

Reprezentarea conceptelor metodologiei OMT în modelarea sistemelor informatice complexe în medicina

Asist.dr. Elena PODOLEANU, Catedra de Informatica Medicală,
UMF "CAROL DAVILA", București

Sistemele construite astăzi sunt de mari dimensiuni, mult mai complexe și mult mai volatile față de cele de acum zece sau douăzeci de ani. Metodele orientate obiect de analiză și proiectare pot să conducă la un sistem mult mai stabil. Folosirea unor metode de modelare orientate obiect pentru astfel de sisteme, oferă o cale mult mai naturală de lucru cu asemenea sisteme. Pentru exemplificarea reprezentării conceptelor informatice ale metodei OMT, am abordat sistemul asigurărilor de sănătate în domeniul stomatologic.

Cuvinte cheie: obiect, clasă de obiecte, atribut, atribut de legătură, legătură, asociere, multiplicitate, nume de rol, generalizare, moștenire, metode, evenimente, operație, funcție, diagramă, modul, scenariu, stare, tranziție, acțiune, activitate, condiție, restricție, proces, actor.

Reprezentarea conceptelor informatice în Modelul Dinamic (MD).

Modelul dinamic prezintă starea și comportamentul sistemelor în timp și reprezintă informația de control într-un sistem de obiecte, privind: secvența de evenimente; secvența de stări; secvența de operații.

Un **eveniment** reprezintă o cale de transmitere a informației în mod unidirecțional de la un obiect la altul. Evenimentul reprezintă o modificare a contextului, care are loc la un anumit moment de timp. Evenimentul nu are durată, se consideră că se produce instantaneu. Două evenimente fără legătură cauzală se numesc *concurrente*. Ele nu au influența unul asupra altuia. Pot să apară simultan. În modelarea sistemului nu trebuie să se stabilească o ordine între evenimentele concurrente, acestea putând surveni într-o ordine aleatoare. Răspunsul la un eveniment depinde de starea obiectului receptor și poate determina: schimbarea stării; înapoierea unui eveniment (răspuns) către obiectul expeditor (emitor); emiterea unui eveniment către un alt obiect.

Deși apar într-o împrejurare unică, evenimentele pot fi grupate în *clase de evenimente* dacă sunt identificate prin: nume care indică o structură de tip ierarhic; comportament comun.

Unele evenimente sunt simple, dar cele mai multe clase de evenimente posedă

atribute indicând informațiile pe care le contin. În model atributele sunt trecute după numele clasei evenimentelor între paranteze rotunde.

Un **scenariu** este o *secvență de evenimente* care se derulează într-un sistem pentru atingerea unui obiectiv sau desfășurarea unui proces. Construirea scenariului poate diferi; el poate să includă toate evenimentele din sistem sau doar cele antrenate în conflicte sau produse de anumite obiecte. Secvența evenimentelor și obiectelor care schimbă aceste evenimente poate apărea într-un "scenariu extins" numit diagramă de urmărire a evenimentelor. În figura de mai jos este prezentat un scenariu posibil de asigurare la o casă de asigurări de sănătate:

În diagrama de mai jos este prezentat un exemplu de *scenariu privind asigurarea de sănătate stomatologică pentru un cetățean din România, care este salariat cu contract de muncă la un angajator*, ce virează contribuția de asigurare de sănătate la o casă de asigurări de sănătate (CAS) aleasă de angajat, în contul deschis de CAS la trezorerie. Drept dovadă a acestei acțiuni, angajatorul eliberează asiguratului o adeverință de plată a contribuției de asigurare de sănătate la CAS, iar CAS eliberează asiguratului dovada de asigurare.

Angajatul completeaza si semneaza Contract de munca cu angajatorul
 Angajatul completeaza si semneaza Cerere de asigurare de sanatate
 Angajatorul transmite Contract de asigurare de sanatate la CAS
 Angajatorul întocmeste si depune liste asigurate la CAS
 CAS elibereaza asiguratului dovada de asigurare
 Angajatorul vireaza lunar, în contul CAS, contributia de asigurare de sanatate a angajatului/asiguratului
 Angajatorul elibereaza asiguratului dovada pentru plata contributie asigurare de sanatate la trezorerie
 Trezoreria confirma încasare la CAS

Scenariu de asigurare de sanatate pentru un cetatean român, din România, angajat în munca.

În diagrama urmatoare este prezentata secventa evenimentelor pentru asigurarea de sanatate pentru un cetatean român, din România, angajat în munca. În aceasta diagrama obiectele au fost simbolizate printr-

o linie verticala iar evenimentele prin linii orizontale terminate cu săgeata (indica obiectul care receptioneaza evenimentul). Desfasurarea în timp a evenimentelor se prezinta de sus în jos pe diagrama, diagrama exprimând doar succesiunea în timp nu si timpul care le separa.

Angajat /Asigurat	Angajator/Asigurator	CAS	Trezorerie
Completeaza si semneaza Contract de munca			
Completeaza si semneaza Cerere de asigurare de sanatate	Transmite Contract de asigurare de sanatate		
	Întocmeste si depune liste		
	Elibereaza asiguratului dovada de asigurare		
Elibereaza adeverinta pentru plata de contributie asig. de sanatate	Vireaza lunar în cont contributia de sanatate a angajatului/asiguratului	asigurare	
		Confirma încasare	

Secventa evenimentelor pentru asigurarea de sanatate pentru un cetatean român, din România, angajat în munca.

