare, lista gradelor didactice, lista în ordinea sosirii a concurenților la o cursă sportivă (doar lista numelor, fără timpul de sosire), etc.
- Pentru astfel de valori putem calcula valoarea modală sau mediana (valoarea mijlocie) dar nu media.
- Valoare sunt în esență categorice dar este mai ușor sa le asignăm valori numerice în caz de conversie.

56

Planul Răspunsului la Vizualizarea UBD-7

## Tipuri de date

- Categorice vs. Numerice
- Scale de măsurare
  - Nominală
  - Ordinală
  - Interval
  - Proportională (eng.: ratio scale)

51

Planul Răspunsului la Vizualizarea UBD-7

## Date categorice și numerice

- Datele categorice constau în numere reprezentând categorii. Exemple: culoare (cu categoriile roșu, verde, albastru și alături) sau gen (masculin, feminin)
- Valorile de acest tip nu sunt de obicei ordonate (nu există o relație de ordine între ele), operațiile posibile fiind testul de egalitate sau de apartenență (inclusiune) la o mulțime.

52

Planul Răspunsului la Vizualizarea UBD-7

## Interval

- Aceste valori sunt numerice.
- În acest caz diferența între două valori are semnificație.
- Exemplu: temperatură în grade Celsius: diferența între 10 și 20 de grade este aceeași ca cea între 40 și 50 de grade (aceeași cantitate de energie în cazul încălzirii unui corp).
- Valoarea zero (0) nu înseamnă "nimic" ci este o valoare arbitrară aleasă. De aceea sunt permise și valori negative.
- Putem calcula media, mediana, valoarea modală, deviația standard și putem utiliza regresia pentru a prognoza noi valori.

57

Planul Răspunsului la Vizualizarea UBD-7

## Proportională (ratio)

- Aceste valori sunt numerice, ca și în cazul interval, dar zero (0) înseamnă "nimic".
- Valoare negativă nu are sens.
- Raportul dintre două valori are semnificație.
- Exemplu: greutatea în kg. Un copil de 10 kg este de două ori mai greu decât unul de 5 kg.
- Alte exemple: temperatură în grade Kelvin, lungimea în metri, varsta în ani, etc.
- Toate operațiile matematice sunt permise.

58

Planul Răspunsului la Vizualizarea UBD-7

## Date categorice și numerice

- Datele numerice constau în numere apartinând unei mulțimi continue sau discrete de valori numerice.
- Valoare sunt ordonate, astfel încât se poate testa ordinea ( $<$ ,  $\leq$ ,  $>$ ,  $\geq$ ).
- Uneori este necesara conversia datelor categorice în date numerice assignând-o valoare numerică (sau cod) pentru fiecare etichetă (categorie).

53

Planul Răspunsului la Vizualizarea UBD-7

## Scale de măsurare

- Stanley Smith Stevens, directorul laboratorului de psihoo-acustică de la Universitatea Harvard a afirmat într-un articol publicat în 1946 în revista Science că toate măsurările din stîntă utilizează patru tipuri diferite de scale de măsurare:
  - Nominală
  - Ordinală
  - Interval
  - Proportională (ratio scale)

54

Planul Răspunsului la Vizualizarea UBD-7

## Date binare

- Uneori un atribut poate avea doar 2 valori: 0 sau 1, masculin sau feminin, da sau nu, etc. În acest caz datele se numesc **binare**. Avem două cazuri:
  1. Binare simetrice: cele două valori au importanță/pondere egală (ca în cazul genului).
  2. Binare asymetrice: una dintre valori are o importanță mai mare decât cealaltă. De exemplu în cazul unei analize pentru depistarea HIV valoarea Pozițiv are o greutate mai mare decât Negativ.

59

Planul Răspunsului la Vizualizarea UBD-7

## Date binare

- Atributele binare pot fi tratate uneori ca fiind de tip interval sau proporțional dar în cea mai mare parte a cazurilor ele vor fi considerate nominale (cele binare simetrice) sau ordinație (cele binare asymetrice).
- Există o serie de funcții de distanță (dissimilaritate) care pot fi utilizate în acest caz.

60

Planul Răspunsului la Vizualizarea UBD-7

## Preprocesarea datelor

- Tipuri de date
- Masurarea datelor**
- Curatarea datelor
- Integrarea datelor
- Transformarea datelor
- Reducerea datelor
- Discretizarea datelor

61

Planul Răspunsului la Vizualizarea UBD-7

## Masurarea datelor

- Masurarea tendinței valorii centrale:**
  - Media
  - Mediana
  - Valoare modală
  - Mijlocul intervalului
- Masurarea dispersiei:**
  - Interval de valori
  - Procentul k
  - IQR
  - Sumarul de 5 numere
  - Deviația standard și varianta

62

Planul Răspunsului la Vizualizarea UBD-7

## Dispersia valorilor (1)

- Intervalul de valori.** Este diferența între cea mai mare și cea mai mică valoare.
- Exemplu: Ior {1, 2, 3, 5, 7, 1001, 2002, 9999} valoarea este 9999 - 1 = 9998.
- Procentul k.** Aceasta este o valoare x, apărând setul de date și care are proprietatea că un procent de k valori din set sunt mai mici sau egale cu ea.
- Exemplu: Mediana este procentul 50.
- Cele mai utilizate procente sunt 25 și 75, numite și **cuartile** (eng. Quartiles). Notație: Q1 pentru 25% și Q3 pentru 75%.

67

Planul Răspunsului la Vizualizarea UBD-7

## Dispersia valorilor (2)

- Intervalul între quartile** (eng. Interquartile range sau IQR) este diferența între Q3 și Q1:
$$IQR = Q3 - Q1$$
- Potențiale puncte izolate (outliers) sunt valori la distanță de cel puțin 1,5 x IQR sub Q1 sau peste Q3.
- Sumarul de 5 numere.** Uneori mediana și cuartile nu sunt suficiente pentru a reprezenta dispersia valorilor. În acest caz adăugăm și valoarea minimă și maximă din setul de date.
- (Min, Q1, Median, Q3, Max) este ceea ce se numește sumarul de 5 numere (eng. five-number summary).

68

Planul Răspunsului la Vizualizarea UBD-7

## Tendinta centrala - Medie

- Fie n valori ale unui atribut:  $x_1, x_2, \dots, x_n$ .
- Media:** Media aritmetică sau valoarea medie este:
$$\mu = (x_1 + x_2 + \dots + x_n) / n$$
- Dacă valoare lui x are ponderile  $w_1, w_2, \dots, w_n$ , atunci **media aritmetică ponderată** este:
$$\mu = (w_1 x_1 + w_2 x_2 + \dots + w_n x_n) / (w_1 + w_2 + \dots + w_n)$$
- În unele cazuri se practică eliminarea celor mai mici și celor mai mari valori (de exemplu cele mai mici 1% și cele mai mari 1%).

63

Planul Răspunsului la Vizualizarea UBD-7

## Tendinta centrala - Mediana

- Mediana:** Valoarea mediană a unui set de n valori ordonate este valoarea din mijlocul sevenței de valori.
- Exemplu: Mediana pentru {1, 3, 5, 7, 1001, 2002, 9999} este 7.
- Dacă n este par (și datele sunt numerice), mediana este media valorilor din mijlocul sevenței:
  - mediana pentru {1, 3, 5, 7, 1001, 2002} este 6 (media valorilor 5 și 7).

64

Planul Răspunsului la Vizualizarea UBD-7

## Dispersia valorilor (3)

- Exemplu:**  
Pentru {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11} Interval de valori = 10; Mijloc interval = 6; Q1 = 3; Q2 = 6; Q3 = 9; IQR = 9 - 3 = 6  
Pentru {1, 2, 3, 4, 5, 6, 7, 8, 8} Interval de valori = 7; Mijloc interval = 5,5; Q1 = 3; Q2 = 5,5 (= 5+6)/2; Q3 = 7; IQR = 7 - 3 = 4  
Pentru {1, 3, 5, 7, 8, 10, 11, 13} Interval de valori = 12; Mijloc interval = 7; Q1 = 4; Q2 = 7,5; Q3 = 10,5; IQR = 10,5 - 4 = 6,6

69

Planul Răspunsului la Vizualizarea UBD-7

## Dispersia valorilor (4)

- Deviația standard** (abaterea standard). Pentru n valori (observați) aceasta este:
$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}, \text{ unde } \bar{x} = \frac{1}{N} \sum_{i=1}^N x_i.$$
- Patratul deviației standard se numește **varianță** (dispersie).
- Deviația standard măsoară raspandirea valorilor în jurul mediului.
- Valoarea 0 este obținuta doar dacă toate valorile sunt egale.

70

Planul Răspunsului la Vizualizarea UBD-7

## Tendinta centrala – Valoare modală

- Valoare modală:** Cea mai frecventă valoare din multimea respectivă.
- Un set de date poate avea mai multe valori mode. Pentru 1, 2 și 3 seturi este denumit ca unimodal, bimodal sau trimodal.
- Cand fiecare valoare apare o singura data setul nu are valoare modală.
- Pentru un set unimodal, valoarea modală măsoară tendința centrală a acestuia. În acest caz există relația empirică:
$$\text{medie} - vmod = 3 \times (\text{medie} - \text{mediana})$$

65

Planul Răspunsului la Vizualizarea UBD-7

## Tendinta centrala – Mijlocul intervalului

- Mijlocul intervalului:** este media aritmetică a celei mai mari și celei mai mici valori.
- Exemplu: pentru {1, 3, 5, 7, 1001, 2002, 9999} valoarea de mijloc a intervalului este 5000 (media lui 1 și 9999).

66

Planul Răspunsului la Vizualizarea UBD-7

## Preprocesarea datelor

- Tipuri de date
- Masurarea datelor
- Curatarea datelor
- Integrarea datelor
- Transformarea datelor
- Reducerea datelor
- Discretizarea datelor

71

Planul Răspunsului la Vizualizarea UBD-7

## Curatarea datelor

- Obiectivele principale ale acestei etape sunt:
  - Înlăturarea (sau eliminarea) valorilor lipsă
  - Înlăturarea zgomotului
  - Eliminarea sau doar identificarea punctelor izolate (eng.: outliers - valori aberante)

72

Planul Răspunsului la Vizualizarea UBD-7

## Valori lipsa

- Anumite atribute pot contine valori lipsa / valori nule (NULL).
- Pot sa apară din diverse motive:
  - Probleme hardware, software sau erori ale operatorului de introducere date.
  - Date care nu au fost colectate într-o anumita perioadă (nu au fost considerate importante).
  - Valori eliminate pentru ca erau inconsistente cu restul.
- Există doar soluții în acest caz:
  - Ignorarea linierelor din tabela de date de intrare care contin valori lipsă. Așa se poate face doar în cazul în care numărul lor este mic și eliminarea nu induce erori în calculele ulterioare.

Foto: Wikipedia/Nume de curs UBD-7

## Valori lipsa

- Pot apărea din diverse motive:
  - Probleme hardware, software sau erori ale operatorului de introducere date.
  - Date care nu au fost colectate într-o anumita perioadă (nu au fost considerate importante).
  - Valori eliminate pentru ca erau inconsistente cu restul.
- Există două soluții în acest caz:
  - Ignorarea linierelor din tabela de date de intrare care contin valori lipsă. Așa se poate face doar în cazul în care numărul lor este mic și eliminarea nu induce erori în calculele ulterioare.

Foto: Wikipedia/Nume de curs UBD-7

## Puncte izolate

- Punctele izolate / valori abnormale (outliers) sunt valori numerice de atribut aflate la o distanță mare de restul datelor.
- Uneori sunt valori corecte: salariajul unui CEO poate fi mult mai mare decât al restului angajaților.
- De cele mai multe ori însă sunt date eronate / zgomot.
- Trebuie identificate și eliminate sau înlocuite, deoarece multi algoritmi sunt sensibili la astfel de puncte – de exemplu k-means da clustere eronate în prezentă lor.

Foto: Wikipedia/Nume de curs UBD-7

## Identificare puncte izolate

- Folosind IQR: am mai spus că potențiale puncte izolate (outliers) sunt valori aflate la o distanță de cel puțin 1.5 \* IQR sub Q1 sau peste Q3.
- Folosind deviația standard: valori care sunt la o distanță mai mare de 2 de valoarea medie sunt potențiale valori izolate.
- Folosind clustering: după gasirea clusterelor, punctele aflate în afara oricărui cluster (sau foarte departe de orice centru de cluster) sunt potențiale puncte izolate.

Foto: Wikipedia/Nume de curs UBD-7

## Netezire zgromot

- Zgromot poate fi definit ca o eroare aleatoare sau variată pentru o valoare măsurată ([Han, Kamber 06]).
- Pentru eliminarea zgromotului se pot folosi diverse tehnici de netezire ca:
  - Regresia (prezentată mai devreme)
  - Binning = Partitionarea în trame

Foto: Wikipedia/Nume de curs UBD-7

## Binning

- Aceasta tehnica poate fi folosita pentru netezirea unui set ordonat de valori. Este bazata pe valoare din vecinătate.
- Sunt 2 pasi:
  - Partitionarea setului ordonat în mai multe trame
  - Netezirea fiecarei trame de date pe baza mediei, medianei sau capetelor intervalului.

Foto: Wikipedia/Nume de curs UBD-7

## Preprocesarea datelor

- Tipuri de date
- Măsurarea datelor
- Curătarea datelor
- Integrarea datelor
- Transformarea datelor
- Reducerea datelor
- Discretizarea datelor

Foto: Wikipedia/Nume de curs UBD-7

## Integrarea datelor

- După ce datele provenind din sursele de date individuale au fost curătate ca mai înainte, are loc integrarea acestora într-un unic ansamblu de date.
- Activități din această categorie includ:
  - 1. Integrarea schemelor.** Problemele sunt de tip sinonime (același lucru apare în două surse cu nume diferite) și omónime (cu același nume se găsesc date de tip diferit în diverse surse).
  - 2. Duplicate și redundanță.** În surse diferite se regăsesc același informații. Duplicatele și redundanța trebuie eliminate în această fază.

Foto: Wikipedia/Nume de curs UBD-7

## Exemplu

- Fie urmatorul set ordonat de valori: 1, 2, 4, 6, 9, 12, 16, 17, 18, 23, 34, 56, 78, 79, 81.

| Tramele inițiale   | Se utilizează media | Se utilizează mediana | Se utilizează capetele intervalului |
|--------------------|---------------------|-----------------------|-------------------------------------|
| 1, 2, 4, 6, 9      | 4, 4, 4, 4, 4       | 4, 4, 4, 4, 4         | 1, 1, 9, 9                          |
| 12, 16, 17, 18, 23 | 17, 17, 17, 17, 17  | 17, 17, 17, 17, 17    | 12, 12, 12, 23, 23                  |
| 34, 56, 78, 79, 81 | 66, 66, 66, 66, 66  | 78, 78, 78, 78        | 34, 34, 81, 81, 81                  |

Foto: Wikipedia/Nume de curs UBD-7

## Rezultat

- Rezultatul netezirii este:
  - Initial: 1, 2, 4, 6, 9, 12, 16, 17, 18, 23, 34, 56, 78, 79, 81
  - Cu medie: 4, 4, 4, 4, 17, 17, 17, 17, 17, 66, 66, 66, 66
  - Cu mediana: 4, 4, 4, 4, 17, 17, 17, 17, 78, 78, 78, 78
  - Cu capetele intervalului: 1, 1, 1, 9, 9, 12, 12, 12, 23, 23, 34, 34, 81, 81, 81

Foto: Wikipedia/Nume de curs UBD-7

## Integrarea datelor

- Inconsistențe.
  - Acum sunt valori aflate în conflict în același set de date.
  - Din exemplu Data nasterii=1.1.1980 și Varsta=30 pentru o același persoană reprezintă o inconsistenta.
  - Pentru detectarea acestora sunt necesare informații suplimentare despre date.

Foto: Wikipedia/Nume de curs UBD-7

## Transformarea datelor

- Aceasta etapa cuprinde procese de transformare și sumarizare a datelor ca:
  - Normalizare
  - Construcție de noi atribute
  - Sumarizare folosind funcțiile statistice

Foto: Wikipedia/Nume de curs UBD-7

## Normalizare

- Toate atributele numerice sunt scalate pentru a avea valori între-un același interval specificat:
  - De la 0 la 1,
  - De la -1 la 1 sau, mai general:  $|v| \leq r$  unde  $r$  este o valoare specificată.
- Normalizarea este necesara cand anumite atribute sunt mai importante decât altale doar pentru că plaja lor de valori este mai întinsă.
- Exemplu: Distanța euclidiană între A(0,5, 10) și B(0,01, 2111) este  $\approx 2010$ , determinată aproape exclusiv pe baza celei de-a doua componente.

Foto: Wikipedia/Nume de curs UBD-7

## Reducerea datelor

- Nu toate informații colectate sunt necesare pentru anumite analize asupra datelor.
- Reducerea înseamnă extragerea unei parti din datele colectate și preprocesate, să anume doar a aceliei parti care conțin informații necesare procesului de analiză respectiv.
- Se realizează de exemplu prin selectarea doar a anumitor atribute (coloane) sau exemple (linii) din matricea de date supusă analizei.

Foto: Wikipedia/Nume de curs UBD-7

## Discretizare

- Nu există algoritmi sănătoși concepuți pentru analiza datelor discrete.
- Pentru a ii folosi, este necesar ca valorile continue să fie înlocuite cu unele discrete.
- Din exemplu Varsta care are valori numerice va fi înlocuită cu un atribut categoric având doar valorile Copil, Adult, Batran.
- Există diverse metode de a efectua operația de discretizare, folosind împărțirea în trame (binning), histograme, intervale bazate pe entropie, clustering, ierarhii de concepte, etc.

Foto: Wikipedia/Nume de curs UBD-7

## Normalizare

- Putem folosi pentru normalizare:
  - Normalizarea min-max:**  $v_{new} = (v - v_{min}) / (v_{max} - v_{min})$
  - Pentru valori pozitive** putem folosi formula:  $v_{new} = v / v_{max}$
  - Normalizarea de tip z-score (o deviație standard):**  $v_{new} = (v - v_{mean}) / \sigma$
  - Scalarea decimală:**  $v_{new} = v \cdot 10^n$
- Unde  $n$  e cel mai mic întreg pentru care toate numerele devin în modul mai mic decât o valoare dată ( $v_{new} \leq 1$ ).

Foto: Wikipedia/Nume de curs UBD-7

## Constructie noi atribute

- Este o tehnică prin care se adaugă noi atribute (în engleză se numește **feature construction**).
- Exemplu: dacă setul de date are un atribut Culoare cu valori (Rosu, Verde, Albastru) putem construi 3 noi atribute numite Rosu, Verde, Albastru conținând doar valori binare (0 sau 1) cu 1 singur pe atributul (culoarea) corespunzător colorii curente.
- Alt exemplu: se poate utiliza un arbore de decizie sau set de reguli pentru a construi noi atribute din cele existente – ca de exemplu clasa.

