



BAZE DE DATE

CURS 3

MODELUL RELAȚIONAL

- Conceput și dezvoltat de **E.F. (Edgar Frank) 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

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 (structurile de date sunt definite de un limbaj de definire a datelor);
 - **operatorii** modelului relațional (manipularea datelor – relațiile pot fi manipulate utilizând un limbaj de manipulare a datelor; în modelul relațional se folosesc operatori specifici algebrei relationale);
 - **regulile de integritate** care guvernează folosirea cheilor în model (constrângeri interne);
- 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 Tuplu Atribut Domeniu	Tablou / Tabel Linie Coloană Tip de date	Fișier Înregistrare Câmp Tip de date

MODELUL RELAȚIONAL

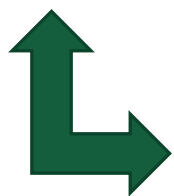
- **DOMENIU** – mulțime de valori care poate fi definită fie enumerând elementele componente, fie definind o proprietate distinctivă a domeniului valorilor.
- Fie D_1, D_2, \dots, D_n domenii finite, nu neapărat disjuncte. **Produsul cartezian** $D_1 \times D_2 \times \dots \times D_n$ al domeniilor 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**.
- Cu alte cuvinte, **domeniul (tipul de date)** este o mulțime finită în care un **atribut (coloana)** poate lua valori. Atributele din exemplul de mai jos sunt: ID, NUME, PRENUME și EMAIL

NUMBER		VARCHAR	
ID	NUME	PRENUME	EMAIL
100	KING	STEVEN	SKING
200	De Haan	Lex	LDEHAAN
300	Hunold	Alexander	AHUNOLD

MODELUL RELAȚIONAL

- 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. Asadar, o **relatie** este o multime finita de tuple.
- **Caracteristicile unei relații:**
 - nu exista tuple identice (intrari/linii);
 - ordinea liniilor si a coloanelor este arbitrara;
 - elementele care fac parte dintr-un domeniu (adica un singur tip de date) sunt de acelasi tip;
 - fiecare coloana apartine unui singur domeniu si nu se poate repeta coloana in cadrul aceleiasi relatii;

**Relatia
ANGAJAT**



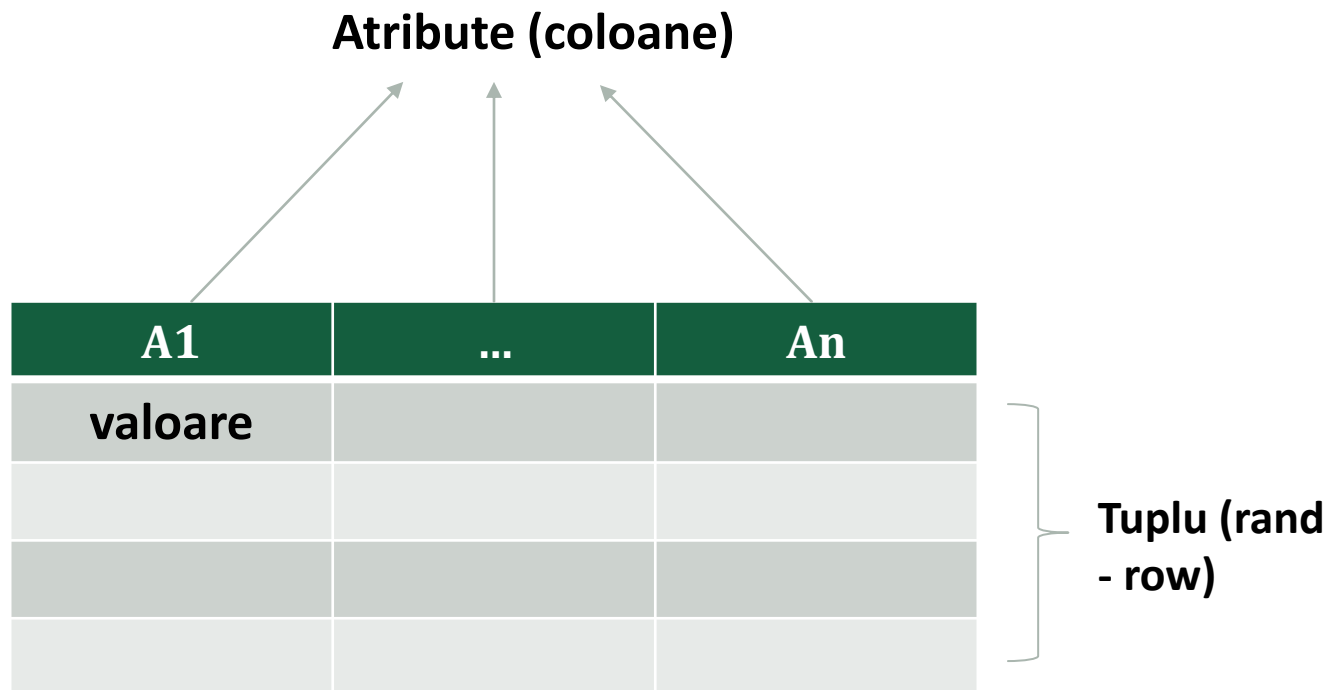
ID	NUME	PRENUME	EMAIL
100	KING	STEVEN	SKING
200	De Haan	Lex	LDEHAAN
300	Hunold	Alexander	AHUNOLD

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**: Angajat(ID, NUME, PRENUME, EMAIL)
- Putem reprezenta o relație printr-un **tabel bidimensional**
 - O coloană corespunde de fapt unui atribut;
 - Numărul atributelor unei relații definește **gradul** relației (**exemplu**: relația ANGAJAT este de grad 4 sau are aritate 4), iar numărul de tupluri din relație definește **cardinalitatea** relației.