Asa cum se observa fiecare obiect poate fi emitator de evenimente cât si receptor de evenimente.

Starea unui obiect este definita de valorile atributelor si a legaturilor lui. În timp, starea unui obiect se schimba datorita interactiunii cu alte obiecte. Raspunsul la un eveniment primit de un obiect poate varia cantitativ, în functie de valorile concrete ale fiecarui atribut, dar în termeni calitativi, el este identic pentru toate valorile aceluiasi eveniment, si poate varia în functie de valorile de stare diferite. Raspunsul poate determina o actiune sau schimbarea starii. O

stare corespunde intervalului între doua evenimente primite de obiect si specifica raspunsul unui obiect la un eveniment. Un eveniment separa doua stari, o stare separa doua evenimente.

Starea se caracterizeaza (se exprima) într-un interval de timp, are o durata si ocupa un interval de timp si adesea este asociata unei activitati care dureaza, sau la o valoare a unui obiect satisfacând o anumita conditie.

De regula, în definirea starilor, nu se tine cont de attributele care nu afecteaza comportamentul obiectelor si se reunesc într-o stare simpla toate combinatiile valorilor atributelor si legaturilor care ofera aceleasi raspunsuri la evenimente. Attributele care

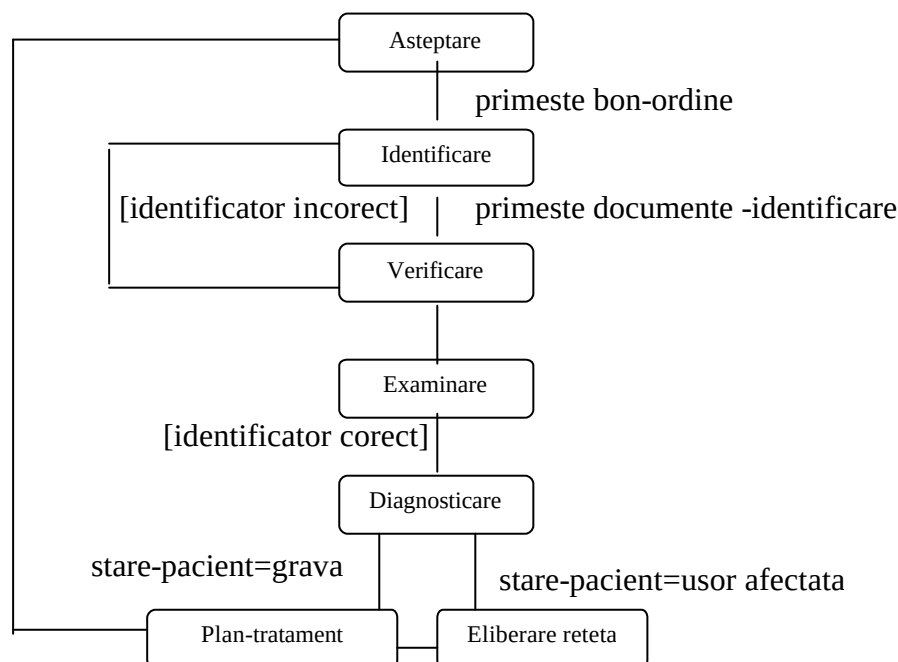
nu afectează structurile de control pot fi considerate ca valori simple de parametri în interiorul unei stări de date.

O stare se poate reprezenta sub forma tabelară, caracterizată prin: nume (care trebuie să fie cât mai sugestiv); descrierea scopului (în limbaj natural); prezentarea evenimentelor declanșatoare a stării; condiții care caracterizează starea (un tabel de prezentare a evenimentelor, acțiunilor și stărilor următoare).

Trebuie subliniat că, în măsura în care legăturile pot fi considerate ca obiecte ele pot avea de asemenea stări. Dacă un obiect se află într-o stare și dacă unul din eveni-

mentele indicate pentru una din tranzițiile sale se produce, atunci obiectul intră într-o nouă stare (starea de sfârșit de tranziție), deci a reușit. Dacă mai multe tranziții ies dintr-o stare, atunci primul eveniment care survine provoacă trecerea la tranziția corespunzătoare. Dacă un eveniment se produce și nu are ca efect tranziția de la starea curentă, atunci evenimentul este ignorat. O secvență de evenimente determină o cale de parcurgere a grafului.

În diagrama de stare prezentată mai jos, este descris serviciul medical, acordat unui pacient asigurat, într-un cabinet medical:



O diagrama de stare descrie comportamentul unei singure clase de obiecte. Deoarece toate instanțele unei clase au același comportament (prin definiție), ele sunt reprezentate de aceleași diagrame de stare. O diagrama de stare poate să se execute o singură dată sau într-o buclă continuă. Diagrama prezentată este o buclă continuă. O diagrama care se execută o singură dată reprezintă obiectele care au o existență finită și se caracterizează printr-o stare inițială și o stare finală.

Starea inițială este intrarea în crearea obiectului este simbolizată printr-un cerc plin. Intrarea în *starea finală* implică distrugerea

(dispariția) obiectului și este simbolizată printr-un cerc cu un punct negru în interior. Stărilor inițiale și finale li se asociază un nume cât mai sugestiv. O diagrama “o singură dată” poate fi considerată ca un subprogram de diagrame de stări putând fi referită în diverse locuri în diagramele de nivel înalt.

