



CURS 2

Modelare entitate - relatie

- Model = reprezentare a obiectelor și evenimentelor lumii reale și a asocierilor dintre ele.
- Abstractizare asupra aspectelor semnificative ale unei „întreprinderi”, ale unui sistem real
- Caz particular al deosebirii uzuale dintre logic și fizic.
- 3 tipuri de modele (care descriu aspecte ... ale procesului modelat):
 - statice
 - dinamice
 - functionale
- Model de date = o colecție integrată de concepte necesare descrierii:
 - datelor
 - relațiilor dintre ele
 - constrângerilor existente asupra datelor sistemului real analizat
- Modelarea unei baze de date = trecerea de la percepția unor fapte din lumea reală la reprezentarea lor prin date
- Modelul de date trebuie:

- să reflecte fidel fenomenele ale lumii reale
- să urmărească evoluția acestei lumi
- să permită comunicarea dintre fenomenele lumii reale
- Modelul de date are 3 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 coerentă datelor
- Modelarea semantică a datelor are 4 etape:
 1. identificarea unei multimi de concepte semantice utile descrierii lumii reale
 - Se presupune că lumea reală (modelul real analizat) este formată din entități care fiecare are anumite proprietăți (identitate, că există legături, corelații între entități).
 2. cautarea unei multimi de obiecte formale, simbolice, utilizate pentru reprezentarea conceptelor semantice anterioare
 3. se dau reguli de integritate formale și generale (constrângeri) care să reflecte restricțiile la care este supus modelul
 4. definirea unei multimi de operatori formali prin care pot fi prelucrate și analizate obiectele formale.

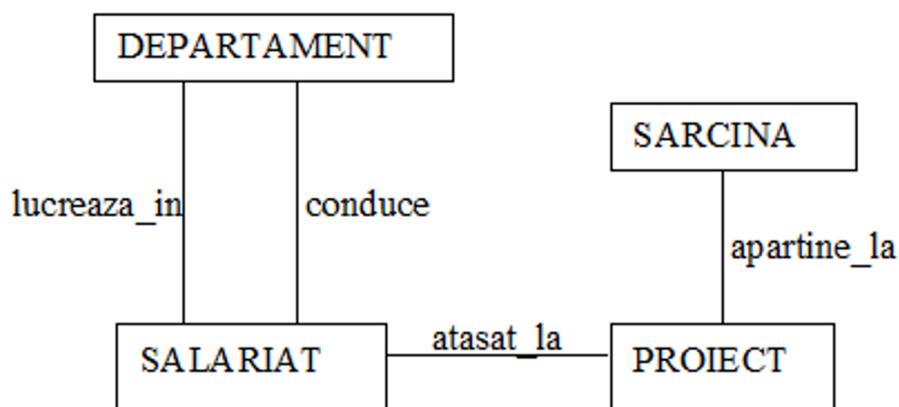
Modelul Entitate-Relație

- abordare a modelării semantice
- model de date conceptual, pt a usura proiectarea BD
- de nivel înalt, independent de platforma hardware utilizată și de tipul SGBD-ului
- reprezentat grafic prin diagrame E/R

- modelul E/R imparte elementele unui sistem real în 2 categorii:
 - entități
 - relații (legături, asocieri) între entități
- entitățile și legăturile au caracteristici (attribute)
- conceptul de relație, în sensul de asociere, care intervine în definirea diagramei E/R ≠ conceptul de relație care este specific modelului relațional

