

Integration DEFinition – IDEF

Studenti: Banica Ioana-Alina (441A)

Nedelcu Adnana Ioana (441A)

Cuprins

1.Introducere.....	3
2.IDEF0: Modelarea Functiilor.....	4
3.IDEF1: Modelarea Informatiilor.....	8
4.IDEF1x: Modelarea Datelor.....	9
5.Modelul Informational vs Modelul Datelor.....	11
6.IDEF3:Modelarea Proceselor.....	12
7.Modelarea Orientata pe Obiect.....	12
8.IDEF5: Modelarea Ontologiilor.....	13
9.IDEF vs UML.....	15
10.Concluzii.....	18
11.Bibliografie.....	19

1. Introducere

1.1 Ce este IDEF?

IDEF (Integration DEFinition) este un nume referitor la clasele de modelare a limbajelor. Acesta este folosit pentru modelarea activitatilor necesare pentru analiza, design-ul, imbunatatirea sau integrarea unui sistem.

IDEF a fost dezvoltat la inceput pentru a imbunatati comunicarea intre oamenii care doreau sa inteleaga un anumit sistem. Acum IDEF este folosit pentru documentare, intelegere, design, analiza, planificare si integrare.

IDEF a fost dezvoltat in cadrul a US Air Force in anul 1970. Modelele IDEF au fost dezvoltate pana la IDEF14:

- IDEF0: Modelare de functii
- IDEF1: Modelare de informatii
- IDEF1X: Modelare de date
- IDEF2: Design model de simulare
- IDEF3: Captura descriere proces
- IDEF4: Design obiect-orientat
- IDEF5: Captura descriere ontologie
- IDEF6: Captura ratiunii de design
- IDEF7: Sistem de audit Informații
- IDEF8: Modelarea interfetei
- IDEF9: Descoperirea constrangerilor de afaceri
- IDEF10: Modelarea arhitecturii de implementare
- IDEF11: Modelarea informației artefact
- IDEF12: Modelarea organizatiei
- IDEF13: Design schema de mapping
- IDEF14: Design de retea

2. IDEF0: Modelarea functiilor (Banica Ioana-Alina)

IDEF0 modeleaza deciziile, actiunile si activitatile unei organizatii sau a unui sistem pentru a comunica perspectiva functionala a unui sistem.

Modelele IDEF0 sunt create ca una dintre primele sarcini in dezvoltarea unui sistem, deoarece acestea descriu:

- functiile care sunt efectuate
- ce este necesar pentru a realiza aceste functii

2.1 Sintaxa

- Diagrama contextului (Context Diagram): este un model al functiei la cel mai inalt nivel de intrari, iesiri si mecanisme.

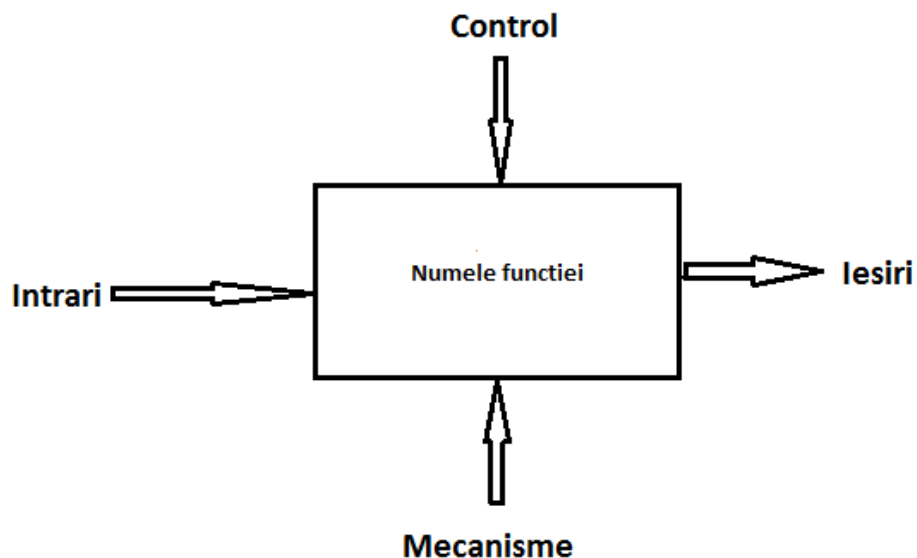


Fig 1. Diagrama contextului

Intrari: elemente care declanseaza activitatea

Control: ghideaza sau regleaza activitatea

Mecanisme: sisteme, oameni, echipamente utilizate pentru a efectua activitatea.

