

## 2. Arhitectura depozitului de date

### Cuprins

2.1. Componente arhitecturale .....	2
Datele operaționale .....	2
Administratorul pentru încărcarea datelor .....	2
Administratorul depozitului de date .....	2
Administratorul cererilor .....	3
Datele detaliate .....	4
Datele agregate .....	4
Datele <i>backup</i> .....	4
Metadatele .....	4
Utilitarele de acces la date pentru utilizatorilor finali .....	6
2.2. Fluxurile de informații dintr-un depozit de date .....	6
Fluxul intern .....	7
Flux ascendent .....	7
Flux descendenter .....	7
Flux extern .....	8
Meta-flux .....	8
2.3. <i>Data mart</i> -uri .....	8
Bibliografie .....	9

## 2. Arhitectura depozitului de date

- În acest capitol sunt prezentate componentele importante ale unui depozit de date.
  - Sunt descrise procesele, utilitarele și tehnologiile asociate cu un depozit de date.

### 2.1. Componente arhitecturale

- Arhitectura tipică unui depozit de date este descrisă în figura *Fig 2.1*.

#### Datele operaționale

- Datele sursă pentru depozitele de date pot fi reprezentate de:
  - date departamentale menținute în baze de date relaționale;
  - date private menținute pe stații și server-e private;
  - date externe, disponibile pe Internet sau în baze de date comerciale publice.

#### Administratorul pentru încărcarea datelor

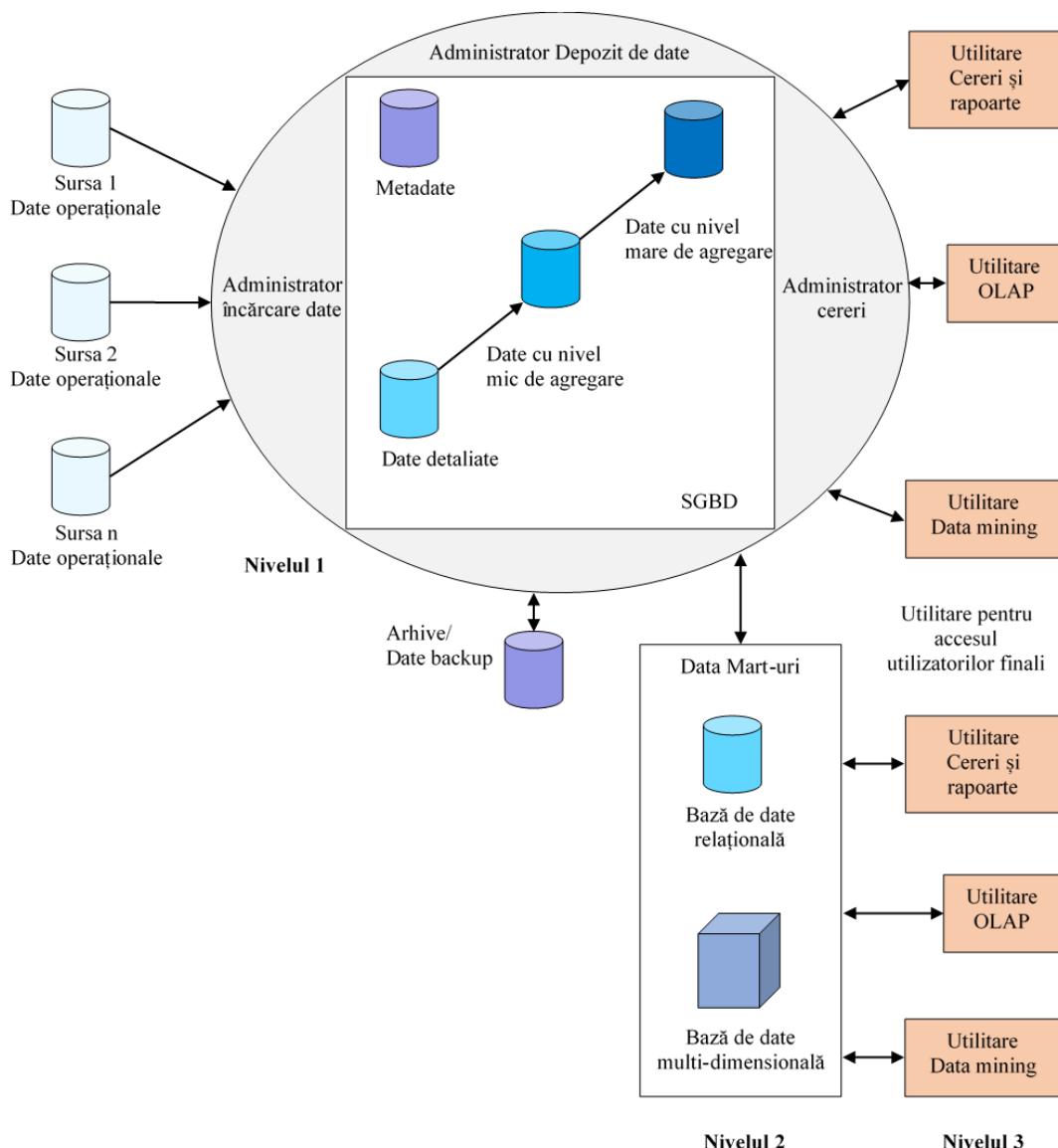
- Este cunoscut și sub denumirea de componentă *font-end* și realizează toate operațiile asociate cu extragerea și încărcarea datelor operaționale într-un depozit de date, incluzând transformări simple ale datelor.
- Complexitatea acestei componente variază în funcție de depozitul de date și poate fi construit utilizând o combinație între utilitarele create pentru încărcarea datelor și programe uzuale.