Diagrame entitate - relație

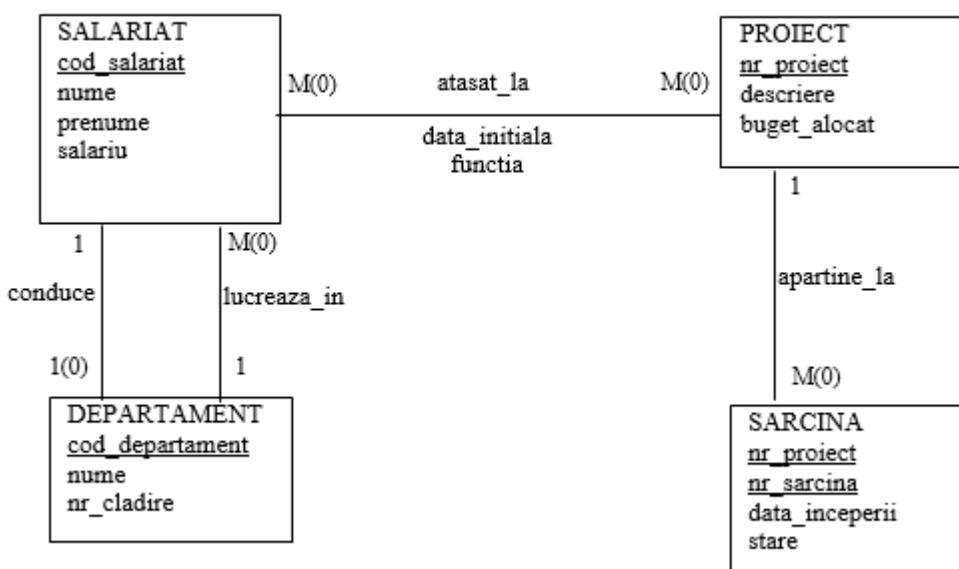
- Diagrama E/R = model de date conceptual de nivel înalt neformalizat pentru reprezentarea unui sistem din lumea reală
- Entitate = persoană, loc, concept, activitate, eveniment care este semnificativ pentru ceea ce modelăm



- Entitățile:
 - vor deveni tabele în modelul relațional
 - scrisă cu litere mari
 - substrantive
 - obligatoriu o descriere detaliată
 - în aceeași diagramă nu pot exista 2 entități cu același nume sau aceeași entitate cu nume diferite

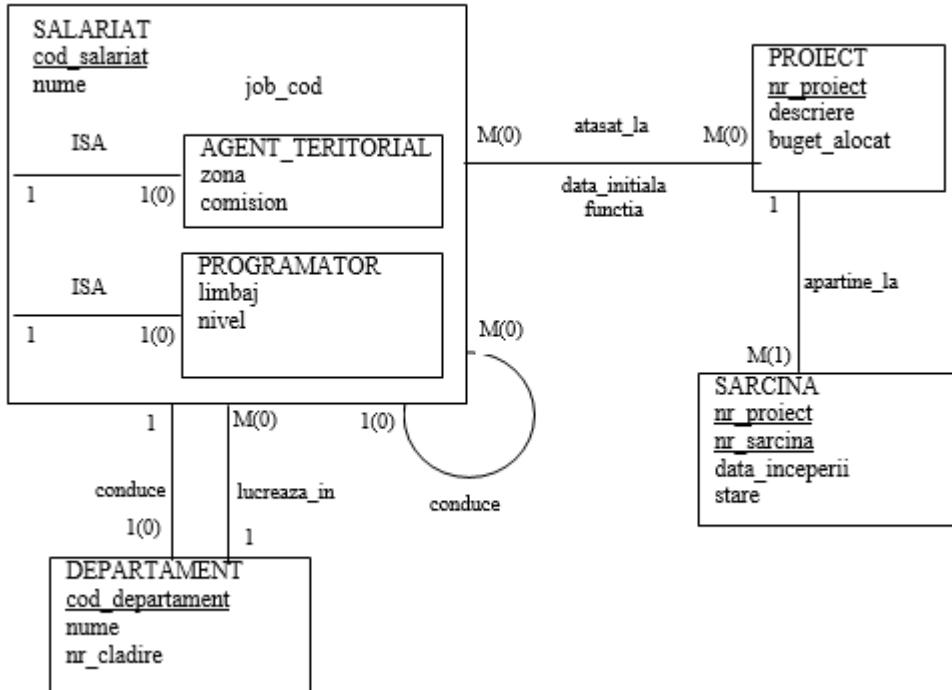
- Cheia primă/PK = un identificator unic în cadrul entității, făcând distincție între valori diferite ale acesteia
- Cheia primara:
 - unica
 - cunoscuta în orice moment
 - fără inf. descriptive
 - simplă, fără ambiguitati
 - stabila
 - controlată de administratorul BD
 - familiară utilizatorului
- Relație = comunicare între 2 sau mai multe entități
- Existența unei relații este subordonată existenței entităților pe care le leagă
- în modelul relational, relațiile devin tabele speciale sau coloane speciale care referă chei primare
- relațiile sunt verbe
- fiecare relație trebuie să aibă o descriere detaliată
- aceeași diagramă poate avea relații diferite cu același nume
- fiecare relație are o cardinalitate (max/min) = nr de tupluri ce aparțin relației
- POATE (cardinalitate maxima) → TREBUIE (cardinalitate minima):
 - Cati salariati **pot** lucra intr-un depart? Multi
 - In cate depart **poate** lucra un salariat? In cel mult unul
 - relația este many-one (n:1)
 - Cati salariati **trebuie** sa conduca un depart? Cel putin unul
 - Cate depart **trebuie** sa conduca un salariat? Zero
 - relația este one-zero (1:0)
- Atribut = proprietate descriptivă a unei entități sau a unei relații
- Atributul:
 - devine coloana în modelele relationale

