
Baze de Date

Curs 2-3. Modelul relational

Ioana Ciuciu
oana@cs.ubbcluj.ro
<http://www.cs.ubbcluj.ro/~oana/>



Planificare

Saptamana	Curs	Seminar	Laborator
S1	1. Concepte fundamentale ale bazelor de date	1. Modelul Entitate-Relatie. Modelul relational	1. Modelarea unei BD in modelul ER si implementarea ei in SQL Server
S2	2. Modelare conceptuala		
S3	3. Modelul relational de organizare a bazelor de date	2. Limbajul SQL – definirea si actualizarea datelor	2. Interogari SQL
S4	4. Gestiunea bazelor de date relationale cu limbajul SQL		
S5-6	5-6. Dependente functionale, forme normale	3. Limbajul SQL – regasirea datelor	3. Interogari SQL avansate
S7	7. Interogarea bazelor de date relationale cu operatori din algebra relationala	4. Proceduri stocate	
S8	8. Structura fizica a bazelor de date relationale		
S9-10-11	9-10-11. Indeksi. Arbori B. Fisiere cu acces direct	5. View-uri. Functii definite de utilizator. Executie dinamica	4. Proceduri stocate. View. Trigger
		6. Formele normale ale unei relatii. Indeksi	
S12	12. Evaluarea interogarilor in bazele de date relationale		
S13	13. Extensii ale modelului relational si baze de date NoSQL	7. Probleme	Examen practic
S14	14. Aplicatii		

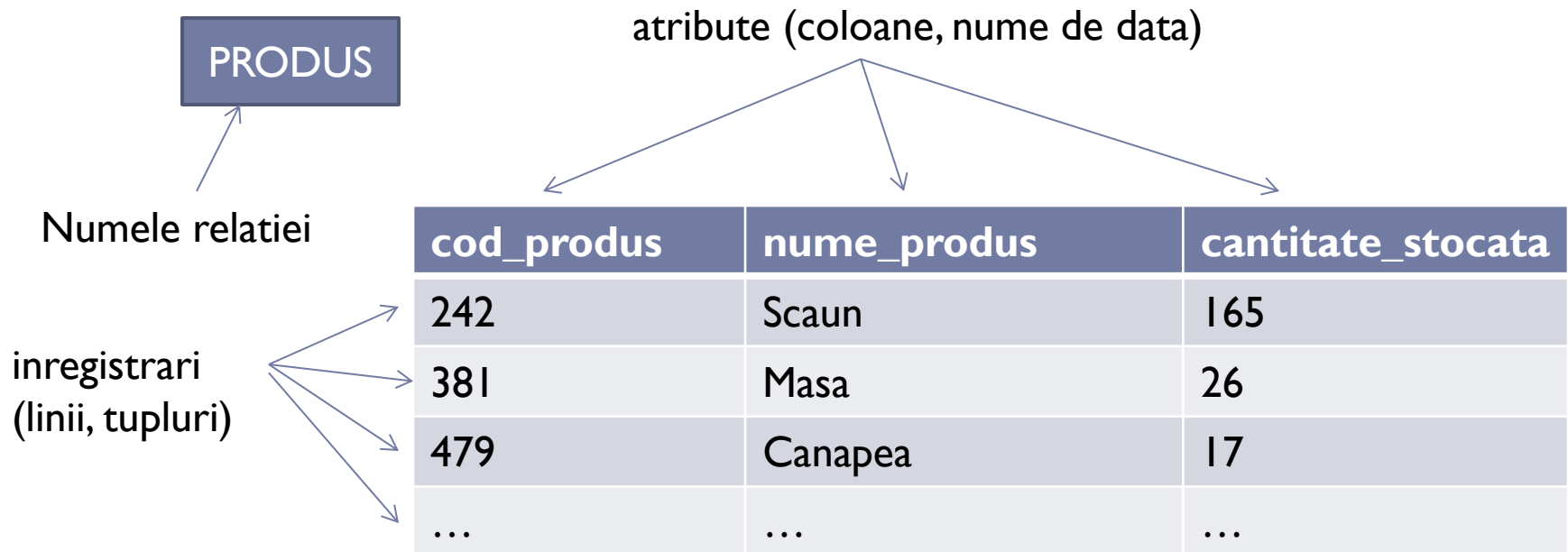
Plan

- ▶ Relatia
- ▶ Restrictii de integritate
- ▶ Reguli de transformare din model conceptual in model relational
- ▶ Baze de date relationale
- ▶ Gestiunea bazelor de date relationale – curs 3, 4