Foto: Wikipedia/Nume de curs UBD-7

## Sumar

- În acest curs am prezentat:
  - O listă de definiții alternative pentru Data Mining și cetera exemple pentru a înțelege ce este Data Mining și cum se face Data Mining.
  - O descriere a principalelor implicații ale Data Mining și despre faptul că Data Mining este de fapt o reunire heterogenă de subdomenii.
  - Pasi principali ai Data Mining process de colectarea de date și a locației diferențiale (deosebit de date, arhive sau sisteme operaționale), para la pasul final de evaluare.
  - O scurtă descriere a subdomeniilor principale ale Data Mining cu exemple pentru fiecare dintre ele.
  - O descriere cuprinzătoare a procesului de preprocesare a datelor.
- Lectia viitoare: Multimi frecvente de articole și reguli de asociere

Foto: Wikipedia/Nume de curs UBD-7

## Bibliografie și referințe

- [Liu 11] Bing Liu, 2011. Web Data Mining, Exploring Hyperlinks, Content, and Queries in Web Databases, Second Edition, Springer.
- [Tan, Steinbach, Kumar 06] Tan, Steinbach, Kumar, 2006. Introduction to Data Mining, Addison-Wesley, 1-18.
- [Kimbala, Ross 02] Ralph Kimball, Margy Ross, 2002. The Data Warehouse Toolkit: Practical Techniques for Building Data Warehouses and Data Marts, John Wiley and Sons, 1-16, 396.
- [Miers, Fischetti 11] Reid Miers and Francesco Fischetti, Data mining tools, 2011, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, Volume 1, Issue 5, DOI: 10.1002/widm.24/pdf
- [Ullman] Jeffrey Ullman, Data Mining Lecture Notes, 2003-2009, web page: <http://infolab.stanford.edu/~ullman/mining/>
- What is the difference between data mining, machine learning and AI? <http://stats.stackexchange.com/questions/5026/what-is-the-difference-between-data-mining-statistics-machine-learning-and-ai>

Foto: Wikipedia/Nume de curs UBD-7

## Sumarizare

- Se folosesc funcții statistică (min, max, sum, avg, count, etc) pentru a adăuga date sumare datelor originale.
- Exemplu: se adaugă suma varianțelor pe zi sau luna sau an, numărul mediu sau total de clienți sau înfrântăci, etc.
- Aceste date vor fi folosite pentru creșterea vitezei de procesare a unor cereri în procesul de data mining.
- Rezultatul este un cub de date și fiecare data sumară este atașată unui nivel de granularitate.

Foto: Wikipedia/Nume de curs UBD-7

## Preprocesarea datelor

- Tipuri de date
- Măsurarea datelor
- Curătarea datelor
- Integrarea datelor
- Transformarea datelor
- Reducerea datelor
- Discretizarea datelor

Foto: Wikipedia/Nume de curs UBD-7

## Bibliografie și referințe

- [Akash Mitra, Data Mining - a simple guide for beginners, 2012, <http://www.akashmitra.com/2012/11/data-mining-11-data-mining-definitions.html>
- [Wikipedia] Wikipedia, the free encyclopedia, en.wikipedia.org
- [Han, Kamber 06] Jiawei Han, Micheline Kamber, Data Mining: Concepts and Techniques, Second Edition, Morgan Kaufmann Publishers Inc., 2006.
- [Stevens June 1946] Stevens, S.S. On the Theory of Scales of Measurement. Science June 1946, 103 (2884): 677–680.
- [Wikipedia] Wikipedia, the free encyclopedia, en.wikipedia.org
- [Liu 11] Bing Liu, 2011. CS 593 Data mining and text mining course notes, <http://www.cs.uic.edu/~iub/teach/cs593-fall-11/cs593.html>

Foto: Wikipedia/Nume de curs UBD-7

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>Capitolul 8</b></p> <p><b>Data mining:</b><br/><b>Reguli de asociere si multimi frecvente de articole (frequent itemsets)</b></p> <p>F. Radulescu, Curs Utilizarea bazelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                  | <p><b>Problema</b></p> <ul style="list-style-type: none"> <li>□ Problema cosului de produse presupune ca avem un mare numar de articole, e.g. "paine", "lalte".</li> <li>□ Cumparatorii pun in cosul lor de produse anumite submultimi de articole iar noi vom afla ce articole sunt cumporate impreuna chiar daca nu si de cine.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazelor de date, anul IV CS.</p>                                                                                                                                                       | <p><b>Reguli de asociere</b></p> <ul style="list-style-type: none"> <li>□ <b>Regulele de asociere:</b> Daca X si Y sunt multimi de articole, o regula este o propozitie de forma <math>X \rightarrow Y</math> inseamnand ca daca gasim toate articolele din X intr-un cos atunci sunt mari sanse sa gasim in acel cos si articolele din Y.</li> <li>□ Probabilitatea de a-l gazi pe Y pentru a accepta aceasta regula este numita <i>incredere</i> acelei reguli.</li> <li>□ In mod normal vom cauta doar reguli care au o incredere peste un anumit prag.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                                                                                                                  | <p><b>Reguli de asociere</b></p> <ul style="list-style-type: none"> <li>□ Putem de asemenea cere ca increderea sa fie semnificativ mai mare decat in cazul in care articolele ar fi plaseate aleator in cos.</li> <li>□ De exemplu putem gasi o regula ca {lalte, urb} <math>\rightarrow</math> paine doar pentru ca foarte multa lume cumpara paine.</li> <li>□ Totusi exemplul bere/scutice arata ca regula {scutice} <math>\rightarrow</math> {bere} este verificata cu o incredere semnificativ mai mare decat a submultimii de cosuri continand bere.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>     |
| <p><b>Problema</b></p> <ul style="list-style-type: none"> <li>□ Vanzatorii utilizeaza aceste informatii pentru asezarea articolelor in magazin, in felul acesta putand sa controleze modul in care anumite clase de cumparatori parcurg raioanele magazinului.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                               | <p><b>Alte utilizari</b></p> <ul style="list-style-type: none"> <li>□ Cos = documente; articole = cuvinte.</li> <li>□ Cuvintele aparand frecvent impreuna in documente pot reprezenta fraze sau concepte legate intre ele.</li> <li>□ Poate fi utilizat pentru colectarea de informatii (intelligence).</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                | <p><b>Scutice si bere</b></p> <ul style="list-style-type: none"> <li>□ În 1992, Thomas Elschnig, managerul unui grup de consultanță pentru vânzări cu ambițiile de la Teradac, și personalul său au proiectat și analizat 1,2 milioane de cosuri de produse de la aproximativ 75 de magazine Oscar Drap.</li> <li>□ Au fost răsuza interogări alezi de date pentru a identifica afinitățile.</li> <li>□ Analiza a descoperit că între orele 17:00 și 19:00 consumatorii cumpără bere și scutice.</li> <li>□ Managerii Oscar NU au exploatat asocierea dintre bere și scutice multănd produsele mai aproape unei de celelalte pe rafturi.</li> <li>□ Acest studiu pentru suportul decizional a fost realizat folosind instrumente de interrogare pentru a găsi asocieri.</li> <li>□ Povestea adevarată este foarte faidă în comparare cu legenda.</li> </ul> <p>Sursa: <a href="http://www.dssresources.com/newsletters/66.php">http://www.dssresources.com/newsletters/66.php</a></p> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p> | <p><b>Cauzalitate</b></p> <ul style="list-style-type: none"> <li>□ <b>Cauzalitate, Ideal:</b> am vrea sa stim daca intr-o regula de asociere prezenta elementele X efectiv "cauzeaza" (determina) cumpararea lui Y.</li> <li>□ <b>"Cauzalitatea"</b> este insa un concept echivoc.</li> <li>□ Exemplu:</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                         |
| <p><b>Alte utilizari</b></p> <ul style="list-style-type: none"> <li>□ Cos = propozitii, articole = documente.</li> <li>□ Doua documente continand aceleasi propozitii pot reprezenta un plagiat sau "mirror sites" pe web.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                   | <p><b>Obiective pentru aceasta clasa de probleme</b></p> <ul style="list-style-type: none"> <li>□ Gasirea de: <ul style="list-style-type: none"> <li>□ Reguli de asociere</li> <li>□ Cauzalitate</li> <li>□ Multimi frecvente de articole</li> </ul> </li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                  | <p><b>Cauzalitate</b></p> <ul style="list-style-type: none"> <li>□ Daca scadem pretul scutecelor si crestem pretul berii putem ademeneii cumparatorii de scutice care au inabilitatea de a cumpara bere din magazin, acoperind astfel pierderile din vanzarea scutecelor.</li> <li>□ Aceasta strategie este valabila deoarece "scutice determina bere".</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <p><b>Cauzalitate</b></p> <ul style="list-style-type: none"> <li>□ Actiunea inversa, micsorarea pretului la bere si marirea pretului scutecelor nu va determina cumparatorii de bere sa cumpere scutice in numar mare si vom pierde bani deoarece regula "bere determina scutice" nu este adevarata.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                           |
| <p><b>Multimi frecvente</b></p> <ul style="list-style-type: none"> <li>□ <b>Multimi frecvente de articole (frequent itemsets)</b>; in multe situatii (dar nu in toate) ne intereseaza doar regulele de asociere si cauzalitatea in ceea ce priveste multimi de articole care apar <i>frecvent</i> in cosuri.</li> <li>□ De exemplu, nu putem conduce o buna strategie de marketing care implica produse pe care oricum nu le cumpara nimeni.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p> | <p><b>Multimi frecvente</b></p> <ul style="list-style-type: none"> <li>□ Astfel, cautarile in date portesc de la premiza ca ne intereseaza doar multimi de articole cu un larg suport;</li> <li>□ Larg suport = ele apar impreuna in multe cosuri de produse.</li> <li>□ Gasim apoi regule de asociere sau cauzalitatea implicand doar articolele cu larg suport (i.e., <math>\{X, Y\}</math> trebuie sa apară in cel putin un anumit procent din cosuri, numit <i>prag de suport</i>)</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p> | <p><b>Abordari</b></p> <ul style="list-style-type: none"> <li>□ Pentru a gasi multimi frecvente de articole avem doua abordari: <ol style="list-style-type: none"> <li>1.Nivel cu nivel (aplicand principiul a-priori)</li> <li>2.Totale multimile de articole - de orice dimensiune - in cateva treceri (2-3 treceri)</li> </ol> </li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <p><b>Nivel cu nivel</b></p> <ul style="list-style-type: none"> <li>□ Procedam nivel cu nivel, gasind intai articolele frecvente (multimi de dimensiune 1), apoi peruchile frecvente, tripletele frecvente, etc.</li> <li>□ In multe cazuri partea cea mai grea este gasirea peruchelor frecvente; continuarea pe nivelele superioare necesita mai putin timp decat gasirea acestora.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                          |
| <p><b>Cadru de cautare</b></p> <ul style="list-style-type: none"> <li>□ Utilizam termenul <i>multimi frecvente de articole (frequent itemsets)</i> pentru "o multime de articole S care apare in cel putin a 's'-a parte din cosuri", unde s este o constanta aleasa, de obicei 0.01 sau 1%.</li> <li>□ Vom presupune ca avem o cantitate de date care nu incepe in memoria centrala a calculatorului.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                       | <p><b>Cadru de cautare</b></p> <ul style="list-style-type: none"> <li>□ Stocare (pentru exemplele urmatoare): <ul style="list-style-type: none"> <li>□ Fie sunt stocate intr-o baza de date relationala (BDR), de exemplu o relatie (tabela) <i>Cosuri(IdCos, articol)</i></li> <li>□ Fie ca un fisier text cu linii de forma <i>(IdCos, articol1, articol2, ..., articol-n)</i>.</li> </ul> </li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                          | <p><b>Nivel cu nivel</b></p> <ul style="list-style-type: none"> <li>□ Algoritmii de acest tip utilizeaza o trecere per nivel.</li> <li>□ Există insă si abordări care nu sunt de tip nivel cu nivel:</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | <p><b>Toate multimile</b></p> <ul style="list-style-type: none"> <li>□ Gasim toate <i>multimile frecvente de articole maximale</i> (i.e., multimi S a.i. nici o multime care include strict pe S nu este frecventa) intr-o singura trecere sau in cateva treceri.</li> <li>□ Tehnicile de acest tip includ fie esantionarea datelor fie citirea lor in transe</li> <li>□ De obicei sunt suficiente 2 treceri prin date (o trecere pentru esantionarea si inca una pentru verificarea finala)</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                   |
| <p><b>Cadru de cautare</b></p> <ul style="list-style-type: none"> <li>□ Cand evaluam timpul de rulare al algoritmilor parametrul optimizat este <i>numarul de treceri prin date</i>.</li> <li>□ Deoarece costul principal este dat adesea de timpul necesar citirii datelor de pe disc, numarul de citiri pentru fiecare data este adesea cea mai buna masura a timpului de rulare al algoritmului.</li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                          | <p><b>Principiul a-priori</b></p> <ul style="list-style-type: none"> <li>□ Exista un principiu cheie, numit <i>monotonicitate</i> (eng. monotonicity) sau <i>principiul a-priori</i> care ne ajuta sa gasim multimi frecvente de articole: <ul style="list-style-type: none"> <li>▪ <b>Daca o multime de articole S este frecventa</b> (i.e., apare cel putin in a 's'-a parte a cosurilor), atunci orice submultime a lui S este de asemenea frecventa."</li> </ul> </li> </ul> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                  | <p><b>Algoritmul a-priori</b></p> <p>Acest algoritm procedeaza nivel cu nivel.</p> <ol style="list-style-type: none"> <li>1.Dandu-se pragul de suport <math>s</math>, in prima trecere gasim articolele care apar in cel putin a 's'-a parte a cosurilor. Aceasta multime este notata <math>L_1</math>, multimea articolelor frecvente.</li> </ol> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <p><b>Algoritmul a-priori</b></p> <ol style="list-style-type: none"> <li>2. Perechile de articole din <math>L_1</math> devin multimea <math>C_2</math> a <i>perechilor candidate</i> pentru a doua trecere. Speram ca dimensiunea lui <math>C_2</math> nu este atat de mare pentru ca altfel nu este suficient spatiu in memoria centrala pentru un contor numeric intreg al paritetiei fiecarei perechi. Perechile din <math>C_2</math> al caror contor ajunge sau depaseste <math>s</math> formeaza multimea <math>L_2</math> a <i>perechilor frecvente</i>.</li> </ol> <p>F. Radulescu, Curs Utilizarea bazeelor de date, anul IV CS.</p> |

## Algoritmul a-priori

3. Tripletele candidate  $C_3$  sunt multimele  $\{A, B, C\}$  pentru care  $\{A, B\}, \{A, C\}$  și  $\{B, C\}$  sunt în  $L_2$ . În a treia trecere sunt numărate aparițiile tripletelor din  $C_3$ ; cele al căror conținut este cel putin  $s$  formează multimea  $L_3$  a tripletelor frecvente.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

25

## Algoritmul a-priori

4. Se poate merge oricât de departe să dorescă (sau până multimile devinvide),  $L_i$  conține multimile frecvente de articole de dimensiune  $i$ ,  $C_{i+1}$  este multimea candidatelor de dimensiune  $i+1$  a.i. fiecare submultime a lor de dimensiune  $i$  este inclusă în  $L_i$ .

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

26

## Efect a-priori

- A-priori "împinge clauza HAVING în jos pe arborele expresiei", determinându-ne în primul rand să înclocuim *Cosuri* cu rezultatul "cererii":

```
SELECT *
FROM Cosuri
GROUP BY articol
HAVING COUNT(*) >= s;
```

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

31

## Efect a-priori

- Cererea corectă care returnează doar linile continând articolele frecvente din cosuri este:
- ```
SELECT * FROM COSURI
WHERE articol IN
(SELECT articol // articole
FROM Cosuri // frecvente
GROUP BY articol
HAVING COUNT(*) >= s);
```

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

32

Algoritmul a-priori

- Oprea algoritmului se face:
1. În cazul în care nici una dintre multimile din C_{i+1} nu are un conținut de apariții mai mare decât pragul de suport.

Sau

2. Nu putem forma nici un element al multimii C_{i+1} din elementele lui L_i .
- De exemplu dacă $L_2 = \{(1, 2), (1, 3)\}$ nu se poate găsi nici un triplet (a, b, c) cu toate submultimile de 2 elemente în L_2

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

27

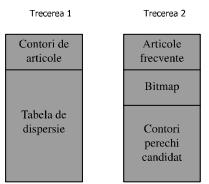
Efect a-priori

- Sa considerăm cererea SQL din slide-ul anterior cu ipotezele:
- Utilizează tabela *Cosuri* (*IdCos*, *articol*)
 - având 10^8 tupluri
 - care contin date despre 10^7 cosuri
 - de către 10 articole fiecare,
 - Presupunem existența a 100,000 articole diferite (tipic pentru o reteaua ca Wal-Mart de exemplu).

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

29

PCY



F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

37

Trecerea 1

- Se numără aparițiile fiecarui articol
- Pentru fiecare cos constant din articolele $\{i_1, \dots, i_n\}$, se aplică funcția de dispersie fiecărei perechi asociind-o unei intrări a unei tabele de dispersie și se incrementă conținutul acestora cu 1.
- La sfârșitul trecerii, se determină L_1 , articolele cu conținut cel puțin s .

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

38

Trecerea 2

- Oparece (i, j) poate fi candidat în C_2 doar dacă toate conditiile următoare sunt adevărate:
1. i este în L_1 ,
 2. j este în L_1 ,
 3. (i, j) este dispersată într-o intrare frecventă (se utilizează bitmapul)
- Ultima condiție diferențiază PCY de a-priori clasice și reduce necesarul de memorie în trecerea 2.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

43

Trecerea 2

- În timpul trecerii 2 luăm în considerare fiecare cos și fiecare pereche de articole din el, efectuând testul de mai sus.
- Dacă o pereche îndeplinește toate cele trei condiții, se incrementă conținutul acesteia din memorie sau se crează unul dacă acesta nu există încă.
- În final perechile numărate care au un conținut cel puțin s formează L_2 .

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

44

Trecerea 1

- De asemenea la sfârșitul trecerii se determină cele intrări din tabela de dispersie care au conținutul cel puțin egal cu s (intrări frecvente).
- Punct cheie: o pereche (i, j) nu poate fi frecventă decât dacă este dispersată într-o intrare frecventă astfel încât perechile care sunt dispersate în alte intrări NU POT fi candidata în C_2 .

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

39

Trecerea 1

- Se înlocuiește tabela de dispersie cu un bitmap având un bit per intrare: 1 dacă intrarea a fost frecventă, 0 altfel.
- Acest bitmap va fi folosit la trecerea 2 prin date.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

40

Q & A

- Q: Când este mai performant PCY decât a-priori?
- A: Cand sunt prea multe perechi de articole din L_1 pentru a încapea într-o tabela de perechi candidate și de conținut asociați în memoria centrală iar numărul de intrări frecvente din algoritmul PCY este suficient de mic pentru a reduce dimensiunea lui C_2 suficient pentru a încapea în memoria centrală (chiar și fără 1/16 din ea consumată de bitmapul).

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

45

Q & A

- Q: Cand o mare parte a intrărilor nu vor fi frecvente în PCY?
- A: Cand sunt puține perechi frecvente iar cea mai mare parte a perechilor sunt atât de puțin frecvente încât chiar dacă sumam conținutul tuturor perechilor care sunt dispersate într-o intrare data nu sunt mari sanse să se obțină o valoare egală sau mai mare ca s .

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

46

Trecerea 2

- Memoria centrală conține o listă cu toate articolele frecvente, i.e. L_1 .
- Tot memoria centrală conține un bitmap reprezentând rezultatele dispersiei din prima trecere.
- Punct cheie: intrările utiliză 16 sau 32 de biți pentru un conținut dar sunt comprimate la un singur bit.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

41

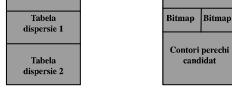
Trecerea 2

- Astfel, chiar dacă tabela de dispersie ocupă aproape întregă memoria centrală la prima trecere, bitmapul nu ocupă mai mult de 1/16 din memoria centrală la trecerea 2.
- În final, memoria centrală conține de asemenea o tabelă cu toate perechile candidat și conținutul asociați lor.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

Tabele de dispersie multiple

- Varianta a PCY. Se folosesc mai multe funcții de dispersie.
- Se împarte memoria între două sau mai multe tabele de dispersie, ca în figura următoare:



F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

47

Tabele de dispersie multiple

- La trecerea 2 se retine în memoria cată un bitmap pentru fiecare dintre acestea; de notat că spațiul necesar pentru aceste bitmape este exact același cu cel necesar în bitmapul unic din PCY deoarece numărul total de intrări reprezentate este același.
- Pentru a fi candidata la C_2 o pereche:
1. Conține articole din L_1 , și
 2. Este dispersată într-o intrare frecventă în fiecare tabelă de dispersie.

