

BAZE DE DATE

CURS 3

Modelarea entitate-relație (continuare).

Modelul relațional.

Observații temă

- Cardinalitățile se pun **la ambele capete** ale relației (nu doar la mijloc).
- În anumite reprezentări, există "**roluri**" ale relației => regăsim verbe la ambele capete => OK (dar se recomandă notația de la curs).
- Aceeași observație pentru alte reprezentări alternative, precum foot-crow. Nu o folosim, ci recurgem la **reprezentarea cu cardinalitate maximă și minimă!**
- Greșeli la cardinalitate (valorile au fost inversate între capete).
- Regulile modelului trebuie să reflecte cât mai bine realitatea.
- **CNP** nu trebuie folosit drept cheie primară – nu este o alegere adecvată din 2 motive: nu este ușor de folosit și reprezintă o informație confidențială!
- Observație referitoare la dinamica modelului: depinde de relevanța intervalului de timp pentru care proiectăm! Vezi exemplul de la laborator: este important acolo să știm istoricul angajaților în companie dar nu avem un istoric al șefilor de departamente (daca nu cumva se poate deduce acest lucru tot din istoricul job-urilor)
 - Ce relație aș pune între entitățile PASAGER și AUTOBUZ? Dar STUDENT-CAMERA?
- Validez întotdeauna punând întrebări modelului !

Exemplul 1

Gestiunea activităților de împrumut dintr-o bibliotecă

- **Entitățile și relațiile** care intervin în acest model sunt următoarele:
 - **CARTE** (entitate independentă) – orice carte care se găsește în inventarul bibliotecii. Cheia primară este atributul *cod_carte*.
 - **CITITOR** (entitate independentă) – orice cititor care poate împrumuta cărți. Cheia primară este atributul *cod_cititor*.
 - **DOMENIU** (entitate independentă) – domeniul căruia îi aparține o carte. Cheia primară este atributul *cod_domeniu*.
 - **IMPRUMUTA** – relație care leagă entitățile CITITOR și CARTE.
 - **APARTINE** – relație care leagă attributele CARTE și DOMENIU.
- **Obs:** S-a presupus (restrictiv) că într-o zi un cititor nu poate împrumuta de mai multe ori aceeași carte -> regulă a modelului.
- **Ce cardinalități au cele două relații?**
- **Reprezentați diagrama E/R a acestui model.**

Exemplul 2

Gestiunea campionatelor de fotbal ale diferitelor țări

- **Entitățile** modelului sunt următoarele:
 - **ECHIPA, SPONSOR, MECI, ETAPA, CAMPIONAT**
- Precizați relațiile care există între aceste entități și cardinalitatea lor.
- Reprezentați diagrama E/R a acestui model.