Relatia (tabela)

- In modelul relational, datele sunt organizate ca o multime de relatii (tabele)



Relatia (tabela)

▶ Precizari:

- ▶ Cardinalitatea relatiei = numarul de linii
- ▶ Gradul relatiei = numarul de coloane
- ▶ Numele atributelor este unic
- ▶ Valorile unui atribut sunt atomice (nu se pot descompune) si scalare (nu se pot memora in tablouri)
- ▶ Liniile si coloanele nu sunt ordonate
- ▶ Liniile din tabel sunt distincte



Relatia (tabela)

- ▶ Domeniul unui atribut

- ▶ Este multimea de valori posibile pentru un atribut

- ▶ Se defineste

- ▶ Fie ca multime

- Culoarea ochilor = {albastri, verzi, negri, caprui}

- ▶ Fie ca interval

- Greutate nou-nascut = [1,5 ; 5,2]



Restricțiile de integritate

▶ Cheia unei relatii

- ▶ Este un atribut (sau o multime de attribute) care pentru oricare doua inregistrari (linii) diferite are valori distincte
- ▶ E vorba deci de un **identificator** al inregistratilor
- ▶ Exemplu:
 - ▶ Pentru relatia MASINA, numarul de inmatriculare al masinii este cheie
 - ▶ Pentru relatia STUDENT, CNP-ul studentului



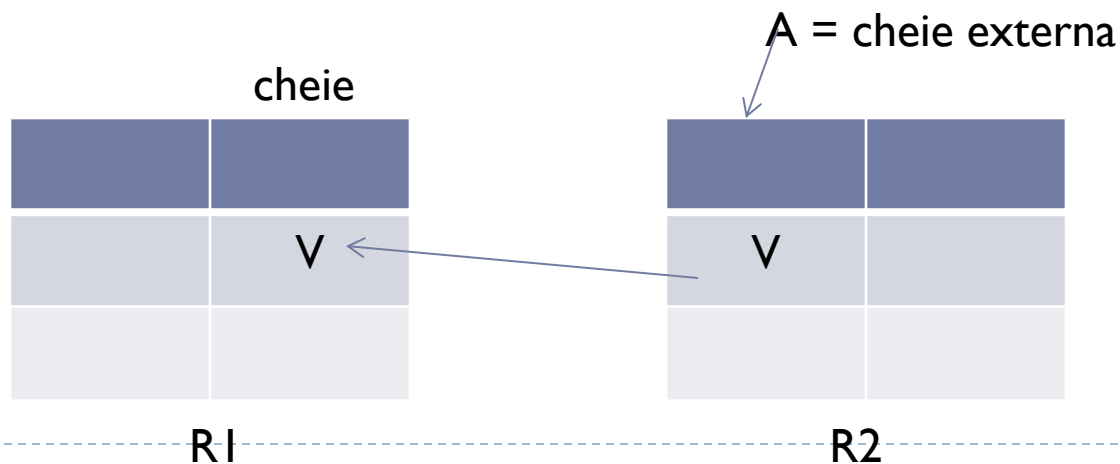
Restricțiile de integritate

- ▶ Pentru o relație pot fi stabilite *mai multe chei*
- ▶ Una dintre chei (un atribut sau un grup de attribute) se alege **cheie principală (primară)**, iar celelalte se vor considera **chei secundare (chei candidat)**
- ▶ Exemplu:
 - ▶ ORAR[Zi, Ora, Sala, Profesor, Formatia, Disciplina], cu orarul pe o săptămână
 - ▶ Se pot alege ca și chei următoarele mulțimi de attribute
 - ▶ {Zi, Ora, Sala}; {Zi, Ora, Profesor}; {Zi, Ora, Formatia}
- ▶ SGBD nu permit existența a două elemente distincte într-o relație cu aceeași valoare pentru **oricare cheie** (principală sau secundară), deci **precizarea unei chei constituie o restricție pentru baza de date**