F. Radulescu, Curse Utilizarea bazei de date, anul IV CS.

48

<p>Tabele de dispersie iterate</p> <ul style="list-style-type: none"> □ Tabele de dispersie iterate <i>Multistage</i>: □ Ca la PCY dar in locul verificarii candidatelor in trecerea 2 se creaza o alta tabela de dispersie (alta functie de dispersie) si sunt dispersate doar acele perechi care indeplinesc conditiile pentru PCY; i.e., ambele articole sunt din L₁, si sunt dispersate intr-un intrare frecventa in prima trecere. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 49</p>	<p>Tabele de dispersie iterate</p> <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 50</p>	<p>Abordarea simplă</p> <ul style="list-style-type: none"> □ <i>Abordarea simplă</i>: Se ia un esantion de date de dimensiunea memoriei centrale. □ Se ruleaza un algoritm nivel cu nivel in memoria centrala (deci nu sunt costuri de I/O) si □ Se spera ca esantionul ne va conduce la adevaratele multimii frecvente. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 55</p>	<p>Abordarea simplă</p> <ul style="list-style-type: none"> □ De notat ca pragul s trebuie scalat prin micsorare; e.g. daca esantionul este 1% din date, se utilizeaza s/100 ca prag de suport (in valoare absoluta). □ Se poate face o trecere completa prin date pentru a verifica daca multimile frecvente de articole ale esantionului sunt cu adevarat frecvente, <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 56</p>
<p>Tabele de dispersie iterate</p> <ul style="list-style-type: none"> □ Intr-o treia trecere, pastram cate un bitmap pentru fiecare tabela de dispersie si tratam o pereche ca fiind candidata (in C₃) doar daca: <ul style="list-style-type: none"> □ Ambele articole sunt in L₁. □ Perechea a fost dispersata intr-o intrare frecventa in prima trecere. □ Perechea a fost dispersata de asemenea intr-o intrare frecventa in trecerea 2. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 51</p>	<p>Q & A</p> <ul style="list-style-type: none"> □ Q: Cand sunt utile tabelele de dispersie multiple? □ A: Cand cele mai multe dintre intrari de la prima trecere a PCY au contori cu mult sub pragul s. Atunci putem dubla contorii intrarilor si totusi cele mai multe vor fi sub prag. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 52</p>	<p>Abordarea simplă</p> <ul style="list-style-type: none"> □ Vor fi pierdute insa multimile de articole care sunt frecvente in ansamblul datelor dar nu in esantion. □ Pentru a minimiza falsele negative se poate scadea putin pragul pentru esantion gasindu-se mai multe candidate pentru trecerea prin ansamblul datelor. □ Risc: prea multe candidate pentru a incepea in memoria centrala. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 57</p>	<p>SON95</p> <ul style="list-style-type: none"> □ <i>SON95</i> (Savasere, Omniecinski and Navathe in VLDB 1995; referit de Toivonen). □ Se citesc submultimi (transite) ale datelor in memoria centrala aplicandu-se abordarea simpla pentru descoperirea multimilor candidat. □ Fiecare cos este parte a uneia dintre aceste submultimi. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 58</p>
<p>Q & A</p> <ul style="list-style-type: none"> □ Q: Cand sunt utile tabelele de dispersie iterate? □ A: Cand numarul intrarilor frecvente din prima trecere este mare (e.g. 50%) - dar nu toate, Atunci, a doua dispersie cu unele dintre perechile ignorante poate reduce semnificativ numarul de intrari. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 53</p>	<p>Toate multimile in 2 treceri</p> <ul style="list-style-type: none"> □ Metodele de mai sus sunt cele mai bune cand dorim perechile frecvente, cazul cel mai comun. □ Daca dorim toate multimile frecvente maximale de articole, inclusiv multimile mari, sunt necesari prea multi pasi. □ Există mai multe abordari pentru obtinerea tuturor multimilor frecvente de articole in doua treceri sau mai putin. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 54</p>	<p>SON95</p> <ul style="list-style-type: none"> □ In a doua trecere o multime este candidata daca a fost identificata ca si candidata in una sau mai multe submultimi ale datelor. □ Punct cheie: O multime nu poate fi frecventa in ansamblul datelor daca nu este frecventa in cel putin o submultime a acestora (oare de ce?). <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 59</p>	<p>Toivonen</p> <ul style="list-style-type: none"> □ Se ia un esantion care incape in memoria centrala. □ Se ruleaza abordarea simpla pe aceste date dar cu un prag micsorat astfel incat sa fie imposibila pierderea unei multimile frecvente de articole (e.g. daca esantionul este de 1% din date se foloseste s/125 ca prag de suport). <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 60</p>
<p>Toivonen</p> <ul style="list-style-type: none"> □ Se adauga candidatelor din esantion <i>marginea negativa</i>: cele multimii de articole S astfel incat S nu este identificata ca frecventa in esantion dar orice submultime stricta maxima a lui S este identificata astfel. □ De exemplu, daca ABCD nu este frecventa in esantion dar ABC, ABD, ACD si BCD sunt frecvente in esantion, atunci ABCD este in marginea negativa. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 61</p>	<p>Toivonen</p> <ul style="list-style-type: none"> □ Se face o trecere prin date, numarand toate multimile frecvente de articole si marginea negativa. □ Daca nici o multime din marginea negativa nu este frecventa in ansamblul datelor, atunci multimile frecvente de articole sunt exact cele candidate care sunt deasupra pragului. <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 62</p>		
<p>Bibliografie</p> <ul style="list-style-type: none"> □ J.D.Ullman - CS345 --- Lecture Notes, primele doua capitulo (Overview of Data Mining, Association-Rules, A-Priori Algorithm) http://infolab.stanford.edu/~ullman/cs345-notes.htm <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 63</p>	<p>Sfârșitul capitolului 8</p> <p>F Radulescu, Curs Utilizarea bazelor de date, anul IV CS. 64</p>		

Capitolul 9

Data mining – date corelate

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

1

Reprezentarea datelor

- ❑Vom continua să considerăm modelul de date "coșuri de produse" și vom vizualiza datele ca o matrice booleană unde:
 - > linii=coșuri și
 - > coloane=articole.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

2

Aplicații

- 2. Linile și coloanele sunt pagini de web:
 $(r, c) = 1$ înseamnă că pagina corespunzătoare coloanei c conține legături către pagina liniei r .
- ❑Acum, coloane similare pot reprezenta copii multiple (mirror) ale unei pagini.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

3

Aserțiuni

1. Matricea este foarte rară; aproape peste tot 0.
2. Numărul de coloane (articole) este suficient de mic pentru a putea stoca în memoria centrală ceea ceva per coloană dar suficient de mare astfel încât nu putem stoca ceva per perche de coloane în memoria centrală (aceeași așteptare pe care am făcut-o până acum privind regulile de asociere).

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

5

Aserțiuni

3. Numărul de linii este atât de mare încât nu putem stoca întreaga matrice în memoria chiar profitând de faptul că e rară și comprimând-o (din nou aceeași așteptare ca întotdeauna).
4. Nu suntem interesați de perchele sau mulțimile de coloane cu larg suport; în schimb dorim perchele de coloane puternic corelate (similare).

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

4

Aplicații

- 4. linile sunt propoziții iar coloanele sunt pagini web.
- ❑Coloane similare pot fi pagini despre același domeniu.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

9

Similaritate

- ❑Am vorbit despre similaritatea coloanelor fără să dam o măsură cantitativă a acesteia.
- ❑Definiție: Să ne gândim la o coloană ca la multimea linilor pentru care coloana conține 1. Atunci **similaritatea** a două coloane C_1 și C_2 este $\text{Sim}(C_1, C_2) = |C_1 \cap C_2| / |C_1 \cup C_2|$.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

10

Aplicații

- ❑Aplicațiile de marketing sunt interesante doar de produsele de larg consum (nu merită să încercăm promovarea obiectelor pe care oricum nu le cumpără mai nimănui).
- ❑Există însă un număr de aplicații care se potriveșc cu modelul de mai sus, de interes fiind în special problema **percheilor** de coloane / articole inclusiv de consum restrâns dar puternic corelate:

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

5

Aplicații

1. Linile și coloanele sunt pagini de web:
 $(r, c) = 1$ înseamnă că pagina corespunzătoare liniei r conține legături către pagina coloanei c .
- ❑Coloane similare pot fi pagini despre același domeniu.
- ❑Obs: Coloane similare = au 1 în cam aceeași poziție.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

Similaritate

- Unde:
➢ $|x|$ reprezintă cardinalul multimii x , deci:
➢ $|C_1 \cap C_2|$ reprezintă numărul de linii în care ambele coloane au valoarea 1
➢ $|C_1 \cup C_2|$ reprezintă numărul de linii în care cel puțin una dintre coloane are valoarea 1.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

11

Exemplu

0	1
1	0
1	1
0	0
1	1
0	1

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

12

Problema

- ❑Deoarece matricea conține foarte multe linii și coloane, testarea similarității este laborioasă (matricea nu încape în memoria centrală).
- ❑Ar fi preferabil ca fiecare coloană să fie reprezentată de o cantitate mult mai mică de informație având proprietatea că testul de similaritate efectuat pe această sa fie relevant pentru similaritatea coloanelor.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

13

Signature

- ❑Idee principală: Se mapează ("disperează") fiecare coloană C într-o cantitate mică de date numita **signature** lui C , (notată $\text{Sig}(C)$) astfel încât:
- $\text{Sig}(C)$ este suficient de mică pentru ca signaturele tuturor coloanelor să încapă în memoria centrală și să se poată efectua testul de similaritate.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

Convenție utilă

- ❑De asemenea vom utiliza a pentru "numărul de linii de tip a ", ș.a.m.d.
- ❑De notat că $\text{Sim}(C_1, C_2) = a / (a + b + c)$.
- ❑Dar cum cele mai multe linii sunt de tip a , într-o selecție de, să spunem, 100 de linii alese aleator toate vor fi de tip a , deci similaritatea coloanelor doar pentru aceste 100 de linii nici nu este definită.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

19

Ce vom prezenta în continuare:

- a. **Dispersia de tip Min** și **dispersia de tip k-min** – metode de calcul signature
- b. **Dispersia sensibilă la localizare (DSL)** – o metodă prin care se minimizează numărul de perechi de signature care se testează pentru similaritate

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

20

Signature

- ❑Când un mod de calcul pentru signature este bun?
- Când coloanele C_1 și C_2 sunt puternic similare dacă și numai dacă $\text{Sig}(C_1) \approx \text{Sig}(C_2)$ sunt puternic similare (dar de notat că este nevoie să definim "similaritatea" pentru signature).

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

32

Exemplu (prost)

- ❑O idee - care însă nu funcționează:
 - >Se iau aleator 100 de linii și șiruul de 100 de biti ai coloanelor pentru acele linii este signaturea fiecărei coloane.
 - ❑De ce nu funcționează?

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

Dispersia de tip Min (Minhashing)

- ❑Este o metodă de calcul signature
- ❑Signature unei coloane va fi un sir de numere întregi.
- ❑Să ne imaginăm linile permute într-o ordine aleatoare. "Disperează" fiecare coloană C în $\text{Sig}(C)$, numărul primei linii în care coloana C are un 1.
- ❑În felul acesta obținem primul întreg al signaturei

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

21

Dispersia de tip Min (Minhashing)

- ❑Probabilitatea ca $\text{H}(C_1) = \text{H}(C_2)$ este $a / (a + b + c)$ deoarece valoarea de dispersie este aceeași dacă prima linie cu un 1 în vreuna din coloane este de tip a și este diferită dacă prima astfel de linie este de tip b sau c .
- ❑De notat că această probabilitate este aceeași cu $\text{Sim}(C_1, C_2)$.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

22

Exemplu (prost)

- ❑Motivul pentru care ideea nu funcționează este că matricea este presupusă că fiind **foarte rară** deci multe coloane vor avea signature identice formate doar din 0 chiar dacă ele nu sunt deloc similare,

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

33

Convenție utilă

- ❑Dându-se două coloane C_1 și C_2 , ne vom referi la linile lor ca fiind de patru tipuri – a , b , c , d – în funcție de bitii lor pe aceste coloane, după cum urmează:

Tip	C1	C2
a	1	1
b	1	0
c	0	1
d	0	0

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

18

Dispersia de tip Min (Minhashing)

- ❑Dacă repetăm experimentul cu o nouă permutare a linilor de un număr mare de ori, să zicem 100, obținem o signature constantă din 100 de numere de linii pentru fiecare coloană.
- ❑"Similaritatea" acestor liste (fractiunea a pozitilorlor în care ele sunt egale) va fi foarte apropiată de similaritatea coloanelor.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

22

Dispersia de tip Min (Minhashing)

- ❑Observație importantă: nu trebuie să permitem fizic linile, ceea ce ar duce la multe treceri prin întreaga cantitate de date.
- ❑În schimb cînd linile într-o ordine oarecare și dispersăm fiecare linie (numărul acesteia) utilizând (să zicem) 100 de funcții de dispersie diferite.

F.Rădulescu, Curs Universitatea Bucureşti de date și VCS

24

DSL Hamming

- ❑ O a doua trecere prin datele originale confirmă care dintr-o candidată sunt întrăveăr similare.
- ❑ Aceasta metodă **exploatează** o idee care poate fi folosită și în alte parti: coloanele similare au un număr similar de 1, deci nu are rost compararea coloanelor al căror număr de 1 este foarte diferit

F. Radulescu, Curs Universitar
de date, anul IV/CS

49

Bibliografie

- ❑ J.D.Ullman - CS345 ---- Lecture Notes, Capitolul 3
(Overview of Data Mining, Association-Rules, A-Priori Algorithm)
<http://csail.mit.edu/~ullman/cs345notes.html>

F. Radulescu, Curs Universitar
de date, anul IV/CS

51

50

Sfârșitul capitolului 9

F. Radulescu, Curs Universitar
de date, anul IV/CS

51

Capitolul 10

Algoritmi de clasificare

Cuprins

- ❑ Ce este învățarea supervizată
- ❑ Evaluare clasificatori
- ❑ Arbori de decizie: ID3 și C4.5
- ❑ Clasificator bayesien (Naive Bayes)
- ❑ Mașini cu vectori suport (Support vector machines)
- ❑ K-nearest neighbors
- ❑ Metode asambliste: Bagging, Boosting, Random Forest

2

Formatul datelor de intrare

- ❑ În majoritatea cazurilor, setul de antrenament (precum și setul de test / validare sau de evaluare) pot fi reprezentate ca un tabel având o coloană pentru fiecare atribut al lui X, iar ultima coloană conține eticheta clasei.
- ❑ Un exemplu foarte folosit este Play-tennis în care sunt folosite condițiile meteorologice pentru a decide dacă jucătorii pot sau nu începe un nou joc.
- ❑ Acest set de date va fi utilizat și în acest curs.

3

4

5

6

7

8

Setul Play tennis

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Strong	No	
Sunny	Hot	High	Weak	Yes	
Overcast	Hot	High	Weak	Yes	
Rain	Hot	High	Weak	Yes	
Rain	Moderate	High	Weak	Yes	
Rain	Moderate	High	Strong	No	
Overcast	Cool	Normal	Strong	Yes	
Rain	Cool	Normal	Weak	Yes	
Sunny	Cool	Normal	Weak	Yes	
Rain	Moderate	Normal	Weak	Yes	
Rain	Moderate	High	Weak	Yes	
Sunny	Moderate	High	Weak	Yes	
Rain	Cool	Normal	Strong	Yes	
Overcast	Cool	High	Strong	Yes	
Sunny	Cool	High	Strong	Yes	
Rain	Moderate	High	Strong	Yes	
Rain	Moderate	High	Weak	Yes	
Sunny	Moderate	Normal	Weak	Yes	
Rain	Hot	High	Strong	No	

9

Obiective

- ❑ Învățarea supervizată este un subdomeniu foarte studiat din Data Mining
- ❑ În acest caz se construiesc noi modele din experiențe trecute (date)

Definiții

- ❑ Învățarea supervizată include:
 - ❑ Clasificare: rezultatele sunt valori discrete (obiectiv: identificarea apartenenței la un grup / clasă).
 - ❑ Regresie: rezultatele sunt valori continue sau ordonate (obiectiv: estimare sau prezicerea unui răspuns).
- ❑ În acest capitol ne concentrăm pe clasificare

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

Clasificarea

Întrare:

- Un set de clase $C = \{c_1, c_2, \dots, c_k\}$ în număr de k .
- Un set de n articole etichetate $D = \{(d_1, c_1), (d_2, c_2), \dots, (d_n, c_n)\}$. Articolele sunt d_1, \dots, d_n , fiecare articol d_i fiind etichetat cu o clasa c_i din C , se numește **dataset de antrenare**.
- Pentru că clasificarea aruncă un **algoritm**, este necesara o mulțime de validate. Aceasta conține de asemenea elemente etichetate care nu sunt incluse în setul de antrenare.

Ieșire:

- Un model sau o metodă pentru clasificarea articolelor noi, Mulțimea de articole pe care vor fi clasificate folosind modelul / metoda obținută se numește **mulțimea/test**.

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

Validarea încriușată

- 2. Validare încriușată folosind 2 subseturi - *2-fold cross validation*.**
Validation. Pentru $k = 2$, metoda de mai sus are avantajul de a folosi seturi mari atât pentru antrenament, cât și pentru testare.

✓ Validarea încriușată este mai bună.

Validarea încriușată

- 3. Validare încriușată stratificată - *Stratified cross validation*.**
 Este o variație de validare încriușată având k subseturi. Fiecare subset are aceeași distribuție a etichetelor.
 De exemplu, pentru exemple pozitive și negative, fiecare subset conține aproximativ aceeași proporție de exemple pozitive și aceeași proporție de exemple negative.

26

✓ Validarea încriușată stratificată este mai bună.

Cuprins

- Ce este învățarea supervizată
- Evaluare clasificatori
- Arbori de decizie: ID3 și C4.5
- Clasificatori bayesieni (Naïve Bayes)
- Mașini cu vectori suport (Support vector machines)
- K-nearest neighbors
- Metode asambliste: Bagging, Boosting, Random Forest

✓ Validarea încriușată stratificată este mai bună.

Ce este un arbore de decizie?

- Un mod comun de a reprezenta un model sau un algoritm de clasificare este un arbore de decizie.
- Având un set de antrenare D și un set A de attribute ale exemplelor, fiecare exemplu etichetat din D are forma: $(a_1 = v_1, a_2 = v_2, \dots, a_n = v_n)$.
- Pe baza acestor attribute se poate construi un arbore de decizie astfel:

32

✓ Validarea încriușată stratificată este mai bună.

Validarea încriușată

- 4. Validație încriușată având subseturi de un element - *Leave one out cross validation*.** Când D conține doar un număr mic de exemple, o valdare încriușată specială poate fi folosită: fiecare exemplu devine set de test și toate celelalte exemple sunt setul de antrenare.

Accuratețea pentru fiecare clasificator este fie 100%, fie 0%.

Media tuturor acestor valori este accuratețea finală.

✓ Validarea încriușată stratificată este mai bună.