- valoarea lui devine valoare in coloane
- substantive
- ii trebuie o descriere completa: ex, contra-ex, caracteristici
- fiecare atribut are specificat:
 - numele
 - tipul fizic (integer, float, char etc.)
 - valori posibile
 - valori implicite
 - reguli de validare
- Reguli (neunice) pentru proiectarea unei diagrame E/R:
 - entitatile → reprezentate prin dreptunghiuri
 - relatiile → reprezentate prin arce neorientate
 - atributele chei primare → subliniate sau marcate prin "#", plasat la sfarsitul numelui acestora
 - card min → indicata de paranteze
 - card max → fara paranteze
 - nu e necesar specifice toate atributele in diagrama



- Cazuri speciale de entitati, relatii, atribute:
 - dependenta
 - Entitate dependenta → nu poate exista in mod independent
 - Cheia primara a unei entitati dependente include cheia primara a sursei (nr_proiect) si cel putin de un atribut caracteristic entitatii (nr_sarcina)
 - Entitatea dependenta se deseneaza prin dreptunghi cu linii mai subtiri
 - mostenirea atributelor
 - Subentitatea/subclasa = submultime a unei entitati (superentitate/superclasa)
 - ex: Salariat ↔ programator
 - Subentitatea este desenata prin dreptunghiuri incluse in superentitate
 - ISA = relatia subentitate-superentitate cu card max 1:1 si card min 1:0
 - Cheile primare, atributele si relatiile unei superentitati sunt valabile pentru orice subentitate (reciproca e falsa)
 - generalizare
 - din entitati similare cu mai multe atribute comune se pot crea superentitati
 - superentitatile contin atribute comune, iar atributele speciale sunt asignate la subentitati
 - pt noile superentitati se introduc chei primare artificiale
 - specializare
 - dupa valorile unor atribute clasificatoare se pot determina clase
 - un grup de subentitati reciproc exclusive defineste o clasa
 - clasele se aliniaza in desen vertical
 - relatii recursive
 - relatii de tip 2,3

