

Proiectarea bazelor de date relaționale (1)

1. Modelarea datelor

Un **model** este o reprezentare a obiectelor și evenimentelor lumii reale și a asocierilor dintre ele. De fapt, el reprezintă o abstractizare asupra aspectelor semnificative ale unei „întreprinderi“, ale unui sistem real, ignorând proprietățile accidentale. Modelul este cel pe care utilizatorii trebuie să-l cunoască; implementarea unui model este cea pe care utilizatorii nu este necesar să o cunoască. Diferența dintre model și implementare este, de fapt, un caz special și important al deosebirii uzuale dintre logic și fizic.

Modelele se impun prin sintaxa și prin semantica lor și, din acest punct de vedere, există trei tipuri fundamentale de modele:

- modele care descriu aspectele statice ale procesului modelat;
- modele care descriu aspectele dinamice ale procesului modelat;
- modele care descriu aspectele funcționale ale procesului modelat.

Un **model de date** reprezintă o colecție integrată de concepte necesare descrierii:

- datelor,
- relațiilor dintre ele,
- constrângerilor existente asupra datelor sistemului real analizat.

Modelarea unei baze de date permite trecerea de la percepția unor fapte din lumea reală la reprezentarea lor prin date. Modelul de date trebuie să reflecte fidel fenomene ale lumii reale, să urmărească evoluția acestei lumi și comunicarea dintre fenomenele lumii reale.

Modelul trebuie să asigure conceptele de bază care permit proiectantului bazei de date și utilizatorilor să comunice, fără ambiguități, cunoștințele lor privind funcționarea și organizarea modelului real analizat. Prin urmare, un model de date trebuie să reprezinte datele și să le facă înțelese.

În esență, modelul de date are trei componente:

- o mulțime de reguli conform cărora sunt construite bazele de date (partea structurală);
- o mulțime de operații permise asupra datelor, care sunt utilizate pentru reactualizarea sau regăsirea datelor (partea de prelucrare);
- o mulțime de reguli de integritate, care asigură coerența datelor.

Abordarea generală a problemei modelării semantice a datelor se face în patru etape.

- Se identifică o mulțime de concepte semantice care sunt utile în descrierea lumii reale. Se presupune că lumea reală (modelul real analizat) este formată din entități care au anumite proprietăți, că fiecare entitate are o identitate, că există legături, corelații între entități. Conceptul de corelație, ca și cel de entitate, este util, în mod intuitiv, la descrierea modelului.

- Se caută o mulțime de obiecte formale, simbolice care sunt utilizate pentru reprezentarea conceptelor semantice anterioare.
- Se dau reguli de integritate formale și generale (constrângeri) care să reflecte restricțiile la care este supus modelul.
- Se definește o mulțime de operatori formali prin care pot fi prelucrate și analizate obiectele formale.

2. Modelul entitate-relație

Una dintre cele mai cunoscute abordări ale modelării semantice (cu siguranță una dintre cele mai utilizate) este cea bazată pe **modelul entitate-relație** (E/R). Acesta a fost introdus de către P.P. Chen în 1976 și rafinat de atunci în diverse moduri de către acesta și de mulți alți cercetători, ca un model de date conceptual, pentru a ușura proiectarea bazelor de date. Pentru reprezentarea grafică a modelului sunt utilizate diagramele E/R, care sunt modele neformalizate pentru reprezentarea unui model, unui sistem din lumea reală.

Diagramele E/R constituie o tehnică de reprezentare a structurii logice a bazei de date, într-o manieră grafică. Aceste diagrame oferă un mijloc simplu și inteligibil de comunicare a caracteristicilor importante ale designului unei anumite baze de date.

Diagrama E/R este un **model de date conceptual de nivel înalt**, independent de platforma *hardware* utilizată și de tipul SGBD-ului. Modelul este constituit din concepte care descriu structura bazei de date și tranzacțiile de regăsire sau reactualizare asociate.

Popularitatea modelului E/R ca modalitate de abordare a proiectării bazelor de date poate fi atribuită în principal tehnicii de realizare a diagramelor E/R. Această tehnică, ca și modelul E/R însuși, a evoluat de-a lungul timpului datorită noilor problematice care au apărut în proiectarea bazelor de date.