Metoda bootstrap

- Metoda **bootstrap**, face parte din metodele cu eșantionare și constă în obținerea setului de antrenare din setul de date original prin eșantionare cu înlocuire (eșantionul nu este înțăritură din set).
- Cazurile care nu sunt alesă în setul de antrenare sunt utilizate ca set de teste.
- De exemplu, dacă D are 1000 de exemple etichetate, alegând să întărimpre un exemplu de 1000 de ori din setul de antrenare. În acesta unele exemple sunt prezente de mai multe ori.

28

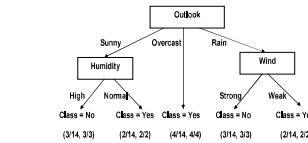
✓ Validarea încriușată stratificată este mai bună.

Ce este un arbore de decizie?

- Nodurile interne sunt attribute (nicio călătoare nu conține de două ori același atribut).
- Ramurile sunt etichetate la valoare discrete (una sau mai multe) sau intervale pentru aceeași atribut. Uneori pot fi utilizate condiții mai complexe pentru ramificare.
- Frunzele sunt etichetate cu clase. Pentru fiecare frunză se poate calcula un suport și o probabilitate, suportul este proporția de exemple care au acelăși atribut ca și rădăcina. Probabilitatea este probabilitatea de a fi corect clasificate. Încrederile sunt următoarele: $P(\text{fiecare frunză} \mid \text{aceeași atribut}) = \frac{\text{suport}}{\text{suport} + \text{suport} \text{ a altor clase}}$
- Încrederile sunt următoarele: $P(\text{fiecare frunză} \mid \text{aceeași atribut}) = \frac{\text{suport}}{\text{suport} + \text{suport} \text{ a altor clase}}$
- De exemplu se poate avea o singură călătoare a arborului (deci o singură frunză = clasa).

33

Exemplu



34

✓ Validarea încriușată stratificată este mai bună.

Metoda bootstrap

- Statistică: 63,2% din exemplele din D sunt alesă pentru setul de antrenare iar 36,8% nu. Aceste 36,8% devin setul de test.
- După construirea unui clasificator și urmarea acestuia pe setul de test este determinată accuratețea și astfel poate fi evaluată metoda de construire a clasificatorului în funcție de valoarea obținută.

Mai multe despre această metodă și alte tehnici de evaluare pot fi găsite în [Sanderson 08].

✓ Validarea încriușată stratificată este mai bună.

De ce 63,2%?

- Din "Data Mining: Concepts and Techniques", să presupunem că avem un set de date cu d exemple. Fiecare exemplu are o probabilitate de $1/d$ de a fi selectat, deci probabilitatea de a nu fi ales este $(1-1/d)^d$.
- Cum alegem d ori, probabilitatea ca un exemplu să nu fie ales în tot acest timp este $(1-1/d)^d$.
- Dacă d este mare, probabilitatea se apropiie de $e^{-1} = 0.368$. Astfel, 36,8% din exemple nu vor fi selectate pentru setul de antrenare și vor ajunge în setul de test iar restul de 63,2% vor forma setul de antrenare.

* J. W. Han, M. Kamber, Data Mining: Concepts and Techniques, Second Edition, 2006, Morgan Kaufmann, pagina 362.

35

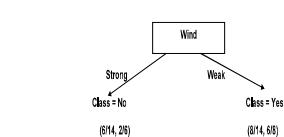
✓ Validarea încriușată stratificată este mai bună.

Exemplu

- Numerele de pe ultima linie sunt suportul și încrederea asociate fiecărei frunze.
- Pentru același set de date pot fi creați mai mulți arbori de decizie.
- De exemplu un alt arbore de decizie pentru același set de date se află în figura următoare (cu valori mai mici ale încrederii decât arborele anterior):

36

Decision trees



✓ Validarea încriușată stratificată este mai bună.

ID3

- ID3 (Iterativ Dichotomiser 3) este un algoritm pentru construirea arborilor de decizie introdus de Ross Quinlan în 1986 (vezi [Quinlan 86]).
- Algoritmul construiește arborele de decizie într-o manieră top-down alegând la fiecare nod atributul „cel mai bun” pentru ramificare.

Mai întâi este ales un atribut rădăcină, construind o ramură separată pentru fiecare valoare diferență a atributului.

✓ Validarea încriușată stratificată este mai bună.

ID3

- Setul de antrenare este de asemenea împărțit, fiecare ramură menținând exemplele care se potrivește cu valoarea atributului ramurii respective.
- Procesul se repetă pentru fiecare descendenta până când toate exemplele au aceeași clasă (în acest caz nodul devine o frunză etichetată cu acea clasă) sau toate attributele au fost utilizate (nodul devine, de asemenea, o frunză etichetată cu valoarea modală = clasa majoritară).
- Un atribut nu poate fi ales de două ori pe aceeași cale; din momentul în care a fost ales pentru un nod, nu va mai fi niciodată testat pentru descendenții aceluia nod.

38

✓ Validarea încriușată stratificată este mai bună.

Exemplu

- Pentru setul de date Play tennis există patru attribute posibile pentru rădăcina arborului decizional: Aspect, Temperatură, Umiditate și Vânt.
- Entropia întregului set de date și valoare ponderată pentru împărțirea utilizării celor patru attribute sunt:

$$\text{entropy}(D) = -\sum_{i=1}^4 \Pr(c_i) \log_2 \Pr(c_i)$$

39

✓ Validarea încriușată stratificată este mai bună.

Pentru fiecare atribut

$$\text{entropy}(D, \text{Humidity}) = \frac{9}{14} \left(-\log_2 \frac{3}{7} - \log_2 \frac{4}{7} \right) + \frac{7}{14} \left(-\log_2 \frac{6}{7} - \log_2 \frac{6}{7} \right) = 0.79$$

✓ La fel:

$$\text{entropy}(D, \text{Wind}) = 0.89$$

$$\text{entropy}(D, \text{Temperature}) = 0.91$$

$$\text{entropy}(D, \text{Outlook}) = 0.69$$

✓ Validarea încriușată stratificată este mai bună.

Cel mai bun atribut

- Esența ID3 este modul în care este descoperit atributul „cel mai bun”. Algoritmul folosește teoria informației înserând să crească puritatea seturilor de date de la nodul tău la descendentul.
- Să luăm în considerare un set de date:

$D = \{e_1, e_2, \dots, e_{14}\}$

cu exemple etichetate cu clase din

$C = \{c_1, c_2, \dots, c_6\}$

Atributele exemplelor sunt A_1, A_2, \dots, A_9 .

✓ Validarea încriușată stratificată este mai bună.

Entropia

- Atunci entropia lui D poate fi calculată astfel:

$$\text{entropy}(D) = -\sum_{i=1}^n \Pr(c_i) \log_2 \Pr(c_i)$$

- Dacă atributul A_1 , având n valori distincte este considerat pentru ramificare, acesta va partaja setul de date în sub集 D_1, D_2, \dots, D_n .

40

✓ Validarea încriușată stratificată este mai bună.

Cel mai bun atribut: Outlook

- Următorul tabel conține valorile pentru entropie și căstig.
- Cel mai bun atribut pentru nodul rădăcină este Aspect (Outlook), cu căstigul maxim de 0,25.

Atribut	entropy	gain
Humidity	0.89	0.16
Wind	0.89	0.05
Temperature	0.91	0.03
Outlook	0.69	0.25

41

✓ Validarea încriușată stratificată este mai bună.

Discuție ID3

- Deoarece este un algoritm greedy ID3 produce un optim local.
- Cel mai bun atribut pentru nodul rădăcină este Aspect (Outlook), cu căstigul maxim de 0,25.

$$\text{gain-ratio}(D, A_k) = \frac{\text{gain}(D, A_k)}{\text{entropy}(D, A_k)}$$

Discuție ID3

- Unele attribute (de exemplu A) pot fi continue. Valoarea pentru A pot fi împărțite în două intervale:
 - Valoarea lui t poate fi după cum urmează:
 - Se sortează exemplul după A .
 - Se alege ca valoare candidat media a două valori consecutive pentru care clasa se schimbă.
 - Pentru fiecare valoare candidat de la pasul anterior se calculează căstigul dacă particionează se face folosind acea valoare. Candidatul cu căstig maxim este ales pentru particionare.

Căstigul informational

- Decarece puritatea seturilor de date crește, entropia (D) este mai mare decât entropia (D, A_k). Diferența dintre ele se numește căstig informational:

$$\text{Gain}(D, A_k) = \text{entropy}(D) - \text{entropy}(D, A_k)$$

- Cel mai bun atribut este cel având căstigul informational maxim.

42

✓ Validarea încriușată stratificată este mai bună.

Discuție ID3

- Unele attribute (de exemplu A) pot fi continue. Valoarea pentru A pot fi împărțite în două intervale:
 - Valoarea lui t poate fi după cum urmează:
 - Se sortează exemplul după A .
 - Se alege ca valoare candidat media a două valori consecutive pentru care clasa se schimbă.
 - Pentru fiecare valoare candidat de la pasul anterior se calculează căstigul dacă particionează se face folosind acea valoare. Candidatul cu căstig maxim este ales pentru particionare.

✓ Validarea încriușată stratificată este mai bună.

Discuție ID3

- În acest fel, atributul continuu este înlocuit cu unul discret (două valori, una pentru fiecare interval).
- Acest atribut concurează cu atributul rămase pentru „cel mai bun” atribut.
- Procesul se repetă pentru fiecare nod, deoarece valoarea partilionară se poate schimba de la un nod la altul.

Calcularea costului înlocuitor la un nod

50

Discuție ID3

5. Când puntemele unele attribute sunt mai scumpe decât altele (măsurăți pe unitatea în bani):
 - Este mai bine ca attributele cu costuri mai mici să fie mai aproape de rădăcina decât cele altibile.
 - De exemplu, pentru o unitate de urgență, este mai bine să testăm pușcul și temperatură mai întâi și numai atunci când este necesar să efectuăm o biopsie.
- Această lucru se poate face prin ponderarea căștișului cu costul:

$$\text{weighted-gain}(A_k) = \frac{\text{gain}(D, A_k)}{\text{cost}(A_k)}$$

Calcularea costului înlocuitor la un nod

Teorema lui Bayes

- Thomas Bayes (1701 - 1761) a fost un cleric presbiterian și un matematician englez.
- Teorema numită după el poate fi exprimată astfel:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

unde $P(A|B)$ înseamnă probabilitatea lui A fiind dat B.

Calcularea costului înlocuitor la un nod

51

C4.5

- C4.5 este versiunea îmbunătățită a ID3 și a fost dezvoltată de același Ross Quinlan. Câteva caracteristici:
 - Sunt permise attribute numerice (continue)
 - Tratează cazul valorilor lipsă
 - Utilizează post-pruning pentru a face față la date cu zgomot.
 - Cele mai importante îmbunătățiri ale ID3 sunt:
 - 1. Atributul sunt ales pe baza raportului de căști (gain ratio) și nu pur și simplu a căștișului informațional.

Calcularea costului înlocuitor la un nod

52

C4.5

2. Folosește post-pruning pentru a reduce dimensiunea arborului. Tărâiera se face numai dacă reduce eroarea estimată. Există două metode de tărîere:
 - Înlătură sub-arbore: Un sub-arbore este considerat numai după toți sub-arborei săi. Aceasta este o abordare de jos în sus,
 - Ridicare sub-arbore: Un nod este ridicat și înlătură este număru superior. Dar în acest caz, unele exemple trebuie reasignate. Această metodă este considerată mai putin importanță și mai lentă decât prima.

Calcularea costului înlocuitor la un nod

53

Soluție

$$\begin{aligned} \Pr(\text{Opt}) &= 0,3 \\ \Pr(3 \text{ rânduri} | \text{Opt}) &= 0,4 \\ \Pr(3 \text{ rânduri}) &= 0,2 \end{aligned}$$

Deci:

$$\Pr(3 \text{ rânduri} | \text{Opt}) * \Pr(\text{Opt}) = \frac{\Pr(3 \text{ rânduri})}{\Pr(3 \text{ rânduri})} = 0,4 * 0,3 / 0,2 = 0,6 \text{ sau } 60\%$$

Calcularea costului înlocuitor la un nod

54

Exemplu

- Studenții din EC004 sunt 70% de la C5 și 30% optionali.
- 20% dintre studenții sunt plasări în primele 3 rânduri, dar pentru optionali acest procent este de 40%.
- Când decanul intră în sală și se așează undeva în primele 3 rânduri, îngă un student, care este probabilitatea ca acest studenț să fie din categoria optional?

Calcularea costului înlocuitor la un nod

55

56

Cuprins

- Ce este învățarea supervizată
- Evaluare clasificatori
- Arboi de decizie: ID3 și C4.5
- Clasificator bayesian (Naive Bayes)
- Mașini cu vectori suport (Support vector machines)
- K-nearest neighbors
- Metode asamblante: Bagging, Boosting, Random Forest

Calcularea costului înlocuitor la un nod

55

Naive Bayes: Generalități

- Această abordare este una probabilistică.
- Algoritmii bazati pe teorema lui Bayes calculează pentru fiecare exemplu de test nu o singură clasă, ci probabilitatea pentru fiecare clasă din C (setul de clase).
- Dacă setul de date are k attribute, A_1, A_2, \dots, A_k , obiectivul este să calculem pentru fiecare clasă $c \in C = \{c_1, c_2, \dots, c_n\}$ probabilitatea ca exemplul de test (a_1, a_2, \dots, a_k) să aparțină clasei c.

$$\Pr(\text{Class} = c | A_1 = a_1, \dots, A_k = a_k)$$

□ Dacă este necesară clasificare, clasă cu cea mai mare probabilitate poate fi atribuită acestui exemplu.

Calcularea costului înlocuitor la un nod

56

Construirea modelului

- Facem următoarea presupunere: „toate attributele sunt condițional independente dându-se clasa $C = c_j$ atunci:

$$\Pr(A_1 = a_1, \dots, A_k = a_k | C = c_j) = \prod_{i=1}^k \Pr(A_i = a_i | C = c_j)$$

□ Din cauza acestei presupuneri metoda este numită „naïvă”.

□ Presupunerea nu este valabilă în toate situațiile.

□ Practica arată însă că rezultatele obținute folosind aceasta presupunere simplificatoare sunt suficiente de bune în majoritatea cazurilor.

Calcularea costului înlocuitor la un nod

57

Construirea modelului

- În final, înlocuind expresia anterioră în formula obținem:

$$\Pr(C = c_j | A_1 = a_1, \dots, A_k = a_k) = \frac{\Pr(c_j) * \prod_{i=1}^k \Pr(A_i = a_i | C = c_j)}{\sum_{k=1}^n \Pr(C = c_k) * \prod_{i=1}^k \Pr(A_i = a_i | C = c_k)}$$

□ Toate probabilitățile din expresia de mai sus pot fi obținute prin numărare!

Calcularea costului înlocuitor la un nod

58

Construirea modelului

- Când este necesară numai clasificare, numitorul expresiei de mai sus poate fi ignorat (este același pentru toate clasele c_j), iar clasa de etichetare se obține prin maximizarea numărătorului:

$$C = \underset{j}{\operatorname{argmax}} \Pr(c_j) * \prod_{i=1}^k \Pr(A_i = a_i | C = c_j)$$

Calcularea costului înlocuitor la un nod

59

Exemplu

- Fie versiunea simplificată a tabelului PlayTennis:

Outlook	Wind	Play Tennis
Overcast	Weak	Yes
Overcast	Strong	Yes
Overcast	Absent	No
Sunny	Weak	Yes
Sunny	Strong	No
Rain	Strong	No
Rain	Weak	No
Rain	Absent	Yes

Calcularea costului înlocuitor la un nod

60

Caz special: valori non-categorice sau absente

Valori non-categorice sau valori absente:

- Toate attributele care nu sunt categorice trebuie discretizate (înlătură cu unele categorice).
- De asemenea, dacă unele attribute au valori lipsă, aceste valori sunt ignorate.

Calcularea costului înlocuitor la un nod

61

Exemplu

- Pr(Yes) = 4/8 Pr(No) = 4/8
- Pr(Overcast | C = Yes) = 2/4 Pr(Weak | C = Yes) = 2/4
- Pr(Overcast | C = No) = 1/4 Pr(Weak | C = No) = 1/4
- Pr(Sunny | C = Yes) = 1/4 Pr(Strong | C = Yes) = 1/4
- Pr(Sunny | C = No) = 1/4 Pr(Strong | C = No) = 2/4
- Pr(Rain | C = Yes) = 1/4 Pr(Absent | C = Yes) = 1/4
- Pr(Rain | C = No) = 2/4 Pr(Absent | C = No) = 1/4
- Calculăm pentru exemplul de test: Sunny Absent

Calcularea costului înlocuitor la un nod

61

Exemplu

For $C = \text{Yes}$

$$\Pr(\text{Yes}) * \Pr(\text{Sunny} | \text{Yes}) * \Pr(\text{Absent} | \text{Yes}) = \frac{4}{8} * \frac{1}{4} * \frac{1}{4} * \frac{1}{32}$$

For $C = \text{No}$

$$\Pr(\text{No}) * \Pr(\text{Sunny} | \text{No}) * \Pr(\text{Absent} | \text{No}) = \frac{4}{8} * \frac{1}{4} * \frac{1}{4} * \frac{1}{32}$$

□ Rezultatul este că ambele probabilități au aceeași valoare și clasa poate fi oricără din ele.

Calcularea costului înlocuitor la un nod

62

SVM: Generalități

- Este prezentată doar ideea generală a metodei de clasificare Support Vector Machines (SVM).

□ SVM-urile sunt descrise în detaliu în multe documentații și cărți, de exemplu [Liu 11] sau [Han, Kamber 06].

□ Metoda a fost descoperită în Uniunea Sovietică în anii '70 de Vladimir Vapnik și a fost dezvoltată în SUA după ce Vapnik s-a alăturat AT&T Bell Labs la începutul anilor '90 (a se vedea [Cortes, Vapnik 95]).

Calcularea costului înlocuitor la un nod

63

SVM: Model

- Considerăm o mulțime de antrenare:

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$$

unde:

➢ $x = (x_1, x_2, \dots, x_n)$ este un vector din \mathbb{R}^n (toate componentele lui x sunt numere reale)

➢ y este eticheta de clasă, cu $y \in \{-1, +1\}$. Dacă x este etichetat cu $+1$ aparține clasei pozitive, altfel aparține clasei negative (-1).

Calcularea costului înlocuitor la un nod

64

Caz special: împărțire la 0

- Expressia modificată este:

$$Pr(A_i = a_i | C = c_j) = \frac{a_i + b}{b + s + r}$$

unde:

➢ $s = 1 / \text{Număr de exemple din setul de antrenare}$

➢ $r = \text{Numărul valorilor distincte pentru } A_i$

□ În acest caz toți termenii expresiei sunt mai mari decât zero.

Calcularea costului înlocuitor la un nod

65

SVM: Model

- Un posibil clasificator este o funcție liniară:

$$f(x) = w \cdot x + b = 0$$

Asigură incă:

$$y_i = \begin{cases} 1 & \text{if } w \cdot x_i + b \geq 0 \\ -1 & \text{if } w \cdot x_i + b < 0 \end{cases}$$

unde:

➢ w este un vector de ponderi (weight vector),

➢ $w \cdot x$ este produsul scalar între vectorii w și x ,

➢ b este un număr real, și

➢ w și b pot fi scalari, după cum se va vedea.

Calcularea costului înlocuitor la un nod

66

SVM: Model

- Semnificația lui f ca hiperplanul

$$< w \cdot x > + b = 0$$

separă punctele setului de antrenare în două:

➢ o jumătate din spațiu conține valori pozitive și

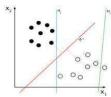
➢ cealaltă jumătate valori negative din D (precum hiperplanul H1 și H2 din figura următoare).