- relatiile relative la 2 entitati → de tip 2
- relatiile relative la 2+ entitati → de tip 3
- 3 relatiile de tip 2 ≠ o relatie de tip 3
- rupand o relatie de tip 3 in 3 relatiile de tip 2 → informatii incorecte
- relatie sau atribut
 - dacă un atribut al unei entități reprezintă cheia primară a unei alte entități, atunci el referă o relație
 - ex: cod_departament în tabelul SALARIAT
- entitate sau relatie
 - se cercetează cheia primară
 - dacă PK combină cheile primare a două entități, atunci este vorba de o relație
 - ex: 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
- Relatiile indirecte → excluse din model pt ca pot conduce la redundanta in BD
- Atributele derivate → eliminate si introduse expresii prin care acestea pot fi calculate
- Algoritmul pt proiectarea diagramei E/R:
 1. identificare entitatilor din cadrul sistemului analizat
 2. identificarea relatiilor dintre dintre entitati si stabilirea cardinalitatii
 3. identificarea atributelor aferente entitatilor si asociierilor dintre ele
 4. stabilirea atributelor de indentificare a entitatilor → stabilirea PKs



- Aceeași realitate poate fi percepță diferit de către diversi analiști pentru un același sistem → modele structurale distincte
- Pasii urmatori ⇒ model relational care va elibera anumite clase anomalii ce pot apărea în proiectarea modelului de date
 - construirea diagramei conceptuale
 - obținerea schemelor relationale
 - normalizarea acestora

Modelul E/R extins

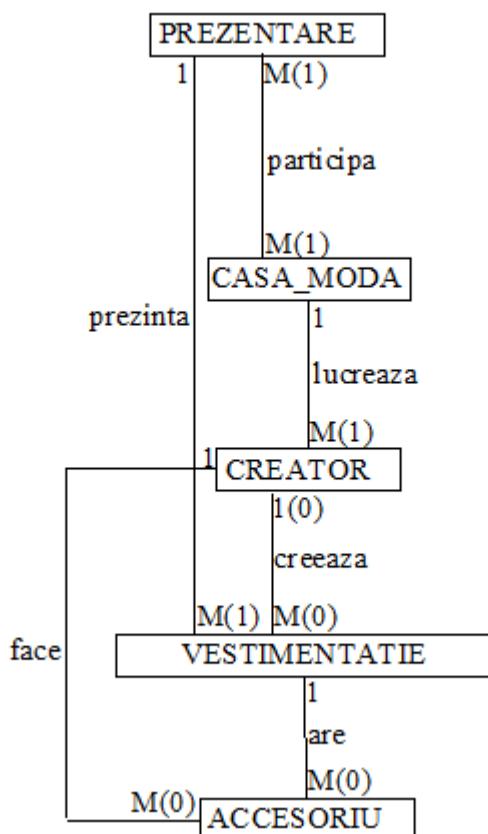
- Conceptele de bază ale modelării E/R nu sunt suficiente pentru a reprezenta cerințe complexe
- Modelul E/R susținut cu concepte semantice adiționale definește modelul E/R extins (EER)
 - include toate conceptele modelului original
 - + concepte adiționale de subclasă, superclasă, moștenire, specializare, generalizare

- Superclasa/superentitatea = o entitate care include subclase/subentități distincte, ce trebuie reprezentate în modelul de date
- Subclasa are un rol distinct și, evident, este membră a unei superclase
- O subclasă, fiind o entitate, poate să posede propriile subclase
- O entitate împreună cu subclasele ei, subclasele acestora și aşa mai departe definește o ierarhie de tip (ierarhie de specializare).
 - ex: ANGAJAT_TEMP reprezintă o superclasă pentru entitatea MODEL
- Specializarea = procesul de maximizare a diferențelor dintre membrii unei entități, prin identificarea caracteristicilor distinctive ale acestora
 - constrângere de disjuncție = dacă subclasele unei specializări sunt disjuncte, atunci o entitate poate fi membră doar a unei subclase a acesteia (constrângere de disjuncție).
 - constrângere de participare = specializare cu participare totală → fiecare entitate din superclasă trebuie să fie membră a unei subclase din specializare
 - o specializare cu participare parțială → nu este necesar ca o entitate să aparțină vreunei subclase a acesteia.
 - ex: există salariați în PERS_CONTACT care nu aparțin niciunei subentități ale acesteia
- Generalizarea = procesul de minimizare a diferențelor dintre entități, prin identificarea caracteristicilor comuneale acestora
- Generalizarea are ca rezultat identificarea unei superclase generalizate din subclasele inițiale