Restricțiile de integritate

- ▶ **Cheie externa:** valorile unor attribute dintr-o relatie care apar in alta relatie
 - ▶ Plecand de la o relatie R2, se pot cauta inregistrările dintr-o relatie R1 dupa valorile unei chei externe (atribut simplu sau compus)
 - ▶ In relatia R2 se stabileste un atribut A, numit **cheie externa**
 - ▶ Valorile atributului A se cauta printre valorile cheii din relatia R1



Restricțiile de integritate

- Schema unei relatii: reprezentarea simbolica a relatiei prin **atributele sale**

PRODUS(cod_produs, nume, cantitate_in_stoc)

Nume
relatie

Atribut cheie
(subliniat)

Atribute non-
cheie

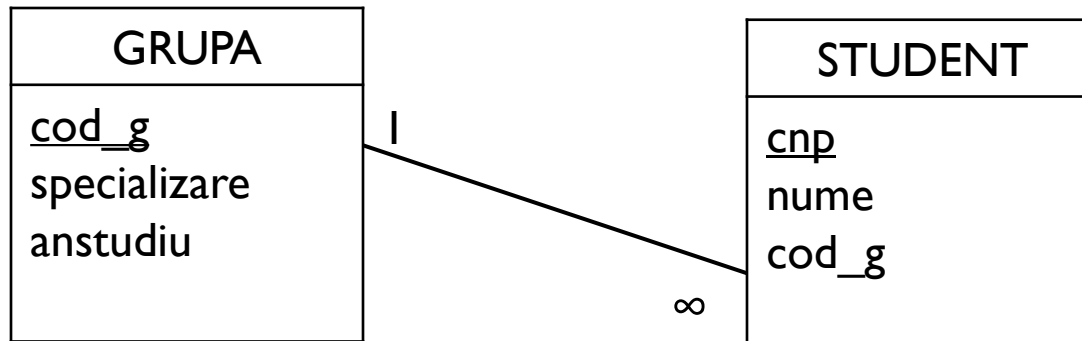


Restricțiile de integritate

► Exemplu:

GRUPA(cod_g, specializare, anstudiu)

STUDENT(cnp, nume, cod_g)



Putem stabili legatura între relația **GRUPA** (considerată ca părinte pentru relație) și relația **STUDENT** (ca membru pentru legatură) prin egalitatea **GRUPA.cod_g = STUDENT.cod_g**

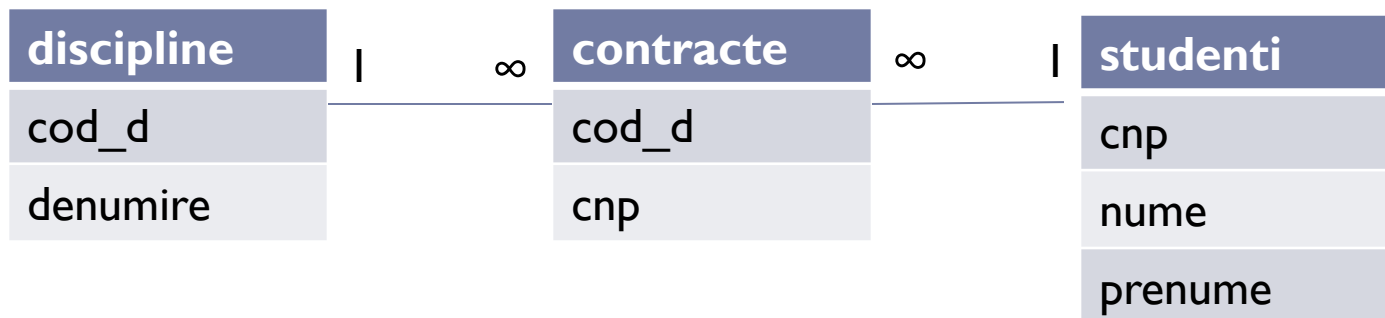
Unei anumite grupe (memorată în relația **GRUPA**), identificată printr-un cod, îi corespund toți studenții din grupa cu codul respectiv.

Atributul **cod_g** din relația **STUDENT** este cheie externă

Prin cheie externă se pot memora **legături 1:n** între entități: la o grupă corespund oricâți studenți, iar unui student îi este asociată cel mult o grupă.