□ Toate exemplele de test pot fi acum clasificate folosind f : valoarea lui f dă eticheta pentru exemplul.

Calcularea costului înlocuitor la un nod

67

Figura 1



Sursa: Wikipedia

[Functii kernel: Ce este și cum se utilizează]

Cel mai bun hiperplan

- ❑ SVM încercă să găsească „cel mai bun” hiperplan de acea formă.
- ❑ Teoria arată că cel mai bun plan este cel care maximizează aşa-numita “margină” (distanță ortogonală minimă între un punct pozitiv și negativ din setul de antrenament - a se vedea figura următoare pentru un exemplu).

73

Functii kernel

- ❑ Dar cum putem găsi această funcție de mapare?
- ❑ În soluționarea problemei de optimizare pentru găsirea hiperplanului de separare liniară în nouă spațiu F toți termenii sunt doar de forma: $\phi(X) \cdot \phi(Y)$.
- ❑ Înlocuind acest produs scalar cu o funcție atât în X_1 cât și în X_2 , nevoia de a găsi ϕ dispără. O astfel de funcție se numește **funcție kernel**: $K(X, X_j) = \phi(X) \cdot \phi(X_j)$
- ❑ Pentru un hiperplanul de separare în F trebuie doar să înlocuim toate produsele scalare cu funcția kernel alesă și apoi să continuăm cu problema de optimizare ca în cazul separabil.

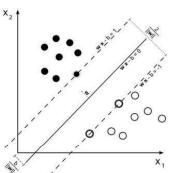
74

Functii kernel

- ❑ Unele dintre cele mai utilizate funcții kernel sunt:
- ❑ Kernel liniar: $K(X, Y) = \langle X \cdot Y \rangle + b$
- ❑ Kernel polinomial: $K(X, Y) = (a * \langle X \cdot Y \rangle + b)^p$
- ❑ Kernel sigmoid: $K(X, Y) = \tanh(a * \langle X \cdot Y \rangle + b)$

80

Figura 2



Sursa: Wikipedia

Separarea non-liniară

- ❑ În multe situații nu există un hiperplan care separă exemplile pozitive de cele negative.
- ❑ În astfel de cazuri este posibilă maparea punctelor din setul de antrenare (exemplul din D) într-un alt spațiu dimensional superior.
- ❑ Aici punctele pot fi liniar separabile.
- ❑ Funcția de mapare primește exemple (vectori) din spațul de intrare X și le mapează în spațiu caracteristic (feature space) F :

$$\phi: X \rightarrow F$$

75

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

Discuție SVM

- ❑ SVM se folosește când atributele au valori reale
- Când în setul de antrenare există attribute categorice, este necesară o conversie la valori reale.
- ❑ Când sunt necesare mai mult de două clase, SVM poate fi utilizat recursiv.
- Prima utilizare separă clasa 1 de clasa 2 și asta mai departe. Pentru N clase sunt necesare ruje N-1.
- ❑ SVM sunt o metodă foarte bună în clasificarea datelor hiper-dimensionale.

81

Cuprins

- ❑ Ce este învățarea supervizată
- ❑ Evaluare clasificatori
- ❑ Arbori de decizie: ID3 și C4.5
- ❑ Clasificatori bayesieni (Naive Bayes)
- ❑ Mașini cu vectori suport (Support vector machines)
- ❑ K-nearest neighbors
- ❑ Metode asambliste: Bagging, Boosting, Random Forest

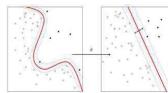
82

Separarea non-liniară

❑ Fiecare punct x din D este mapat în $\phi(X)$. După maparea tuturor punctelor avem alt set de antrenare conținând vectori din F și nu din X , cu: $\dim(F) \geq n = \dim(X)$:
 $D = \{(\phi(X_1), y_1), (\phi(X_2), y_2), \dots, (\phi(X_n), y_n)\}$

❑ Pentru un ϕ corespunzător aceste puncte sunt liniar separabile,

Figura 3



76

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

kNN

- ❑ Metoda “Ceai mai apropiat k vecinii” - K-nearest neighbors (kNN) nu produce un model, ci este o metodă simplă pentru determinarea clasei unui exemplu bazat pe etichetele vecinilor săi aparținând setului de antrenare.
- ❑ Pentru rularea algoritmului este necesară o funcție de distanță pentru calcularea distanței de la exemplul de test până la exemplele din setul de antrenare.

83

kNN

- ❑ O funcție $f(x, y)$ poate fi utilizată ca funcție de distanță dacă sunt îndeplinite patru condiții:

 1. $f(x, y) \geq 0$
 2. $f(x, x) = 0$
 3. $f(x, y) = f(y, x)$
 4. $f(x, y) \leq f(x, z) + f(z, y)$.

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

Algoritm

Întrare:
❑ Un set de date D care conține exemple etichetate (setul de antrenare)
❑ O funcție de distanță f pentru măsurarea distanței între două exemple
❑ Un întreg $k >$ parametrul care spune căi vecini sunt luați în considerare
❑ Un exemplu de test t

Iesire:❑ Clasa lui t **Metodă:**

❑ Se folosește f pentru a calcula distanța dintre t și fiecare punct din D
❑ Se selectează cei mai apropietăți k vecini
❑ Se alege punctul cu clasa majoritară din setul de cei mai apropietăți k vecini.

Exemplu

- ❑ $K = 3 \rightarrow$ Roșu
- ❑ $K = 5 \rightarrow$ Albastru



- ❑ kNN este foarte sensibil la valoarea parametrului k .
- ❑ Cea mai bună valoare pentru k poate fi găsită, de exemplu, prin validare încrușitată.

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

86

Bagging

În ce constă tehnica Bagging:

- ❑ Părind de la setul de date original, construim n seturi de date de antrenare prin eșantionare cu înlocuire (eșantionare bootstrap).
- ❑ Pentru fiecare set de date de antrenare, construim un clasificator folosind același algoritm de învățare (se colț în clasificatorul slab).
- ❑ Clasificatorul final se obține prin combinarea rezultatelor clasificatorilor slab (prin vot, de exemplu).
- ❑ Bagging ajută la imbunătățirea preciziei pentru algoritmi instabili de învățare: arbori de decizie, rețele neurale, etc.
- ❑ Nu ajută în cazul kNN sau Naive Bayes.

87

Boosting

- ❑ Tehnica Boosting constă în construirea unei secvențe de clasificatori slabii și sădăgăarea lor în strucția clasificatorului final puternic.
- ❑ Clasificatorii slabii sunt ponderați în funcție de acuratețe.
- ❑ De asemenea, datele sunt reevaluate după ce construim fiecare clasificator slab, astfel încât exemplurile clasificate incorrecții să crească în greutate.
- ❑ Rezultatul este că următorii clasificatori slabii din secvență se concentrează mai mult pe exemplare pe care le-au clasificat incorrecționat clasificatorii slabii precedenți.

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

88

Cuprins

- ❑ Ce este învățarea supervizată
- ❑ Evaluare clasificatori
- ❑ Arbori de decizie: ID3 și C4.5
- ❑ Clasificatori bayesieni (Naive Bayes)
- ❑ Mașini cu vectori suport (Support vector machines)
- ❑ K-nearest neighbors
- ❑ Metode asambliste: Bagging, Boosting, Random Forest

Metode asambliste

- ❑ Metodele asambliste combină mai mulți clasificatori “slabi” (weak) pentru a obține unul mai bun (mai “puternic” – strong).
- ❑ Clasificatorii combinați sunt similari (utilizează aceeași metodă de învățare), dar setările de antrenare și ponderile exemplilor din ele sunt diferite.

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

89

Random forest

- ❑ Tehnica Random forest este un clasificator asamblist format dintr-un set de arbori de decizie. Clasificatorul final produce valoarea modală a claselor de ieșire ale fiecărui arbor.
- ❑ Algoritm este următorul:

1. Alegeri T – numărul de arbori de construit.
2. Alegeri m – numărul de variabile utilizate pentru ramificare la fiecare nod, $m < M$, unde M este numărul de variabile de intrare.

90

Random forest

1. Construim T arbori. Pentru fiecare:
- Construim un set de antrenare prin eșantionare cu înlocuire. Din el se crează un arbor de decizie.
- La fiecare nod, selecțiem în variabile la întărirea (attribute) și le folosim pentru a găsi cea mai bună ramificare.
- Crescem arborul la maxim. Nu există tăieri (pruning).
4. Se prezic rezultatele agregând predicțiile arborilor (de ex., medie pentru clasificare, medie pentru regresie).

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

94

Bagging

❑ Numele Bagging vine de la Bootstrap Aggregating.
❑ Așa cum am văzut mai devreme metoda bootstrap face parte din metodele de eșantionare și constă în obținerea unui set de antrenare din datele etichetate inițiale prin eșantionare cu înlocuire.

Exemplu

Setul original	a	b	c	d	e	f
Set antrenare 1	a	b	b	c	e	f
Set antrenare 2	b	b	c	c	d	e
Set antrenare 3	a	b	c	c	d	f

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

95

Referințe

- Liu [1] Bing Liu, 2011, *Web Data Mining: Exploring Hypertext, Contents, and Usage Data*, Springer, chapter 3.
- Han, Kamber [6] Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques*, Second Edition, Morgan Kaufmann Publishers, 2006.
- Cortes, Vapnik [5] Cortes, Corinna, and Vapnik, Vladimir N., "Support-Vector Networks", *Machine Learning*, 20(3):273-297, 1995, <http://dx.doi.org/10.1007/BF00050937>
- Wikipedia [4] Wikipedia, <http://en.wikipedia.org/>
- Sanderson [8] Robert Sanderson, 2008, *Data mining course notes*, Dept. of Computer Science, University of Liverpool 2008, *Classification: Evaluation*, <http://www.cs.liv.ac.uk/~azeroth/courses/current/comps227/lectures/comp227-13.pdf>
- Quinlan [8] Quinlan, J. R. 1986, *Induction of Decision Trees*, *Machine Learn.*, 1, 1 (Mar. 1986), 81–106, <http://www.cs.cmu.edu/~ml/mlcourse/mlcourse.html#quinlan.pdf>

96

[Functii kernel: Ce este și cum se utilizează]

[Functii kernel: Ce este și cum se utilizează]

97

Capitolul 11

Data mining – clustering

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

1

Problema

- Dându-se puncte într-un spațiu oarecare – deseori un spațiu cu foarte multe dimensiuni – grupează punctele într-un număr mic de *cluster*, fiecare cluster constând din puncte care sunt "apropiate" într-un anume sens.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

2

Exemple de aplicatie

- Documentele pot fi percepute ca puncte într-un spațiu multi-dimensional în care fiecare dimensiune corespunde unui cuvânt posibil.
- Positia documentului într-o dimensiune este data de numărul de ori care cuvântul apare în document (sau doar 1 dacă apare, 0 dacă nu).
- Clusterile de documente în acest spațiu corespund deseori cu grupările de documente din același domeniu.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Distanta

- Pentru a discuta dacă o mulțime de puncte sunt suficiente de apropiate pentru a fi considerate un cluster avem nevoie de o *măsură a distanței* $D(x, y)$ care spune cât de departe sunt punctele x și y .
- Nu orice funcție poate fi utilizată ca funcție de măsurare a distanței.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

8

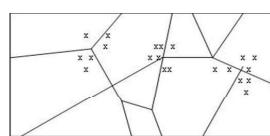
Exemple de aplicatie

- Cu mulți ani în urmă, în timpul unei izbucniri a holerei în Londra, un medic a marcat localizarea cazurilor pe o hartă, obținând un desen care arăta ca în figura urmatoare:

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

3

Exemple de aplicatie



F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

4

Distanta

- Axiomile uzuale pentru o măsură a distanței D sunt următoarele:
 - $D(x, y) \geq 0$
 - $D(x, x) = 0$. Un punct este la distanță 0 de el însuși.
 - $D(x, y) = D(y, x)$. Distanța e simetrică.
 - $D(x, y) \leq D(x, z) + D(z, y)$.

Inegalitatea triunghiului.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Exemple de aplicatie

- Cu multă așteptare, datele au indicat că aparițiile cazurilor se grupează în jurul unor intersecții, unde existau puturi infestate, arătând nu numai cauză holerică ci indicând și ce e de făcut pentru rezolvarea problemei.
- Din păcate nu toate problemele de data mining sunt atât de simple, deseori deoarece clusterurile sunt în atât de multe dimensiuni încât vizualizarea este foarte dificilă.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

5

Exemple de aplicatie

- SkyCat* a grupat în cluster 2×10^9 obiecte cerești în steluță, galaxii, quasari, etc.
- Fiecare obiect era un punct într-un spațiu cu 7 dimensiuni, unde fiecare dimensiune reprezintă nivelul radiației într-o bandă a spectrului.
- Proiectul Sloan Sky Survey este o încercare mult mai ambitioasă de a cataloga și grupa înregul univers vizibil.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

6

Distanta comuna

- Distanța comună (sau norma L2) este data de formula cunoscută:

$$\sqrt{\sum_{i=1}^k (x_i - y_i)^2}$$

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Distanta Manhattan

- Distanța Manhattan (sau norma L1) este data de formula următoare:

$$\sum_{i=1}^k |x_i - y_i|$$

- Ea poate fi folosită de exemplu si pentru calculul distanței între două puncte pe o placă cu circuite imprimate multistrat.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

10

Maximul pe o dimensiune

- Este data de formula:

$$\max_{i=1}^k |x_i - y_i|$$

- Aceasta funcție verifică toate cele 4 condiții pentru a fi funcție de distanță.
- Poate fi folosită de exemplu pentru spații euclidiene hiperdimensionale (număr de dimensiuni foarte mare)

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

13

Alte distante

- Unde nu există un spațiu euclidian în care să plăsăm punctele gruparea devine mult mai dificilă.
- Iată un exemplu în care are sens: o măsură a distanței în lipsa unui spațiu euclidian

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

14

Hiperdimensionalitatea

- Cu toate acestea, să presupunem k foarte mari, să zicem 100.000. Indiferent de folosirea L_2 , L_1 , sau L_∞ , suntem că:
 $D(x, y) \geq \max_i |x_i - y_i|$ pentru $x = [x_1, x_2, \dots]$ și $y = [y_1, y_2, \dots]$.
- Pentru k foarte mare, e foarte posibil să existe o dimensiune astfel încât x și y sunt diferențe aproape de maximul posibil, chiar dacă x și y sunt foarte apropiate în alte dimensiuni.
- Astfel $D(x, y)$ va fi foarte apropiată de 1.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Hiperdimensionalitatea

- O altă consecință interesantă a hiper-dimensionalității este că toți vectorii, $x = [x_1, x_2, \dots]$ și $y = [y_1, y_2, \dots]$ sunt aproape ortogonali.
- Motivul este că dacă proiecțiem x și y pe oricare dintre cele C_k plane formate de două dintre cele k axe va exista unu în care proiecțiile vectorilor sunt aproape ortogonale (probabilitatea sa existe crește cu numărul de dimensiuni).

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

20

Alte distante

- Șirurile de caractere, cum sunt secvențele ADN, pot fi similară chiar și dacă există unele inserări și ștergeri precum și modificări ale unor caractere.
- De exemplu, *abcde* și *bcdxye* sunt destul de similară chiar dacă nu au nici o poziție comună și nu au nici chiar aceeași lungime.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

15

Alte distante

- Astfel, în loc să construim un spațiu euclidian cu câte o dimensiune pentru fiecare poziție, putem defini funcția distanță:
$$D(x, y) = |x_1 - y_1| + |x_2 - y_2| + \dots + |x_n - y_n|$$
 unde LCS este cea mai lungă subsecvență comună lui x și y .
- În exemplul nostru *LCS(abcede, bcdxye)* este *bcd* de lungime 4, deci $D(abcede, bcdxye) = 5 + 6 - 2 \times 4 = 3$; i.e., șirurile sunt destul de apropiate.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Abordări clustering

- La nivel final, putem împărti algoritmii de grupare în două mari clase:
- 1. Abordarea tip centroid: "ghicim" centroizii sau punctele centrale pentru fiecare cluster și asignăm punctele la clusterul având cel mai apropiat centroid.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Abordări clustering

- Abordarea ieherarhică:
 - Începem prin a considera că fiecare punct formează un cluster.
 - Comasăm repetat clusterurile apropiate de unele măsuri pentru apropierea a două clusterelor (e.g., distanța dintre centroizii lor), sau pentru cătă de bun va fi rezultatul (e.g., distanța medie de la punctele din cluster la nouul centroid rezultat).

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

22

Alte distante

- Aceasta funcție de distanță arată cătă caractere trebuie stersă sau adăugată unuia dintre șiruri pentru a obține celalalt șir.
- Intr-adevar, pentru a obține de exemplu pe *abcede* din *bcdxye* trebuie să:
 1. Adaugam un *a* în prima poziție
 2. Stergem *x* de la poziția 4
 3. Stergem *y* de la poziția 5

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

17

Hiperdimensionalitatea

- O consecință mai puțin intuitivă a lucrului în spații hiperdimensionale este că aproape toate percherile de puncte sunt la o depărtare aproape egală cu media distanțelor între puncte.
- Exemplu:
 - Să presupunem că aruncăm aleator puncte într-un cub k -dimensional.
 - Pentru $k=2$, ne astăptăm ca punctele să fie răspândite în plan cu unele foarte apropiate între ele și unele percheri aproape la distanță maximă posibilă.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Algoritmul k-means

- Acest algoritm este un algoritm popular care *ține datele în memoria centrală*
- Pentru acest algoritm se bazează pe altăl algoritmi de clustering (ex.: BFR)
- k*-means alege aleator k centroizi de cluster dintre punctele care se grupează și asignează celelalte puncte la această alegând centroidul cel mai apropiat de punctul respectiv.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

Observație

- Dupa asignarea tuturor punctelor se recalculează centroizii clusterelor ca centrul de greutate al punctelor din fiecare cluster.
- El nu este de obicei unul din punctele care formează clusterul ci un punct intre ele în spațiu euclidian respectiv.
- Procesul de asignare puncte – recalculare centroizi se repeta pana se indeplinește o conditie de oprire.

