

# ***Modelul entitate-asociere extins***

*prof. dr. ing. Mircea Petrescu*

## ***Introducere***

Modelarea conceptuală este procesul în care se elaborează o descriere semantică a unui sistem (de exemplu, o organizație, o întreprindere, etc.), care trebuie reflectat de (modelat în ...) structura generală (design) și de implementarea unei aplicații de baze de date. Modelarea conceptuală este independentă față de baza de date; ea implică analiza cerințelor aplicației și elaborarea unei structuri semantice generale (design) de nivel înalt a conținutului bazei de date și a restricțiilor specifice aplicației.

Restricțiile care decurg din natura și cerințele aplicației definesc (determină) reguli pentru asigurarea validității (validity!) datelor (sau informațiilor) conținute de baza de date. Ca urmare, implementarea bazei de date trebuie să respecte restricțiile introduse în structura conceptuală proiectată, precum și alte eventuale constrângeri suplimentare ce pot fi formulate față de structura conceptuală.

Modelul entitate-asociere (EA), introdus de Peter Chen în 1976, este una din cele mai cunoscute abordări privind proiectarea structurii conceptuale a unei baze de date. Modelul EA descrie entitățile asociate cu o aplicație de baze de date într-o manieră independentă de baza de date propriu-zisă. În același mod independent modelul descrie asocierile între entități și restricțiile asupra acestora.

Ulterior apariției modelului EA, în scopul extinderii capacității de modelare a cunoștințelor din modelul EA, au fost propuse mai multe modele semantice pentru date. În cele mai multe dintre acestea, a fost urmărită includerea unor modalități de modelare care să permită ca în proiectarea structurii conceptuale să se țină seama la un anumit nivel superior de restricțiile semantice. În acest fond a apărut modelul entitate asociere extins (MEAE), care dezvoltă modelul EA original prin folosirea unor proceduri mai avansate de modelare conceptuală, proceduri aplicate de modelele semantice de date.

## ***Generalizarea și specializarea***

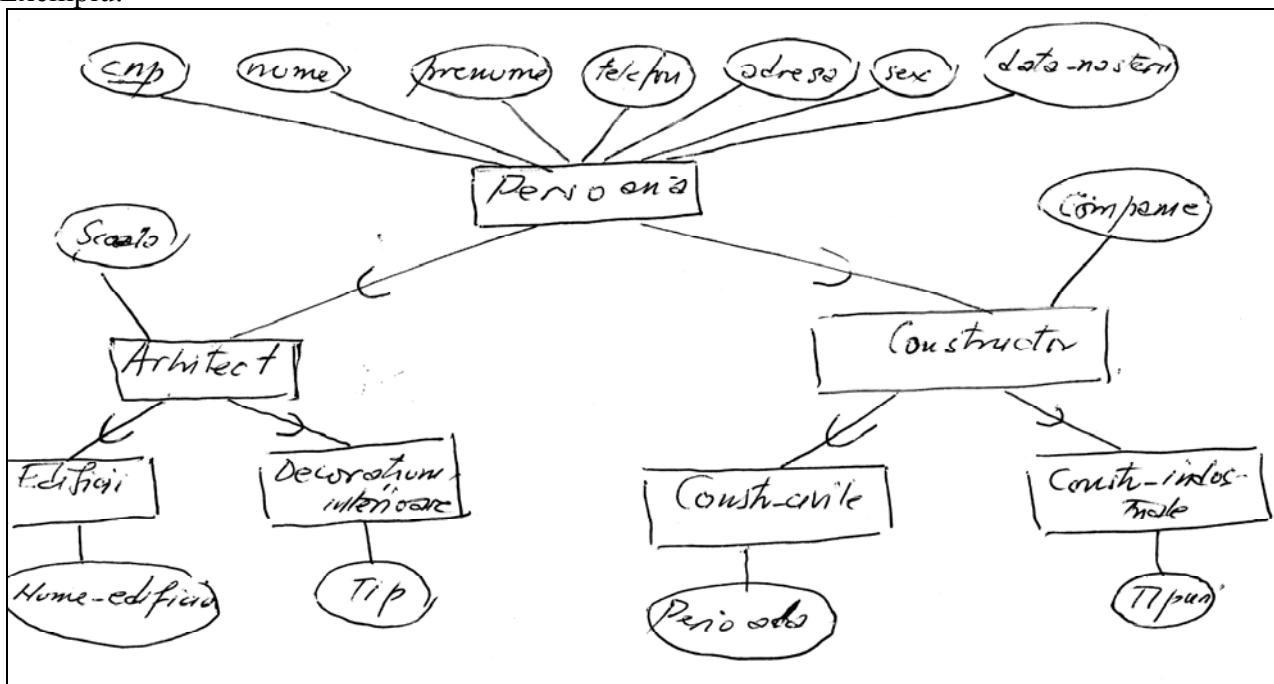
Notă – una din extensiile cele mai importante este posibilitatea de *a modela tipuri de entitate* (mulțimi entitate) *în cadrul unei ierarhii de tipuri ISA*, sau ierarhia de clase ISA. În modelul EA extins (EAE), tipurile de entitate (sau mulțimile entitate) poartă numele de clase, ele putând forma superclase și subclase, legate printr-o asociere ISA. O asociere ISA este un concept în care o subclasă este un tip specializat a unei superclase, iar o superclasă este un tip mai generalizat al unei subclase.