Modelul dinamic este constituit ca o colecție de diagrame de stări integrate unele cu altele prin intermediul evenimentelor partajate (comune), constituind structura de control a sistemului. Diagramele de stare pot fi structurate pentru a permite descrieri concise a sistemelor complexe.

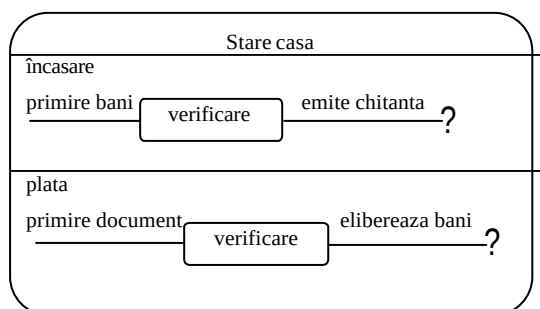
Imbricarea este echivalenta generalizarii si agregarii. Asemănător generalizării, imbricarea permite descrierea unei activități la un nivel înalt (general), pentru ca apoi sa fie rafinata la un nivel inferior, adăugând amănunte. Mai mult, generalizarea permite stărilor si evenimentelor sa fie organizate într-o ierarhie de generalizare cu mostenirea structurii si a comportamentului comun. Agregarea este echivalenta concurenței stărilor. Concurența stărilor corespunde adesea agregării obiectelor sau întregului sistem, înglobând parti interactive. O activitate, în anumite situații, se poate întinde într-o diagrama de stare la cele mai mici nivele, fiecare stare reprezentând un pas al activității. Activitățile imbricate sunt diagrame de stare “*o singura data*” cu tranziții de intrare si iesire, utilizate într-o diagrama de nivel înalt si precizata, în mod detaliat, separat într-o subdiagrama.

Evenimentele pot de asemenea sa se întinda pe mai multe diagrame de stare subordonate.

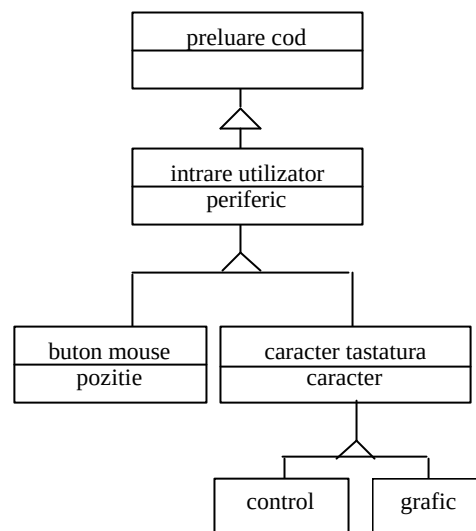
Generalizarea stărilor este o diagrama de stare imbricata. Generalizarea exprima “relatia sau”. Stările din diagramele imbricate sunt rafinări ale diagramelor de nivel înalt. Ele pot avea substarile care mostenesc tranzițiile superstarilor lor. Fiecare tranziție sau actiune aplicata la o stare se aplica la toate substarile lor, mai puțin dacă vor fi rescrise pentru o tranziție echivalenta pentru o substare.

O **superstare** se reprezintă printr-un dreptunghi mare cu colturile rotunjite care încadrează substarile sale. În dreptunghiul superstarii pot fi cuprinse alte generalizări de nivel mai mic care se vor simboliza la fel.

În diagrama de mai jos se prezintă o generalizare a stării unui pacient.



Generalizarea evenimentelor este o organizare a evenimentelor într-o ierarhie de generalizare cu mostenirea atributelor evenimentelor si permite utilizarea diferitelor nivele de abstractizare la diferite nivele de precizie ale modelului. În diagrama de mai jos se prezintă evenimentul “preluare cod” care se poate face tastându-l, sau selectându-l dintr-o fereastră activa.



O **conditie** este o functie logica a valorii obiectelor (având ca parametrii obiecte) si poate avea durata în timp (spre deosebire de condițiile evenimentelor care nu au durata în timp). Condiția este simbolizata printr-o expresie logica pusa între paranteze drepte si se plasează după numele evenimentului.

Condițiile pot fi utilizate ca “filtre” pentru tranziții. O tranziție se produce când are loc un eveniment doar dacă este îndeplinita si o anumita condiție.

Operatiile atasate pot fi definite ca activități sau ca actiuni. Operatiile atasate stărilor si evenimentelor sunt executate ca răspuns la stările sau evenimentele corespunzătoare.

Diagramele de stare ar fi prea puțin utile dacă nu ar prezenta reacția obiectului la perceperea evenimentului respectiv.

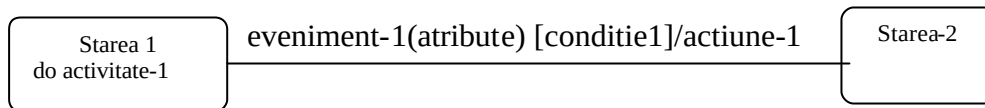
O **activitate** este o operatie care necesita un anumit timp de executie; este atasata unei stări si se execută în starea respectiva. Activitățile cuprind *operatii continue* (de exemplu: afisează fereastra pe ecran) sau

operatii *secventiale* care se repeta, revenind tot la ele dupa un interval de timp (executarea unui calcul).