F. Radulescu, Curs: Utilizarea bazelor de date, anul IV CS.

24

Observatie

- Conceptul de centroid nu este valabil decat in cazul clusteringului in spatiu euclidian.
- Pentru cazurile in care nu avem un spatiu euclidian (punctele nu au coordonate numerice) exista o serie de variante ale algoritmului precum k-medoids sau k-modes.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

25

Exemplu

- Un exemplu foarte simplu cu cinci puncte in două dimensiuni.
- Consideram pentru k valoarea 2 (k este un parametru de intrare in algoritm).
- Presupunem ca alegem punctele 1, 2 ca centroi initiali.
- Aaignam punctele 3, 4 si 5 la cel mai apropiat centroid.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

26

Oprire k-means

- Criteriile de oprire - continuare:
- 4. Scădere SSD (suma pătratelor distanțelor) este sub un prag dat.
- SSD măsoară cat de compacte sunt clusterile (ca ansamblu).
- Este suma pătratelor distanțelor de la fiecare punct la centroidul său:

$$SSD = \sum_{i=1}^k \sum_{p \in Cluster_i} Dist(p, Centroid(Cluster_i))^2$$

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

31

Aplicare k-means

- Dacă nu suntem siguri de valoarea lui k putem incerca valori diferite pentru k.
- Procesul se opreste când găsim cel mai mic astfel încât mărimea lui k nu măsoarează prea mult distanța medie a punctelor față de centroidul lor.
- Exemplul urmator ilustrează acest lucru.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

32

Exemplu

- Rezultatul este urmatorul:



F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

27

Exemplu

- Când considerăm punctul 3, presupunem ca acesta este mai apropiat de 1, deci 3 se adaugă clusterului conținând 1.
- Presupunem că atunci când asignăm 4 găsim că 4 este mai aproape de 2 decât de 1, deci 4 se alătură lui 2 în clusterul acestuia.
- În final, 5 este mai aproape de 1 decât de 2, deci el se adaugă la clusterul {1, 3}.
- Recalculăm acum centroizi și gasim noli centroizi C1 și C2.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

28

Alt exemplu

- Să considerăm datele din figura urmatoare.
- În mod clar k=3 este numărul corect de clustere dar să presupunem că întâi încercăm k=1.
- În acest caz toate punctele sunt într-un singur cluster și distanța medie la centroid va fi mare.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

33

Alt exemplu - cont

- Presupunem că apoi încercăm k=2.
- Unul dintre cele trei clustere va fi format iar celelalte două vor fi forțate să creze împreună un singur cluster, asa cum arată linia punctată.
- Distanța medie a punctelor la centroid de va măsura astfel considerabil.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

34

Aplicare k-means

- Repetam acum operația și asignam toate cele 5 puncte la cea mai apropiată centroid (dintre C1 și C2).
- Obținem aceleasi clustere și aleiasem centroizi.
- Este evident că dacă vom continua nu obținem altceva. Procesul se încheie.
- Am obținut clusterele {1, 3, 5} și {2, 4}.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

29

Oprire k-means

- Criteriile de oprire pot fi:
 1. Clusterele nu se schimbă de la o iteratie la alta (ca în exemplul anterior).
 2. Modificările clusterelor sunt sub un prag fixat (de exemplu, nu mai mult de ρ puncte schimbă clusterul între două iterări succesoare).
 3. Mișcarea centroizoilor este sub un prag dat (de exemplu, suma distanțelor dintre pozițiile vechi și cele noi pentru centroizi nu depășește între două iterări succesoive un prag fixat).

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

30

Alt exemplu - cont

- Dacă luăm k=3 atunci fiecare dintre clusterele vizibile va forma un cluster iar distanța medie de la puncte la centroizi se va măsura din nou, aşa cum arată graficul din figura.
- Totuși, dacă mărim k la 4 unul dintre adevaratele clustere va fi particionat artificial în două clustere apropiate, aşa cum arată liniile continui.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

35

Alt exemplu - cont

- Distanța media la centroid va scădea puțin dar nu mult.
- Acest eșec de a merge mai departe ne arată că valoarea k=3 este corectă chiar dacă datele sunt în atât de multe dimensiuni încât nu putem vizualiza clusterele.
- În acest fel aflăm valoarea corecta a lui k – numărul de clustere

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

36

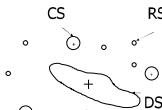
Algoritmul BFR

- Bazat pe k-means, acest algoritm citește datele o singură dată în transe egale cu memoria centrală disponibilă la fiecare pas.
- Algoritmul lucrează cel mai bine dacă clusterele sunt normal distribuite în jurul unui punct central, eventual cu o deviație standard diferită în fiecare dimensiune.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

37

Reprezentare clustere în BFR



Cei care au creat acest algoritm și au reprezentat clusterele ca pe niște galaxii.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

38

Reprezentare - cont

- De notat că aceste trei tipuri de informație, totalizând în cazul în care avem k dimensiuni $2k+1$ numere sunt suficiente pentru a calcula statistici importante pentru un cluster sau subcluster.
- Este mai convenabil de menținut pe măsură ce punctele sunt adăugate la cluster decât, să spunem, media și varianța în fiecare dimensiune,

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

43

Reprezentare - cont

- Cordonata μ_i a centroidului clusterului i în dimensiunea N este SUM_i/N
- Varianța (dispersia) în dimensiunea N este:

$$\frac{SUMSQ_i}{N} - \left(\frac{SUM_i}{N} \right)^2$$
- iar deviația standard σ este rădăcina pătrată a acesteia.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

44

Reprezentare clustere în BFR

- Un cluster constă din:
 1. Un nucleu central numit **Discard set - DS**.
 2. Multimea acestor puncte este considerată ca aparținând în mod sigur clusterului.
- Notă: deși numărul punctelor "de aruncat" acestora au de fapt un efect semnificativ pe parcursul executiei algoritmului de vreme ce determină colectiv unde este centrul și care este deviația standard a clusterului în fiecare dimensiune.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

39

Reprezentare clustere în BFR

- 2. Galaxii înconjurătoare, numite colectiv **Compression set - CS (Multimea comprimată)**.
- Fiecare subcluster din CS constă într-un grup de puncte care sunt suficiente de apropiate unele de altel încât pot fi înlocuite cu statisticile lor, la fel ca și DS.
- Totuși, ele sunt suficiente de departe de orice centroid de cluster încât nu suntem încă siguri de care cluster aparțin.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

40

Procesare

- La prima încărcare cu date a memoriei centrale, BFR selectează /comprimă/ punctele care vor fi utilizate în algoritm carească lucrările în memoria centrală, c.e.g., se ia un eşantion de puncte, se optimizează exact clusterul și se aleg centroizi lor ca centroizi inițiali.
- O memorie centrală de puncte este procesată la fel în toate încărcările cu date următoare după cum urmează:

1. Se determină care puncte sunt suficiente de apropiate de un centroid curent astfel încât pot fi luate în DS și statisticile lor ($N, SUM, SUMSQ$) combinate cu statisticile anterioare ale clusterului.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

45

Procesare

2. În memoria centrală se încearcă gruparea punctelor care nu au fost încă plasate în DS, inclusiv puncte ale RS din pașii precedenți.
- Dacă găsim un cluster de puncte a căror varianță este sub un prag ales, atunci vom privi aceste puncte ca un subcluster, le înlocuim cu statisticile lor și le considerăm parte a CS.
- Toate celelalte puncte vor fi plasate în RS.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

46

Reprezentare clustere în BFR

- Stăle individuale care nu sunt parte a nici unei galaxii sau subgalaxii, **Multimea refuzată (Retained set - RS)**.
- Aceste puncte nici nu pot fi asignate vreunui cluster și nici grupate în vreun subcluster al CS.
- Ele sunt stocate în memoria centrală ca puncte individuale împreună cu statisticile pentru DS și RS.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

41

Reprezentare comprimată

- Statisticile utilizate pentru a reprezenta fiecare cluster din DS și fiecare subcluster din CS sunt:
 1. Conținutul numărului de puncte N .
 2. Vectorul sumelor coordonatelor punctelor în fiecare dimensiune. Vectorul este notat cu SUM iar componenta pentru dimensiunea i cu SUM_i .
 3. Vectorul sumelor pătratelor coordonatelor punctelor în fiecare dimensiune notat cu $SUMSQ$. Componenta pentru dimensiunea i cu $SUMSQ_i$.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

42

Procesare

- Luăm în considerare unirea unui subcluster nou apărut cu un subcluster anterior din CS.
- Testul pentru a vedea dacă este de dorit să facem asta este că multimea combinată de puncte să albă o varianță sub un anumit prag.
- De notat că statisticile tijunte pentru subclusterurile din CS sunt suficiente pentru a calcula varianța mulțimii combinante.

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

47

Procesare

- Dacă este ultimul pas, i.e. nu mai sunt date, atunci vom asigna subclusterelor din CS și punctele din RS la cel mai apropiat cluster de ele chiar dacă ele vor fi destul de departe de orice centroid de cluster.
- În felul acesta obținem distrelere finale produse de algoritm

F. Radulescu, Curs: Utilizarea bazeelor de date, anul IV CS.

48

Scalarea multidimensională

- În multe cazuri nu avem un spațiu euclidian ci doar o mulțime de puncte și distanța între oricare două dintre acestea.
- Exemplu: un graf în care cunoaștem lungimea fiecarui arc. Din aceste lungimi putem afla distanța între oricare două noduri ca fiind lungimea drumului minim între ele

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

49

Scalarea multidimensională

- Se poate demonstra că având N puncte și distanțele între oricare 2 dintre ele putem crea un spațiu cu $N-1$ dimensiuni
- În acest spațiu punctele sunt plasate exact (distanța calculată din coordonatele după plasare este aceeași cu distanța de la care s-a plecat pentru orice percheie de puncte)

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

50

Fastmap

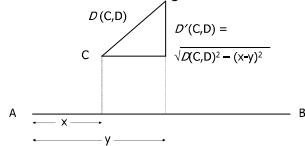
1. Se aleg două puncte aflate la distanță cat mai mare, a și b . Acestea devin o axă de coordonate (cu originea în a).
2. Pentru orice punct c din cele N se calculează coordonata pe această axă conform formulei anterioare:

$$x = (D^2(a, c) + D^2(a, b) - D^2(b, c)) / (2 * D(a, b))$$

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

55

Fastmap



Scalarea multidimensională

- Problema este că în cazul unui număr mare de puncte rezultă un număr mare de dimensiuni (spațiu hiperdimensional).
- Ideal ar fi să plasăm cat mai exact cele N puncte într-un spațiu având K dimensiuni unde $K << N$.
- Acum procesul se numește scalare multidimensională.
- Plasarea celor N puncte nu este 100% exactă (distanțele calculate din coordonatele rezultante nu sunt total exacte cu cele de la care s-a pornit)

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

51

Scalarea multidimensională

- Formula de bază folosită este cea prin care având două puncte putem afla distanțele proiecției unui al treilea punct pe segmentul format de primele două puncte,
- Formula este obținută din teorema lui Pitagora generalizată (teorema cosinusului)

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

52

Fastmap

3. Pentru următoarele axe se vor folosi nu distanțele initiale dintre puncte ci distanțe reportate în modul următor:

$$D'^2 = D^2 - (x - y)^2$$

4. Procesul se sfârșește după calculul numărului dorit de coordonate sau când nu mai pot fi alese noi axe de coordonate.

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

57

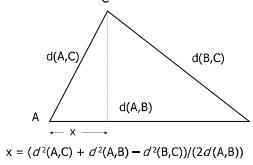
Probleme în cazuri reale

- În cazurile reale (matricea nu este euclidiană) se poate întâmpla că patratul lui D' calculat cu formula anterioară să dea un număr negativ.
- În astfel de cazuri pentru a putea continua se poate lua $D' = 0$.
- Aceasta alegeră duce însă la erori care se propagă.
- Iată un exemplu de rulare pentru algoritm în cazul în care sunt considerate 2000 de noduri,

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

58

Proiecția lui C pe AB



53

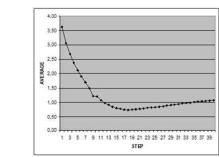
Fastmap

- Algoritmul Fastmap este unul dintre algoritmii de scalare multidimensionale.
- Aceasta este un algoritm prin care se calculează succesiv coordonatele punctelor, ceea ce corespondă (o dimensiune) la fiecare pas.
- Pasul algoritmului este următorul:

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

54

Probleme în cazuri reale



F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

59

Probleme în cazuri reale

- Figura arată media diferențială între dreal și dcălculat unde:
- Dreala este distanța între puncte de la care s-a pornit (cunoscută prin ipoteza problemei)
- Dcălculat este distanța dintre puncte calculată pe baza coordonatelor obținute pana la pasul respectiv.
- Se observă că există un minim după 18 pași (spațiu optim are deci pentru acest exemplu 18 dimensiuni, $k=18$)

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

60

Probleme în cazuri reale

- În mod normal graficul ar trebui să tindă asimptotic către 0.
- Faptul că nu se întâmplă asa e datorat erorilor induse de considerarea lui $D' = 0$ în cazul în care patratul este negativ.
- Erorile acumulate fac ca după al 18-lea pas graficul să înceapă să crească.

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

61

Bibliografie

- J.D.Ullman - CS345 --- Lecture Notes, Clustering I, II
<http://info.stanford.edu/~ullman/cs345notes.html>

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

62

Sfârșitul capitoului 10

F.Radulescu, Curs: Ullman's notes de date, anii IV-CS.

63

Capitolul 12

Data mining – căutare pe web

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

1

Căutarea pe web

- ❑ Puncte importante :
 1. **Rangul paginii**, pentru descoperirea celor mai "importeante" pagini de web, utilizat de Google.
 2. **Indeci și autorități**, o evaluare mai detaliată a importanței paginilor web utilizând o varianta a calculului de valori proprii utilizată pentru rangul paginii.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

2

Exemplul 1

- ❑ Fie $[n, m, a]$ vectorul importanței pentru cele trei pagini : Netscape, Microsoft respectiv Amazon.
- ❑ Atunci ecuația care descrie valorile asymptotice ale acestor trei variabile este :

$$\begin{bmatrix} n \\ m \\ a \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \\ 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} n \\ m \\ a \end{bmatrix}$$

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

3

Rangul paginii

- ❑ Intuitiv rezolvăm problema definiției "importeantei" recursiv : o pagină este importantă dacă pagini importante conțin legături către ea.
- ❑ Creăm o matrice stochastică a Internetului astfel :
 - > Fiecare pagină / corespunde liniei și coloanei / a matricii.
 - > Dacă pagina j are n succesiuni (legături), atunci elementul j, j al matricii este $1/n$ dacă pagina / este unul dintre acești succesiuri ai paginii j și 0 altfel.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

4

Rangul paginii

- ❑ Intuiția care stă în spatele acestor matrici este :
- ❑ Să ne imaginăm că inițial fiecare pagină are o unitate de importanță.
- ❑ La fiecare pas fiecare pagină își împarte importanța între succesișorii săi și primește noi fracțiuni de importanță de la predecesorii săi.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

5

Exemplul 1

- ❑ Primele patru iterări dă următoarele estimări :

$$\begin{aligned} n &= \begin{bmatrix} 1 & 1 & 5/4 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{bmatrix} \\ m &= \begin{bmatrix} 1 & 1/2 & 3/4 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{bmatrix} \\ a &= \begin{bmatrix} 1 & 3/2 & 1 \\ 0 & 1/8 & 17/16 \\ 1/2 & 0 & 0 \end{bmatrix} \end{aligned}$$

❑ La limită, soluția este $n = a = 6/5 ; m = 3/5$.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

6

Observații

- ❑ De notat că nu putem să obținem niciodată valorile absolute ale lui n, m și a ci **doar raportul lor**, de vreme ce asertjunea inițială că fiecare a pornit de la 1 a fost arbitrară.
- ❑ Deoarece matricea este stochastică (suma pe fiecare coloană este 1), procesul de **relaxare** de mai sus converge către **vectorul principal de valori proprii** al matricii.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

10

Rangul paginii

- ❑ După mai multe iterări, importanța fiecărui pagină atinge o limită care este compoziția corespunzătoare ei din vectorul principal de valori proprii al matricii.
- ❑ Această importanță este de asemenea probabilitatea ca un navigator pe web, pornind de la o pagină aleatorie și urmând legături aleator alese din fiecare pagină, să ajungă la pagina în discuție după o lungă serie de legături.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

5

Exemplul 1

- ❑ În 1839 Internetul constă din doar trei pagini : Netscape, Amazon și Microsoft. Legăturile între aceste trei pagini erau ca în figura următoare:



F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

6

Probleme cu grafuri reale

- ❑ 2 tipuri de probleme:
 1. **Dead end** : o pagină care nu are succesiuri nu are către cine să-și trimită importanță. În final totă importanța "se va scurge" din Internet.
 2. **Capcane** : un grup de una sau mai multe pagini care nu au legături către pagini din afara grupului vor acumula la final totă importanță din Internet.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

11

Exemplul 2: Dead end

- ❑ Să presupunem că Microsoft încercă să profite că este un monopol înăsturând toate legăturile din situl său. Noul Internet este ca în figura următoare:



F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

12

Exemplul 2

- ❑ Ecuatia matricială este în acest caz:

$$\begin{bmatrix} n \\ m \\ a \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 \\ 1/2 & 0 & 0 \end{bmatrix} * \begin{bmatrix} n \\ m \\ a \end{bmatrix}$$

❑ Se observă că suma pe coloane nu mai este intotdeauna 1 (coloane cu suma nula)

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

13

Exemplul 2

- ❑ Primii patru pași ai soluției iterative sunt :

$$\begin{aligned} n &= \begin{bmatrix} 1 & 1 & 3/4 \\ 0 & 1 & 1/4 \\ 1/2 & 0 & 0 \end{bmatrix} \\ m &= \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1 & 1/4 \\ 1/2 & 0 & 0 \end{bmatrix} \\ a &= \begin{bmatrix} 1 & 1/2 & 1/2 \\ 0 & 1/8 & 3/16 \\ 1/2 & 0 & 0 \end{bmatrix} \end{aligned}$$

❑ În acest caz, fiecare dintre n, m și a ține catre 0, i.e. toată importanță se scurge afară.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

14

Prevenire dead end și capcane

- ❑ Ecuatia cu taxare:

$$\begin{bmatrix} n \\ m \\ a \end{bmatrix} = 0.8 \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} n \\ m \\ a \end{bmatrix} + \begin{bmatrix} 0.2 \\ 0.2 \\ 0.2 \end{bmatrix}$$

❑ Soluția acestei ecuații este $n = 7/11 ; m = 21/11 ; a = 5/11$.

❑ De notat că suma celor trei valori nu este 3 dar obținem o distribuție mult mai rezonabilă a importanței decât în Exemplul 3.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

19

Spam

- ❑ "Spamming" este în acest context încercarea multor situri web de a părea că sunt despre un subiect care atrage vizitatorii fără că într-adevăr să fie deosebit de interesant.
- ❑ Google, ca și alte motoare de căutare, încercă să potrivescă cuvintele din cererile de căutare cu cuvinte din pagini web.

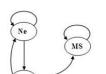
❑ Cu toate acestea, Google, spre deosebire de alte motoare de căutare, tinde să creată o pagină web facând mai greu pentru aceasta să pară că fiind despre ceva ce nu este.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

20

Exemplul 3

- ❑ Microsoft decide să nu folosească decât legături către el însuși de acum încolo. Acum, Microsoft a devenit un capcană. Noul Internet este în figura următoare:



15

Exemplul 3

- ❑ Ecuatia matricială este în acest caz:

$$\begin{bmatrix} n \\ m \\ a \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 0 & 1 & 1/2 \\ 1/2 & 0 & 0 \end{bmatrix} * \begin{bmatrix} n \\ m \\ a \end{bmatrix}$$

❑ Suma pe coloane este 1 dar apar valori de 1 pe diagonala principala a matricii.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

16

Spam

- ❑ Utilizarea rangului paginii pentru a măsura importanța în locul unei măsuri mult mai naivă ca "numărul de legături către acea pagină" protejează de asemenea împotriva spamului.

❑ Măsura naivă poate fi insățată de un spammer care creează 1000 de pagini care se referă între ele în timp ce rangul paginii recunoaște că nici una dintre acestea nu are importanță reală.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

21

Indeci și autorități

- ❑ Intuitiv, definim "index" și "autoritate" într-un mod mutual recursiv: un index conține legături către multe autorități iar o autoritate este referită de mulți indeci.

❑ Autoritățile pot fi pagini care oferă informații despre un subiect.