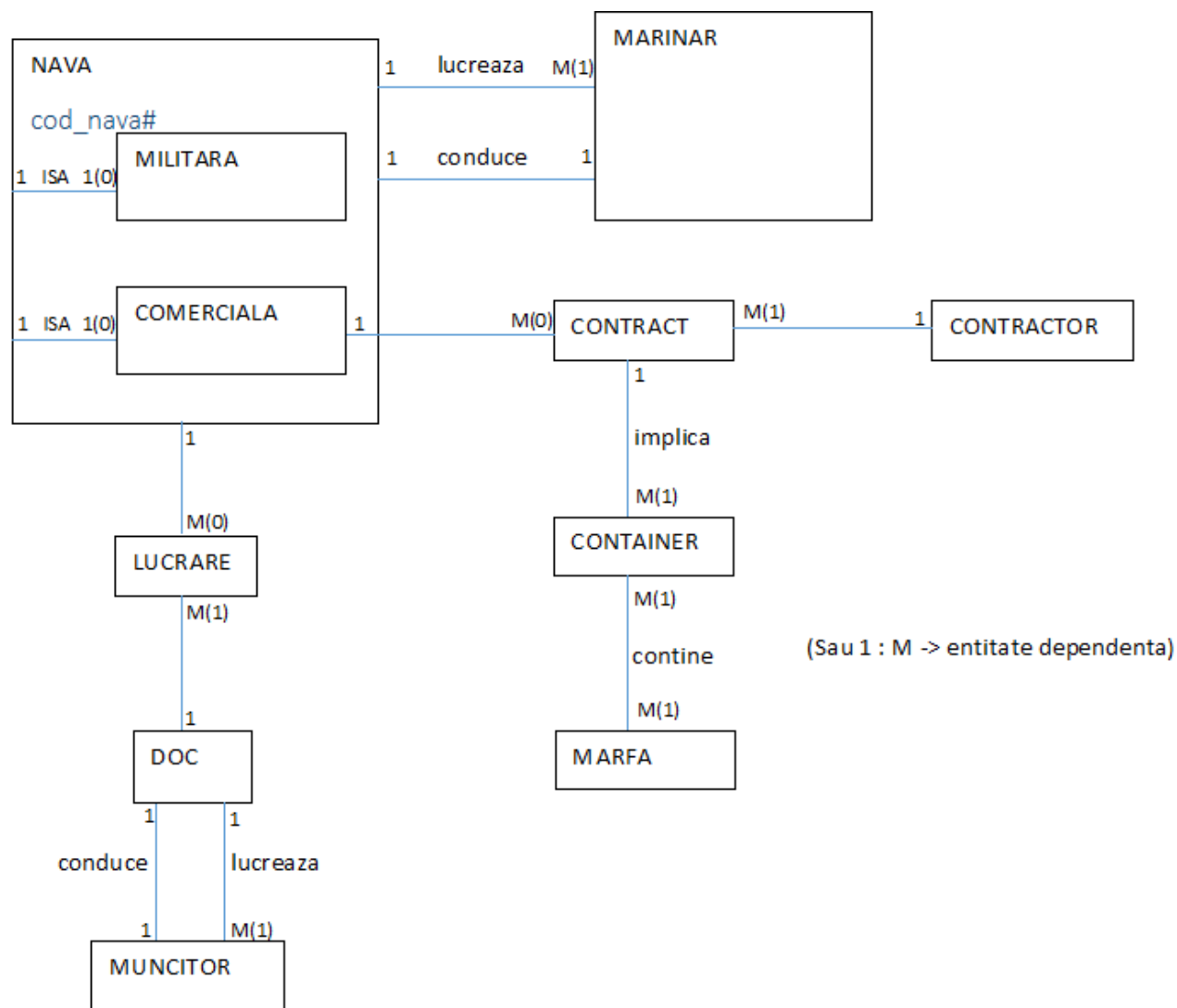
Exemplul 3

Gestiunea activităților dintr-un port referitoare la servirea navelor

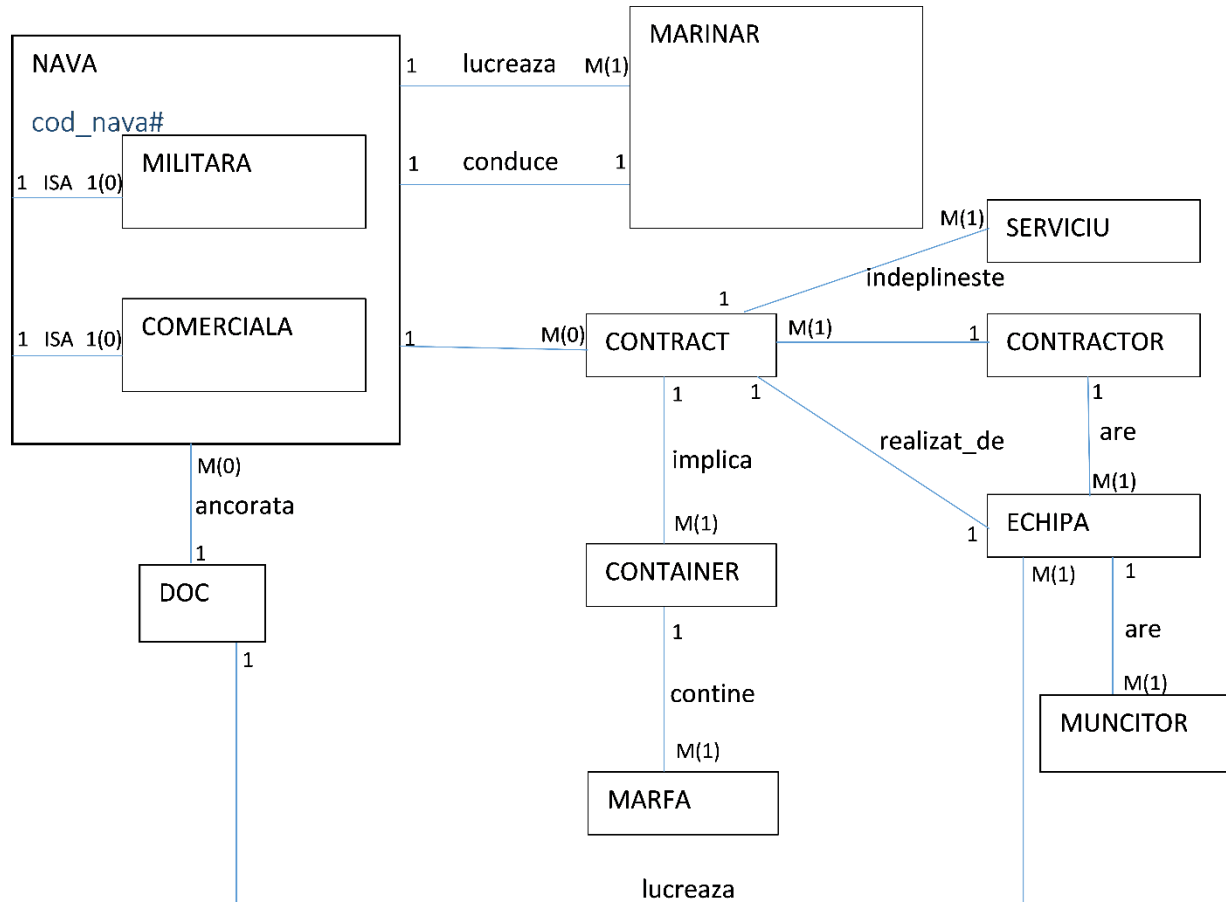
- Întrebări al căror răspuns trebuie să se regăsească în descrierea și regulile de funcționare ale modelului:
 - Ce tipuri de nave există?
 - Ce servicii pot fi oferite navelor (tipuri de lucrări)?
 - Ce attribute are entitatea DOC?
 - Ce tipuri de personal pot intra în componența diferitelor echipe?
 - Etc .
 - Exemple de specificații:
 - ☐ Pentru nave sunt semnate contracte de încărcare/descărcare cu contractori
 - ☐ Marfa ajunge în port în containere
 - Dați exemple de alte întrebări al căror răspuns trebuie specificat în descriere și reguli.
- **Reprezentați diagrama E/R a unui model ce tratează cel puțin aspectele de mai sus.**

Exemplul 3 (continuare)

Care dintre
variantele
următoare este
corectă? Corectați.



Exemplul 3 (continuare)



PROIECTAREA BAZELOR DE DATE RELAȚIONALE



Modelarea entitate-
relație (E/R)



Diagrame entitate-
relație

Modelul relațional

Modelul relațional



Modelul relațional

- Conceput și dezvoltat de E.F. **Codd**
- Model formal de organizare conceptuală a datelor, destinat reprezentării legăturilor dintre date, bazat pe **teoria matematică a relațiilor**.
- **Modelul relațional este alcătuit numai din relații** și prin urmare, orice interogare asupra bazei de date este tot o relație.
- Cercetarea în domeniu → 3 mari proiecte (*System R, INGRES, PRTV*)

Modelul relațional

- **Calități:**
 - este **simplu**;
 - **riguros** din punct de vedere matematic;
 - **nu este orientat** spre sistemul de calcul.
- **Definirea unui SGBD relațional presupune:**
 - prezentarea datelor în **tabele** supuse anumitor operații de tip **proiecție, selecție, reuniune, compunere, intersecție** etc.
 - un sistem de baze de date ce suportă un limbaj de tip **SQL** – *Structured Query Language*;
 - un sistem de baze de date care respectă **principiile modelului relațional** introdus de E.F. Codd.