O activitate se simbolizeaza precedând-o de cuvântul “do” (executa) în dreptunghiul de prezentare a starii si exprima faptul ca activitatea începe de la intrarea în starea

respectiva si înceteaza la trecerea la o noua stare (pentru activitatile continue) sau ca activitatea secventiala începe la intrarea în starea respectiva si înceteaza când este gata.

Exemplu de sintaxa generala:



Daca un eveniment provoaca o tranzitie de stare înainte ca activitatea sa se termine, atunci activitatea este terminata prematur. O activitate continua poate fi asimilata unei activitati secventiale care dureaza nelimitat.

O **actiune** este o operatie instantanee, se asociaza unui eveniment si se simbolizeaza prin “/” (slash) urmat de numele actiunii care succede evenimentului care a produs-o. Desi în lumea reala nu exista actiuni instantanee totusi durata lor este nesemnificativa în modelul de control. Daca totusi se acorda importanta timpului, atunci ea va fi tratata ca activitate atasându-i: un eveniment de început; un eveniment final; evenimente intermediare posibile.

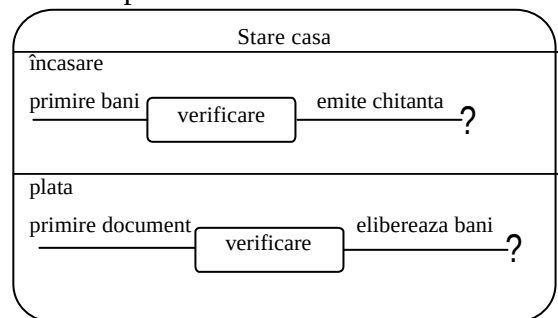
Actiunile pot de asemenea sa reprezinte: operatii de control intern; modificari ale atributelor sau producerea altor evenimente.

Aceste actiuni nu au un echivalent în lumea reala, dar sunt mecanisme de control în structurile interne de implementare.

Concurenta. Un model dinamic descrie ansamblul de obiecte concurente, fiecare având propriile stari si diagrame de stare. Obiectele într-un sistem sunt, în mod inerent concurente si își pot schimba starea independent. Starea unui sistem nu poate fi reprezentata printr-o stare unica într-un obiect unic; este produsul tuturor starilor, al tuturor obiectelor. În multe sisteme, numărul obiectelor se poate schimba dinamic. O diagrama de stari pentru un ansamblu este o serie de diagrame de stare, câte una pentru fiecare componenta. Agregarea implica concurenta. Starea întregului este sta-

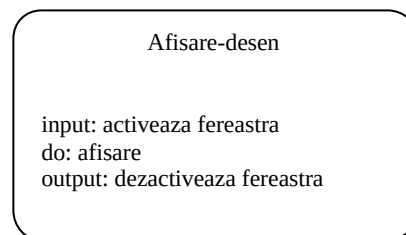
rea primei diagrame si starea celei de-a doua etc. Starile componente pot interaciona. Tranzitiile unui obiect pot depinde de un alt obiect într-o stare data. Aceasta permite o interactiune între diagramele de stare, fiecare pastrându-si modularitatea.

Concurenta poate surveni si în interiorul unui obiect, când obiectul poate fi partitionat în subansamble de attribute, fiecare din ele având propriile subdiagrame. Starea obiectului înglobeaza starea fiecărei subdiagrame. Nu este necesar ca fiecare subdiagrama sa fie independenta; acelasi eveniment poate provoca tranzitii în mai multe subdiagrame. Concurenta în interiorul unei stari compuse a unui unic obiect se simbolizeaza prin înscrierea fiecărei diagrame de substari despartita prin linii orizontale într-un dreptunghi cu colturi rotunjite. În diagrama urmatoare se prezinta stările obiectului “casa” care pot fi : “încasare” sau “plata”:

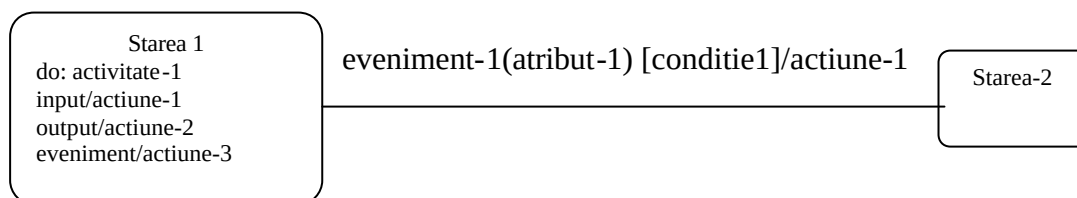


Actiunile de intrare si de iesire. Uneori actiunile atasate evenimentelor care determina o tranzitie pot fi introduse în casuta de stare. Daca aceste actiuni se executa la intrarea într-o stare indiferent de tranzitia care a realizat-o si daca nu pot fi întrerupte, sunt precedate de cuvântul “input” si “/”;

Daca aceste actiuni se executa la iesirea dintr-o stare de fiecare data pentru tranzitia la o noua stare si daca nu pot fi întrerupte, sunt precedate de cuvântul “**output**” si “/”. Trebuie subliniat ca daca un eveniment atasat tranzitiei are la intrare si o actiune, aceasta se executa înaintea actiunilor “input”, iar operatiile spacificate prin “do” se executa dupa actiunile “input”, dar înaintea operatiunilor “output” si pot fi întrerupte. Actiunile de intrare si de iesire sunt utile în diagramele de stare imbricate caci permit o clarificare mai buna a prioritatilor executarii actiunilor.