Baza de date poate fi definită ca o mulțime de date ce modelează un sistem real. Acest sistem este format din obiecte legate între ele.

Modelul E/R împarte elementele unui sistem real în două categorii:

- **entități**
- **relații** (legături, asocieri) între aceste entități.

Entitățile și legăturile au anumite caracteristici, numite **atribute**.

Nu trebuie confundat conceptul de relație, în sensul de asociere, care intervine în definirea diagramei E/R cu conceptul de relație care este specific modelului relațional.

Observație: Diagramele E/R vor fi reluate și exemplificate mai jos, în capitolul 4 al acestui material. În general, pe lângă materialul furnizat pe moodle, mai multe studii de caz și exemple vor prezentate la curs!

3. Deficiențe ale modelului E/R

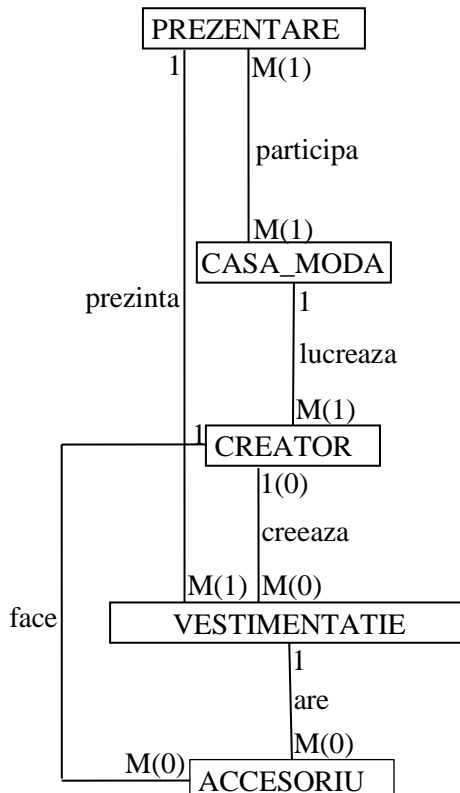
În proiectarea unui model de date pot apărea diverse probleme datorită unei interpretări eronate a sensului unei relații. Aceste probleme sunt denumite **capcane de conectare**.

Unele dintre aceste capcane pot să nu fie semnificative pentru modelul particular considerat, în timp ce altele cer restructurarea modelului. Există două clase importante de capcane: de întrerupere și în evantai.

O capcană de întrerupere poate să apară acolo unde modelul sugerează existența unei relații între entități, dar nu există o cale între anumite apariții ale entităților. Această capcană poate să apară acolo unde există o relație cu participare parțială (0 la cardinalitatea minimă), care face parte din calea dintre entitățile ce sunt legate.

O capcană în evantai poate să apară acolo unde modelul ia în considerare o relație între entități, dar calea dintre anumite apariții ale entităților este ambiguă. Aceste capcane apar când două sau mai multe relații *one_to_many* provin din aceeași entitate.

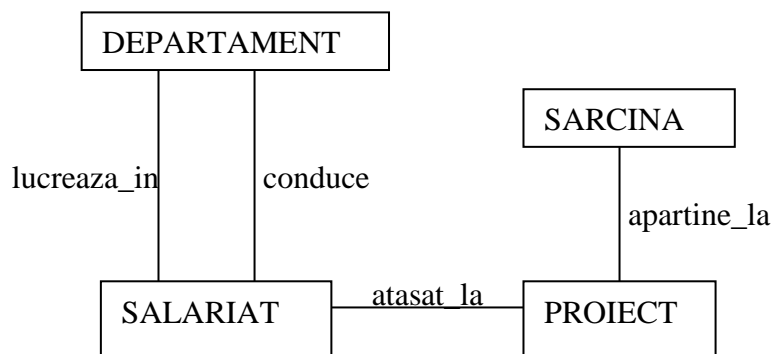
Practic, aceste capcane generează situațiile în care, așa cum a fost proiectat modelul de date, el nu poate să răspundă la anumite interogări. De exemplu, pentru a afla pentru ce prezentare de modă a fost creată o anumită vestimentație, a fost necesară introducerea unei legături între entitățile PREZENTARE și VESTIMENTATIE, care însă a generat redundanță în modelul de date:



4. Diagrame entitate-relație

Ca o consecință a celor prezentate anterior, putem afirma că diagrama E/R este un model neformalizat pentru reprezentarea unui sistem din lumea reală. Reamintim că aceasta este asociată modelului de date conceptual de nivel înalt dezvoltat de Chen (1976).