Iesiri: rezultatele efectuării activității.

- Diagrama de descompunere (Decomposition Diagram): alatura diagramele contextuale (Context Diagrams).

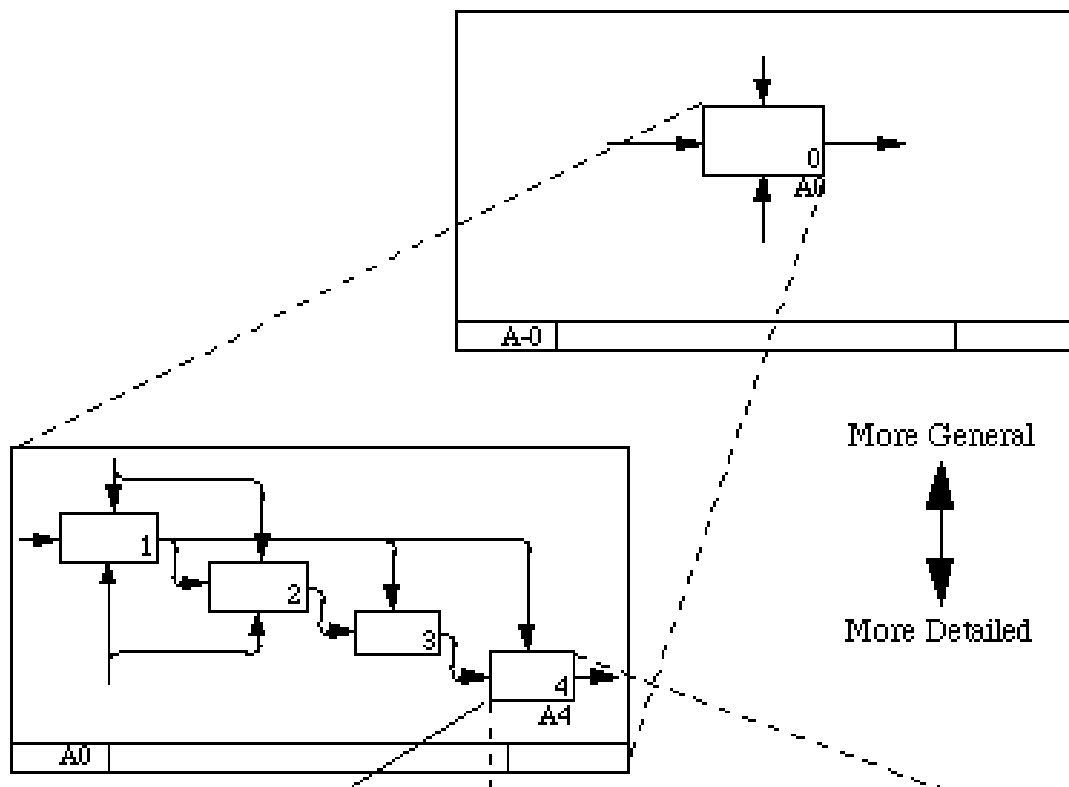


Fig2. Decomposition Diagram

Sursa: <https://sewiki.iai.uni-bonn.de/teaching/lectures/oosc/start>

2.2 Puncte forte IDEF0

- modelul s-a dovedit eficient in detalierea activitatilor sistemului pentru modelarea functiilor.
- modelele IDEF0 ofera o abstractizare fata de sincronizare, secventiere si logica deciziei.
- este usor de folosit IDEF0 pentru modelarea activitatilor secventiale oricand este nevoie (ordoneaza activitatile de la stanga spre dreapta in diagrama de descompunere).
- ofera o descriere concisa a sistemului (foloseste ICOMS – Inputs, Controls, Outputs, Mechanism).
- naturan ierarhica a lui IDEF0 permite sistemului sa fie analizat in detaliu, pana cand modelul este descris indeajuns pentru activitatea de luare a deciziilor.