Un eveniment poate provoca executia unei actiuni fara sa provoace schimbarea unei stari. Aceste evenimente se înscriu în interiorul dreptunghiului de stare urmate de “/” si de actiunea care se întreprinde la aparitia evenimentului. Forma completa de simbolizare a diagramelor de stare este:



Reprezentarea conceptelor informatice în Modelul Funcțional (MF)

Modelul functional descrie calculele din interiorul sistemului; arata ce se întâmpla în interiorul sistemului. Cu ajutorul lui observam cum se obtin valorile de iesire pornind de la valorile de intrare, fara sa se tina seama de ordinea în care ele sunt calculate (aspect reflectat de modelul dinamic). Modelul functional contine multiple diagrame de flux de date (*data flow diagrams*), de la intrarile provenite din exterior, traversând operatiile si colectiile interne de date. De asemenea, contine si restrictiile între valorile interne ale modelului obiect. Diagramele de **flux de date** nu contin informatii de control sau de structura a obiectelor. Specifica atât semnificatia operatiilor si a restrictiilor din modelul obiect cât si semnificatia actiunilor si a activitatilor din modelul dinamic

O diagrama a fluxului de date arata relatii functionale între valorile calculate de un sistem, si cuprinde: valorile de intrare ; valorile de iesire; structurile datelor interne (colectiile de date interne).

Un flux de date leaga iesirea unui obiect sau a unui proces cu intrarea altui obiect sau proces. El reprezinta o valoare de date

intermediare într-un calcul; nu schimba datele.

Aceiasi valoare poate fi transmisa în locuri diferite. Aceasta situatie se simbolizeaza prin doua sau mai multe sageti care pornesc din acelasi punct. Sagetile nu au etichete fiindca au aceiasi valoare.

Se simbolizeaza printr-o sageata care leaga producatorul si consumatorul informatiei. Sageata este însoțita de o eticheta care descrie datele (tipul datelor sau denumirea lor).

Alteori, o valoare de data agregata este divizata în partile componente, fiecare parte având alta destinatie. Se reprezinta acest proces prin doua sau mai multe sageti care pornesc din acelasi loc, fiecare eticheta poarta numele partii componente (combinarea mai multor parti componente într-o valoare unica se face în sens invers – sagetile componente se unesc în acelasi loc determinând valoarea rezultata).

Fiecare sageata de date reprezinta o valoare a unei etape de evaluare. Fluxurile de date interne ale diagramei reprezinta valori intermediare în interiorul unui calcul si nu au în mod obligatoriu o semnificatie în lumea reala.

Sagetile care sunt la frontiera diagramei fluxurilor de date sunt intrarile si iesirile sale. Aceste sageti pot sa nu fie conectate (daca diagrama reprezinta o parte a unui sistem complet), sau pot sa fie conectate obiectelor.

O diagrama a fluxului de date este un graf care arata sensul în care sunt transformate ("curg") datele, fluxul valorilor datelor începând de la sursa lor din obiecte, trecând prin prelucrările (procesele) care le transforma, spre destinatia lor în alte obiecte.

O diagrama de flux contine: *proces* care transforma datele; *obiecte (actori)* care produc si consuma datele; *structuri (colecții) de date* care stocheaza datele.

Procesele (prelucrările) transforma valorile datelor. Prelucrările de cel mai jos nivel sunt functii pure (care doar calculeaza) fara efecte laterale (exemplu.: adunarea a doua valori, împartirea cu rest etc.)

O prelucrare poate avea efecte laterale daca contine componente nefunctionale ca de exemplu obiecte externe sau structuri de date. Modelul functional cu efecte colaterale nu indica numai rezultatele unei prelucrări ci si drumurile functionale posibile (variantele posibile fara a o indica pe cea care va fi urmata). Rezultatele unei anumite prelucrări depind de comportamentul sistemului, dupa cum este precizat în modelul dinamic.

Procesul se simbolizeaza printr-o elipsa (sau un cerc) continând o descriere a prelucrării. Fiecare proces are un numar fix de sageti de intrare si de iesire, fiecare din ele simbolizeaza o valoare a unui tip de data.

Intrarile sau iesirile pot fi denumite în scopul evidentierii rolului lor în evaluare, dar de cele mai multe ori specificarea tipului valorilor este suficienta. Procesele sunt implementate ca metode (sau fragmente de metode) ale operatiilor claselor de obiecte. Un **actor (terminator)** este un obiect activ care dirijeaza graful fluxului de date producând sau consumând informatie. Actorii