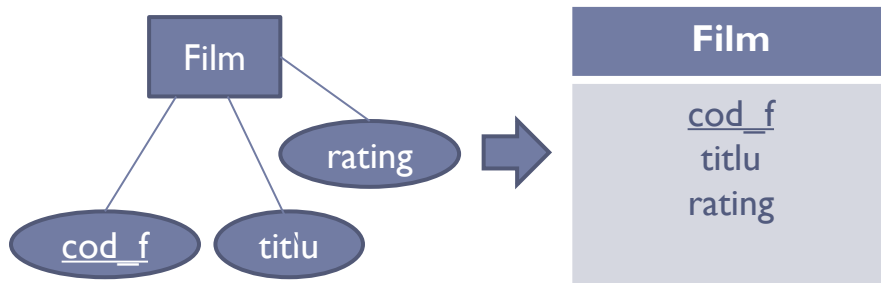
Restricțiile de integritate

- ▶ Cheia externa se poate folosi și pentru a memora legături m:n între entități
- ▶ Fie două entități, discipline și studenți. La o disciplină sunt înscriși mai mulți studenți, iar un student are asociate mai multe discipline;
 - ▶ Varianta de memorare cuprinde o relație intermediară



Transformare din model E-R in model relational

- ▶ Reguli de transformare pentru tipurile de entitate
 - ▶ Un tip de entitate devine un tabel
 - ▶ Fiecare atribut al tipului de entitate devine atribut in tabel
 - ▶ Atributul cheie devine cheia primara a tabelului
 - ▶ Exemplu:



cod_f	titlu	rating
123	Warcraft	5
455	Nerve	4
355	Suicide squad	3

Transformare din model E-R in model relational

► Reguli de transformare pentru tipurile de relatie

► Relatie **1:1 (one-to-one)**

► Trei optiuni posibile

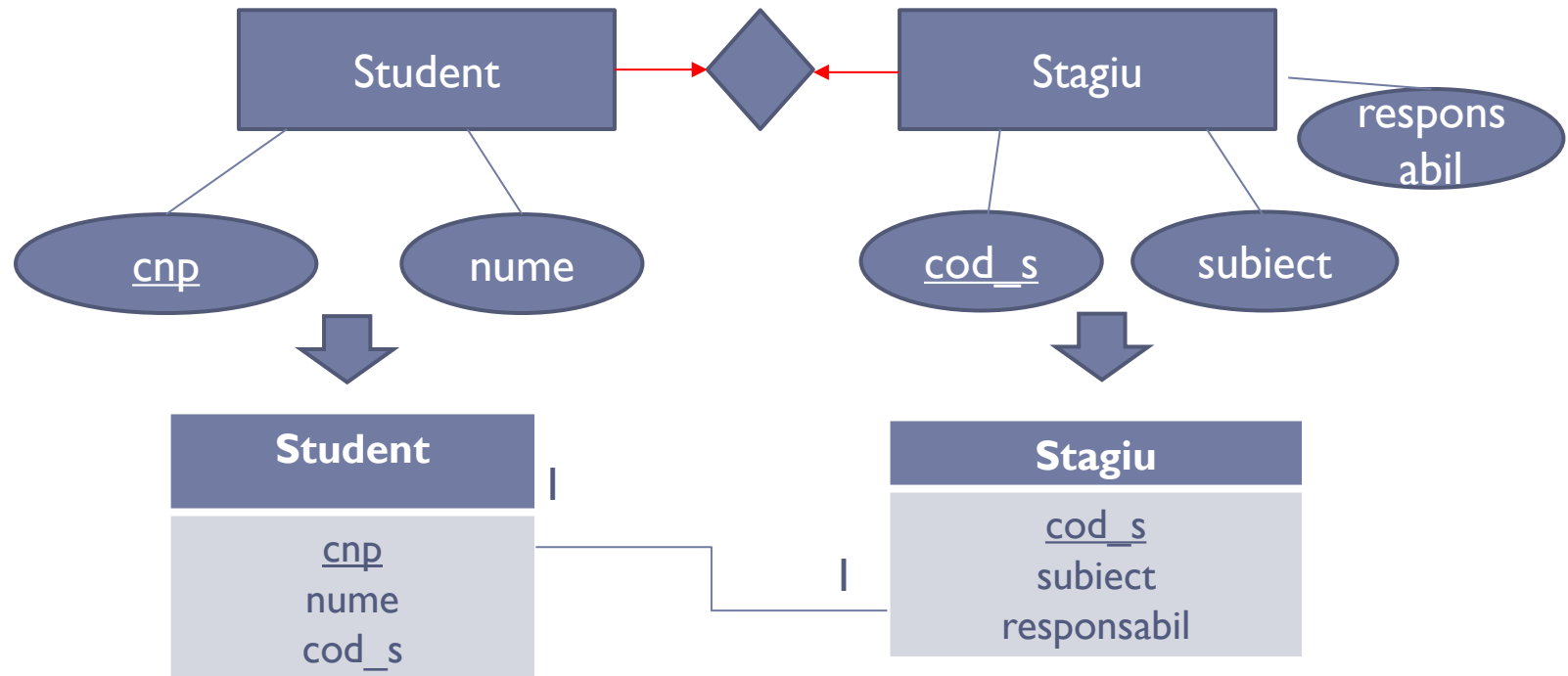
- ☐ a. Se include atributul cheie al primului tip de entitate ca si cheie externa in tabelul corespunzator celui de al doilea tip de entitate
- ☐ b. Se include atributul cheie al celui de al doilea tip de entitate ca si cheie externa in tabelul corespunzator primului tip de entitate
- ☐ c. Ambele operatii de mai sus (se creeaza, eventual, un tabel de legatura)