Modelul relațional

- **Caracteristicile** unui model relațional:

- **structura** relațională a datelor;
- **operatorii** modelului relațional;
- **regulile de integritate** care guvernează folosirea cheilor în model.

➤ Aceste trei elemente corespund celor trei componente ale ingineriei *software*:

➤ **informație**

➤ **proces**

➤ **integritate.**

Modelul relațional

Structura datelor

- Definirea noțiunilor de **domeniu**, **relație**, **schemă relațională**, **valoare null** și **tabel vizualizare** (*view*).
- Conceptele utilizate pentru a descrie **formal**, **uzual** sau **fizic** elementele de bază ale organizării datelor:

Formal	Uzual	Fizic
Relație	Tabel / tabelă	Fișier
Tuplu	Linie	Înregistrare
Atribut	Coloană	Câmp
Domeniu	Tip de date	Tip de date

Modelul relațional

- **Domeniu** – mulțime de valori care poate fi definită fie enumerând elementele componente, fie definind o proprietate distinctivă a domeniului valorilor.
- Fie D_1, D_2, \dots, D_n domenii finite, nu neapărat disjuncte. **Produsul cartezian** $D_1 \times D_2 \times \dots \times D_n$ al domeniilor D_1, D_2, \dots, D_n este definit de mulțimea **tuplurilor** (V_1, V_2, \dots, V_n) , unde $V_1 \in D_1, V_2 \in D_2, \dots, V_n \in D_n$. Numărul n definește **aritatea tuplului**.
- O **relație** R pe mulțimile D_1, D_2, \dots, D_n este o **submulțime a produsului cartezian** $D_1 \times D_2 \times \dots \times D_n$, deci este o **mulțime de tuple**.
- Caracteristicile unei relații → comentat curs!

Modelul relațional

- Definirea unei relații se referă la mulțimi care variază în timp.
- Este necesar un element invariant în timp: structura relației (schema relațională).
- Mulțimea numelor atributelor corespunzătoare unei relații R definește schema relațională a relației respective.
- Vom nota schema relațională prin $R(A_1, A_2, \dots, A_n)$. Exemplu!
- Putem reprezenta o relație printr-un tabel bidimensional.
 - O coloană corespunde de fapt unui atribut.
 - Numărul atributelor definește gradul relației, iar numărul de tupluri din relație definește cardinalitatea relației.

Modelul relațional

Exemplu (crearea unui tabel în SQL):

```
CREATE TABLE salariat (  
    cod_salariat      NUMBER(4) PRIMARY KEY,  
    nume              VARCHAR2(25),  
    prenume           VARCHAR2(20),  
    salariu           NUMBER(8, 2),  
    sef               NUMBER(4),  
    job_cod            VARCHAR2(6),  
    cod_departament   NUMBER(3));
```


Modelul relațional

Valoare *null*

- Când se inserează tupluri într-o relație, de multe ori un atribut este **necunoscut** sau **neaplicabil**.
- Pentru a reprezenta acest atribut a fost introdusă o valoare **convențională** în relație, și anume valoarea ***null***.
- Este necesară o aritmetică și o logică nouă care să cuprindă acest element.
 - Rezultatul operatorilor aritmetici sau logici este *null* când unul din argumente este *null*. Comentat excepții!
 - Prin urmare, „*null* = *null*” are valoarea *null*, iar $\neg \text{null}$ este *null*.