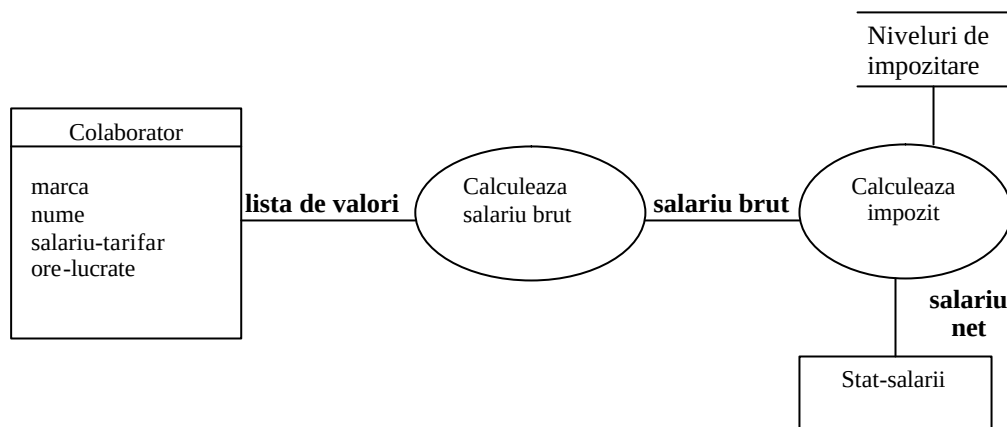
sunt atasati intrarilor sau iesirilor unei diagrame a unui flux de date. Actiunile actorilor sunt la exteriorul domeniului diagramei fluxului de date, dar trebuie sa faca parte din modelul dinamic. *Un actor se simbolizeaza printr-un dreptunghi* pentru a arata ca se refera la un obiect.

O structura (colectie) de date este un obiect pasiv în interiorul unei diagrame a fluxului de date care stocheaza date pentru un acces ulterior. Diferit de un actor, o structura de date nu angajeaza operatii din proprie initiativa, dar raspunde cererilor de stocare sau accesare a datelor. Colectiile de date permit accesul la valori într-o ordine diferita de cea în care au fost produse. Accesul se realizeaza prin ordinea inserarilor sau al cheilor index.

Colectiile de date se simbolizeaza prin doua linii paralele ce contin numele colectiei. Liniile care intra indica informatiile si operatiile care modifica datele stocate (stergere, adaugare, modificare), iar liniile care ies indica informatiile preluate din colectie. Structura reala a obiectului trebuie descrisa în modelul obiectelor unde trebuie sa se prezinte printre altele, operatiile de actualizare si tipurile de acces autorizate.

Actorii si colectiile de date sunt obiecte. Acestea se deosebesc datorita comportamentului si utilizarii lor diferite, dar ca o caracteristica pot fi implementate ca obiecte într-un limbaj obiect. Pe de alta parte o structura (colectie) de date poate fi implementata ca un fisier si un actor ca un dispozitiv extern. Anumite fluxuri de date pot fi de asemenea obiecte, cu toate ca se manifesta, în cele mai multe cazuri ca valori pure, ca un întreg (ansamblu) fara identitate proprie.

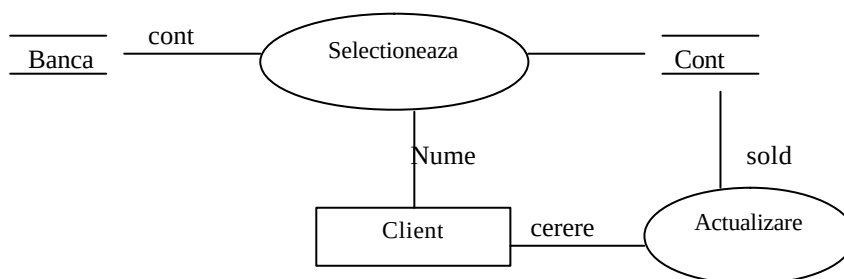
Exista diferente între a vedea un obiect ca pe o valoare unica si a-l vedea ca pe o colectie de date continând mai multe valori. Exemplu de diagrama de flux de date pentru calculul impozitului pe salariu al unui colaborator:



Diagramele fluxului de date sunt în particular utilizate pentru a prezenta funcția de cel mai înalt nivel a sistemului și descompunerea lui în unități functionale mai mici (motiv pentru care mai sunt cunoscute și ca diagrame de nivel). O prelucrare poate fi dezvoltată într-o nouă diagramă de flux de date (imbricare). Fiecare intrare sau ieșire a prelucrării este o intrare sau o ieșire a noii diagrame. Noua diagramă a fluxului de date poate avea colecții (structuri) de date absente din diagrama de nivel înalt. Diagramele pot fi imbricate la un număr arbi-

trar de rafinări și ansamblul complet de diagrame imbricate formează un arbore. Nivelul cel mai scăzut de imbricare se termină prin funcții simple. Imbricarea diagramelor de date permite fiecărui nivel să fie coerent și inteligibil. O diagramă care se referă tot la ea, simbolizează un calcul repetitiv.

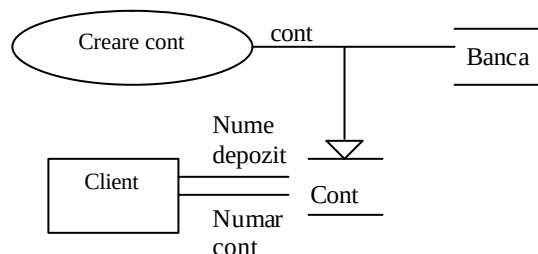
În diagrama următoare, numele clientului selectează un cont la bancă. Rezultatul acestei operații este obiectul cont, care este utilizat ca o colecție de date în operația de actualizare.