O **entitate** reprezintă o persoană, un loc, un obiect, un concept, o activitate, un eveniment care este semnificativ pentru ceea ce modelăm.



Observații:

- Entitățile devin tabele în modelele relaționale.
- În general, entitățile se scriu cu litere mari.
- Entitățile sunt substantive, dar nu orice substantiv este o entitate.
- Pentru fiecare entitate este obligatoriu să se dea o descriere detaliată.
- Nu pot exista, în aceeași diagramă, două entități cu același nume, sau o aceeași entitate cu nume diferite.

Pentru fiecare entitate este necesar să se specifice cheia sa primară.

Cheia primară este un identificator unic în cadrul entității, făcând distincție între valori diferite ale acesteia.

Cheia primară:

- trebuie să fie unică și cunoscută la orice moment;
- trebuie să fie controlată de administratorul bazei;
- trebuie să nu conțină informații descriptive, să fie simplă, fără ambiguități;
- să fie stabilă;
- să fie familiară utilizatorului.

O **relație** (asociere) reprezintă o comunicare între două sau mai multe entități. Existența unei relații este subordonată existenței entităților pe care le leagă.

Observații:

- În modelul relațional, relațiile devin tabele speciale sau coloane speciale care referă chei primare.
- Relațiile sunt verbe, dar nu orice verb este o relație.
- Pentru fiecare relație este important să se dea o descriere detaliată.
- În aceeași diagramă pot exista relații diferite cu același nume. În acest caz, le diferențiază entitățile care sunt asociate prin relația respectivă.

Pentru fiecare relație trebuie stabilită cardinalitatea sa.

Există două tipuri de **cardinalitate** a unei relații: maximă și minimă. Pentru stabilirea valorilor acestora, se formulează întrebări al căror predicat este verbul:

- „poate” (pentru cardinalitatea maximă)
- „trebuie” (pentru cardinalitatea minimă)

Exemplu:

Câți salariați **pot** lucra într-un departament? Mulți!

În câte departamente **poate** lucra un salariat? În cel mult unul!

➔ Relația SALARIAT_lucraza_in_DEPARTAMENT are cardinalitatea maximă *many-to-one* (m:1).

Exemplu:

Câți salariați **trebuie** să conducă un departament? Cel puțin unul!

Câte departamente **trebuie** să conducă un salariat? Zero!

➔ Relația SALARIAT_conduce_DEPARTAMENT are cardinalitatea minimă *one-to-zero* (1:0).

Un **atribut** este o proprietate descriptivă a unei entități sau a unei relații.

Observații:

- Trebuie făcută distincția între tipul atributului (devine coloană în modelele relaționale) și valoarea acestuia (devine valoare în coloane).
- Atributele sunt substantive, dar nu orice substantiv este atribut.
- Fiecărui atribut trebuie să i se dea o descriere completă (exemple, contraexemple, caracteristici).
- Pentru fiecare atribut trebuie specificat numele, tipul fizic (*integer*, *float*, *char* etc.), valori posibile, valori implicite, reguli de validare, tipuri compuse.

Pentru reprezentarea **diagramei entitate-relație** au fost stabilite anumite reguli (care nu sunt unice):

1. entitățile sunt reprezentate prin dreptunghiuri;
2. relațiile dintre entități sunt reprezentate prin arce neorientate;
3. attributele care reprezintă chei primare trebuie subliniate sau marcate prin simbolul „#”, plasat la sfârșitul numelui acestor attribute;
4. cardinalitatea minimă este indicată în paranteze, iar cardinalitatea maximă se scrie fără paranteze;
5. nu trebuie specificate toate attributele.