Transformare din model E-R in model relational

► Reguli de transformare pentru tipurile de relatie

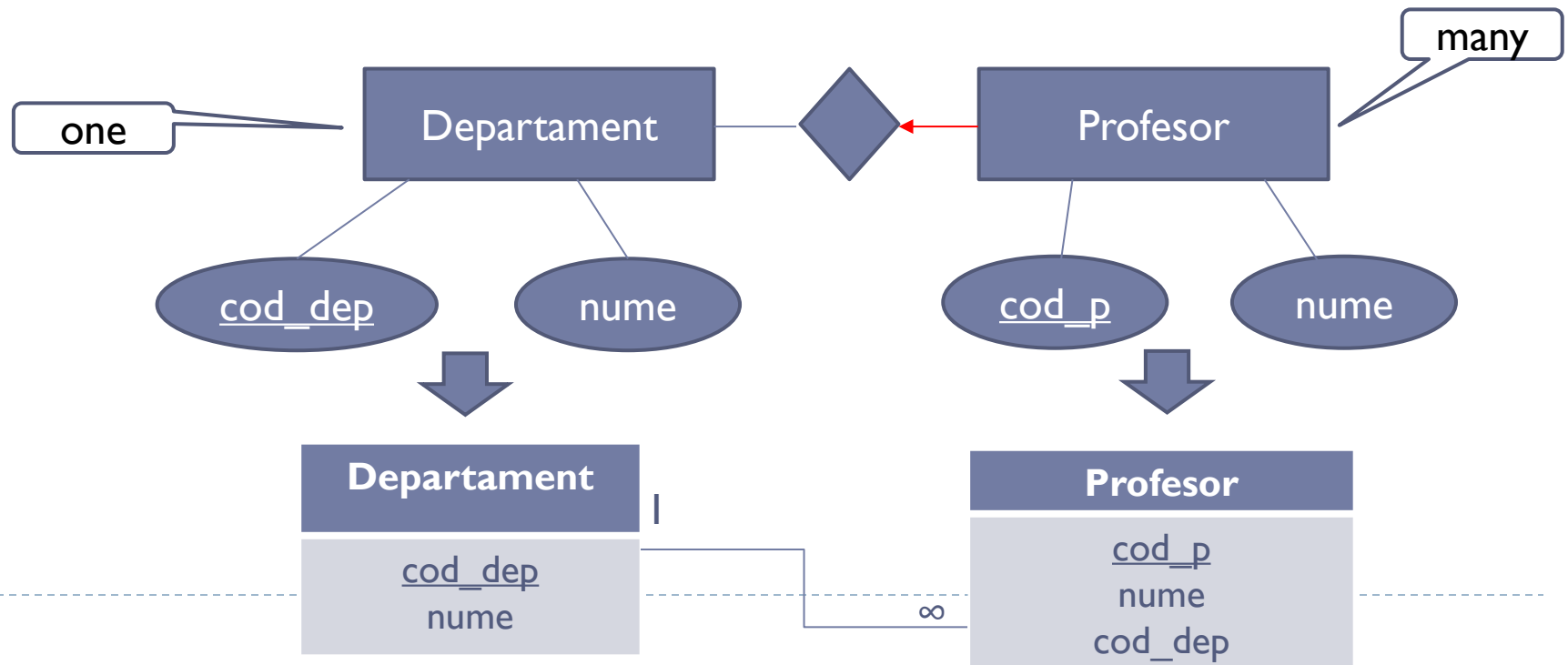
► Relatie 1:1 (one-to-one)