Un flux de date care generează un obiect utilizat ca țintă într-o altă operație este simbolizat printr-un triunghi (sageata goală) la sfârșitul fluxului, către obiectul generat (selectat).

Exemplu de adăugare a unui nou cont într-o bancă în care obiectul cont este văzut ca o valoare de date (stocată în bancă) cât și ca o colecție de date (utilizat pentru a stoca și a furniza informații).

Rezultatul prelucrării "creare cont" este un nou cont, blocat în bancă. Numele și depozitul clientului sunt stocate în cont. Numarul noului cont este transmis clientului.



Un **flux de control** este o valoare logică ce permite să se știe dacă o prelucrare are loc. Valoarea logică nu este o dată de intrare în prelucrarea propriu-zisă și se simbolizează printr-o linie punctată, plecând dintr-o prelucrare care produce o variabilă logică, către o prelucrare controlată (prelucrarea res-

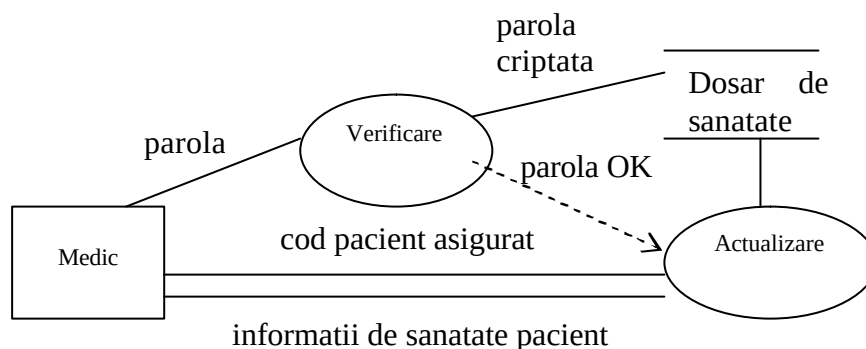
pectiva se executa doar în anumite conditii).

O diagrama de flux de date arata toate variantele posibile de calcul a valorilor; ea nu arata nici ce variabila se executa si nici în ce ordine.

Deciziile si secventele sunt probleme de control si sunt prezentate de modelul dinamic. O decizie permite sa se stie daca una sau mai multe functiuni sunt executate mai repede decât de a furniza o valoare a

prelucrării. Totusi daca functiile nu au valori de intrare provenind de la aceste functii de decizie, este uneori util sa fie înglobate în modelul functional, pentru a nu le uita si pentru a exprima dependentă lor fata de date.

Exemplu de diagrama de flux de date pentru actualizarea unui dosar de sanatate al unui pacient asigurat, de catre medicul specialist stomatolog. Medicul furnizeaza parola si codul pacientului.



Accesul la dosarul de sanatate al pacientului nu poate avea loc decât daca este corecta parola. Deci executarea operatiei de actualizare a dosarului de sanatate al pacientului este conditionata de cunoasterea parolei. Evident ca dosarul pacientului nu poate fi identificat si actualizat daca codul pacientului nu este corect.

Prelucrările (procesele) din diagramele fluxului de date trebuie la sfârșitul modelării să fie implementate ca operatii pe obiecte. Fiecare prelucrare atomica sau la nivelul cel mai scăzut este o operatie.

Prelucrările de nivel înalt pot fi de asemenea considerate ca operatii, dar în rafinările ulterioare pot fi implementate diferit pentru anumite nivele (redefinite). Operatia este de obicei inclusa în modelul obiect pentru a arata structura de mostenire. Fiecare operatie poate fi precizata în diferite feluri: functii matematice; tabele de valori de intrare/iesire; ecuatii specificând iesirile în functie de intrari; pre si post conditii (definitii axiomatice); tabele de decizie; pseudo-cod; limbaj natural.

Specificarea unei operatii presupune o semnatura ("amprenta") si o prelucrare (transformare).

Semnatura definește interfata operatiei: argumentele cerute la intrare (numar, ordine tip); valorile returnate (numar, ordine, tip). Semnaturile tuturor metodelor implementate pentru o operatie trebuie să fie compatibile. Transformarea definește efectul operatiei (ce face operatia); valorile rezultate ca functii ale valorilor de intrare; efectele asupra obiectelor operanzii. Specificarea externa a unei operatii descrie în mod unic schimbarile vizibile la exteriorul operatiei. În timpul implementării unei operatii, valorile interne (intermediare) pot fi create din motive de optimizare sau comoditate. Astfel de detalii sunt "private" (proprii) unei operatii si nu apar în specificatiile externe. Obiectivul specificatiilor este de a indica ce trebuie să facă operatia si nu cum va face.

În consecinta și starea unui obiect trebuie să fie divizata în informatii vizibile în exterior și informatii "private" interne. Schimbarile stărilor interne ale unui obiect care

nu sunt vizibile la exterior, nu modifica valoarea (starea obiectului).

Marea majoritate a limbajelor orientate obiect disting între atribute și operații publice (vizibile din exterior) și operații private (accesibile doar din cadrul obiectului respectiv).