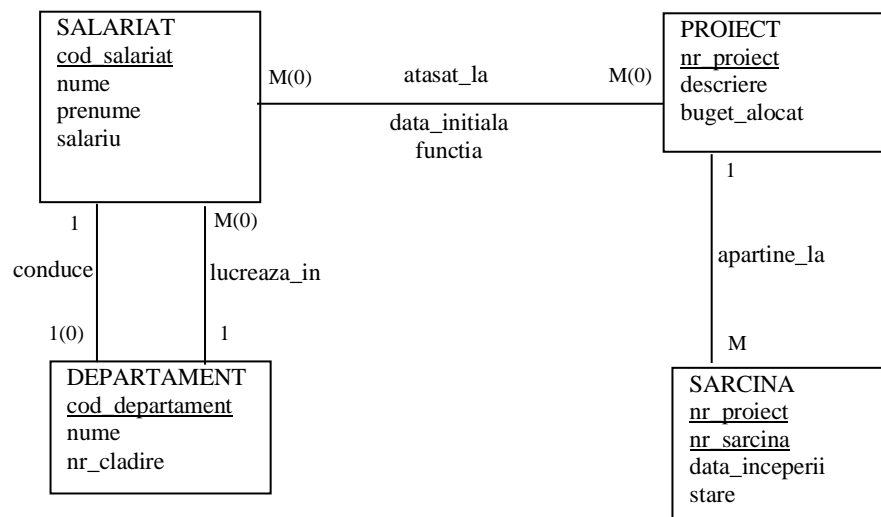


Diagrama E/R.

5. Cazuri speciale de entități, relații, attribute

1. **Entitate dependentă** – nu poate exista în mod independent (SARCINA depinde de PROIECT). Cheia primară a unei entități dependente include cheia primară a entității de care depinde (*nr_proiect*) și cel puțin un atribut specific entității (*nr_sarcina*). (Entitatea dependentă se desenează prin dreptunghiuri cu linii mai subțiri.)
2. **Moștenirea atributelor. Subentitate** (subclasă) – submulțime a unei alte entități, numită **superentitate** (superclasă) (SALARIAT <—> PROGRAMATOR). Subentitatea se desenează prin dreptunghiuri incluse în superentitate. Există o relație între o subentitate și o superentitate, numită ISA („is a”), care are cardinalitatea maximă 1:1 și minimă 1:0. Cheile primare, attributele și relațiile unei superentități sunt valabile pentru orice subentitate. Afirmția reciprocă este falsă.
3. **Generalizare.** Din entități similare care au mai multe attribute comune se pot crea superentități. Aceste superentități conțin attributele comune, iar attributele speciale sunt asignate la subentități. Pentru noile superentități se introduc chei primare artificiale.

4. **Specializare.** După valorile unor atribute clasificatoare se pot determina **clase**. Un grup de subentități reciproc exclusive definește o clasă. Clasele se aliniază în desen vertical.
5. Într-o diagramă E/R se pot defini **relații recursive**. Exemplu din modelul de la laborator: relația de subordonare dintre angajați.
6. Unele relații sunt relative la două entități și le numim de **tip 2**, iar dacă relațiile implică mai mult de două entități, le vom numi de **tip 3**. Trei relații de tip 2 sunt diferite de o relație de tip 3. Rupând o relație de tip 3 în trei relații de tip 2, pot apărea informații incorecte. Exemplul prezentat la curs: relația de tip 3 dintre entitățile ANGAJAT, PROIECT, FUNCTIE.
7. Trebuie **excluse** din model **relațiile indirecte** deoarece ele pot conduce la redundanță în baza de date.
8. **Atributele derivabile** trebuie **eliminate** și introduse expresii prin care aceste atribute pot fi calculate.
9. **Relație sau atribut?** Dacă un atribut al unei entități reprezintă cheia primară a unei alte entități, atunci el referă o relație (*cod_departament* în tabelul SALARIAT).
10. **Entitate sau relație?** Se cercetează cheia primară. Dacă aceasta combină cheile primare a două entități, atunci este vorba de o relație. (cheia primară a relației *asociat_la* combină *cod_salariat* cu *nr_proiect*, prin urmare, *SALARIAT_asociat la_PROIECT* va defini o relație și nu o entitate).
11. Un **atribut indirect** este inoportun. El nu descrie real relația sau entitatea. Prin urmare, atributele indirecte trebuie **reassignate**. De fapt, un atribut indirect este un caz special de relație indirectă care trebuie eliminată pentru că introduce redundanță în date (numărul clădirii în care lucrează un salariat este un atribut al entității DEPARTAMENT și nu este o caracteristică a entității SALARIAT).
12. Există **attribute opționale**, a căror valoare este uneori necunoscută, alteleori neaplicabilă. Aceste atribute trebuie introduse la **subentități** (comisionul pentru deplasare și zona de lucru sunt atribute specifice unui agent teritorial și trebuie introduse la subentitatea AGENT_TERITORIAL).