Modelul relațional

- Tabelele de adevăr pentru operatorii AND și OR:

AND	T	F	Null
T	T	F	Null
F	F	F	F
Null	Null	F	Null

OR	T	F	Null
T	T	T	T
F	T	F	Null
Null	T	Null	Null

Modelul relațional

Tabelul vizualizare

- **View**, filtru, relație virtuală, vedere
- constituie un filtru relativ la unul sau mai multe tabele, care conține numai informația necesară unei anumite abordări sau aplicații.
 - Securitate, reactualizări → comentat la curs!
- Vizualizarea este **virtuală** deoarece datele pe care le conține nu sunt în realitate memorate într-o bază de date. Este memorată numai **definiția** vizualizării.
 - Vizualizarea nu este definită explicit, ca relațiile de bază, prin mulțimea tuplurilor componente, ci **implicit**, pe baza altor relații prin intermediul unor expresii **relaționale**.
 - Stabilirea efectivă a tuplurilor care compun vizualizarea se realizează prin **evaluarea expresiei** atunci când utilizatorul se referă la această vizualizare.

Modelul relațional

Exemplu (crearea unei vizualizări în *SQL*):

```
CREATE VIEW programator(nume,departament)
    AS SELECT      nume, cod_departament
        FROM      salariat
        WHERE      job_cod='IT_PROG';
```

Modelul relațional

Reguli de integritate

→ **asertiuni** pe care datele conținute în baza de date trebuie să le satisfacă.

➤ Trebuie făcută **distincția** între:

- **regule structurale** inerente modelării datelor;
- **regule de funcționare** specifice unei aplicații particulare.
- Există **trei tipuri de constrângeri structurale**:

- de cheie
- de referință
- de entitate

constituie **mulțimea minimală de reguli de integritate**
pe care **trebuie** să le respecte un SGBD relațional

Restricțiile de integritate minimale sunt definite în raport cu **noțiunea de cheie a unei relații**.

Modelul relațional

- O **mulțime minimală** de attribute ale căror valori **identifică unic** un tuplu într-o relație reprezintă o **cheie** pentru relația respectivă.
 - Fiecare relație are cel puțin o cheie.
 - Una dintre **cheile candidat** va fi aleasă pentru a identifica efectiv tupluri și ea va primi numele de **cheie primară**.
 - Cheia primară nu poate fi reactualizată.
 - Attributele care reprezintă cheia primară sunt fie subliniate, fie urmate de semnul # în diagrama E/R și în schemele relaționale.

Modelul relațional

- O **cheie** identifică linii și este diferită de un **index** care localizează liniile.
- O **cheie secundară** este folosită ca index pentru a accesa tupluri.
- Un grup de atribute din cadrul unei relații care conține o **cheie** a relației poartă numele de **supercheie**.
- Fie schemele relaționale **$R1(P1, S1)$** și **$R2(S1, S2)$** , unde **$P1$** este cheie primară pentru **$R1$** , **$S1$** este cheie secundară pentru **$R1$** , iar **$S1$** este cheie primară pentru **$R2$** . În acest caz, vom spune că **$S1$** este **cheie externă** (cheie străină) pentru **$R1$** .

$R1(P1\#, S1)$



$R2(S1\#, S2)$

Modelul relațional

Modelul relațional respectă **trei reguli de integritate structurală**.

- **Regula 1 – unicitatea cheii**. Cheia primară trebuie să fie **unică** și **minimală**.
- **Regula 2 – integritatea entității**. Atributele cheii primare trebuie să fie **diferite de valoarea null**.
- **Regula 3 – integritatea referirii**. O cheie externă trebuie să fie ori **null în întregime**, ori să corespundă unei **valori a cheii primare asociate**.