Din alt punct de vedere operațiile pot fi împartite în:

Operațiile triviale sunt operațiile de acces, care doar citesc sau scriu atribute sau legături ale unui obiect. Nu este necesar de a enumera sau a preciza operațiile de acces în timpul analizei căci ele sunt evidente. În timpul proiectării, va trebui să se noteze operațiile publice și cele care vor fi private unei clase de obiecte. Ratiunea restricțiilor de acces nu ține de coerență logică; se procedează așa mai curând pentru a permite încapsularea datelor.

Operațiile netriviale pot fi împartite în trei categorii: interogări (query); acțiuni; activități.

O interogare este o operație care nu are efecte laterale asupra stării exterioare a unui obiect; este o funcție pură, care doar calculează niste valori.

O interogare fără parametri (cu excepția obiectului tinta) este un atribut derivat (el este aparent un atribut, dar nu este obligatoriu să fie implementat). De exemplu, dacă obiectul *Pacient* conține data nașterii, *vârsta* reprezintă un atribut derivat.

În modelele obiect, operațiilor interogări le pot fi asociate atribute, dar statutul lor derivat trebuie indicat deoarece ele nu furnizează informații suplimentare asupra stării obiectelor. În numeroase cazuri, alegerea atributelor ca atribute de bază sau derivate este arbitrară. Operațiile *interogări* sunt derivate pornind de la variantele modelului obiect sau recalculând datele pornind de la același model.

O *acțiune* este o transformare având efecte laterale la frontiera obiectului tinta sau asupra altor obiecte din sistem, putând fi atinse pornind de la obiectul tinta – dar fără durată în timp (“instantanee din punct de vedere logic”).

Așa cum starea unui obiect este definită de atributele și legăturile sale, acțiunile trebuie să fie definite în termenii modificărilor atributelor de bază și a legăturilor. O acțiune poate fi definită în termenii stării sistemului înainte și după acțiune. Acțiunile sunt în general derivate pornind de la procesele (prelucrările) modelului funcțional. Se pot descrie acțiuni în diferite moduri: ecuații matematice; arbori de decizie; tabele de decizie; enumerarea tuturor intrărilor posibile; calculul predicatului; limbaj natural.

Este important ca o specificație să fie clară și neambiguă dar nu în mod obligatoriu formularizată.

Descrierea acțiunii presupune specificarea următoarelor elemente: numele funcției; intrările; ieșirile; prelucrările (ce face funcția, inclusiv efecte laterale); restricțiile care trebuie urmărite (constrângerile asupra valorilor de intrare).

Exemplu:

Funcția: avizarea plății serviciilor stomatologice sigurate Intrări: Lista serviciilor medicale asigurate și tarifele aferente, Lista asiguraților pe grupe de vârstă, lista serviciilor stomatologice furnizate Ieșiri: suma de platit Prelucrări: se compară serviciile asigurate cu serviciile executate pentru fiecare pacient, se analizează modul și termenul de plată Restricții: dacă au fost depuse autorizația și acreditarea furnizorului de servicii medicale stomatologice

O *activitate* este o operație a unui obiect care are durată și are efecte laterale datorită desfășurării în timp asupra obiectelor (actori sau colecții de date). Detaliile unei activități sunt precizate atât de modelul dinamic cât și de cel funcțional. În cele mai multe cazuri, o activitate corespunde diagramei de stare din modelul dinamic.

O *restricție (constrângere)* face să apară fie relația între două obiecte la același moment de timp (ca, de exemplu frecvența și lungimea de undă), fie relația între valori diferite ale aceluiași obiect, dar la momente diferite.

O restricție între două valori poate fi exprimată:

- printr-o functie partiala (o valoare este marginita prin alta, dar nu este precizata în mod în mod complet);
 - printr-o functie totala (o valoare este precizata în mod complet prin alta.
 - Restrictiile pot aparea în fiecare tip de model:
 - restrictiile în modelul obiect specifica dependenta obiectelor, partiala sau totala, fata de alte obiecte;
 - restrictiile în modelul dinamic, specifica relatiile (legaturi) între starile sau evenimentele diferitelor obiecte;
 - restrictiile în modelul functional indica restrictii asupra operatiilor.
- O restrictie între valorile unui obiect pe toata durata de viata se numeste invariant.

Bibliografie

1. *LE MOIGNE, J.I.*, La modernization des system complexes, Ed. Bordas, Paris, 1990
2. *RUMBAUGH J., BLAHA M., PREMERLANI W., EDDY F., LORENSEN W.* Object-oriented modeling and desing, Ed. Prentice-hall International, Inc., 1991
3. *C. SPIRCU, LOPATAN I.*, Analiza, proiectarea si programarea orientate spre obiecte, ed. Teora, 1995
4. *E. PODOLEANU.*, Posibilitati de utilizare a informaticii de gestiune în modernizarea asistentei medicale în România, Teza de doctorat, ASE, Bucuresti, iulie 2000
5. *** ACSIOME Modelization dans la conception des systemes d'information, Ed. Masson, Paris, 1989
6. *** Legea asigurarilor sociale de sanatate nr. 145/1997, Revista Economistul, Editie speciala, Nr. 295, august 1997
7. *** Hotarârea Guvernului nr. 312/1999, Ordin pentru aprobarea Normelor metodologice de aplicare a Contractului-cadru privind conditiile acordarii asistentei medicale în cadrul sistemului asigurarilor sociale de sanatate pe anul 1999, Monitorul Oficial al României, Partea I, Anul XI, Nr. 436, 7 septembrie 1999