2.3 Puncte slabe IDEF0

- modelele IDEF0 pot fi atat de concise incat numai expertii domeniului il pot intelege.
- uneori modelele IDEF sunt interpretate gresit ca fiind secvente de activitati.

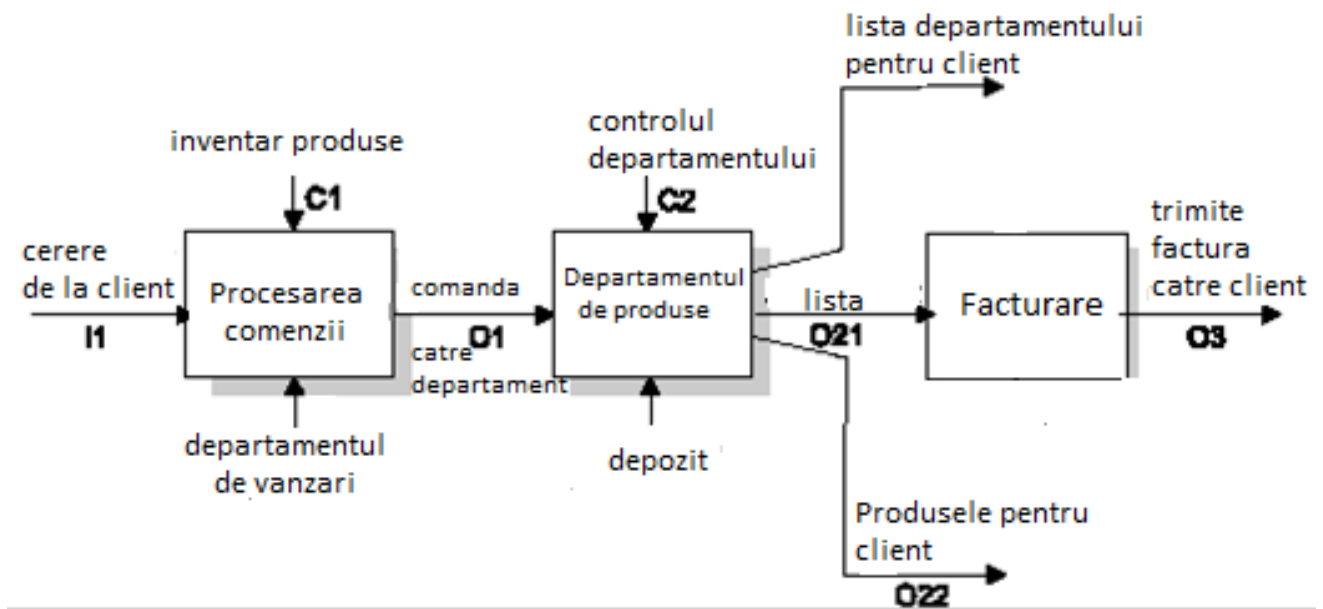


Fig3. Diagrama IDEF0 a unui proces simplu de vanzare

I - intrare

C - control

O - iesire

3. IDEF1: Modelarea informatiilor

IDEF1 este o metoda pentru a analiza si a comunica structura si semantica informatiei a unui sistem.

Modelele IDEF1 sunt folosite pentru:

- a identifica ce informatie este curent folosita pentru gestionarea unui sistem real (ce poate fi format atat din componente automate cat si de obiecte neautomatizate - oameni).
- a identifica deficientele in gestionarea curenta a informatiei
- a specifica ce informatie va fi folosita pentru punerea in aplicare a sistemului.

Pricipalele concepte IDEF1:

Entitati: reprezinta informatia pastrata intr-un sistem despre obiectele reale.

Relatii: reprezinta asocierile intre entitati.

3.1 Puncte forte IDEF1:

- IDEF1 reprezinta o metoda eficienta pentru documentarea cerintelor informationale ale unei activitati oferind o baza pentru proiectarea bazelor de date.
- impune modularitatea ceea ce presupune eliminarea caracterului incomplet, imprecizia, inconsecventele si inexactitatile gasite in procesul de modelare.

4. IDEF1x: Modelarea Datelor

IDEF1x este o metoda de proiectare a bazelor de date relationale. Aceasta este mai utila pentru proiectarea bazelor de date dupa ce cerintele de informare sunt cunoscute si decizia de a pune in aplicare o baza de date relationala a fost facuta.