*Specializarea* este un proces în care o superclasă formează mai multe clase mai specializate. În procesul de *generalizare*, mai multe subclase formează o superclasă, care conține atribute și asocieri comune ale subclaselor. Remarcăm că în procesul de specializare ies în evidență atribute și asocieri ale subclaselor, care nu există la nivelul superclasei.

## Formarea ierarhiilor de clase

(Note: Persoana este clasa rădăcină, sau clasa de bază. Simbolul  $\Psi$  reprezintă direcția asocierii ISA)

Exemplu:



Cu ierarhia de clase este asociat termenul de moștenire, în sensul că fiecare subclasă moștenește toate atributele definite la nivelul superclasei corespunzătoare. Această organizare a claselor permite ca atributele comune la nivelul unei clase și la nivelurile subclaselor acesteia, să fie exprimate la nivelul cel mai comun de superclasă, în loc de a fi repetate la fiecare nivel al ierarhiei de clase.

Remarcăm, de asemenea, că fiecare subclasă introduce atribute noi, care nu se regăsesc la nivel de superclasă. Fiecare subclasă are atributele sale proprii și moștenește atributele tuturor superclaselor sale. În ceea ce privește instanțele claselor, remarcăm că instanțele unei superclase includ toate instanțele imediate ale clasei, precum și toate instanțele subclaselor acesteia. Așadar, fiecare instanță în baza de date poate fi văzută la niveluri diferite în ierarhie. În acest context, observăm că deoarece atributele nu sunt moștenite ascendent în ierarhie, în exemplul dat, atunci când un arhitect este văzut ca o instanță a clasei Persoana va fi incorect să ne referim la atributul “școala” deoarece, în general, o instanță a clasei Persoana nu este în mod necesar o instanță a clasei Arhitect. Problema distincției ce trebuie făcută între instanțele imediate ale unei clase și instanțele subclaselor sale, din punct de vedere al interogărilor, va fi tratată separat.

Faptul că o instanță a unei subclase este întotdeauna o instanță a unei superclase, este cunoscut ca *restricția ISA*. Această restricție presupune că eliminarea unei instanțe de superclasă trebuie să atragă după sine eliminarea acestei instanțe din toate subclaselor din care face ea parte. Dacă se permite ca o instanță de subclasă să existe fără ca aceasta să fie, în același timp, o instanță a superclasei sale, restricția ISA este încălcată.

Totuși, o instanță de superclasă poate fi creată, fără a o specializa în una din subclaselor corespunzătoare. De exemplu, mai sus se poate crea o instanță a clasei Persoana fără a se specifica faptul că persoana este un Arhitect sau un Constructor.

## Restricții asupra specializării

În cele prezentate anterior s-a presupus că în ierarhiile de clasă o instanță a unei subclase este totodată o instanță a unei superclase. Trebuie însă precizat că în practică, cu privire la apartenența la o subclasă se pot formula mai multe restricții specifice. Ne putem întreba, de pildă, dacă o persoană poate fi în același timp un arhitect specializat în realizarea de edificii sau de decorațiuni interioare, sau dacă putem crea o instanță a clasei Constructor, fără să precizăm dacă avem în vedere un specialist în construcții civile sau în construcții industriale (sau în instalații).