6. Algoritmul pentru proiectarea diagramei entitate-relație

1. Identificarea entităților din cadrul sistemului analizat;
2. Identificarea relațiilor dintre entități și stabilirea cardinalității;
3. Identificarea atributelor aferente entităților și asocierilor dintre entități;
4. Stabilirea atributelor de identificare a entităților (stabilirea cheilor).

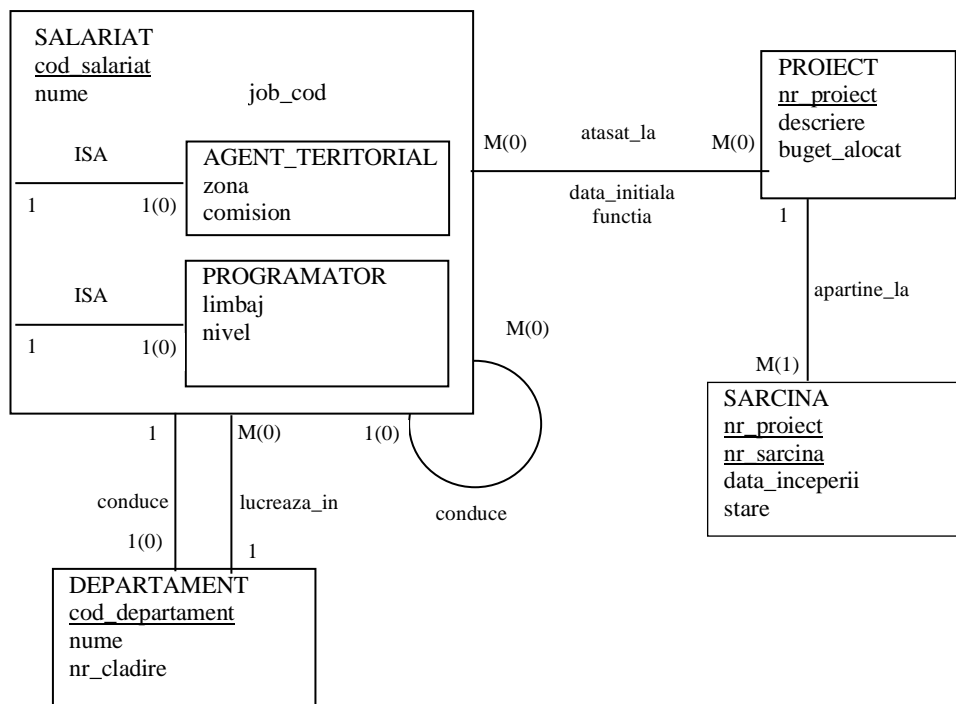


Diagrama E/R.

Modelul EER (modelul E/R extins) = Modelul E/R + concepte adiționale (subclasă, superclasă, moștenire, specializare, generalizare).