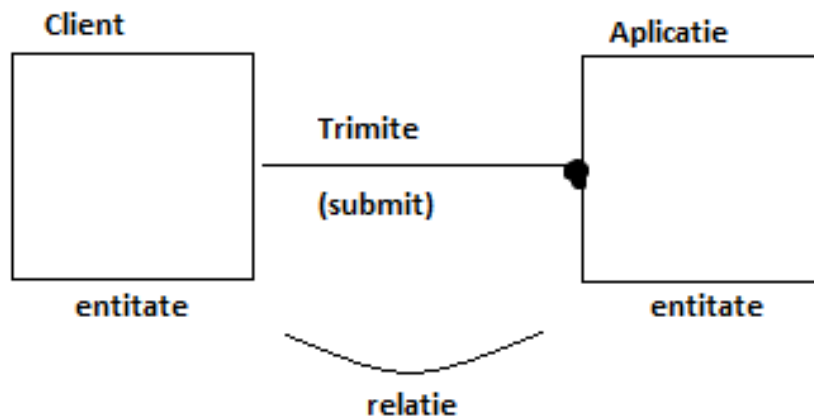


Fig4. Exemplu IDEF1x – modelarea datelor

4.1 Puncte forte IDEF1x:

- reprezinta un instrument puternic pentru modelarea de date
- descrie normele care reglementeaza gestionarea informatiilor
- este folosit pentru a valida conceptele din modelul IDEF0.
- ajuta la descoperirea cauzelor care stau la baza unor probleme.

5. Modelul informational (IDEF1) vs. Modelul datelor (IDEF1x)

Modelul informational se concentreaza pe colectarea, stocarea si gestionarea informatiei, in timp ce IDEF1x se concentreaza pe datele reale dintr-o baza de date relationala si pe reprezentarea si structurarea acestor date.

Modelul informational (IDEF1) este folosit pentru identificarea problemelor, definirea cerintelor, proiectarea sistemului de informatii, iar IDEF1x este folosit pentru proiectarea logica a bazelor de date si aplicatiilor si pentru proiectarea fizica a bazei de date.

6. IDEF3: Modelarea proceselor (Nedelcu Adnana-Ioana)

Structura de organizare: Un scenariu poate fi considerat ca o situatie recurenta, un set de situatii care descriu o clasa tipica de probleme abordate de catre o organizatie sau sistem.

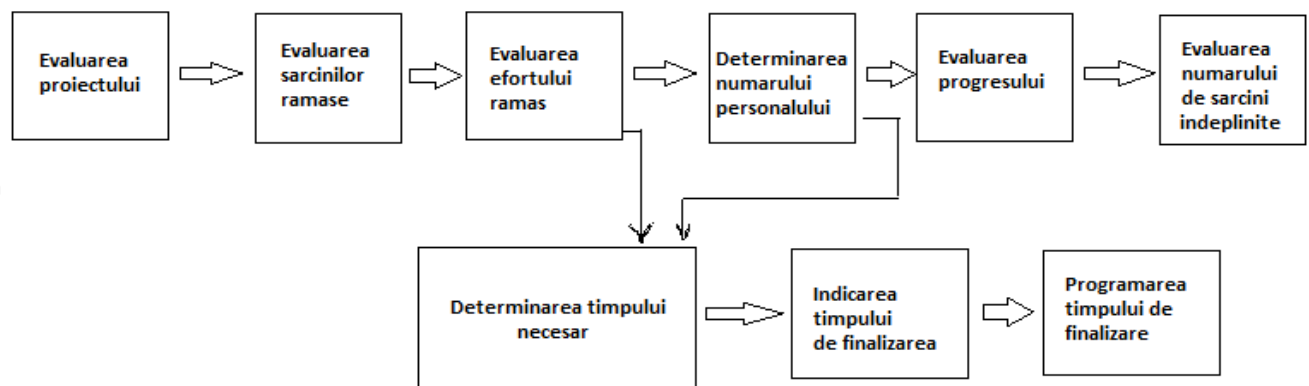


Fig 5. Exemplu de modelare a proceselor

7. IDEF4: Modelarea Orientata pe Obiect

Modelul IDEF4 este conceput pentru a ajuta in crearea programelor software orientate pe obiect. Acest model ofera un cadru (este un framework) pentru evolutia unei proiectari obiect-orientate.