obs: acest exemplu presupune ca stagiul exista deja la crearea unui student

Transformare din model E-R in model relational

- ▶ Reguli de transformare pentru tipurile de relatie
 - ▶ Relatie 1:n (**one-to-many**)
 - ▶ Se include atributul cheie al tipului entitate aflat pe partea cu *one* a tipului de relatie ca si cheie externa in tabelul corespunzator tipului de entitate care se afla pe partea de *many* al tipului de relatie



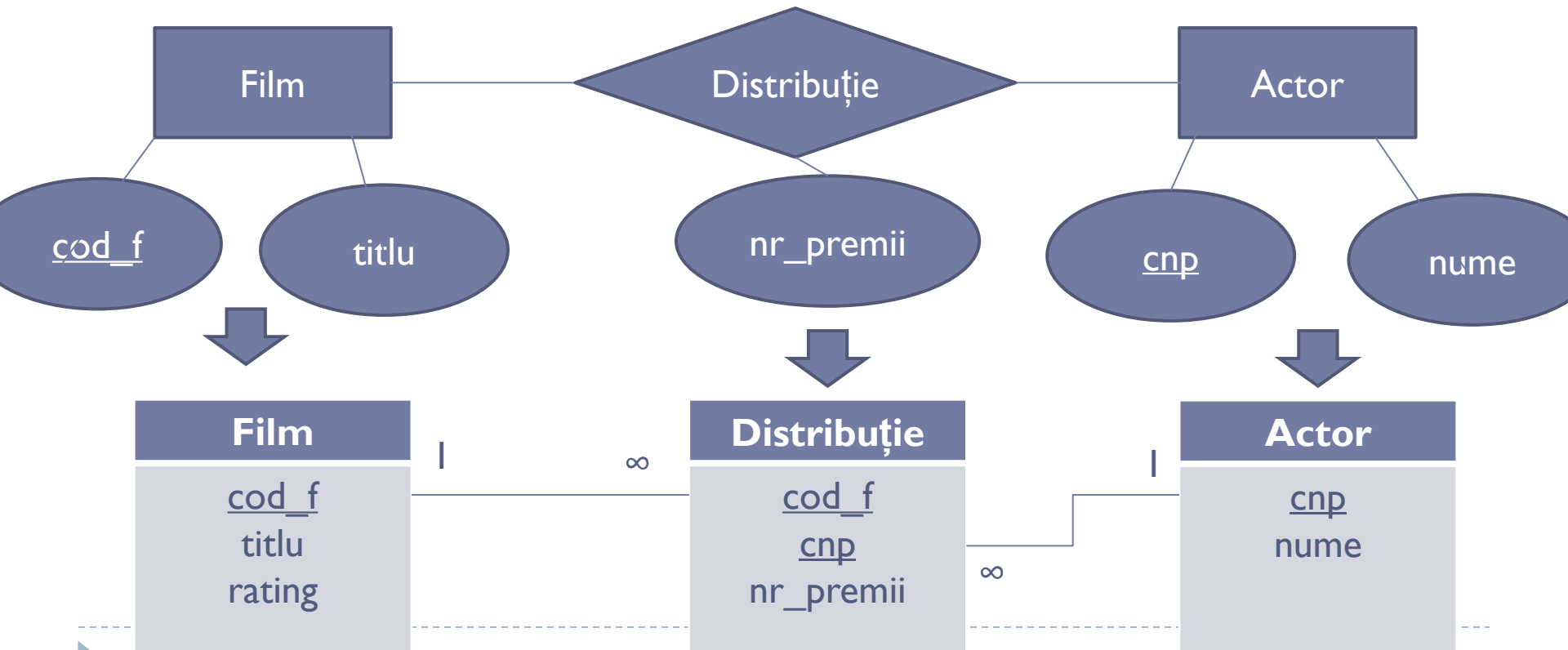
Transformare din model E-R in model relational