❑ Indecii sunt pagini care nu furnizează informații ci spun unde se găsesc informații.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

22

Exemplul 3

- ❑ Primii pași ai algoritmului produc valori:

$$\begin{aligned} n &= \begin{bmatrix} 1 & 1 & 3/4 \\ 0 & 1 & 1/4 \\ 1/2 & 0 & 0 \end{bmatrix} \\ m &= \begin{bmatrix} 1 & 3/2 & 7/4 \\ 0 & 1 & 3/4 \\ 1/2 & 0 & 0 \end{bmatrix} \\ a &= \begin{bmatrix} 1 & 1/2 & 1/2 \\ 0 & 3/8 & 5/16 \\ 1/2 & 0 & 0 \end{bmatrix} \end{aligned}$$

❑ Se observă acumularea pe linia m.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

17

Prevenire dead end și capcane

- ❑ În loc de a aplica matricea direct, "taxăm" fiecare pagină cu o fracțiune din importanța sa curentă și distribuim importanța taxată în mod egal tuturor paginilor.
- ❑ Dacă folosim o taxă de 20% ecuația din exemplul 3 devine cea de pe transparentul urmator.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

18

Indeci și autorități

- ❑ Utilizează o formalizare matricială similară cu cea de la rangul paginii dar fără restricția stochastică. Numărăm fiecare legătură ca 1, indiferent de cătă succesorii sau predecesorii are o pagină.
- ❑ Aplicarea repetată a matricii duce la divergență, dar putem introduce un factor de scalare pentru a ţine valourile calculate pentru gradul de "autoritate" sau de "indexare" pentru fiecare pagină între limite finite.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

23

Indeci și autorități

- ❑ Definim matricea A ale cărei linii și coloane corespund paginilor web având elementul $A_{ij} = 1$ dacă pagina i referă pagina j și 0 altfel.
- ❑ De notat că A^T , transpusa lui A , arată că matricea utilizată pentru calculul rangului paginilor din A^T are 1 acolo unde matricea pentru rang are fracții.

F.Rădulescu, Curs Utilizarea bazei de date, anul IV CS.

24

Indecsi si autoritati

- Fie a si h doi vectori iar componenta lor corespunde gradului de autoritate respectiv indexare a paginii. Fie λ si μ factorii de scalare corespunzători care vor fi calculați mai târziu. Atunci putem afirma că:
- $h = \lambda A a$. Adică gradul de indexare al fiecărei pagini este suma gradelor de autoritate ale tuturor paginilor referente, scalată cu λ .
- $a = \mu A^T h$. Adică gradul de autoritate al fiecărei pagini este suma gradelor de indexare ale tuturor paginilor care o referă, scalată cu μ .

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

25

Indecsi si autoritati

- Din ecuațiile (1) și (2) putem deduce folosind substituția, două ecuații care leagă vectorii a și h doar de ei însăși:
- $a = \lambda A^T A a$
- $h = \lambda \mu A^T h$
- Ca urmare, putem calcula h și a prin relaxare, obținând vectorul principal de valori proprii al matricilor $A A^T$ și respectiv $A^T A$

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

26

Extragerea de cunoștințe din web

- Puncte importante:
- 1. **Numărarea dinamică a mulțimilor de articole:** Căutarea de mulțimi interesante de articole într-un spațiu mult prea mare pentru a se putea lăua în considerare fiecare pereche de articole.
- 2. **"Carti și autori":** Intrigantul experiment al lui Sergey Brin de extragere de date relaționale din web.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

31

Numarare dinamica

- Problema este de a găsi mulțimi de cuvinte care apar "neobișnuit de des" împreună pe web, e.g. "New" și "York" sau "Ducesa, de, York".
- "Neobișnuit de des" poate fi definit în diverse moduri pentru a încorpora ideea că numărul de documente web conținând mulțimea de cuvinte este mult mai mare decât cel așteptat în cazul în care cuvintele ar fi fost alese la întâmplare, fiecare cuvânt cu probabilitatea sa de apariție într-un document.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

32

Exemplul 4

- Fie graful următor:



F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

27

Exemplul 4

- Matricele sunt:

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \quad AA^T = \begin{bmatrix} 3 & 1 & 2 \\ 0 & 0 & 1 \\ 2 & 0 & 2 \end{bmatrix}, \quad A^T A = \begin{bmatrix} 2 & 2 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 2 \end{bmatrix}$$

- Dacă utilizăm $\lambda = \mu = 1$ și considerăm că vectorii h = $[h_n, h_m, h_a]$ și a = $[a_n, a_m, a_a]$ sunt inițial fiecare $[1, 1, 1]$, primele trei iterări ale ecuațiilor pentru a și h sunt:

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

28

Numarare dinamica

- Un mod adecvat este **entropa per cuvânt din mulțime**. Formal, **interesul** unei mulțimi de cuvinte S este:

$$\log \left(\frac{\prod_{w \in S} prob(w)}{|S|} \right)$$

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

33

Numarare dinamica

- De notat că împărțirea dimensiunii lui S pentru a evita "efectul Bonferroni", în care sunt atât de multe mulțimi de o dimensiune dată încât unele, din motive probabilistice, par a fi corelate,

- Exemplu: Daca a , b și c (cuvinte) apar fiecare în 1% din toate documentele și $S = \{a, b, c\}$ apar în 0,1% din documente, interesul lui S este $(\log(0,001/0,01 \times 0,01 \times 0,01))/3 = \log(2)(1000/3)$ adică aproximativ 3,3.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

34

Exemplul 4

- Secesiunea de valori pentru a este:

a_n	=	1	5	24	114
a_m	=	1	5	24	114
a_a	=	1	4	18	84

- iar pentru h :

h_n	=	1	6	28	132
h_m	=	1	2	8	36
h_a	=	1	4	20	96

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

29

Exemplul 4

- Vectorul a , **scalat corespunzător**, va converge către un vector în care:

➤ $a_n = a_m$ și

➤ fiecare dintre aceste numere este mai mare ca a_a în raportul $1 + \sqrt{3}/2$ sau aproximativ 1,36

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

30

Numarare dinamica

- Problema tehnică: interesul nu este monoton sau "închis în jos" în modul de la produse cu larg suport.

- Astăzintă putem avea mulțimi S cu o valoare mare a interesului și totuși unele sau chiar toate submulțimile sale stricte să nu fie interesante.

- Prin contrast, dacă S are suport larg, atunci toate submulțimile sale au cel puțin același suport.

- Observație: Cu mai mult de 10^6 cuvinte diferite apărând pe web nu este posibil nici măcar să considerăm toate perechile de cuvinte,

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

35

DICE

- DICE (dynamic itemset counting engine) vizitează repetat paginile web într-un mod de tip "round-robin" ("zborul prigorului/prigoriei").



F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

36

DICE

- De fiecare dată numărării aparițiile anumitor mulțimi de cuvinte și ale fiecărui cuvânt din aceste mulțimi.
- Numarul de mulțimi numărate este suficient de mic încât contorii lor încap în memoria centrală.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

37

DICE

- Din cînd în cînd, să spunem la fiecare 5000 de pagini, DICE își reconsideră mulțimile pentru care numără. Înălță acele mulțimi care au cel mai mic interes și le înlocuiește cu alte mulțimi.

- Alegera noilor mulțimi se bazează pe proprietatea numita **heavy edge** care este o observație justificată experimental că acele cuvinte care apar în mulțimi cu interes ridicat au probabilitatea mai mare să apară în alte mulțimi cu interes ridicat.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

38

Cum lucreaza

1. Se pornește de la un eşantion al tuplurilor (liniilor) care se doresc găsite.

- În exemplul discutat în lucrarea lui Brin au fost folosite cinci exemple de perechi cu titluri de cărți și autori acestora.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

43

Cum lucreaza

2. Pe baza exemplelor cunoscute, se caută pagini unde apar aceste date pe web.

- Dacă se găsește un şablon care identifică un număr de tupluri cunoscute și este suficient de specific încât și puțin probabil să identifice prea mult, atunci se acceptă acest şablon.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

44

DICE

- Astfel, când selectează noi mulțimi pentru a începe numărarea, DICE este direcțional în favoarea cuvintelor care apar deja în mulțimi cu interes ridicat.
- Totuși, nu se bazează pe aceste cuvinte altfel nu ar putea niciodată să găsească mulțimi cu interes ridicat compuse din multele cuvinte pe care nu le-a considerat niciodată.
- Unele (dar nu toate) din construcțiile pe care le utilizează DICE pentru crearea noilor mulțimi sunt:

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

39

DICE: noi mulțimi

1. Două cuvinte aleatorii. Aceasta este singura regulă independentă de assertiunea "heavy edge" și ajuta noi cuvinte să ajungă în mulțime.
2. Un cuvânt dintr-o mulțime interesantă și un cuvânt aleator.
3. Două cuvinte din două mulțimi interesante diferite.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

40

Cum lucreaza

3. Fiind dată o mulțime de şabloane acceptate, se caută date care satisfac aceste şabloane și se adaugă la mulțimea datelor cunoscute.

4. Se repetă pașii (2) și (3) de un număr de ori. În exemplul citat au fost utilizate patru citări care au dus la 15,000 de tupluri; aprox. 95% au fost perechi adevarate titlu-autor.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

45

Ce este un şablon

- Are 5 componente:
 1. **Ordinea**, i.e., dacă titlul apare în text înaintea autorului sau vice-versă. Într-un caz general, în care tuplurile au mai mult de 2 componente, ordinea va fi dată de permutea componentelor.
 2. **Prefixul adresei web (URL)**.
 3. **Prefixul/textul**, care apare înaintea primului dintre titlu și autor

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

46

Exemplu

- Un şablon posibil poate consta din următoarele:
 1. **Ordinea**: titlu și apoi autor.
 2. **Prefixul URL**: www.stanford.edu/dass
 3. **Prefixul/textul**, care urmează după al doilea dintr-o serie de date. Atât prefixul căt și sufloxul au fost limitate la 10 caractere.
- Aici `<L>><titlu></>` de **autor** `</P>`
- Titlul este orice apără între `prefix` și `mijloc`; autorul este orice apără între `mijloc` și `sufixul`.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

47

DICE: noi mulțimi

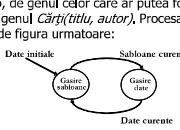
4. Reuniunea a două mulțimi interesante a căror intersecție are dimensiunea 2 sau mai mult.
5. $\{a, b, c\}$ dacă toate mulțimiile $\{a, b\}$, $\{a, c\}$ și $\{b, c\}$ sunt și găsite ca fiind interesante.
- Bineînțeles, în general sunt mult prea multe opțiuni de a aplica cele de mai sus în toate modulele posibile astfel încât se utilizează o selecție aleatoare a opțiunilor dând o anumită sensă fiecăreia dintre ele.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

41

Carti si autori

- Ideea principală este de a căuta pe web faptele de un anumit tip, de genul celor care ar putea forma o relație de genul **Cartă(titlu, autor)**. Procesarea este sugerată de figura următoare:



F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

42

Ce este un sablon

- Are 5 componente:
 1. **Mijlocul** text care apare între cele două elemente de date,
 2. **Sufixul**/textului care urmează după al doilea dintre cele două elemente de date. Atât prefixul căt și sufloxul au fost limitate la 10 caractere.

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

47

F. Radulescu, Curs Utilizarea webului de date, anul IV CS.

48

Tuning sablon

- Definim **specificitatea** unui şablon ca fiind produsul lungimilor prefixului, mijlocului, sufixului şi prefixului URL.
- În mare, specificitatea măsoară cât de posibil este să găsim date care corespund şablonului; cu cât specificitatea este mai mare, cu atât ne aşteptăm la mai puține aparitii ale acestuia în date.

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

49

Tuning sablon

- Şablonul trebuie să îndeplinească două condiții pentru a fi acceptat:
 1. Trebuie să fie cel puțin 2 elemente de date cunoscute care apar conform aceluia şablon.
 2. Produsul specificității şablonului cu numărul de aparitii de date conform acestuia trebuie să depășească un anumit prag T (nespecificat în articolul original).

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

50

Constructia sabloanelor

- Prefixul comun este www.stanford.edu/class/cs
- Dacă trebuie să spargem grupul, atunci următorul caracter, 3 sau 1, sparge grupul în două, cu acele aparitii ale datelor din prima pagină (pot fi multe astfel de aparitii) mergând într-un prim grup și aparitiile din celelalte două pagini în ceeaலălt:
 - >www.stanford.edu/class/cs345/index.html
 - >www.stanford.edu/class/cs340/readings.html
- Să...
 - >www.stanford.edu/class/cs145/index.html

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

55

Gasirea aparitiilor din sabloane

1. Se găsesc toate URL-urile care se potrivesc cu prefixul URL al cel puțin unui şablon.
2. Pentru fiecare astfel de pagină se parcurge textul folosind o expresie regulată construită din prefixul, mijlocul și sufixul şablonului.
3. Se extrage din fiecare potrivire titlul și autorul, după ordinea specificată în şablon.

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

56

Pasii executiei

1. Găsirea aparitiilor pornind de la datele cunoscute
2. Construcția sabloanelor din aparitiile de date
3. Găsirea aparitiilor pornind de la sabloane

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

51

Aparitie

- O aparitie a unui tuplu (existent in tabela) constă în:
 1. Un anumit titlu și autor.
 2. Adresa Internet completa (URL) și nu doar prefixul ca în cazul şablonului,
 3. Ordinea, prefixul, mijlocul și sufixul şablonului după care au apărut titlul și autorul respectiv.

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

52

Bibliografie

- J.D.Ullman - CS345 --- Lecture Notes: PageRank, Hubs-and-Authorities , Web Mining <http://infolab.stanford.edu/~ullman/cs345-notes.html>

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

57

Sfârșitul capitolului 12

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

58

Constructia sabloanelor

1. Se grupează aparitiile de date după ordinea și mijlocul lor. De exemplu, un grup din acest "group-by" poate corespunde ordinii "titlu-apoi-autor" și mijlocului "</>" de ".
2. Pentru fiecare grup se găsește cel mai lung prefix, sufix și prefix URL comun.
3. Dacă testul de specificitate pentru acest şablon este îndeplinit, se acceptă şablonul.

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

53

Constructia sabloanelor

- Dacă testul de specificitate nu este îndeplinit, se încearcă spargerea grupului în două prin extinderea lungimii prefixului URL cu un caracter și apoi se repetă pasul (2). Dacă este imposibil să spargem grupul (pentru că există doar un URL) atunci am eşuat în a produce un nou şablon din acel grup.
- Exemplu:** Să presupunem că grupul conține trei URL-uri:
 - >www.stanford.edu/class/cs345/index.html
 - >www.stanford.edu/class/cs145/index.html
 - >www.stanford.edu/class/cs340/readings.html

F.Rădulescu, Curs Ulliman, baza de date, anul IV CS

54

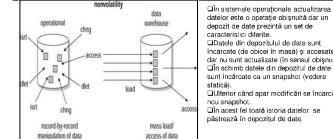
Depozite de date și Modelarea dimensională

Câteva definiții

▪ Wikipedia:

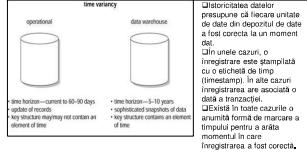
- Un depozit de date este locul de stocare al datelor electronice ale unei organizații. Depozitele de date sunt concepute pentru a facilita raportarea și analiza (din carteau lui Inmon spun că [1]).
- Un depozit de date găzduiește o formă standardizată, consistentă, curată și integrată a datelor proveniente din diverse sisteme operaționale utilizate în aceea organizație, structurată pentru a aborda în mod specific cerințele de raportare și de analiză.

Colecție nevolatilă



2

Date istorice (time variant)



8

Câteva definiții

▪ R. Kimball (vezi [2, 3]):

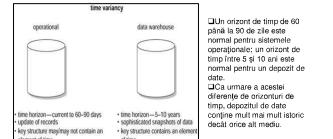
- Un depozit de date este o copie a datelor tranzacționale structurate special pentru interogare și analiză.
- Conform acestei definiții:
 - Forma datelor stocate (SGBDR, fișier) nu este legată de definiția unui depozit de date.
 - Depozitarea datelor nu este legată exclusiv de „factorii de decizie” sau utilizată doar în procesul de luare a deciziilor.

Câteva definiții

▪ Inmon (vezi [1, 3]):

- Un depozit de date este o colecție de date pentru asistență decizională factorilor de conducere care este:
 - orientată pe subiecte,
 - integrată,
 - nevolatilă,
 - istorică (time variant)

Date istorice (time variant)



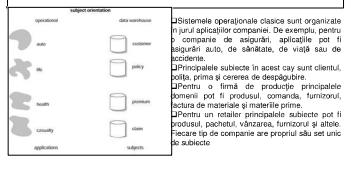
4

Date normalizate vs. Modelare dimensională

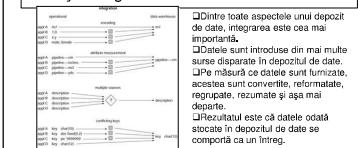
- Există două abordări principale pentru stocarea datelor într-un depozit de date:
 1. Abordarea normalizată (Inmon - [1])
 2. Abordarea dimensională (Kimball - [2])
- Aceste abordări nu se exclud reciproc și există și alte abordări. Abordările dimensionale pot implica normalizarea datelor într-un anumit grad.
- Cursul de azi se bazează pe abordarea dimensională și cartea lui Kimball & Ross - vezi [2].

10

Colecție orientată pe subiecte



Colecție integrată



6

Date normalizate vs. Modelare dimensională

- În abordarea normalizată, datele din depositul de date sunt stocate urmărind, într-o anumită măsură, regulile cunoșute privind normalizarea din domeniul bazelor de date.
- Tabelele sunt grupate pe subiecte care reflectă categoriile generale de date (de exemplu date despre client, produse, finanțe etc.).
- Principalul avantaj al acestei abordări este că este simplu să adaugi noi informații în baza de date.

1

Date normalizate vs. Modelare dimensională

- Un dezavantaj al acestei abordări este faptul că, din cauza numărului mare de tabele implicate, poate fi dificil pentru utilizatorii să reușească prin join date din diferite surse pentru a obține informații semnificative și precise.
- Să acceseze informații fără o înțelegere precisă a sursei de date și a structurii depozitului de date.

12

Date normalize vs. Modelare dimensională

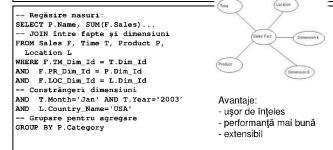
- În abordare dimensională, datele tranzacționale sunt împărțite în fapte sau măsuri (date numerice ale tranzacțiilor) și dimensiuni (informații referitoare care conferă contextul faptelor).
- De exemplu, o tranzacție de vânzare poate fi defășurată în lăpt prezentând codul produsului și prețul plătit pentru produse și în dimensiuni prezentând date comenzi, numele clientului, codul produsului, locația expedierii, locația de facturare și agenții de vânzări responsabili de primirea comenzii.

Date normalize vs. Modelare dimensională

- Un avantaj esențial al unei abordări dimensionale este că depositul de date este mai ușor de înțeles și de folosit. De asemenea, regăsirea datelor în depositul de date poate să funcționeze foarte rapidă.
- Principalele dezavantaje ale abordării dimensionale sunt:
 1. Pentru a crea o colecție integrată, tabelelor de fapte și dimensiuni înaintea depozitării datei cu date din diferite sisteme operaționale este mai complicat.
 2. Este mai dificil de modificat structura depositului de date dacă organizația care adoptă abordarea dimensională schimbă modul în care își desfășoară activitatea.

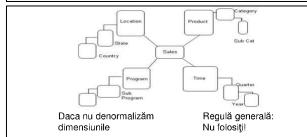
14

Scheme stea



9

Schema Fulg de nea (snowflake)



20

Modelare dimensională

- Este o modalitate de structurare pentru un depozit de date
- Bazat pe:
 - Tabele de fapte (sau măsuri)
 - Tabele de dimensiuni

Tabelă de fapte (măsuri)

- Reprezintă un proces de business
- Contine măsurătorile sau valourile sau faptele proceselor de business
- De exemplu „cantitate” și „preț total” în cazul unei companii de retail
- Majoritatea atributelor (coloanele) sunt additive (vânzări în această lună), unele sunt semi-additive (sold la data de azi), atele nu sunt additive (cod produs)
- Nivelul de detaliu se numește „granularitatea” tabelului – ce reprezintă o linie din tabela de fapte respectivă.
- Conține chei strânsă pentru fiecare tabelă de dimensiuni cu care se relatează.