IDEF4 imparte activitatea de proiectare in doua parti distincte. O sintaxa grafica evidentiaza deciziile de proiectare care trebuie efectuate si impactul lor asupra altor perspective ale proiectului.

Modelarea Orientata pe Obiect presupune existenta a doua submodele: submodelul clasei si submodelul functiei. Aceste doua structuri cuprind toata informatia reprezentata intr-un model de proiectare.

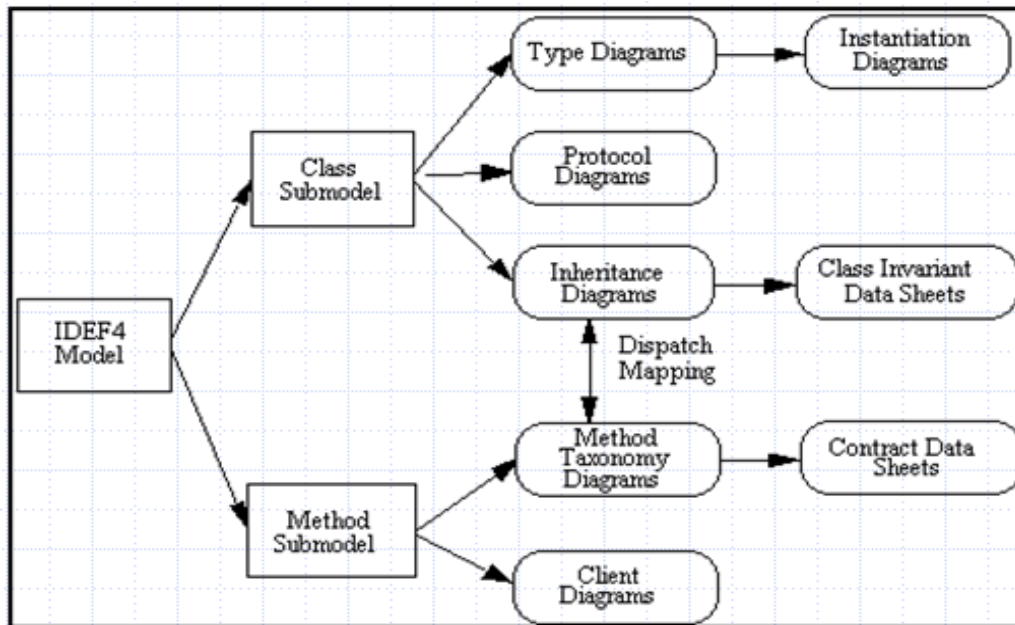


Fig6. Exemplu de modelare IDEF4

Sursa:

<http://www.idi.ntnu.no/emner/tdt4252/2013/Additional%20reading/ml-vs-idef-griffithsuniversity.pdf>

8. IDEF5: Modelarea ontologiilor

O ontologie este un vocabular complet ce contine un set cu definitii precise sau axiome care limiteaza semnificatiile termenilor pentru a permite o interpretarea coerenta a datelor.

Pasii construirii unei otologii:

1. termenii se catalogheaza

2. identificarea constrangerilor care guverneaza modul in care acesti termeni pot fi folositi pentru a face declaratii descriptive despre domeniu.
3. construirea modelului

Procesul de dezvoltare IDEF5 este format din urmatoarele activitati:

1. Colectarea datelor: date necesare pentru dezvoltarea ontologiei.
2. Analiza datelor: pentru a facilita extragerea ontologiei.
3. Dezvoltarea initiala a ontologiei: dezvoltarea unei ontologii initiale din datele adunate.
4. Rafinarea si validarea ontologiei.