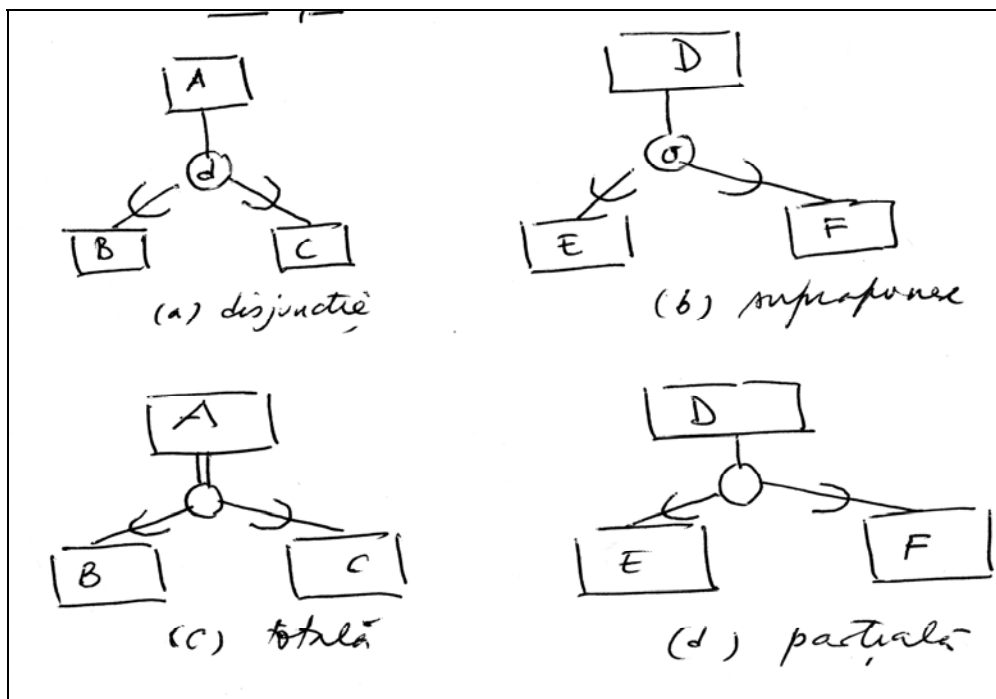
Pornind de la întrebări ca cele de mai sus, în modelul EAE ierarhiile de clasă pot fi completate cu restricții care rafinează semantica asocierilor ISA. Se introduc în modelul EAE noțiunile:

- *restricție disjunctivă* arată dacă o instanță a unei superclase este limitată la a fi o instanță numai a unei subclase;
- *restricție de specializare cu suprapunere* se referă la specializarea ne-disjunctivă, permițându-se unei instanțe de superclasă să fie o instanță a mai multor subclase ale sale;
- *restricția de completitudine* indică faptul că o instanță a unei superclase trebuie în mod necesar să fie o instanță a cel puțin unei subclase.

Atunci când o instanță a unei superclase trebuie să fie o instanță a unei subclase, *specializarea* se numește *totală*. În caz contrar, avem o *specializare parțială*. Specializarea totală se numește și *restricție de acoperire*.

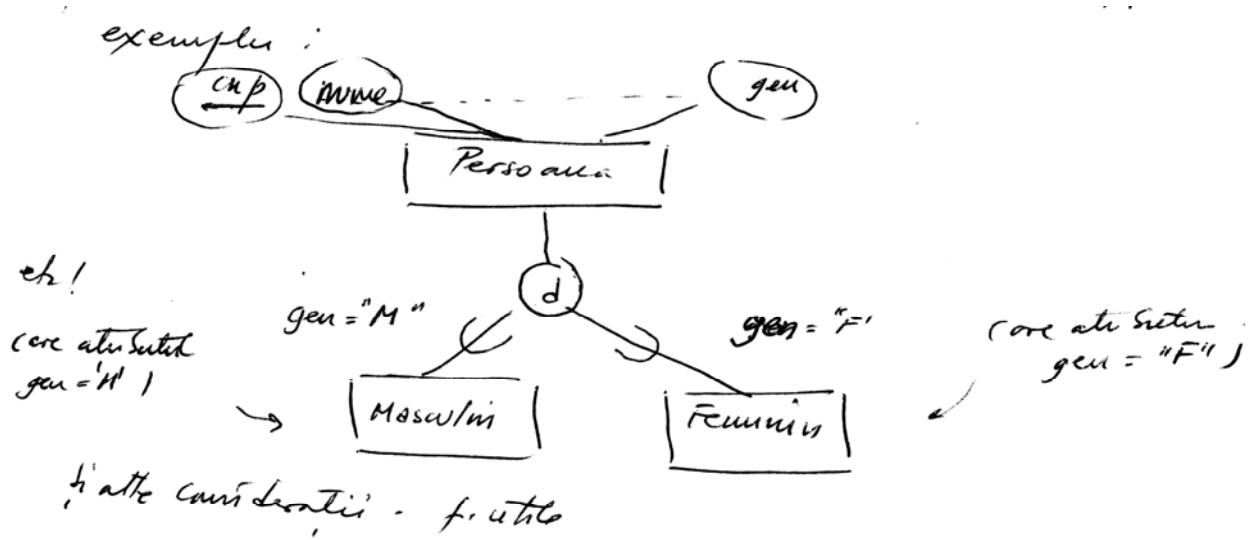
Pornind de la cele de mai sus, este important de precizat că restricțiile de disjuncție și de completitudine pot fi folosite împreună în definirea semanticii apartenenței la o subclasă în raport cu o superclasă.