16

Cei patru pași ai modelării dimensionale

- Obiectiv: proiectarea unei baze de date dimensionale, luând în considerare patru pași în această ordine:
 1. Selectarea procesul de business de modelat.
 2. Declarația gradului de detaliere (declare the grain).
 3. Stabilirea dimensiunilor pentru tabelele de fapte.
 4. Identificarea atributelor tabelelor de fapte.

(Ralph Kimball, Margy Ross - The Data Warehouse Toolkit, Second Edition, Wiley & Sons, 2002, pp.26-25)

1

1. Selectare proces de business

- Un proces este o activitate naturală de business desfășurată într-o organizație
- De obicei este susținut de un sistem de colectare a datelor.
- Exemple de procese de afaceri includ:
 - Aprovisionare,
 - Comenzi,
 - Livrări,
 - Facturare,
 - Inventar (stocuri),
 - Contabilitate

22

Tabelele de dimensiuni

- Reprezintă cine, ce, unde, când și cum pentru o măsură
- Reprezintă entități din lumea reală și nu procese de business
- Dau contextul unei măsuri (subiect)
- De exemplu pentru tabelul de fapte Vânzări, caracteristicile măsurii „cantitate” pot fi locație (unde), timp (când), produs vândut (ce).
- Atributele dimensiunilor sunt coloanele din respectivul tabel.

Tabelele de dimensiuni

- În dimensiunea Locație, atributele pot fi codul locației, județul, țara, codul poștal.
- În general, atributele dimensiunilor sunt utilizate în etichetele raportărilor și în condiții cum ar fi Tara = "USA".
- Înainte de a crea o depozitări de date trebuie decis ce conține acesta. Să zicem că se dorește un depozit de date care să conțină vânzările pe diverse locații ale magazinelor, pe timp (data) și pe produse. Atunci dimensiunile vor fi: Locație, Dată și Produs.

18

2. Declarația gradului de detaliere

- Declarația gradului de detaliere înseamnă specificarea exactă a ceea ce reprezintă o linie din tabelul de fapte.
- Aceasta oferă răspunsul la întrebarea „Cum descrieți o singură linie din tabelul de fapte?”

19

2. Declarația gradului de detaliere

- Exemplu – ce este o linie din tabelul de fapte:
 - O linie de pe bonul de casă al unui client, generat de POS
 - O linie de pe factură primită de la o firmă
 - Un permis de îmbărcare pentru a urca într-un zbor
 - Valoarea zilnică a stocului pentru fiecare produs dintr-un depozit
 - Soldul la sfârșit de lună pentru un cont bancar

24

2. Declarația gradului de detaliere

- Alegere necorespunzătoare a gradului de detaliere va compromite implementarea depozitului de date.
- Declarația gradului de detaliere este un pas critic care nu poate fi făcut ușor.
- Dacă în etapele 3 sau 4 vedem că declarația gradului de detaliere este greșită, trebuie să revenim la pasul 2, să redescoperim corect gradul de detaliere și să redevenim pași 3 și 4 din nou.

Evaluare: 1/100 | Scris de: user

5

3. Stabilirea dimensiunilor

- Dacă este clar gradul de detaliere dimensiunile pot fi identificate destul de ușor.
- Ele reprezintă toate descrierile posibile formate din valori singulare ale unei linii din tabela de fapte.

Evaluare: 1/100 | Scris de: user

26

Studiul de caz: firmă de retail

- Restul de 5.000 de SKU-uri provin din departamente cu produse vrac precum carne, brânzetură, panificație, flori, etc.
- Deși aceste produse nu au CUP-uri recunoscute la nivel național, lanțul alimentar le atribuie coduri SKU.
- Deoarece lanțul nostru alimentar este automatizat, etichetele generate se lipesc pe multe articole din aceste departamente.

Evaluare: 1/100 | Scris de: user

31

Studiul de caz: firmă de retail

- Deși codurile de bare nu sunt CUP-uri, sunt cu siguranță coduri SKU.
- Datele sunt colectate în mai multe locuri din magazin. Unele dintre cele mai ușe date sunt colectate la casele de marcat pe măsură ce clientii cumpără produse.
- Sistemul POS se află la ieșirea magazinului alimentar unde este înregistrat conținutul cosului de cumpărături.
- Ușa din spate, în care furnizorii fac livrări, este un alt punct interesant de colectare a datelor.

Evaluare: 1/100 | Scris de: user

32

3. Stabilirea dimensiunilor

- Exemple de dimensiuni comune:

- Data,
- Produs,
- Client,
- Tip tranzacție,
- Stare tranzacție.

Evaluare: 1/100 | Scris de: user

7

4. Identificare atribute tabela de fapte

- Faptele sunt determinate răspunzând la întrebarea „Ce măsurăm?”
- Toate faptele/măsurile candidate din proiect trebuie să fie adecvate pentru gradul de detaliere stabilit la pasul 2.
- Faptele/măsurile care aparțin în mod clar unui alt grad de detaliere trebuie puse într-un tabel de fapte separat.
- Faptele/măsurile tipice sunt numérică și additive. ex. cantitatea comandată sau valoarea totală.

Evaluare: 1/100 | Scris de: user

28

Studiul de caz: firmă de retail

- Altă managementul magazinului, cât și departamentul de marketing de la sediul central petrec o mare cantitate de timp cu stabilirea prejудиilor și a promociilor.
- Promociile într-un magazin alimentar includ reduceri temporare de preț, anunțuri în ziare, modul de plasare în magazin (inclusiv plasarea pe cap de rând) și cupoane.

Evaluare: 1/100 | Scris de: user

31

Studiul de caz: firmă de retail

- Cea mai directă și eficientă metodă de a crește volumul de vânzări este să adăunăm dimensiuni.
- O reducere de 50 de centuri a prețului serviciilor de hârtie, în special atunci când este cumpărată cu un anumit sau o plasare preferențială poate determina creșterea serviciilor de hârtie cu un factor de 10.
- Din pacat, o reducere astfel de mare a prețului, de obicei, nu este sustenabilă, deoarece serviciile probabil se vând în perdere.
- Pe baza acestui exemplu vom începe să proiectăm modelul dimensional al datelor.

Evaluare: 1/100 | Scris de: user

34

Studiul de caz: firmă de retail

- Să luăm în considerare un lanț de magazine alimente cu 100 de magazine distribuite pe o zonă de cinci state (USA).
- Fiecare magazin are o gamă completă de departamente, inclusiv alimente, alimente congelate, lăptate, carne, produse, panificație, produse florale și auxiliare pentru sănătate / frumusețe.
- Fiecare magazin are aproximativ 60.000 de produse individuale pe rafturile sale.

Evaluare: 1/100 | Scris de: user

9

Studiul de caz: firmă de retail

- Produsele individuale se numesc stock keeping units (SKU).
- Aproximativ 55.000 dintre SKU-urile provin de la producători externi și au coduri de bare înscrise pe pachetul de produse. Aceste coduri de bare se numesc coduri universale de produse (CUP).
- CUP-urile sunt în același nivel cu SKU-urile individuale.
- Fiecare variație de impachetare a unui produs are un CUP separat și, prin urmare, este un SKU separat.

Evaluare: 1/100 | Scris de: user

30

1. Selectare proces de business

- Primul pas în proiectare este de a decide ce proces (e) de afaceri vor fi modelate, combinând o înțelegere a cerințelor de afaceri cu o înțelegere a datelor disponibile.
- În primul model dimensional construit ar trebui să fie cel cu cel mai mare impact - ar trebui să răspundă la cele mai prezente întrebări de afaceri și să fie ușor accesibil pentru extragerea datelor

Evaluare: 1/100 | Scris de: user

5

1. Selectare proces de business

- În studiu de vânzare cu amănuntul, managementul dorește să înțeleagă mai bine procesul de cumpărare al clientilor înregistrat de sistemul POS.
- Astfel, procesul de business pe care îl vom modela este cel de vânzări prin POS.
- Aceste date ne vor permite să analizăm ce produse se vând în ce magazine în ce zile în ce condiții promoționale.

Evaluare: 1/100 | Scris de: user

36

2. Declarația gradului de detaliere

- De preferat, ar trebui să dezvoltăm modele dimensionale pentru cele mai atomice informații captate de un proces de business .
- Datele atomice sunt cele mai detaliate informații colectate; aceste date nu pot fi divizate în continuare.
- Datele atomice sunt în final dimensionale - deci este o poveste perfectă pentru abordarea dimensională.
- Modelul cu date mai puțin detaliate este vulnerabil la solicitările neașteptate ale utilizatorilor de a explora datele.

Evaluare: 1/100 | Scris de: user

7

2. Declarația gradului de detaliere

- Datele sumare (agregate: sume, medii, numărări, etc.) joacă un rol important ca instrument de ajustare a performanței, dar nu constituie un substitut pentru a oferi utilizatorilor accesul la cele mai mici detali.
- Un depozit de date necesită aproape întotdeauna date exprimate la cel mai înalt grad de detaliere posibil pe fiecare dimensiune.
- În studiu nostru de caz, cele mai granulare date sunt cele de pe 9 linie a bonului de casă.

Evaluare: 1/100 | Scris de: user

38

4. Identificare atribute tabela de fapte

- Profilul brut se obține prin scăderea costului produsului din valoarea totală.
- Toate aceste fapte (cantitatea, suma totală, costul total și profitul brut) sunt additive pe toate dimensiunile.
- Procentajele și raporturile cum ar fi marja brută, sunt nonadditive (marja brută = profitul brut / valoarea totală)
- Număratorul și numitorul ar trebui stocate în tabelul de fapte.
- Raportul poate fi calculat la nevoie.
- Profitul unitar este, de asemenea, un fapt nonadditiv (este un raport: valoare totală/cantitate)

Evaluare: 1/100 | Scris de: user

3

Atribute dimensiuni : Data

- Dimensiunea data este singura dimensiune aproape garantată a fi găsită în fiecare schemă stocărească practic aceasta este o serie de tip.
- Dimensiunea timp (al zilei) este diferită de cea a datei.

Evaluare: 1/100 | Scris de: user

44

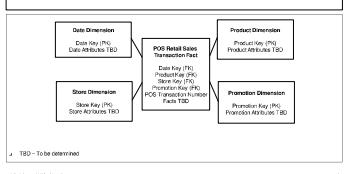
3. Stabilirea dimensiunilor

- În studiu nostru de caz identificăm 3 dimensiuni principale: data, produsul și magazinul.
- În plus, putem adăuga noi dimensiuni, ca promoția sub care se vinde produsul.
- În studiu nostru de caz, decindem următoarele dimensiuni descriptive: data, produsul, magazinul și promoția.
- În plus, vom include numărul bonului de casă ca dimensiune specială (descrișă mai târziu)

Evaluare: 1/100 | Scris de: user

9

3. Stabilirea dimensiunilor



Evaluare: 1/100 | Scris de: user

40

Atribute dimensiuni: Data

Date Key (PK)	Calendar Quarter
Date	Calendar Year
Date Day	Calendar Year
Day of Week	Weekday
Day Number in Epoch	Day of Year
Day Number in Epoch	Day of Week Number in Year
Month Number in Epoch	Month Number in Year
Month Number in Epoch	Month Number in Year
Year Number in Epoch	Year Number in Year
Year Number in Epoch	Fiscal Year
Day Number in Fiscal Year	Fiscal Year Number
Last Day in Week Number	Week Number in Year
Last Day in Week Number	Weekday Number in Year
Calendar Month	Month Name
Calendar Month	Season
Calendar Year	Year Name
Calendar Year	Season
Calendrical Month	Month Name
Calendrical Month	Season

Evaluare: 1/100 | Scris de: user

5

Atribute dimensiuni: Data

KeyDate	Date	Full Date	Description	Day of Week	Calendar Month	Calendar Year	Fiscal Year	Month	Holiday Indicator	Weekday Indicator
1	01-01-2020	January 1, 2020	January 1, 2020	Sunday	January	2020	2020	January	Yes	Yes
2	02-01-2020	January 2, 2020	January 2, 2020	Monday	January	2020	2020	January	No	No
3	03-01-2020	January 3, 2020	January 3, 2020	Tuesday	January	2020	2020	January	No	No
4	04-01-2020	January 4, 2020	January 4, 2020	Wednesday	January	2020	2020	January	No	No
5	05-01-2020	January 5, 2020	January 5, 2020	Thursday	January	2020	2020	January	No	No
6	06-01-2020	January 6, 2020	January 6, 2020	Friday	January	2020	2020	January	No	No
7	07-01-2020	January 7, 2020	January 7, 2020	Saturday	January	2020	2020	January	No	No
8	08-01-2020	January 8, 2020	January 8, 2020	Sunday	January	2020	2020	January	Yes	Yes

Evaluare: 1/100 | Scris de: user

46

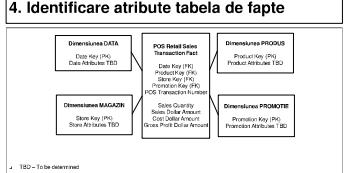
4. Identificare atribute tabela de fapte

- Faptele trebuie să existe pentru gradul de detaliere ales. (în acest caz, linie individuală din tranzacția POS).
- Faptele colectate de sistemul POS includ cantitatea, prețul unitar și valoarea totală (= cantitatea * prețul unitar).
- Unele sisteme POS oferă un cost standard pentru produs, și anume cel cu care a fost livrat de furnizor.

Evaluare: 1/100 | Scris de: user

1

4. Identificare atribute tabela de fapte



Evaluare: 1/100 | Scris de: user

42

Atribute dimensiuni: Data

- Datele de date au întotdeauna nevoie de o dimensiune Data explicită.
- Există multe atribute ale datei calendaristice care nu sunt calculate de la data de la data inclusiv perioadele fiscale, anotimpurile, sărbătorile și week-endurile.
- În loc să încercăm să calculăm aceste informații calendaristice non-standard într-o interogare, ar trebui să le căutăm într-un tabel ("dimensiunea Data").
- Date și ora sunt aproape complet independente.
- Dacă am combina cele două dimensiuni, dimensiunea ar crește semnificativ.

7

Atribute dimensiuni: Data

KeyDate	Date	Full Date	Description	Day of Week	Calendar Month	Calendar Year	Fiscal Year	Month	Holiday Indicator	Weekday Indicator
1	01-01-2020	January 1, 2020	January 1, 2020	Sunday	January	2020	2020	January	Yes	Yes
2	02-01-2020	January 2, 2020	January 2, 2020	Monday	January	2020	2020	January	No	No
3	03-01-2020	January 3, 2020	January 3, 2020	Tuesday	January	2020	2020	January	No	No
4	04-01-2020	January 4, 2020	January 4, 2020	Wednesday	January	2020	2020	January	No	No
5	05-01-2020	January 5, 2020	January 5, 2020	Thursday	January	2020	2020	January	No	No
6	06-01-2020	January 6, 2020	January 6, 2020	Friday	January	2020	2020	January	No	No
7	07-01-2020	January 7, 2020	January 7, 2020	Saturday	January	2020	2020	January	No	No
8	08-01-2020	January 8, 2020	January 8, 2020	Sunday	January	2020	2020	January	Yes	Yes

Evaluare: 1/100 | Scris de: user

48

Atribute dimensiuni: Produs

- Dimensiunea Produs descrie fiecare SKU din magazinul alimentar.
 - Un magazin obisnuit din lanț poate stoca 60.000 de SKU-uri, dar atunci când avem în vedere diferite scheme de merchandising din lanț și produse istorice care nu mai sunt disponibile, dimensiunea produsului ar avea cel puțin 150.000 de linii sau mai mult.

F. Fabiano - UCB - Novembre

Atribute dimensiuni: Produs

- o Product Key (PK)
 - o Product Description
 - o SKU Number (Natural Key)
 - o Brand Description
 - o Category Description
 - o Department Description
 - o Package Type Description
 - o Package Size
 - o Fat Content
 - o Diet Type
 - o Weight
 - o Weight Units of Measure
 - o Storage Type
 - o Shelf Life Type
 - o Shelf Width
 - o Shelf Height
 - o Shelf Depth

50

Atribute dimensiuni: Magazin

- Store Name
 - Store Number (Natural Key)
 - Store Street Address
 - Store City
 - Store County
 - Store State
 - Store Zip Code
 - Store Manager
 - Store District
 - Store Region
 - Floor Plan Type
 - Photo Processing Type
 - Financial Service Type
 - Selling Square Footage
 - Total Square Footage
 - First Open Date
 - Last Remodel Date

1

Atribute dimensiuni: Promoție

- Dimensiunea **Promotie** descrie condițiile de promovare în care a fost vândut un produs.
 - Condițiile de promovare includ reducere temporare de preț, plasare preferențială, anunțuri în ziare și cupoane.
 - Această dimensiune este adesea numită dimensiune cauzală - causal dimension (spre deosebire de o dimensiune obiectivă - *casus dimension*), deoarece descrie factorii gândiți să provoace o schimbare a cifrelor de vânzări.

56

Atribute dimensiuni: Produs

1

Atribute dimensiuni: Produs

- ❑ Un tabel rezonabil de tip **Produs** are 50 sau mai multe atribute descriptive.
 - ❑ Fiecare atribut este o sursă bogată pentru construirea rapoartelor.

52

Atribute dimensiuni: Promoție

- Promotion Key (PK)
 - Promotion Name
 - Price Reduction Type
 - Promotion Media Type
 - Ad Type
 - Display Type
 - Coupon Type
 - Ad Media Name
 - Display Provider
 - Promotion Cost
 - Promotion Begin Date
 - Promotion End Date

1

Număr tranzacție: Dimensiuni degenerate

- Numerele de control operațional, cum ar fi numerele de ordine, numerele facturări și numerele de facturare, de obicei, dă naștere la dimensiuni goale și sunt reprezentate ca dimensiuni degenerate (adică avem chei strânsă în tabela de fapte fără tabele de dimensiune corespunzătoare).

58

Atribute dimensiuni: Magazin

- Dimensiunea **Magazin** descrie fiecare magazin din lantul nostru.
 - Dimensiunea **Magazin** este dimensiunea geografică principală în studiu nostru de cauză.
 - Fiecare magazin poate fi găsită căruia o locație. Din această cauză, putem asocia un magazin cu orice atribut geografic, cum ar fi codul poștal, județul și statul din Statele Unite.
 - De obicei, magazinele se plasează în districte și regiuni.

1

Atribute dimensiuni: Magazin

- ❑ De obicei, magazinele se plasează în districte și regiuni.
 - ❑ Aceste două ierarhii diferențe sunt ambele reprezentate cu urșinjă în dimensiunea magazinului, deoarece atât ierarhile geografice cât și cele regionale ale magazinului sunt bine definite pentru o singură linie din dimensiunea Magazin.
 - ❑ Nu este neobișnuit să reprezintăm mai multe ierarhii într-un tabel cu dimensiuni. În mod ideal, numele și valorile atributului ar trebui să fie unice pentru ierarhile multiple.

1

Bibliografie

1. W.H. Inmon - Building The Data Warehouse, Third Edition, Wiley & Sons, 2002
 2. Ralph Kimball, Margy Ross - The Data Warehouse Toolkit, Second Edition, Wiley & Sons, 2002
 3. Dimitra Vistia, CS 680 Course notes
 4. Wikipedia – Pages on Data Warehouse, etc.
 5. <http://searchsqlserver.techtarget.com>
 6. Vincent Raineri, Building a Data Warehouse with Examples in SQL Server, Springer, 2008

1