- ▶ Reguli de transformare pentru tipurile de relatie
 - ▶ Relatie **m:n (many-to-many)**
 - ▶ Tipul de relatie **many-to-many** se transforma intr-un tabel, numit si tabel de legatura, in care se includ attributele chei pentru fiecare tip de entitate care participa la tipul de relatie, precum si attributele tipului de relatie
 - ▶ Attributele chei pentru fiecare tip de entitate care participa la relatie sunt chei externe in tabelul de legatura si impreuna pot fi alese drept cheie primara a tabelului



Transformare din model E-R in model relational

- ▶ Reguli de transformare pentru tipurile de relatie
 - ▶ Relatie **m:n (many-to-many)**
 - ▶ Exemplu:



Transformare din model E-R in model relational

Model E-R	Model relational
Relatie 1:1	Se adauga atributul cheie al oricarui tip de entitate ca si cheie externa in tabelul corespunzator celui alt tip de entitate, sau se face acest lucru pentru ambele tipuri de entitate (eventual prin intermediul unei tabele de legatura)
Relatie 1:n	Se adauga atributul cheie al tipului de entitate aflat pe partea de <i>one</i> ca si cheie externa in tabelul corespunzator tipului de entitate de pe partea de <i>many</i> a relatiei
Relatie m:n	Se creeaza un tabel aditional de legatura care are cheia primara compusa din attributele cheie corespunzatoare celor doua tipuri de entitate; in acest tabel se includ, de asemenea, attributele tipului de relatie



Bazele de date relationale

- ▶ Bazele de date relationale se bazeaza pe modelul relational de reprezentare a datelor
- ▶ Datele dintr-o baza de date relationala sunt stocate in tabele, asa cum s-a aratat in slide-urile precedente
- ▶ Determinarea relatiilor din baza de date
 - ▶ Pentru o problema data, solutia nu este unica



Bazele de date relationale

- ▶ Exemplu de baza de date relationala pentru evidenta unor informatii din cadrul unei facultati
 - ▶ Informatii care se gestioneaza
 - ▶ **Studenti**: cateva date generale (nume, prenume, cnp, sectia si numarul matricol)
 - ▶ **Grupele** de studiu pentru mai multi ani universitari si apartenenta studentilor la aceste formatii ('traectoria' studentilor)
 - ▶ **Rezultatele** profesionale ale studentilor: disciplina, nota si numarul de credite, anul universitar si semestrul