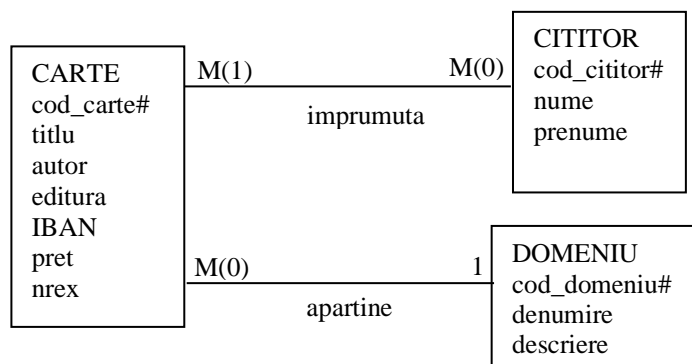
7. Studii de caz

7.1 Gestionarea activităților de împrumut dintr-o bibliotecă

S-a presupus (restrictiv) că într-o zi un cititor nu poate împrumuta, de mai multe ori, aceeași carte. Din motive didactice, modelul prezintă unele anomalii. Care sunt acestea?

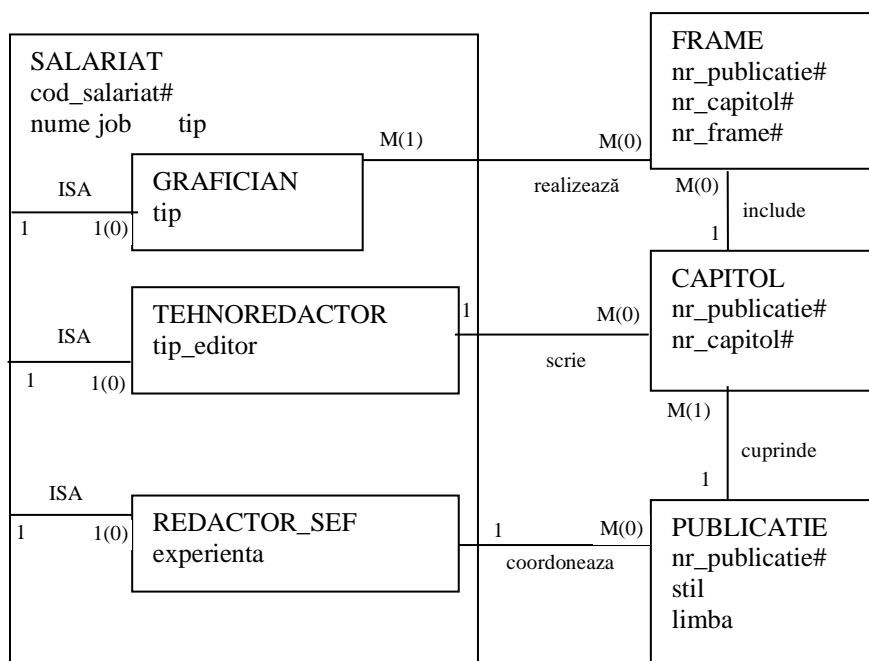
Entitățile și relațiile care intervin în acest model sunt următoarele:

1. CARTE (entitate independentă) – orice carte care se găsește în inventarul bibliotecii. Cheia primară este atributul *cod_carte*.
2. CITITOR (entitate independentă) – orice cititor care poate împrumuta cărți. Cheia primară este atributul *cod_cititor*.
3. DOMENIU (entitate independentă) – domeniul căruia îi aparține o carte. Cheia primară este atributul *cod_domeniu*.
4. IMPRUMUTA – relație având cardinalitatea *m:m* care leagă entitățile CITITOR și CARTE.
5. APARTINE – relație care leagă attributele CARTE și DOMENIU. Relația are cardinalitatea maximă *m:1*, iar cardinalitatea minimă *1:1*.



7.2 Gestionarea activităților de editare dintr-o editură

Se analizează activitatea dintr-o editură referitoare la tehnoredactarea textelor, realizarea elementelor grafice, coordonarea unor publicații.



7.3 Gestionarea activităților unei companii de construcții

Modelul de date prezentat furnizează informații legate de obiective de execuție, investitori, executanți, șantiere, contracte etc., necesare unui manager al unei companii de construcții.

STANȚA (nr_stanță, împărat emitent, valoare nominală, an emitere, monedă, legenda de pe avers, legenda de pe revers) == > attribute ale entității **STANTA**

Completați cardinalitatea!

Temă:

Repartiția studenților în căminele universității – diagrama E/R (entități și chei primare, relații și cardinalități, attribute ale entităților și relațiilor).