#### Administratorul depozitului de date

- Această componentă realizează toate operațiile legate de administrarea datelor din depozit.
- Operațiile realizate de componenta de administrare a depozitului de date includ:
  - analiza datelor pentru a asigura consistență;
  - transformarea datelor sursă din structurile temporare de stocare și includerea acestora în tabelele depozitului de date;
  - crearea indecșilor și a vizualizărilor asupra tabelelor de bază;
  - generarea denormalizării (dacă este necesar);
  - generarea agregărilor;
  - crearea arhivelor și a *backup-urilor*.

## Administratorul cererilor

- Este cunoscut și sub denumirea de componentă *back-end* și realizează toate operațiile legate de administrarea cererilor utilizator.
- În mod tipic, această componentă conține:
  - utilitare de acces la datele disponibile utilizatorilor finali;
  - utilitare de monitorizare a depozitului de date;
  - facilități oferite de sistemul de gestiune a bazei de date.
- Operațiile realizate de această componentă includ
  - direcționarea cererilor pe tabelele necesare;
  - planificarea execuției acestora.



**Fig 2.1.** Arhitectura depozitului de date

## Datele detaliate

- Această zonă a depozitului de date stochează toate datele detaliate din baza de date.
  - În cele mai multe cazuri, aceste date sunt aggregate la primul nivel următor detaliului.
- Datele detaliate sunt adăugate în depozitul de date pentru a suplimenta datele aggregate.
- Sunt necesare atunci când la nivel managerial se doresc rapoarte cu alt nivel de detaliu sau de agregare al datelor.

## Datele aggregate

- Această zonă a depozitului stochează toate aggregările predefinite de date, având diferite niveluri de agregare.
- Scopul menținerii datelor aggregate este de a mări performanța cererilor care necesită aggregări.
- Datele aggregate sunt actualizate permanent, pe măsură ce sunt încărcate date noi în depozit.

## Datele *backup*

- Această zonă a depozitului de date menține datele detaliate și datele aggregate pentru arhivare și restaurare.
- Deși datele aggregate sunt generate din datele detaliate, acestea sunt menținute pentru cazurile în care este necesară restaurarea aggregărilor.

## Metadatele

- Această zonă a depozitului de date stochează toate definițiile metadatelor (informațiile despre date) utilizate de procesele bazei de date depozit (procesele de extragere și încărcare a datelor, procesul de administrare a depozitului de date, procesul de administrare a cererilor etc).
- Structura metadatelor diferă în funcție de procesul care le utilizează, deoarece scopul lor este diferit de la un proces la altul.
  - Aceasta înseamnă că depozitul de date menține mai multe copii ale metadatelor care descriu același articol de date din depozit.
- Metadatele asociate cu procesul de transformare și încărcare a datelor trebuie să descrie datele sursa și orice schimbări care au avut loc asupra lor.
  - De exemplu, pentru fiecare câmp sursă trebuie să existe un identificator unic, un nume, un tip de date și locația (inclusiv sistemul și numele obiectului), împreună cu tipul datei destinație și numele tabelului destinație.