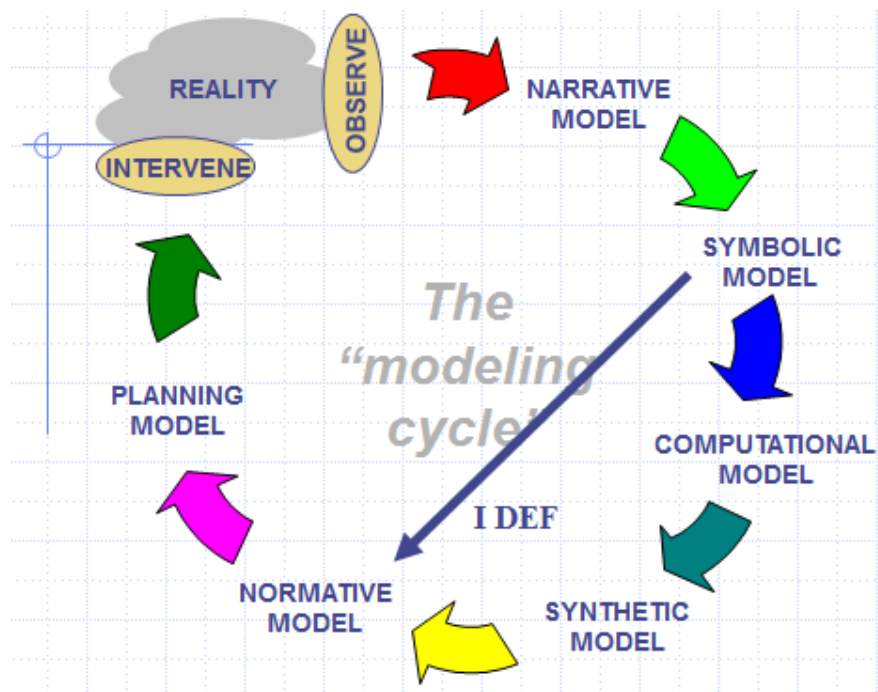


Fig7. Exemplu de modelare a ontologiilor (IDEF5)

Sursa: <http://searchsoa.techtarget.com/definition/IDEF>

9. IDEF vs. UML

Atat abordările UML cât și IDEF pot fi folosite pentru a modela aproape orice vedere utilă a unei afaceri.

UML este un limbaj de modelare mai nou față de IDEF. Acesta poate fi folosit eficient când este complementat cu :

- modelarea de design care permite propagarea cunoștințelor de proiectare și utilizare
- extensii specifice care permit să capteze în mod eficient procesele de afaceri.

Este neobisnuit ca un limbaj de modelare care include noua tipuri diferite de diagrame să mai are nevoie de extensii și nu are o bază de cunoștințe.

Pe partea pozitivă UML este ușor de învățat și poate introduce utilizatorii interesați în concepte precum meta-modelul. Utilizatorii nu trebuie să folosească toate diagramele și sunt liberi să pună în aplicare modelul lor în orice mod doresc.

În continuare voi da câteva exemple de diagrame UML pentru modelarea unei aplicații care presupune un sistem de gestionare a profesorilor și a activităților acestora în universități. Pe de o parte aplicația gestionează apartenența unui profesor la anumite facultăți, pe de altă parte oferă un suport de predare și evaluare