Proiectarea modelului relațional

Proiectarea modelului relațional

Transformarea entităților

- **Entitățile independente** devin **tabele independente**.
 - Cheia primară nu conține chei externe.
- **Entitățile dependente** devin **tabele dependente**.
 - Cheia primară a entităților dependente conține cheia primară a entității de care depinde (cheie externă) **plus** unul sau mai multe attribute adiționale.
- **Subentitățile** devin **subtabele**.
 - Cheia externă se referă la supertabel, iar cheia primară este această cheie externă (cheia primară a subentității PROGRAMATOR este *cod_salariat* care este o cheie externă).

Proiectarea modelului relațional

Transformarea relațiilor (1/2)

- Relațiile **1:1** și **1:n** devin **chei externe**.
 - Relația **1:1** plasează cheia externă **în tabelul cu mai puține linii**.
 - Relația **1:n** plasează cheia externă **în tabelul corespunzător cardinalității *many***.
 - Relația *conduce* devine coloană în tabelul DEPARTAMENT, iar relația *lucreaza_in* devine coloană în tabelul SALARIAT.
 - Simbolul „**x**” indică plasamentul cheii externe, iar simbolul „**x**” exprimă faptul că această cheie externă este conținută în cheia primară.

Proiectarea modelului relațional

Transformarea relațiilor (2/2)

- Relația **m:n** devine un tabel special, numit **tabel asociativ**, care are **două chei externe** pentru cele două tabele asociate.
 - Cheia primară este compunerea acestor două chei externe **plus eventuale coloane adiționale**.
 - Tabelul se desenează punctat.
- Relațiile **de tip trei** devin **tabele asociative**.
 - Cheia primară este compunerea a trei chei externe plus eventuale coloane adiționale.

Proiectarea modelului relațional

Transformarea atributelor

- **Un atribut singular** devine o **coloană**.
- **Atributele multiple** devin **tabele dependente** ce conțin cheia primară a entității și atributul multiplu.
 - Cheia primară este o cheie externă, **plus** una sau mai multe coloane adiționale.
- Ce devin **atributele relațiilor**?
 - Pentru relații **1:1** și **1:n**, atributele relațiilor vor aparține **tabelului care conține cheia externă**
 - Pentru relații **m:n** și **de tipul trei**, atributele vor fi plasate **în tabelele asociative**.

Proiectarea modelului relațional

- Cele patru tipuri de tabele (**independente**, **dependente**, **subtabele** și **asociative**) se deosebesc prin structura cheii primare.

Tabel	Reprezintă	Cheie primară
Independent	Entitate independentă	Nu conține chei externe
Subtabel	Subentitate	O cheie externă
Dependent	Entitate dependentă	O cheie externă și una sau mai multe coloane adiționale
	Atribut multiplu	
Asociativ	Relație m:n	Două sau mai multe chei externe și (opțional) coloane adiționale
	Relație de tip 3	

- **Diagramele conceptuale pentru proiectarea modelelor relaționale comentate vor fi construite din diagramele E/R prin adăugarea tabelelor asociative și prin marcarea cheilor externe.**

Exemplu

- Proiectați diagrama E/R pentru repartizarea studenților în căminele universității.

TEMĂ

- Proiectați diagrama E/R și diagrama conceptuală pentru baza de date a unui site de stagii pentru studenți (*internship-uri*). Specificați schemele relaționale corespunzătoare diagramei conceptuale obținute.
 - Lista minimală a entităților și atributelor care vor apărea în model este următoarea: angajator, persoana de contact, universitate, facultate, student, domeniu de studiu, an de studiu, stagiu, competente necesare, coordonator stagiu, interviu.
- Observații:
 - Va fi creat un assignment Teams prin intermediul căruia vor fi trimise soluțiile până pe 7 aprilie 2023, ora 23:59
 - Diagramele vor respecta modul de reprezentare folosit la curs.

Bibliografie

- Popescu, I., Velcescu, L., ***Proiectarea bazelor de date***, Editura Universității din București, 2008 – Capitolele 2, 3
- Connolly, T.M., Begg, C.E., Database Systems: ***A Practical Approach to Design, Implementation and Management***, 6th edition, Pearson Education, 2015 – Capitolele 16, 17