- Metadatele asociate cu administrarea datelor descriu datele aşa cum sunt stocate în depozitul de date.
  - Fiecare obiect al bazei de date trebuie să fie descris.
  - Aceste informaţii sunt menţinute în dicţionarul datelor.
  - Totuşi pentru bazele de date depozit sunt necesare informaţii suplimentare.
    - De exemplu, metadatele trebuie să descrie orice câmpuri care sunt asociate cu agregări, incluzând o descriere a aggregării care a fost realizată.
    - În plus, partiiile tabelelor trebuie şi ele descrise, incluzând informaţii despre cheia de partiţionare şi domeniul partii.
- Metadatele descrise mai sus sunt necesare şi administratorului cererilor.
  - Administratorul cererilor generează metadate despre cererile care pot fi utilizate pentru a crea un istoric al tuturor cererilor şi profilului cererilor pe utilizator, pe grupuri de utilizatori sau la nivel de bază de date.
- În plus, cele mai multe utilitare pentru administrarea copierii datelor şi accesul utilizatorilor la date folosesc propriile versiuni de metadate.
  - De exemplu, utilitarele de copiere a datelor folosesc metadatele pentru a determina regulile de mapare a datelor sursă în datele din depozit, iar utilitarele pentru accesul utilizatorilor la date folosesc metadatele pentru a determina planul de execuţie a unei cereri.



❖ Exemplu comun de metadate ce descriu un atribut al unei tabele

**Nume:** tip\_plata

**Descriere:** Modalitatea particulară de a achita o factură (plată cu numerar, transfer bancar, dispoziţie de plată, bilet la ordin, plată cu card bancar, CEC)

**Tip de date și dimensiune:** varchar2(4)

**Valori permise:** NUM (plată cu numerar), TB (transfer bancar), DP (dispoziţie de plată), BO (bilet la ordin), CARD (plată cu card bancar), CEC (CEC)



❖ În cazul depozitelor de date sunt necesare metadate complexe, care descriu atât datele menţinute în depozit, cât şi procesele ce sunt utilizate pentru a crea depozitul. Acestea conţin informaţii ce sunt utilizate pentru a mapa datele între sistemele sursă şi depozitul de date şi, de asemenea, pot conţine suplimentar reguli de transformare a datelor.

- ❖ Metadatele sunt definite în prima fază a procesului de design a depozitului de date.
- ❖ Metadatele se pot defini și administra manual sau automat cu ajutorul utilitarelor (de exemplu, *Oracle Warehouse Builder*, *Analytical Workspace Manager* etc).

### Utilitarele de acces la date pentru utilizatorilor finali

- Scopul principal al depozitelor de date este de a oferi informațiile necesare utilizatorilor pentru luarea deciziilor strategice de marketing.
- Acești utilizatori interacționează cu depozitul de date prin utilitarele de acces, care sunt împărțite în cinci categorii principale:
  - utilitate pentru rapoarte și cereri;
  - utilitate pentru dezvoltarea aplicațiilor;
  - utilitate pentru procesarea analitică online;
  - utilitate *data mining*;
  - utilitate caracteristice sistemelor informatiche pentru conducerea executivă (sunt considerate o formă specializată a sistemelor suport de decizie).

## 2.2. Fluxurile de informații dintr-un depozit de date

- În mediile depozitelor de date există cinci fluxuri principale de informații:
  - flux intern;
  - flux ascendent;
  - flux descendent;
  - flux extern;
  - meta-flux.
- Procesele asociate cu fiecare dintre aceste fluxuri includ:
  - extragerea și încărcarea datelor din sistemele sursă în depozitul de date (flux intern);
  - adăugarea de valori datelor din depozit, prin operații de agregare și de distribuire (flux ascendent);
  - arhivarea și menținerea datelor pentru restaurări (flux descendent);
  - realizarea disponibilității datelor pentru utilizatorii finali (flux extern);
  - administrarea metadatelor (meta-flux).

## Fluxul intern

- Este axat pe procesul de preluare a datelor din sistemele sursă și încărcarea acestora în depozitul de date. De obicei, datele sursă trebuie reconstruite pentru a putea fi utilizate de depozit.
- Reconstrucția datelor implică:
  - restructurarea datelor astfel încât acestea să fie potrivite cerințelor depozitului de date, de exemplu adăugarea sau eliminarea unor câmpuri, denormalizarea datelor;
  - asigurarea consistenței datelor.
- Pentru a putea administra eficient fluxul intern, trebuie să se identifice mecanismele prin care să se determine când se începe extragerea datelor și când se realizează verificările consistenței.
- Atunci când se extrag datele din sistemele sursă este important să se asigure că datele respective sunt în stare consistentă pentru a putea genera o singură vizualizare consistentă asupra lor.

## Flux ascendent

- Activitățile asociate cu acest flux de informații includ:
  - agregarea datelor (prin selecții, proiecții, *join*-uri și grupări asupra datelor) în vizualizări care sunt utile utilizatorilor finali;
  - convertirea datelor detaliate sau aggregate în alte formate (documente text, reprezentări grafice, animații etc.);
  - distribuirea datelor în grupuri corespunzătoare pentru a mări disponibilitatea și accesibilitatea.