Diagrama de clasa

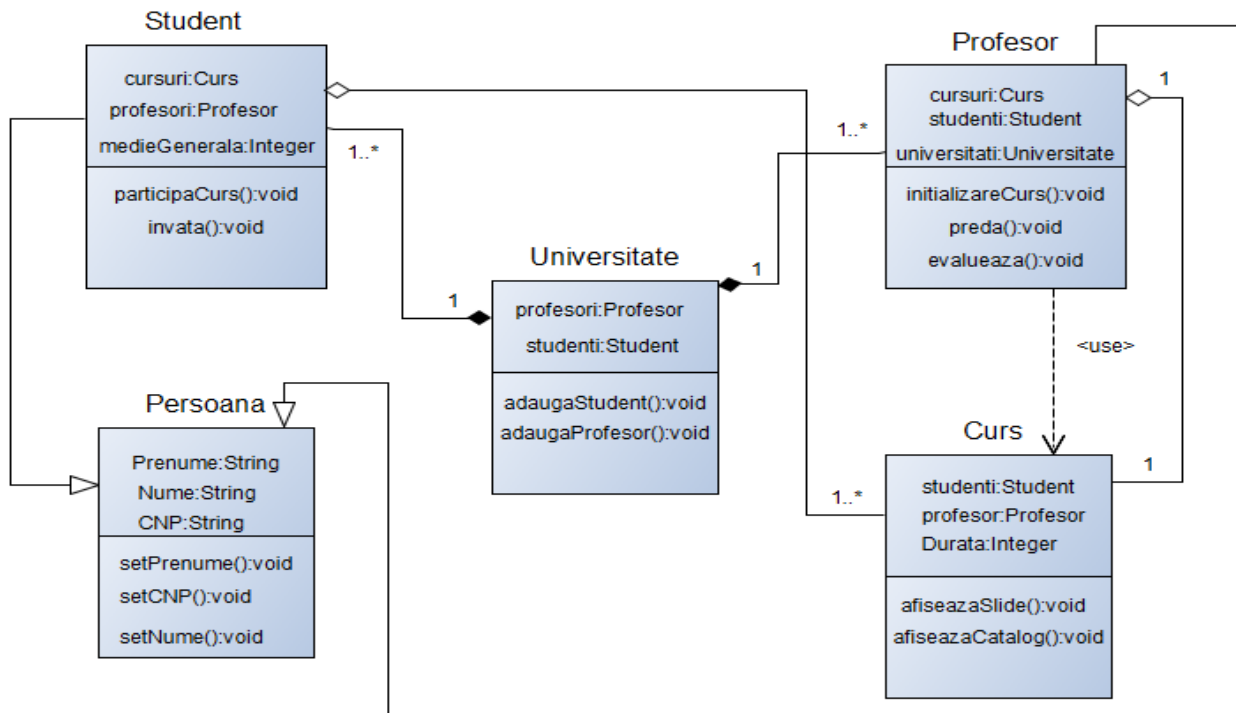


Diagrama de stare

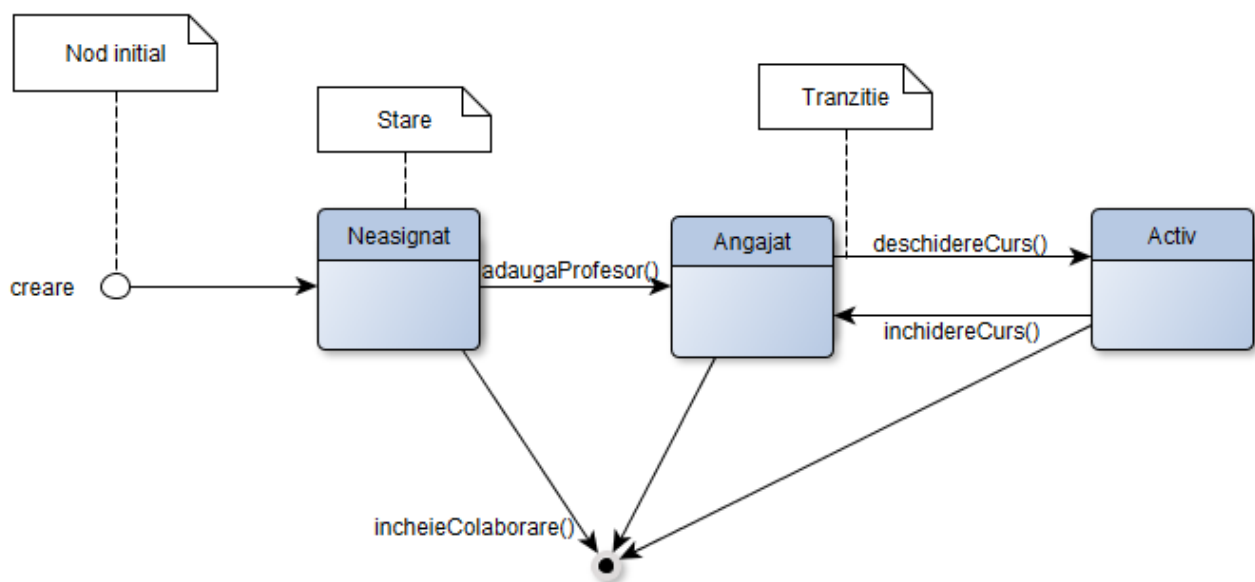
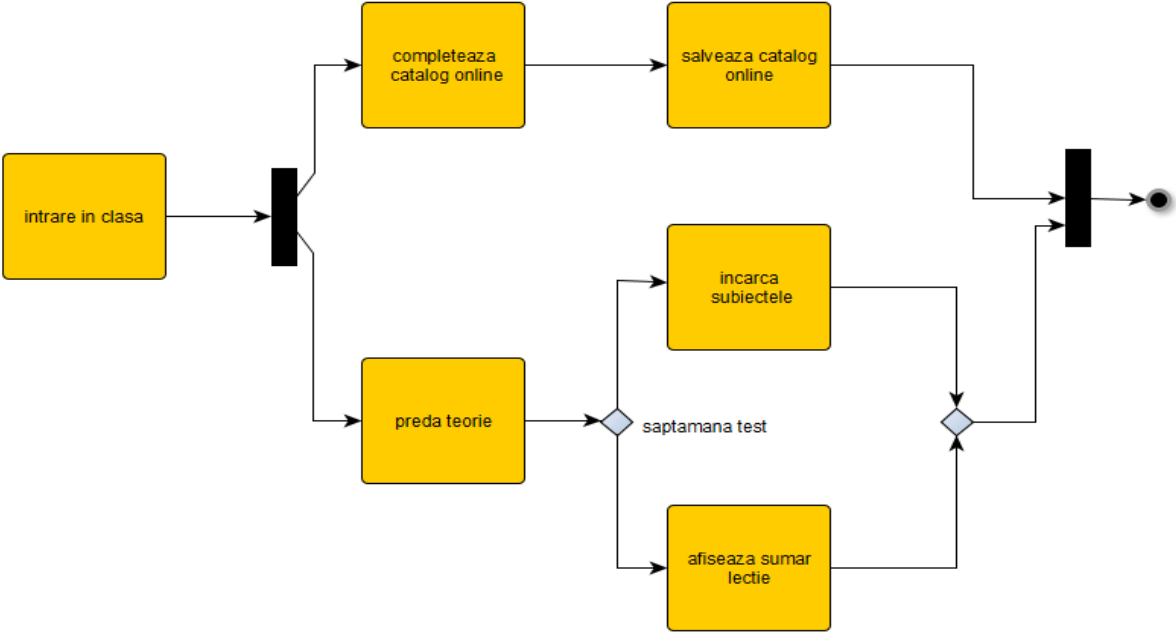