Notății:



## Specializare definită prin atribut

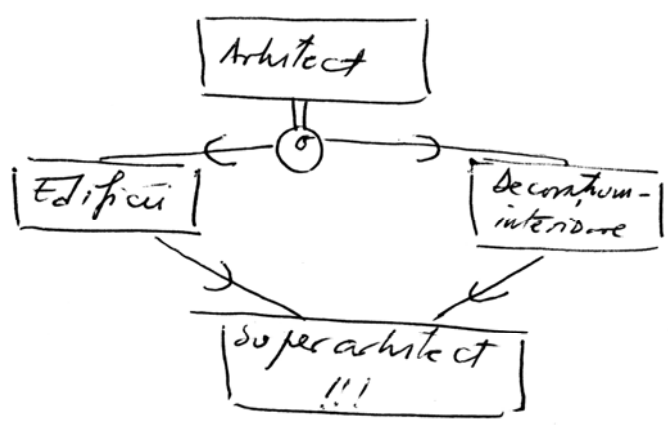
Formele de specializare anterioare sunt definite de utilizator. Cu alte cuvinte, utilizatorul indică la ce subclasă aparține o instanță de superclasă.



### Moștenire multiplă

În unele cazuri de modelare este posibil ca o subclasă să aibă mai multe superclase, formând astfel o *latică de specializare*, în care o clasă cu mai mult de o superclasă se numește *subclasă partajată*.

Exemplu:



### Categorii, categorizare

Moștenirea multiplă este folosită pentru a modela intersecția a două subclase cu o clasă rădăcină comună. Categoria se introduce pentru a modela reuniunea a două sau mai multor tipuri diferite de clase. Prin urmare, o *categorie* este o *colecție heterogenă de instanțe de entități*, spre deosebire de alte clase, care sunt colecții omogene. O instanță a unei subclase de tip categorie trebuie să fie o instanță a cel puțin uneia din superclasele sale, dar nu poate fi (aparține) în mod necesar tuturor superclasele sale.

Categoriile trebuie să satisfacă fie restricția totală de categorie, fie restricția parțială de categorie, astfel:

Restricție de categorie	Sensul restricției
Totală	Restricția definește instanțele unei categorii ca fiind egale cu reuniunea superclasele sale.
Parțială	Restricția definește instanțele unei categorii ca fiind o submulțime a reuniunii superclasele sale.

(vezi exemplul următor)

Admitem că un proiect poate fi sponsorizat fie de o persoană, fie de o companie. În loc de a construi două “asocieri de sponsorizare” între Persoană și Proiect, respectiv între Companie și Proiect, este util să definim sponsorul ca o categorie, în care clasa Sponsor este o submulțime a reuniunii claselor Persoană și Companie. Atunci, o asociere SponsorizatDe se definește între clasa Sponsor și clasa Proiect. În această situație, clasa Sponsor este definită ca o categorie parțială pentru a semnala faptul că fiecare instanță a clasei Persoană și fiecare instanță a clasei Companie nu trebuie să fie un sponsor. Prin contrast, o instanță a clasei Sponsor reprezintă o entitate care este fie o instanță a clasei Persoană, sau o instanță a clasei Companie, dar nu o instanță a ambelor clase Persoană și Companie, ca în cazul moștenirii multiple și a subclaselor partajate.

