



# BAZE DE DATE

CURS 3

# Observații temă

1. Cardinalitățile se pun la ambele capete ale relației (nu doar la mijloc)
2. Relațiile diferite trebuie să fie semantic diferite. De exemplu: CAMIN se afla în LOCATIE, LOCATIE are CAMIN este o singură relație, nu o dublez în diagrama
3. În anumite reprezentări, există "roluri" ale relației => regăsim verbe la ambele capete => OK
4. Nu punem atribute ce au valori multiple. Exemplu: FACULTATE nu are atribute precum specializări, profesori
5. Gardian, administrator, personal curățenie - o entitate cu atribut job și eventual subentități. Când definesc subentități?
6. Greșeli la cardinalitate (valorile au fost inversate între capete).
7. Deoarece universitatea este sistemul al cărui aspect îl modelează => NU este entitate în model! Ar fi fost dacă modelam evidența caminelor din universitățile din România sau dintr-un oraș anume
8. Unele modele au presupus (restrictiv) că un student poate urma o singură facultate și un camin aparține în întregime unei singure facultăți.
9. Mai multe modele care au presupus că un student poate urma mai multe facultăți nu răspund la întrebarea: pe locurile carei facultăți este cazat studentul?

# EXEMPLU

## 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 dependentă) – 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.**

# EXEMPLU

## 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.

# EXEMPLU

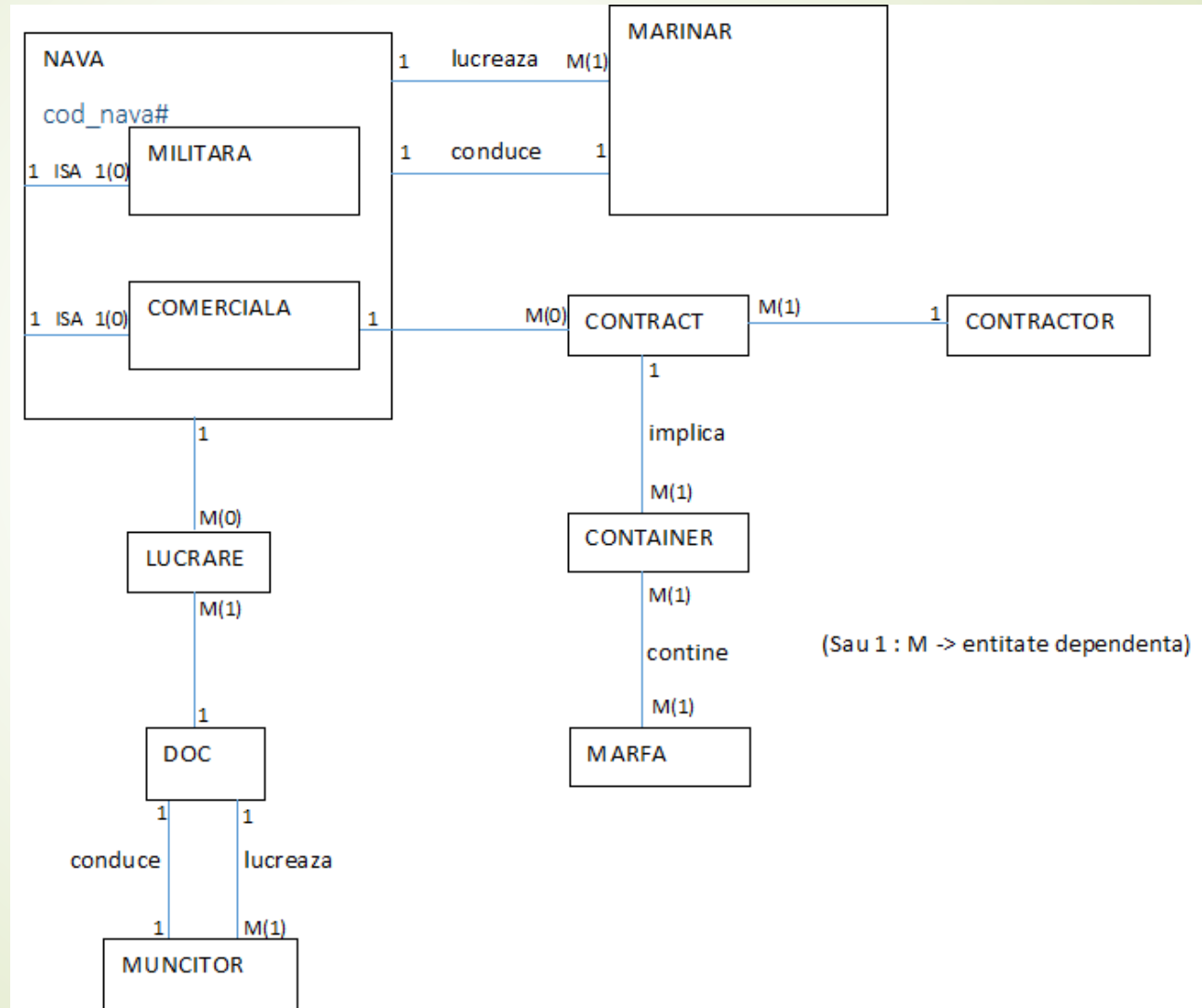
## Gestiunea activităților dintr-un port referitoare la servirea navelor