Diagrama de tip activitate



10. Concluzii

IDEF (Integration DEFinition) este folosit pentru modelarea activitatilor necesare pentru analiza, designul,, imbunatatirea sau integrarea unui sistem.

Limbajul de modelare IDEF a fost dezvoltat pentru a raspunde la trei cerinte principale de modelare a unei activitati :

- modelarea a ceea ce se cunoaste despre lumea reala și relatiile dintre oameni, evenimente, etc.
- cerinte de modelare existente și viitoare de gestionare a informațiilor
- sprijinirea metodelor de proiectare a sistemelor de raspuns la nevoile anterior identificate.

IDEF0 modeleaza functii separate, dar nu sustin specificatiile unui proces si nici nu specifica ordinea in timp in care se executa acestea (cum se intampla in cazul diagramei secventiale a limbajului de modelare UML).

IDEF1 nu poate fi folosit direct in faza de implementare. Acesta este foarte util in modelarea informatiei in cadrul activitatii, independent de constrangerile de implemenare.

In cazul modelului IDEF3 problema de baza este necesitatea de a distinge intre o descriere a ceea ce un sistem ar trebui sa faca si un model de simulare reprezentativ care va anticipa ce va face un sistem.

IDEF4 a fost dezvoltat ca un instrument de proiectare pentru dezvoltatorii de software care folosesc limbaje orientate pe obiecte, cum ar fi Common Lisp Object System, Flavors, Smalltalk, Objective-C, C++, și altele.

Metoda IDEF5 are trei componente principale: un limbaj grafic pentru a sprijini analiza ontologiei conceptuale, un limbaj text structurat pentru caracterizarea ontologiei detaliate, precum și o procedura sistematica pentru a captura ontologia eficient.

11. Bibliografie

<http://it.toolbox.com/blogs/enterprise-design/idef-modelling-pros-and-cons-10043>

<http://www.idi.ntnu.no/emner/tdt4252/2013/Additional%20reading/uml-vs-idef-griffithsuniversity.pdf>

<http://searchsoa.techtarget.com/definition/IDEF>

<http://en.wikipedia.org/wiki/IDEF>