ID	NUME	PRENUME	EMAIL
100	KING	STEVEN	SKING
200	De Haan	Lex	LDEHAAN
300	Hunold	Alexander	AHUNOLD

MODELUL RELAȚIONAL



Domeniu = multimea de valori in care poate lua valori un atribut

Relatie = Tabel

Numele relatiei = Denumirea tabelului

Tuplu = o succesiune de valori de diferite tipuri

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

Valoarea *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 este *null* când unul din argumente este *null*.
- 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
- 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 particular;

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

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**.

MODELUL RELAȚIONAL

O mulțime minimală de attribute ale căror valori identifică unic un tuplu (adica un rand/inregistrare din baza de date) 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 simbolul #**;

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 $R1$.

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 referă cheia primară din 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 „×” indică plasamentul cheii externe, iar simbolul „⊗” 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.

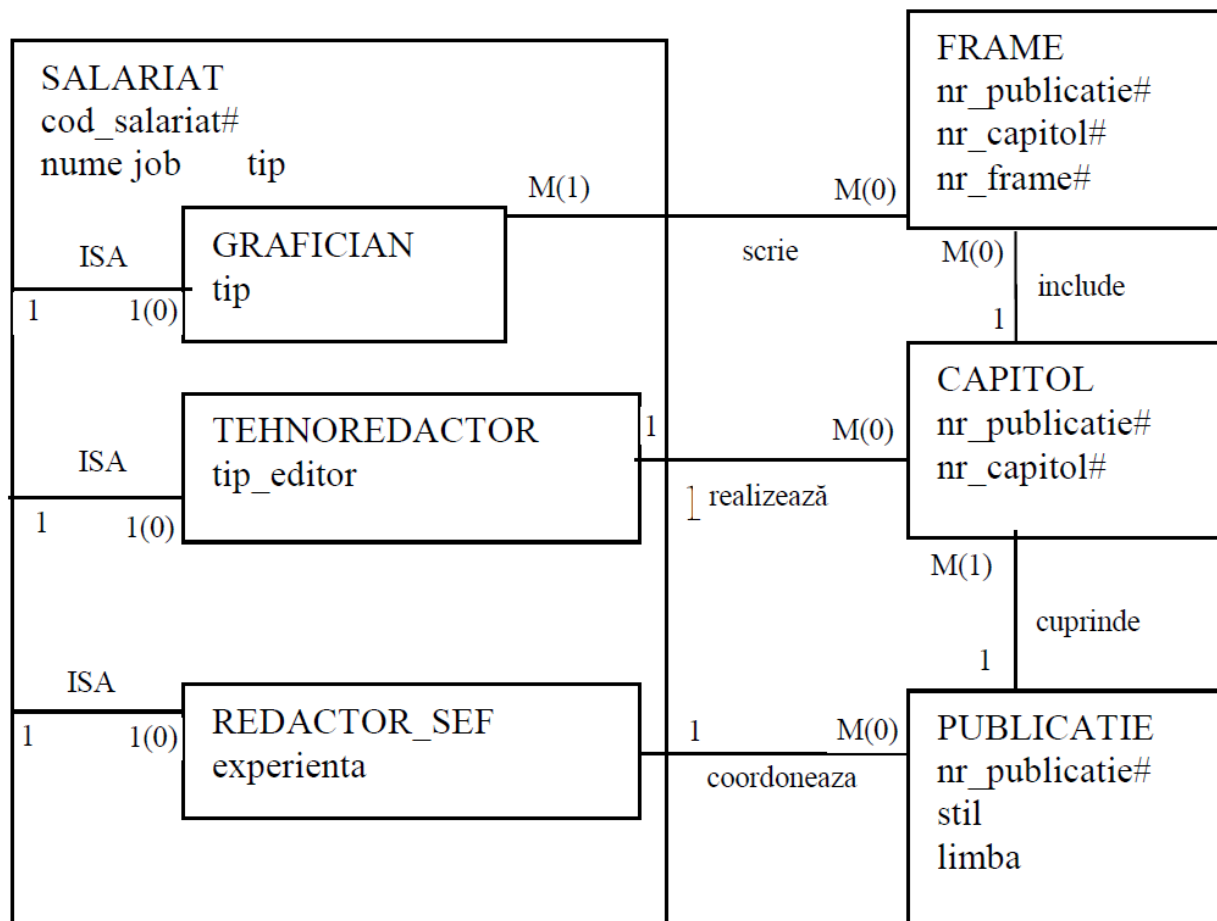
Tabel	Reprezintă	Cheie primară
Independent	entitate independentă	nu conține chei externe
Subtabel	Subentitate	o cheie externă
Dependent	entitate dependentă	o cheie externă și una sau mai
	atribute multiple	multe coloane adiționale
Asociativ	relație m:n	două sau mai multe chei
	relații de tip 3	externe 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

- Sa se studieze notiunile parcurse in cadrul cursului si sa se analizeze Diagrama Conceptuala proiectata pentru **Gestiunea activitatilor de editare dintr-o editura**, plecand de la Diagrama Entitate/Relatie.
- Studiatii regulile de transformare si modul in care acestea au fost aplicate.

TEMA – Diagrama E/R



TEMA – Diagrama Conceptuala