- 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?
- ❑ Pentru nave sunt semnate contracte de încărcare/descărcare cu contractori
- ❑ Marfa ajunge în port în containere

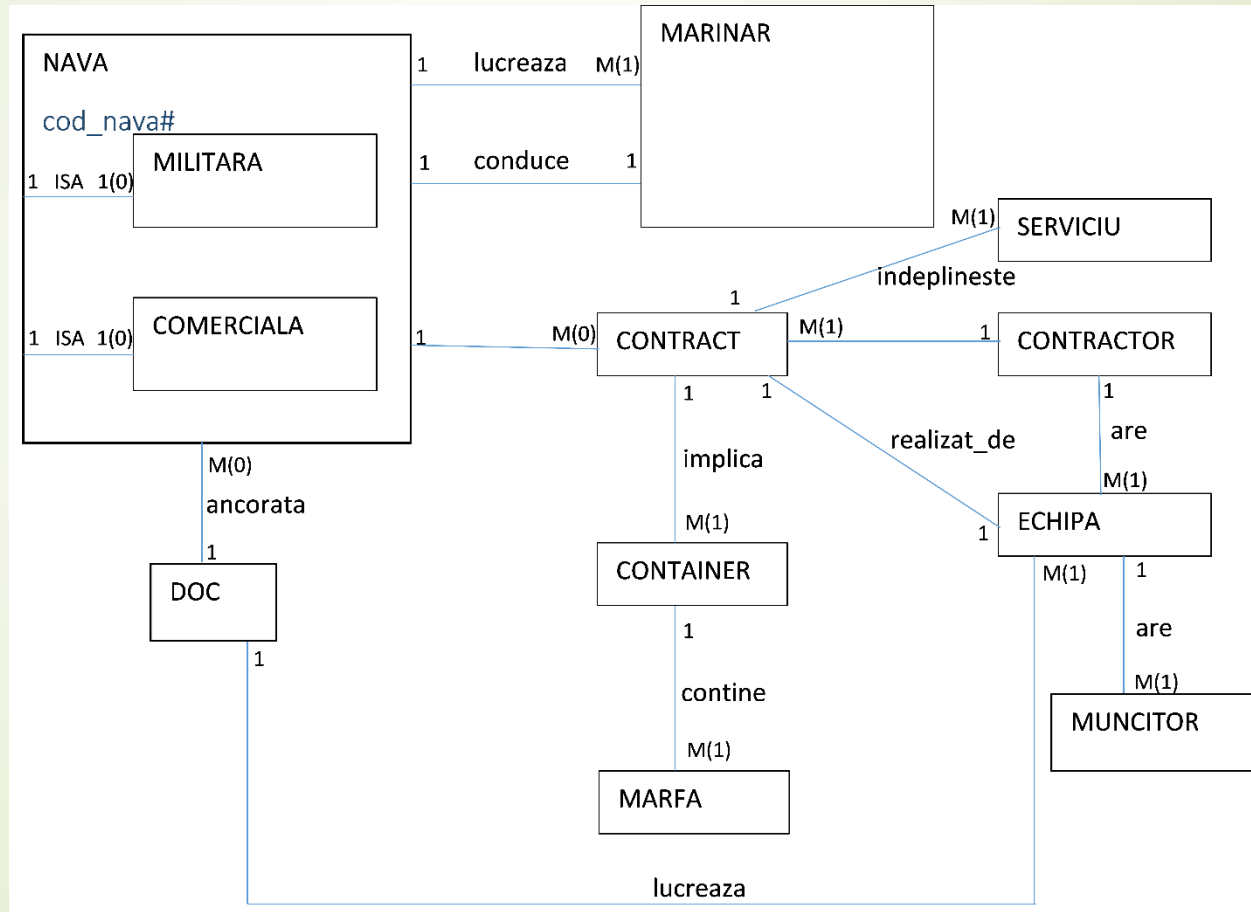
**Reprezentați diagrama E/R a unui model ce tratează cel puțin aspectele de mai sus.**