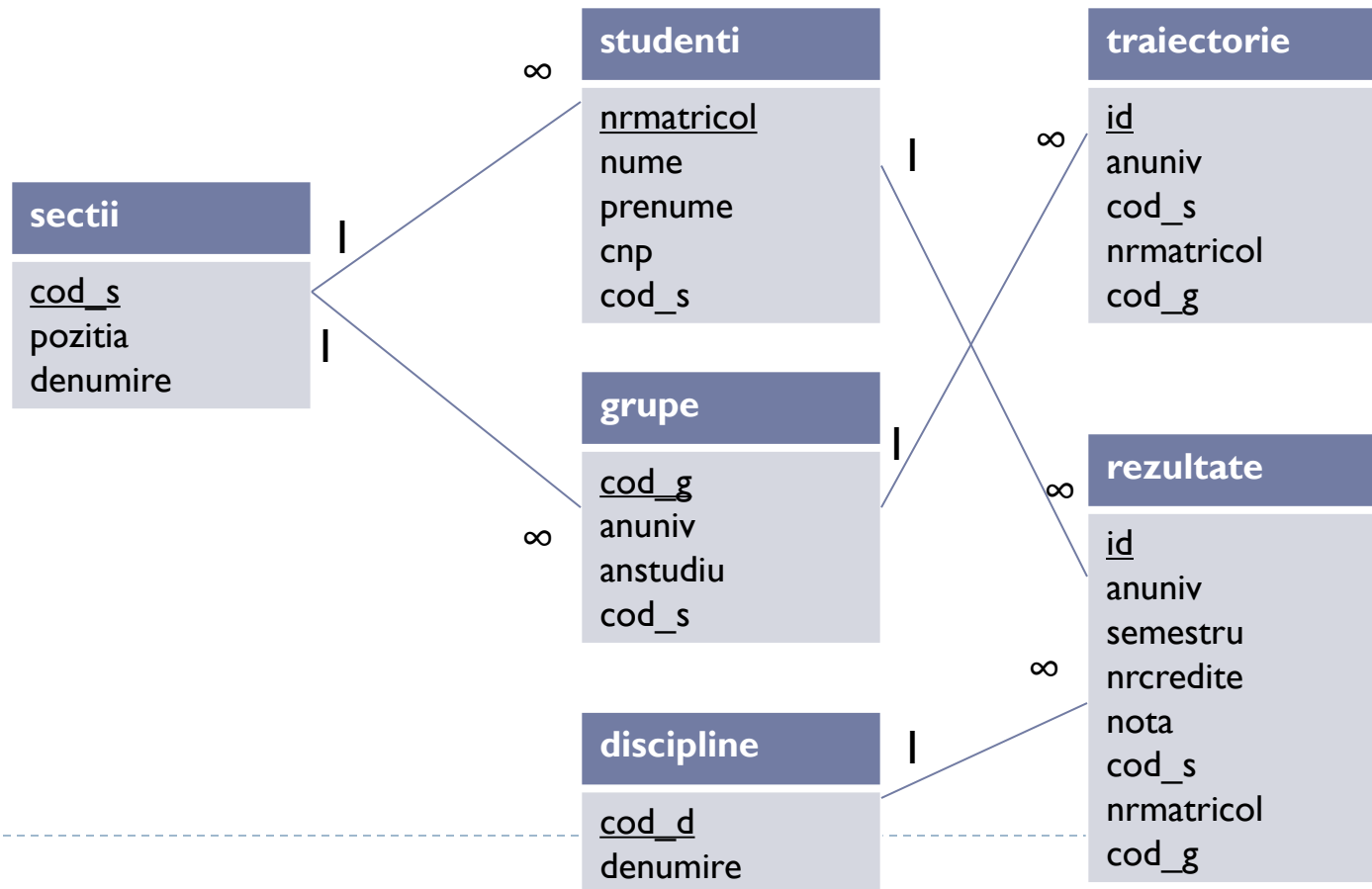
Bazele de date relationale

- ▶ Exemplu de baza de date relationala pentru evidenta unor informatii din cadrul unei facultati
 - ▶ Relatii posibile pentru acest exemplu (solutia nu este unica):
 - ▶ Studenti
 - ▶ Grupele de studiu
 - ▶ Sectiile
 - ▶ Apartenenta studentilor la grupe
 - ▶ Rezultatele la examene
 - ▶ Disciplinele



Bazele de date relationale

- ▶ Exemplu de baza de date relationala pentru evidenta unor informatii din cadrul unei facultati



Gestiunea bazelor de date relationale

- ▶ Definirea datelor - Data Definition Language
- ▶ Manipularea datelor - Data Manipulation Language
- ▶ Cursurile 3 si 4



Referinte

- ▶ [Da04] DATE, C.J., An Introduction to Database Systems (8th Edition), Addison-Wesley, 2004.
- ▶ [Ga08] GARCIA-MOLINA, H., ULLMAN, J., WIDOM, J., *Database Systems: The Complete Book*, Pearson Prentice Hall, 2008
- ▶ [Mi09] MIU, L., OZSU, M.T., *Encyclopedia of Database Systems*, Springer 2009 (3818 pages).
- ▶ [Ra07] RAMAKRISHNAN, R., *Database Management Systems*. McGraw-Hill, 2007, <http://pages.cs.wisc.edu/~dbbook/openAccess/thirdEdition/slides/slides3ed.html>
- ▶ [Si10] SILBERSCHATZ A., KORTZ H., SUDARSHAN S., *Database System Concepts*, McGraw-Hill, 2010, <http://codex.cs.yale.edu/avi/db-book/>
- ▶ [Ul11] ULLMAN, J., WIDOM, J., *A First Course in Database Systems* (3rd Edition), Addison-Wesley + Prentice-Hall, 2011
- ▶ Leon Tambulea, curs de Baze de Date, UBB Cluj-Napoca
- ▶ Surdu Sabina, seminar de Baze de Date, Matematica-Informatica, UBB Cluj-Napoca, 2016
- ▶ Sara BouchenaK, Curs Bases de données et systemes d'information, licenta MIAGE, Université Joseph Fourier Grenoble, 2012