Deficiente ale modelului E/R

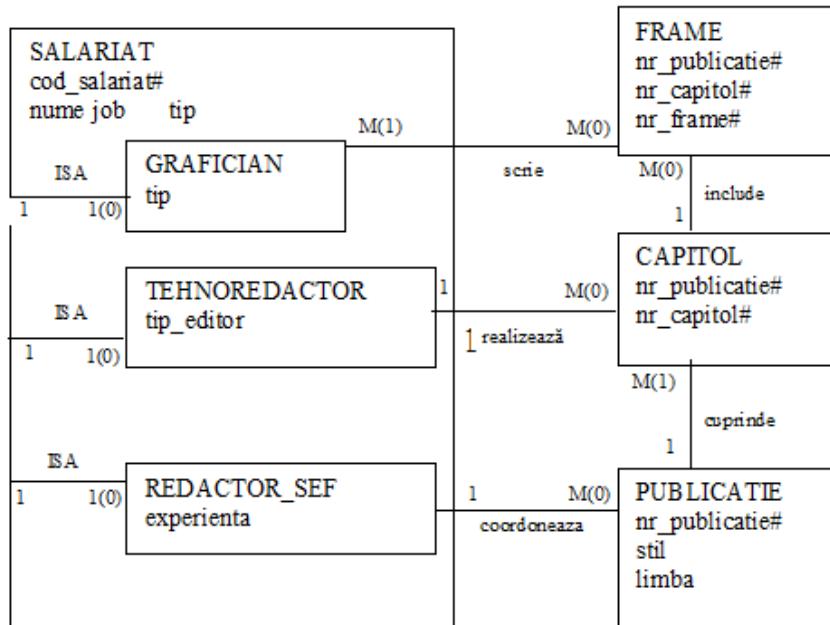
- Cauza: interpretare eronată a sensului unei relații → capcane de conectare
- Posibil să necesite restructurarea modelului
- 2 clase de capcane de conectare:

- de intrerupere
- în evantai
- Capcane de intrerupere = modelul sugerează existența unei relații între entități, dar nu există o cale între anumite apariții ale entităților.
 - apare acolo unde există o relație cu participare parțială (card min=0), care face parte din calea dintre entitățile ce sunt legate
- Capcane în evantai = modelul ia în considerare o relație între entități, dar calea dintre anumite apariții ale entităților este ambiguă
 - apar când două sau mai multe relații one-to-many provin din aceeași entitate
- Aceste capcane generează situațiile în care, așa cum a fost proiectat modelul de date, el nu poate să răspundă la anumite interogări
- Ex capcana de intrerupere: 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