## Flux descendant

- Arhivarea datelor vechi joacă un rol important în menținerea performanței depozitului de date.
  - Datele vechi se transferă pe structuri de stocare adiacente (discuri magnetice etc).
- Fluxul descendant de informații include procese care asigură că starea curentă a depozitului de date poate fi restaurată în cazul în care au loc defecțiuni la nivel de sistem sau se pierd anumite date.
  - Datele arhivate trebuie să fie stocate în aşa fel încât să permită restaurarea sistemului atunci când este necesar.

### Flux extern

- Acest flux de informații include:
  - operații de accesare a datelor necesare pentru a satisface cererile utilizatorilor finali; problema principală este crearea unui mediu care să permită utilizatorilor să folosească utilitare pentru accesarea datelor sursă corespunzătoare;
  - operații de livrare a datelor către stația utilizatorului final care a inițiat cererea de acces la date.
- Ca parte a administrării fluxului extern de informații, sistemul trebuie să determine cea mai eficientă cale de a răspunde unei cereri. Performanța cererii depinde de volumul datelor întâi și de existența datelor precalculate.

### Meta-flux

- Fluxurile anterioare descriu administrarea depozitului de date din perspectiva mutării datelor în, respectiv din depozit.
- Meta-fluxul este un proces prin care se mută metadate (informații despre celelalte fluxuri).
- Metadatele au câteva funcționalități în depozitul de date legate de procesele de transformare și încărcare a datelor, administrare a depozitului de date și de generare a cererilor.
- Pentru a răspunde necesităților afacerii, sistemele se modifică în mod constant. Deci administrarea unui depozit de date implică modificări în funcție de evoluția afacerii. Meta-fluxul de date trebuie actualizat continuu în concordanță cu aceste schimbări.

## 2.3. Data mart-uri

- Un *data mart* reprezintă o submulțime de date care sunt necesare unui departament particular al companiei sau unei anumite funcții a afacerii.
- Există câteva motive pentru care se creează *data mart*-uri:
  - cu scopul de a furniza datele într-o formă necesară anumitor grupuri de utilizatori;
  - pentru a îmbunătăți timpul de răspuns prin micșorarea volumului de date care trebuie accesate;
  - pentru a furniza structuri de date în forma în care sunt necesare utilitarelor de acces;
  - deoarece costurile implementării *data mart*-uri sunt mai mici decât cele necesare stabilirii întregului depozit.

## Bibliografie

1. Connolly T.M., Begg C.E., *Database Systems: A Practical Approach to Design, Implementation and Management*, 5th edition, Pearson Education, 2005
2. Dollinger R., Andron L., *Baze de date și gestiunea tranzacțiilor*, Editura Albastră, Cluj-Napoca, 2004
3. Inmon W.H., *Building the Data Warehouse*, 4th Edition, Wiley, 2005
4. Kimball R., *The Data Warehouse Toolkit*, 3rd Edition, Wiley, 2013
5. Kimball R., Ross M., Thorntwaite W., Mundy J., Becker B., *The Data Warehouse Lifecycle Toolkit*, 2nd Edition Wiley, 2008
6. Oracle and/or its affiliates, *Oracle Database Concepts*, 1993, 2024
7. Oracle and/or its affiliates, *Oracle Database Administrator's Guide*, 2001, 2025
8. Oracle and/or its affiliates, *Oracle Database Performance Tuning Guide*, 2013, 2025
9. Oracle and/or its affiliates, *Oracle Database SQL Language Reference*, 1996, 2025
10. Oracle and/or its affiliates, *Oracle Database PL/SQL Language Reference*, 1996, 2025
11. Oracle and/or its affiliates, *Oracle Database: SQL Tuning Workshop*, 2010, 2025
12. Oracle and/or its affiliates, *Oracle OLAP Customizing Analytic Workspace Manager*, 2006, 2019
13. Oracle and/or its affiliates, *Oracle OLAP DML Reference*, 1994, 2019
14. Oracle and/or its affiliates, *Oracle OLAP User's Guide*, 2003, 2019
15. Oracle and/or its affiliates, *Oracle Warehouse Builder Concepts*, 2000, 2021
16. Oracle and/or its affiliates, *Oracle Warehouse Builder Data Modeling, ETL, and Data Quality Guide*, 2000, 2021
17. Oracle and/or its affiliates, *Oracle Database Data Warehousing Guide*, 2001, 2021
18. Oracle University, *Oracle Database: PL/SQL Fundamentals, Student Guide*, 2009, 2025
19. Poe V., Klauer P., Brobst S., *Building A Data Warehouse for Decision Support*, 2nd Edition, Prentice Hall; 1997
20. Popescu I., Alecu A., Velcescu L., Florea (Mihai) G., *Programare avansată în Oracle9i*, Ed. Tehnică, 2004