# EXEMPLU

- Care dintre variantele următoare este corectă? Corecți.



# EXEMPLU



# 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.

## ► Modalități pentru **definirea unui SGBD relațional**:

- 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;
  - **regulele 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
<b>relație</b> <b>tuplu</b> <b>atribut</b> <b>domeniu</b>	Tablou /tabel linie coloană tip de date	fișier înregistrare câmp 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 tupluri. 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		OR	T	F	Null
T	T	F	Null		T	T	T	T
F	F	F	F		F	T	F	Null
Null	Null	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 acest tabel.

# 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:

- **regulile structurale** inerente modelării datelor;
- regulile 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 #.

# 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 attribute 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  $R1$ .

# 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

## 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

- Relațiile 1:1 și 1: $n$  devin **chei externe**.
  - Relația *conduce* devine coloană în tabelul DEPARTAMENT, iar relația *lucreaza\_in* devine coloană în tabelul SALARIAT.
  - Simbolul „ $\times$ ” indică plasamentul cheii externe, iar simbolul „ $\underline{\times}$ ” exprimă faptul că această cheie externă este conținută în cheia primară. Relația 1:1 plasează cheia externă în tabelul cu mai puține linii.
- 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
<b>Independent</b>	entitate independentă	nu conține chei externe
<b>Subtabel</b>	Subentitate	o cheie externă
<b>Dependent</b>	entitate dependentă	o cheie externă și una sau mai
	atribute multiple	multe coloane adiționale
<b>Asociativ</b>	relație m:n	două sau mai multe chei
	relații de tip 3	externe și (opțional) coloane adiționale

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

# TEMA

- **Soluțiile vor fi trimise până pe 11 martie 2020 (finalul zilei) prin Dropbox File Request (link-ul va fi disponibil pe moodle!)**
- **Numele fișierelor încărcate vor fi de forma <grupa>\_Nume\_Prenume\_<descriere>.<extensie>**
- **Diagrama E/R pentru modelarea activităților dintr-o agenție de turism, pe baza specificațiilor următoare:**
  - Într-o agenție de turism lucrează ghizi, șoferi, agenți de vânzări.
  - Din oferta agenției fac parte sejururi și excursii.
  - Ghizii conduc excursii, la care sunt atașați șoferi.
  - Clienții agenției achiziționează sejururi sau excursii.
  - Clienții sunt deserviți de către agenții de vânzări ai agenției.
  - Un sejur se desfășoară într-o anumită locație.
- **Gestiunea unui club sportiv: diagrama E/R și diagrama conceptuală**