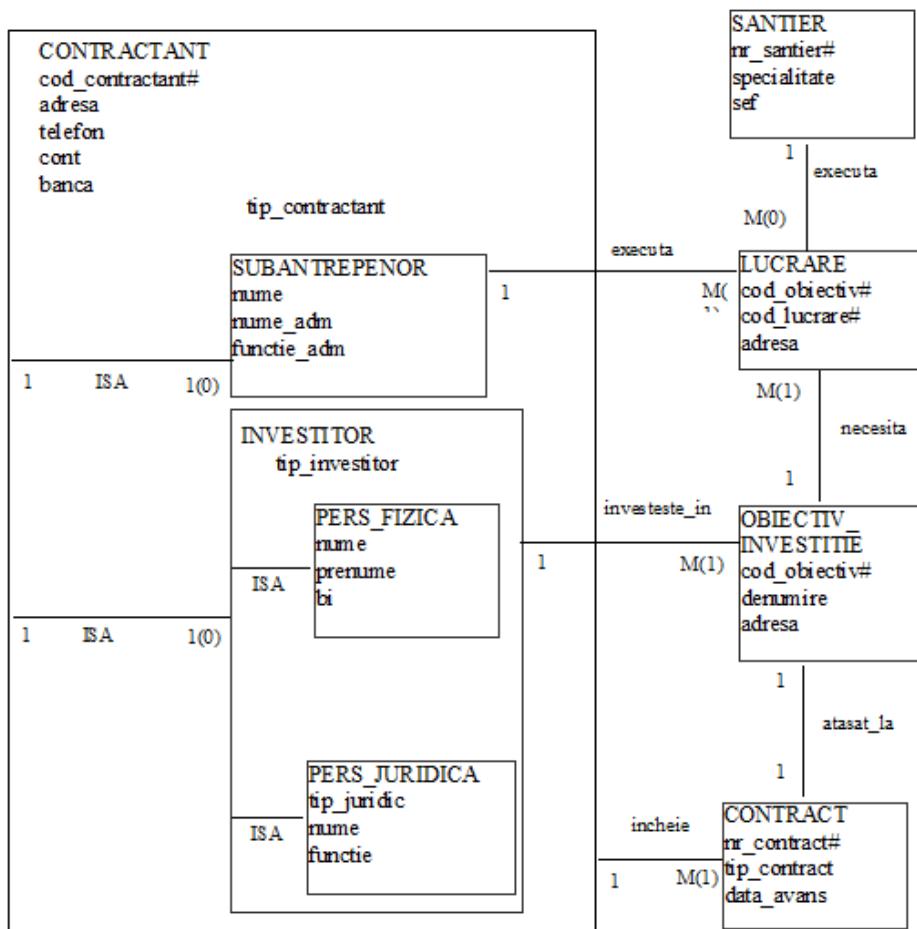


Exemple de diagrame E/R

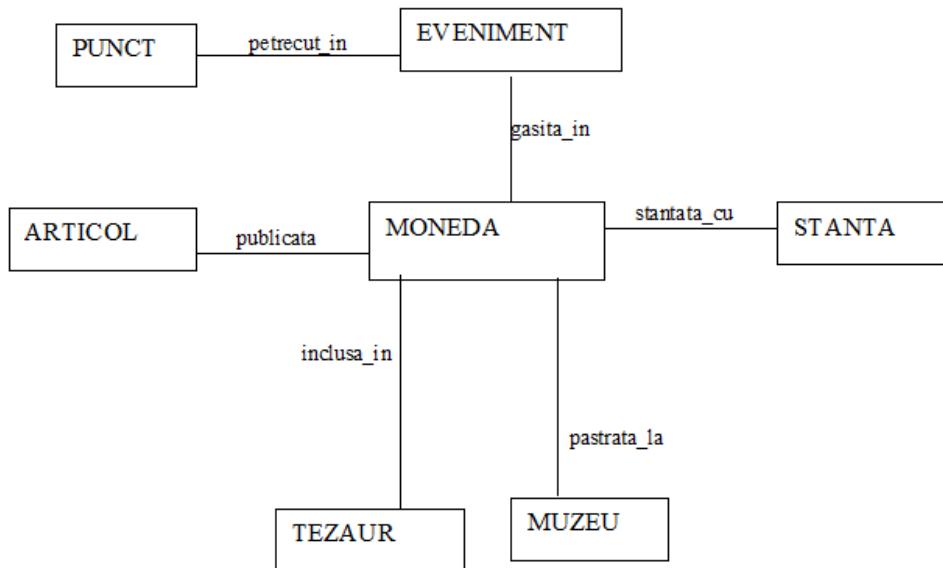
- Exemplu 1 → gestiunea activitatilor de editare dintr-o editura



- Exemplul 2 → gestiunea unei firme de constructii



- Exemplul 3 → descoperiri de monede antice din România
 - STANȚA (nr_stanță, împărat emitent, valoare nominală, an emitere, monetăria, legenda de pe avers, legenda de pe revers) → atrbute ale entității STANTA



- Exemplul 4 → evidența școlilor de șoferi din România
 - Completăți relațiile (lucreaza_la, conduce, sustine, asista, instruiește) dintre entități și specificați cardinalitatea

